# SIEMENS

## SIMATIC S5

## Counter Module
## – IP 242A With
##   FB 178/179/180/181/182
## – IP 242B With
##   FB 183/184

**Equipment Manual**

**Release: 02**

Order No: 6ES5 998–0KM21

| | |
|---|---|
| Introduction | |
| General Function Description | 1 |
| Hardware Settings | 2 |
| Software Settings (Registers) | 3 |
| Guidelines for Interrupt Processing | 4 |
| Putting the Counter Module into Operation | 5 |
| Control Words | 6 |
| Special Functions | 7 |
| Calculation Functions | 8 |
| Operation Modes | 9 |
| Technical Specifications | 10 |
| Programming Instructions, FB 178/179 | 11 |
| Program Example for IP 242A | 12 |
| Programming Instructions, FB 180/181/182 | 13 |
| Programming Instructions, FB 183/184 | 14 |
| Program Example for IP 242B | 15 |
| Function Blocks FB 38 and FB 39 (Only for PLC S5–115U) | 16 |
| Glossary | 17 |
| Index | 18 |
| IP 242B with S7-400 (Appendices A, B, C) | 19 |
| Suggestion Form | 20 |

ENVIRONMENTAL
PROTECTION
IN ACTION

## Information Concerning
## Packaging Material/Notes on Disposal

**Dear Customer !**

Our high–quality products cannot reach you safely without effective protective packaging. The size of the packaging is kept to an absolute minimum.

All our packaging materials are harmless to the environment and can be disposed of without danger.

Wood is not chemically treated.

Cardboard is made primarily of waste paper which can then be torn up and given to a waste paper collection.

Sheeting is made of polyethylene (PE), tapes of polypropylene (PP) and CFC–free padding of foamed polystyrene (PS).

These materials are pure hydrocarbons and can be recycled. Please dispose of these valuable se–condary raw materials at a recycling center.

Recycling saves raw materials and cuts down on the amount of trash.

Ask your city administration for the address of the recycling center nearest you to dispose of packing materials and discarded devices.

**Thank you for your help !**

# SIEMENS

# Warning

**Risks involved in the use of so–called SIMATIC–compatible modules of non–Siemens manufacture**

"The manufacturer of a product (SIMATIC in this case) is under the general obligation to give warning of possible risks attached to his product. This obligation has been extended in recent court rulings to include parts supplied by other vendors. Accordingly, the manufacturer is obliged to observe and recognize such hazards as may arise when a product is combined with products of other manufacture.

**For this reason, we feel obliged to warn our customers who use SIMATIC products not to install so–called SIMATIC–compatible modules of other manufacture in the form of replacement or add–on modules in SIMATIC systems.**

Our products undergo a strict quality assurance procedure. We have no knowledge as to whether outside manufacturers of so–called SIMATIC–compatible modules have any quality assurance at all or one that is nearly equivalent to ours. These so–called SIMATIC–compatible modules are not marketed in agreement with Siemens; we have never recommended the use of so–called SIMATIC–compatible modules of other manufacture. The advertising of these other manufacturers for so–called SIMATIC–compatible modules wrongly creates the impression that the subject advertised in periodicals, catalogues or at exhibitions had been agreed to by us. Where so–called SIMATIC–compatible modules of non–Siemens manufacture are combined with our SIMATIC automation systems, we have a case of our product being used contrary to recommendations. Because of the variety of applications of our SIMATIC automation systems and the large number of these products marketed worldwide, we cannot give a concrete description specifically analyzing the hazards created by these so–called SIMATIC–compatible modules. It is beyond the manufacturer's capabilities to have all these so–called SIMATIC–compatible modules checked for their effect on our SIMATIC products. If the use of so–called SIMATIC–compatible modules leads to defects in a SIMATIC automation system, no warranty for such systems will be given by Siemens.

In the event of product liability damages due to the use of so–called SIMATIC–compatible modules, Siemens is not liable since we have taken timely action in warning users of the potential hazards involved in so–called SIMATIC–compatible modules."

# Table of Contents

# Introduction

## Principle of Counting

Counting is primarily the acquisition and addition of events. In the field of electronics, this is the addition of pulses.

### Counting Up
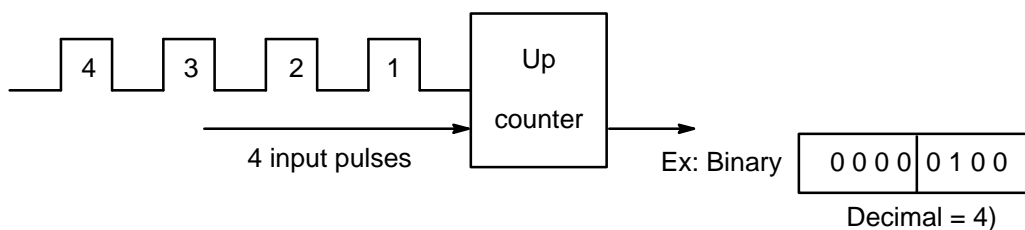


This type of counting is used, for example, for the acquisition of a simple piece count.

The AM 9513A counting chip on the IP 242A/242B, however, allows the counter to be used in a great variety of ways.

### Counting Down

Starting with an initial value stored in a register (memory location) on the IP, the counter is decremented when a pulse occurs.
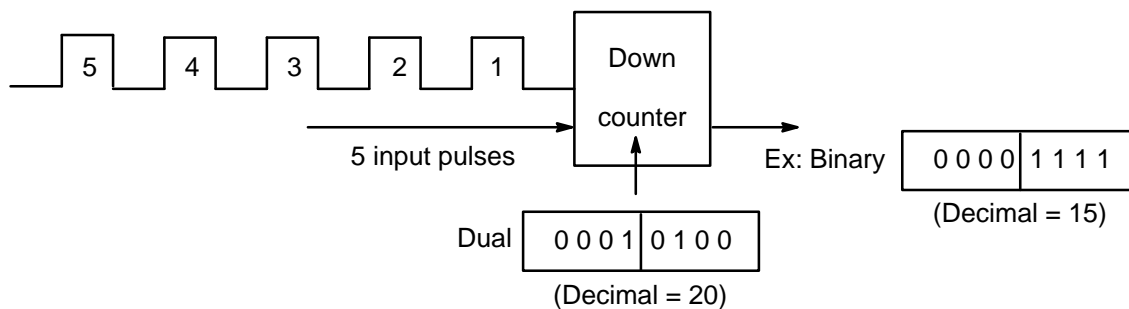


This type of counting is used, for example, when you want to count out an exact quantity and then package it. The counter is equipped with an output signal which allows you to close a valve, for example, when the counter reaches zero.

**Counting with Software Start (Or Software Stop)**

It is often necessary to link the pulses starting at a defined point in time depending on other input values, and to start the counter with this collective information.
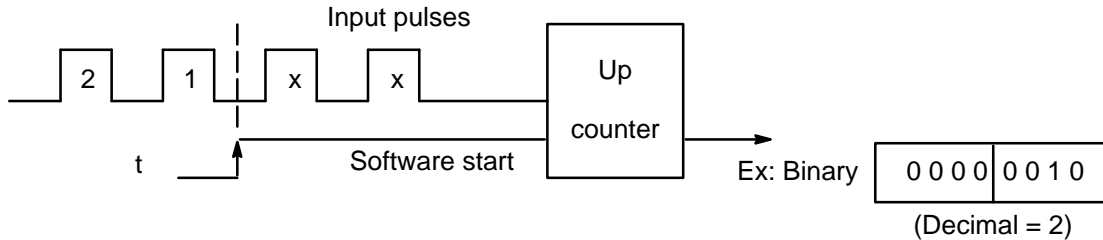


This is used, for example, when you know that the first products will have a different shape, color, quality of material or other deviations from the serial product.

**Counting with Gate Signal**

When a hardware signal directly from the system (i.e., from the process) is to start a high–speed counting procedure, it is best to connect this signal directly to the counter. This allows counting without occupying the S5 cycle.
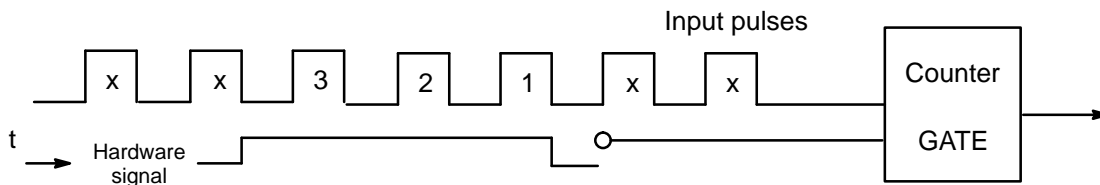


Here, the gate provides a defined stop of the counting procedure.

**One–Time Counting**

After a software start, the counter is started with the value stored in the load register, and begins counting starting with this value.

The counter is stopped by the following.

– Overflow of the counter
– Underflow of the counter

The counter remains stopped until another software start occurs.

**Cyclic Counting**

After a software start, the counter is started with the value stored in the load register, and begins counting starting with this value.

When a counting range limit is exceeded (overflow or underflow), the counter is loaded with the load value again, and resumes counting starting with this value. Another software start is not required here.

## Setup of This Manual

This equipment manual furnishes extensive information on the installation and operation of the IP 242A and IP 242B modules.

Before starting work with the module, please take the time to look through this equipment manual. Feel free to spend more time on the passages which are of particular interest to you. We want to give you a general feel of the subject matter and an overview of the information this manual provides.

Each section is self–contained and provides information on one of the following aspects of the module.

– Installation

– Programming (parameterization)

– Handling (operating)

In particular, you will find information concerning your special applications of the module in the following sections.

| User group | IP 242A | IP 242B |
|---|---|---|
| First–time user | 2, 3, 5, 6, 7, 9, 11, 12, 13 | 2, 3, 5, 6, 7, 8, 9, 14, 15 |
| Experienced user | 3, 6, 7, 11, 12, 13 | 3, 6, 7, 8, 14, 15 |
| Specialist | IP 242A Short Instructions | IP 242B Short Instructions |

The following criteria have been used to divide the users into groups.

**Experienced users** have accumulated experience in handling SIMATIC S5 controllers and counter mo–dules. **Specialists** are experienced users who have had long years of experience in handling SIMATIC S5 controllers and IP modules.

All steps required to commission the module are listed in section 5. These steps must be performed in the order given.

| 1. Definition of the counting task | ⇒ | 2. Hardware settings (DIL switches and jumpers) | ⇒ | 3. Software settings (parameteri–zation and con-trol) | ⇒ | Process |
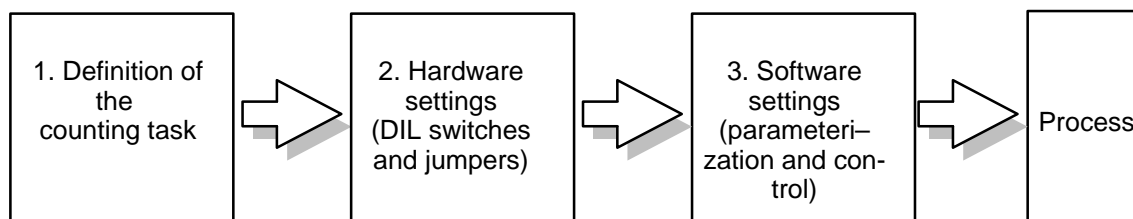
Section 13 contains the programming instructions for **standard function block FB 180, FB 181 and FB 182**. These blocks can be used for fast handling of the IP 242A module in the interrupt branch (e.g., OB 2). The commands which use these function blocks have significantly shorter load times than the call in standard function blocks FB 178 and FB 179.

**Function blocks FB 38 and FB 39** are used to save or load the scratchpad flags/system data, and the page frame number during interrupt processing in PLC S5–115U. They can be used for both the IP 242A and IP 242B.

## Differences Between the IP 242A and IP 242B

Counter modules IP 242A and IP 242B are equipped with the same hardware.

In functionality, the IP 242B module is a direct extension of the IP 242A. This applies in particular to the following areas.

– Implementation of calculation functions directly on the IP 242B

– Expansion of the counting length of counters 6 and 7 from 24 bits to 32 bits

– Storage of up to eight parameterization data records in the EEPROM of the IP 242B

– Provision of a measured value memory with trace function (collection of past values)
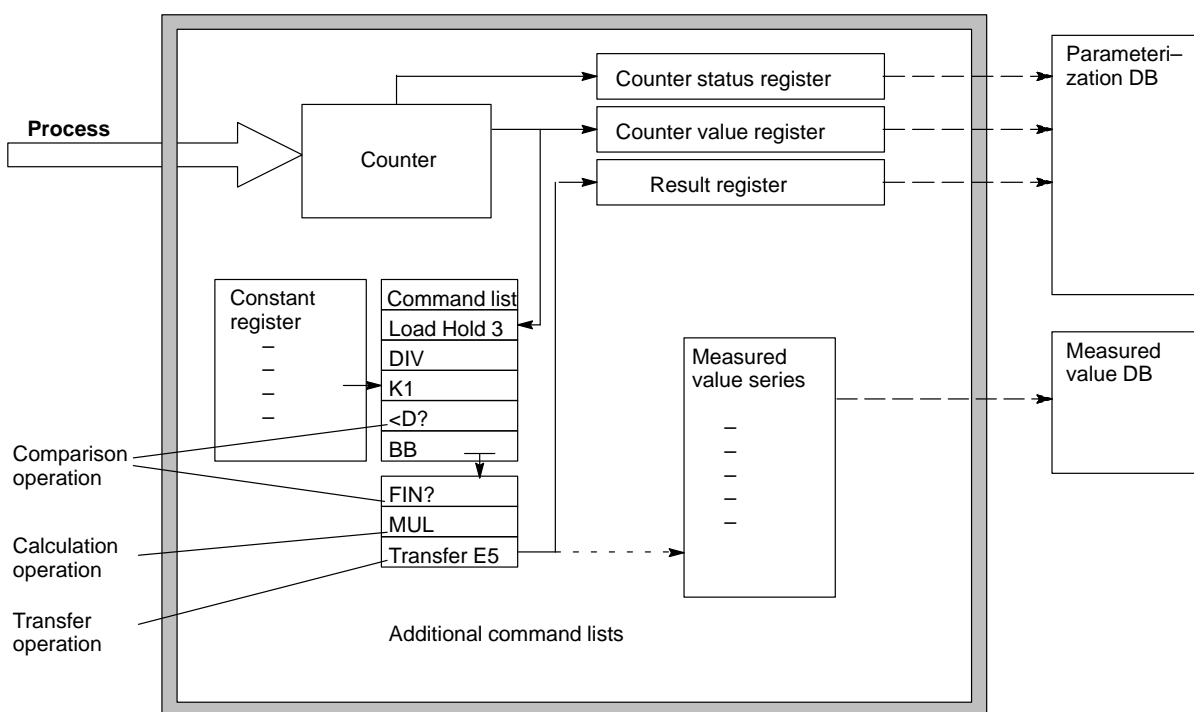
Standard function blocks FB 178 to FB 182 for the IP 242A cannot be used with the IP 242B. Similarly, function blocks FB 183 and FB 184 for the IP 242B cannot be used with the IP 242A.

| Module | Corresponding FB |
|--------|------------------|
| IP 242A | FB 178 to FB 182 |
| IP 242B | FB 183 and FB 184 |

The IP module does not respond to function blocks other than those assigned to it as stated above.

## Upgrading an IP 242A to an IP 242B

Since the IP 242A and IP 242B modules are based on the same hardware, the firmware of the IP 242B can also be run on the IP 242A.

This requires that the firmware EPROMs on your IP 242A be replaced. In addition, FB 183 and FB 184 must also be integrated in the S5 since the firmware of one module will not accept the function blocks of the other.

To make the change easier, the structure of the parameterization data block has been retained, and the new registers make use of the previously unused positions.

An upgrading kit is available from WKF under order number 764 22 716 for "upgraders".

## Notes for First–Time Users

The IP 242A and IP 242B counter modules are used for the acquisition and conditioning of high–speed counting pulses.

They contain seven independent counters. Counters 1 to 5 can be switched on and off via external addressing of counting gate inputs or software starts/stops.

The initial values must be specified for the counters in the S5 program. The point in time at which these values are loaded in the counters can also made dependent directly on the external signals.

Each counter is equipped with an output with which it can report events such as, for example, overflow or end of counting. This message can be reported to the programmable controller both via an interrupt and externally via a hardware output.

This allows the counter module to be used as a link between high–speed events in the process, fast reactions and the program in the programmable controller.

However, not a single pulse is counted when counting and gate signals are applied to the front plug connector.

> Before the counter module can report events, it must be provided with the following settings (after the counting task has been defined).
>
> – Hardware settings (DIL switches and jumpers)
> – Software settings (parameterization and control)

1. Definition of the counting task

2. ...

3. ...
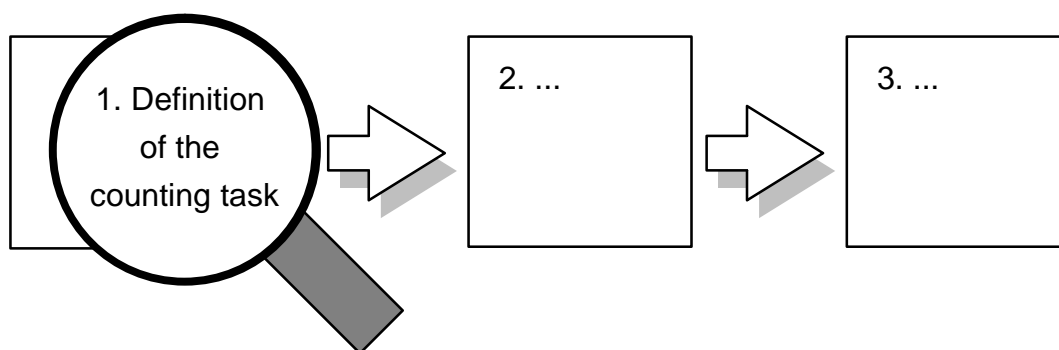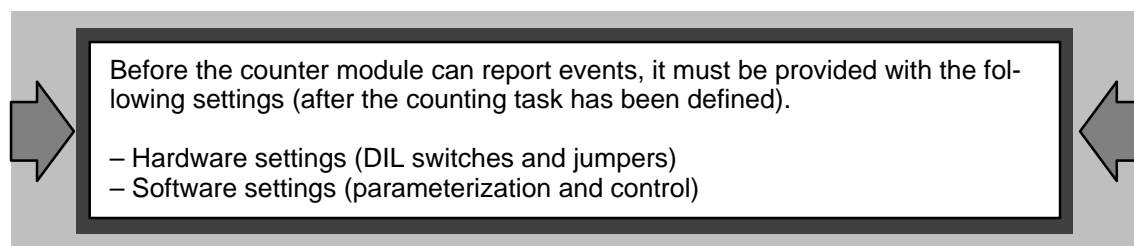
Why is such a conditioning module needed when counters programmable in STEP5 are available for all programmable controllers ?

A simple consideration makes the limits of these software counters clear. For example, counting 50 Hz pulses requires that the programmable controller process the programming for the counter in time intervals of less than 10 msec. (Twice the processing speed is required since the software counter also has to acquire the falling edge of the counting signal in order to be able to recognize the rising edge of the same signal.) This means that the programming for counting (with direct access to the periphery) must be called several times.

Additional Reasons for Using the IP 242A/242B

– Separate gate inputs are available for each of counters 1 to 5.
– Edge or level–dependent counting procedures can be easily controlled on the counting and gate inputs.
– A choice of internal or external counting procedures is available. In addition, these can be combined.
– Counters 1 to 5 can be cascaded.
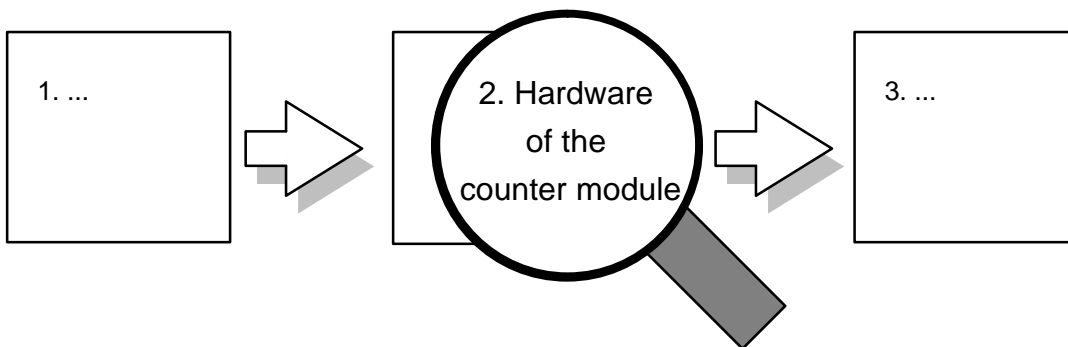


The counter module is equipped with seven counters. Each counter can be addressed externally and counters 1 to 5 can also be addressed internally.

External Circuiting

– See section 2 for circuiting the inputs (from connection via signal conditioning to circuiting of the outputs).
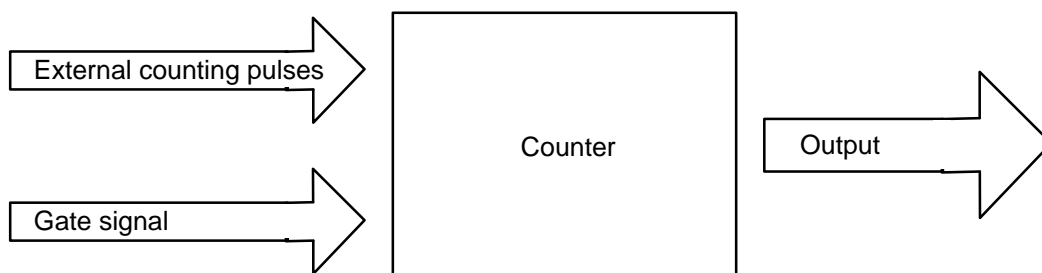
Use of the Internal Signals (For Counters 1 to 5 Only)

– Outputs of scaler 1 which scales the 1 MHz clock pulse of the quartz
– Output of scaler 2 (Possible inputs for scaler 2 are the outputs of scaler 1 or the external counting/gate inputs.)
– Output of counter n–1 (counter cascading see * below and section 7.6)
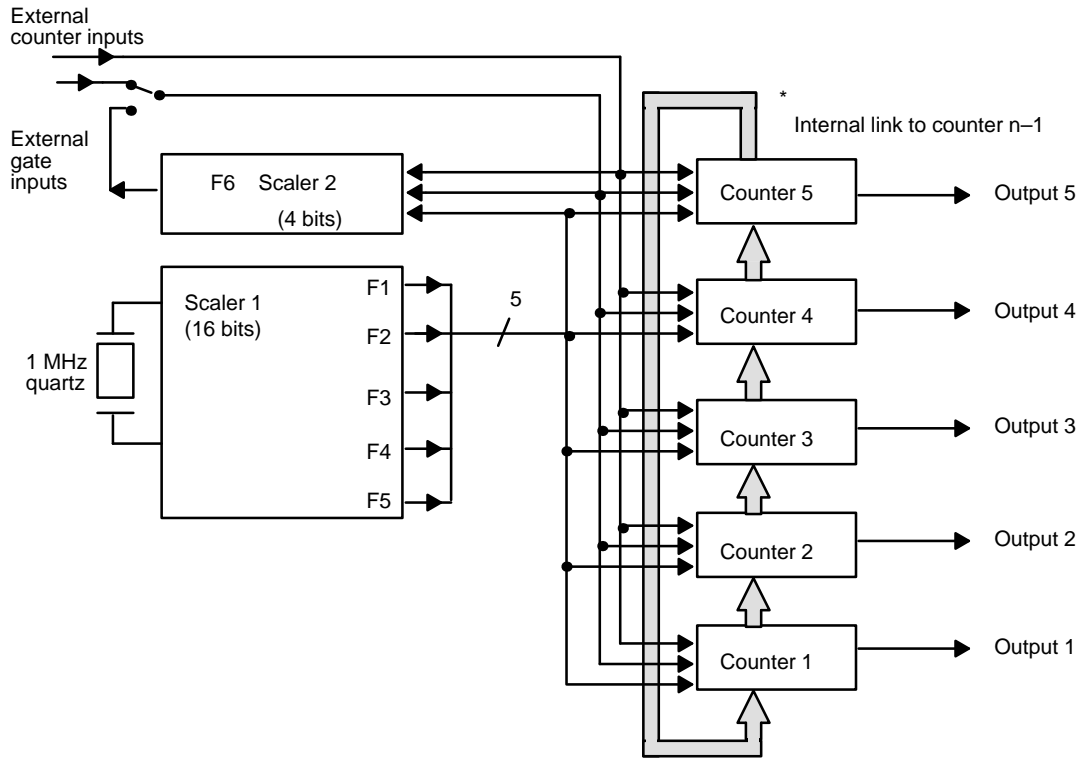


Use of the Internal Signals (For Counters 6 and 7 Only)
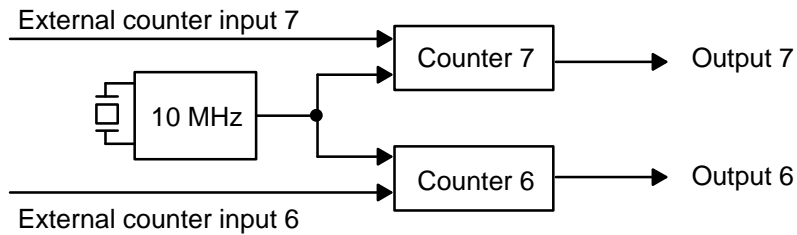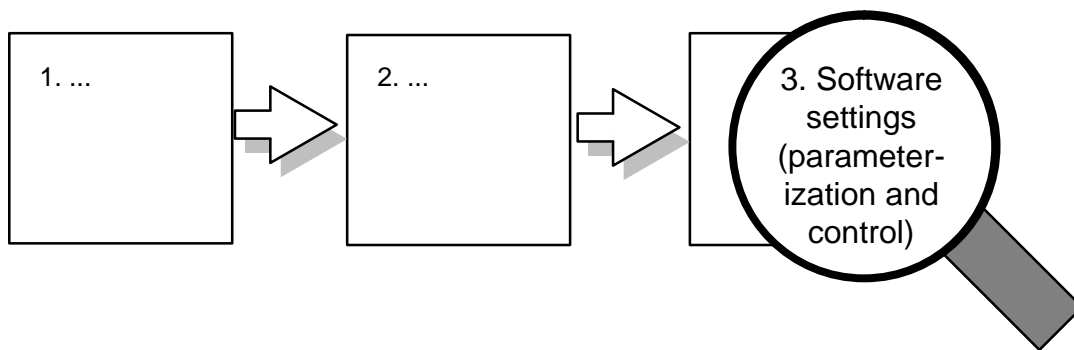
Each communication between the S5 and the IP 242A/242B is handled by a transfer memory (dual port RAM).
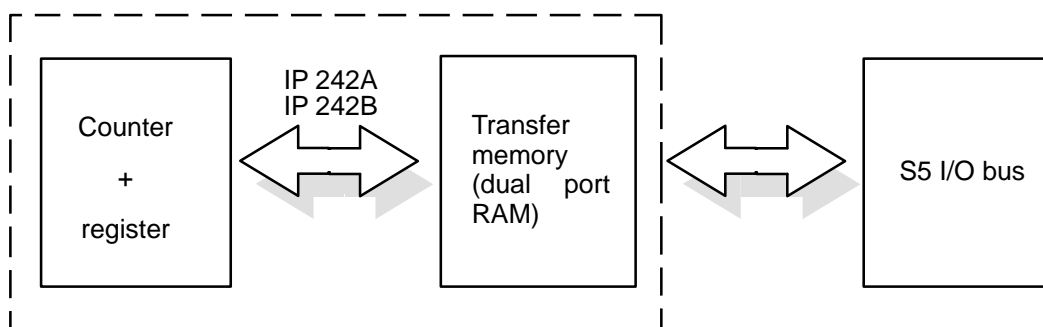


This data communication is handled as follows.

– Parameters are transferred from the CPU to the IP.

– Data are fetched by the CPU from the IP to the PLC.

The standard function blocks handle the organization of all accesses.

**Software Support**

In addition to the required hardware settings, each individual counter is supplied with the parameterization of the registers.

Support of the Counting Task
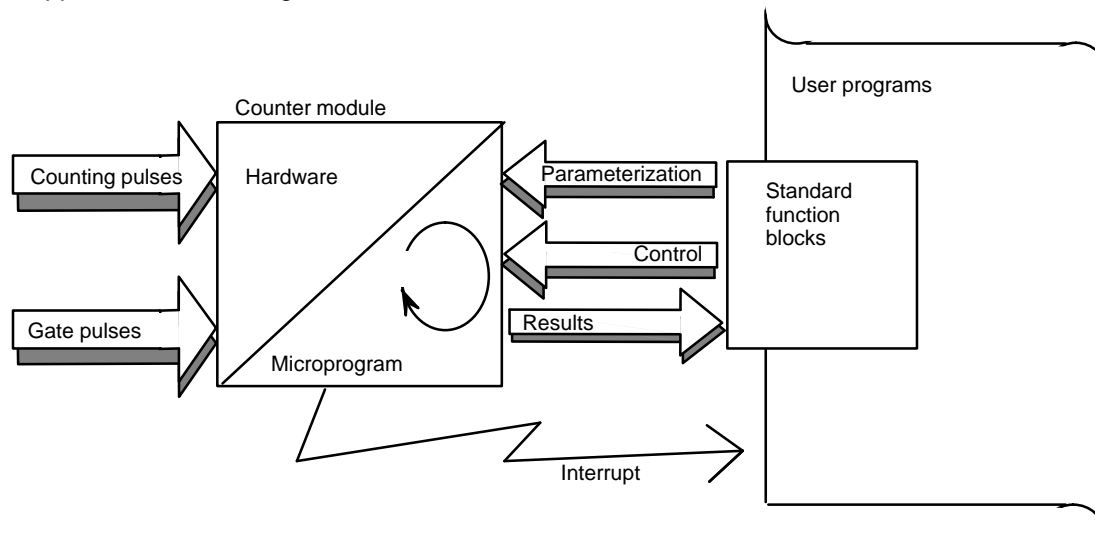


Because of the many conditions which must be adhered to (i.e., order, time requirements, etc.), writing your own program for the interaction between user program and counter module may cause problems.

Because of this, standard function blocks are provided for the user program–counter module interface.

Available Standard Function Blocks

IP 242A:     FB 178 to FB 182
IP 242B:     FB 183 and FB 184

A significant advantage of using the function blocks is the improved readability of the program. Complex relationships between user program and counter module are reduced to the defined environment of the function blocks and their calls.

The following must be specified by the user, however.

– Points in time of the call
– Conditions under which the call is made

Support of the Communication with the Programmable Controller
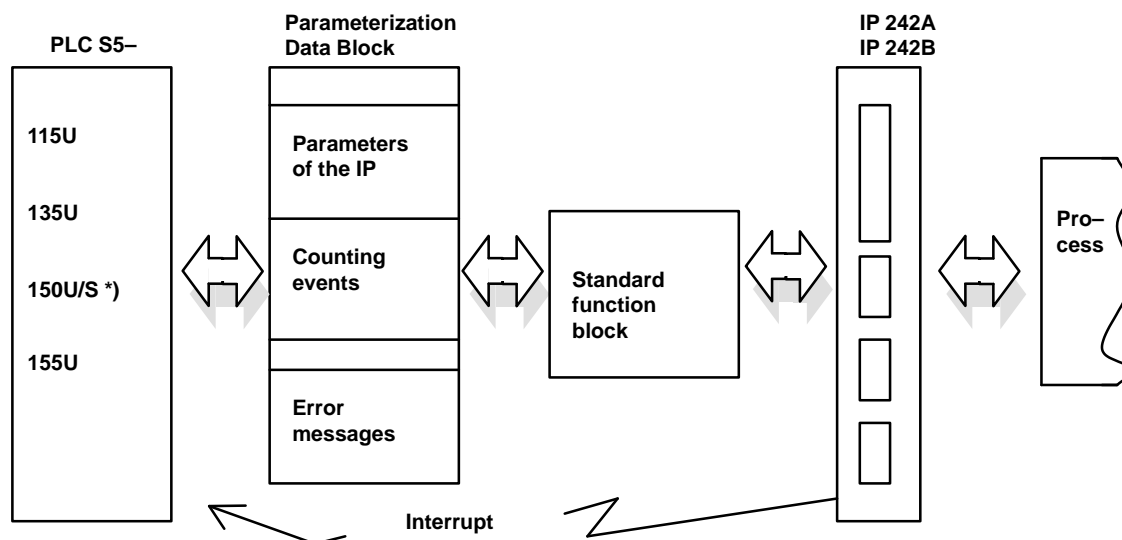


\*) With IP 242A only

The IP 242A/242B uses standard function blocks to communicate with the programmable controller (PLC). In addition, interrupt lines (interrupt) can be used to react immediately to an event in the process (interrupt processing). The function blocks access the parameterization data block (i.e., the IP 242A/242B can be parameterized, controlled and monitored with these data blocks and the standard function blocks).

The parameterization data block must be set up in the PLC by the user, and provided with parameters.

> Sections 11, 13 and 14 contain an explanation of the method of function and the use of the standard function blocks. A program example included on the re-spective floppy disk provides you with an example of simple commissioning and step–by–step familiarization.
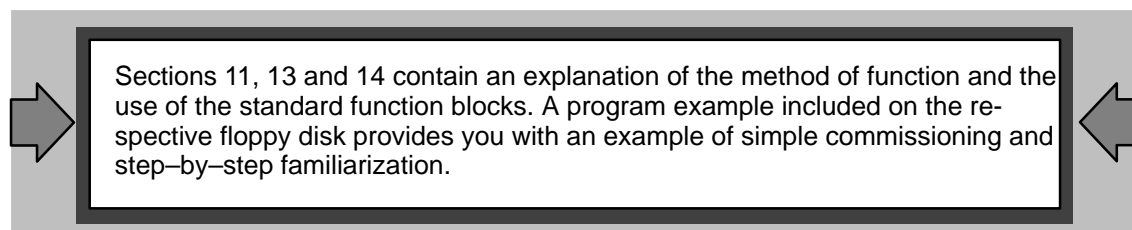
## Stipulations

**Emphasis of Safety Notes**

Notes are identified as follows in this manual.

> **!** Texts in these boxes contain important information or instructions which must absolutely be adhered to to ensure safe functioning and protection of the module. **!**

> ⇨ Texts in these boxes contain information and notes which require particular attention. ⇦
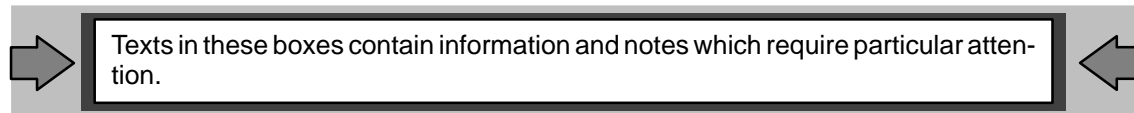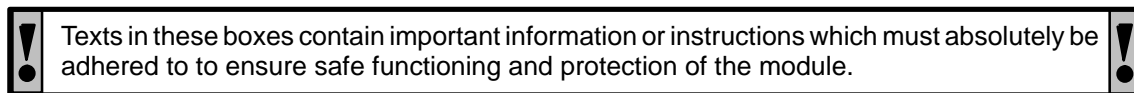
**Abbreviations**

Abbreviations which are not part of everyday usage are written out in full the first time they appear. See the glossary for a list of the abbreviations used.

**Cross References**

Cross references are not made to parts of other sections unless repetition of the information would require too much space, and it can be assumed that the description at another location is sufficient. Cross references to parts of other sections are made by specifying the section number (e.g., "→ section 2.1").

**Symbols Used**

The symbols **A** and **B** in the margin of the text indicate sections of the text which only apply to the

corresponding module.

– **A** for IP 242A

– **B** for IP 242B

In addition, the title of each section includes a note to the right of the title indicating which module this section applies to (if differences exist). See examples below.

| 3.3.2 | **Prescaler Register (VTR)** |
|---|---|
| | **IP 242A/242B** |

| 3.3.6 | **Interrupt Filter Register (AFR)** |
|---|---|
| | **IP 242B** |
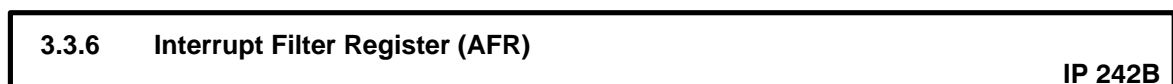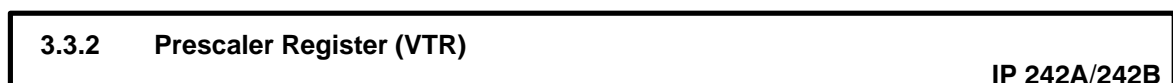
All unmarked passages apply without restrictions to both the IP 242A and the IP 242B.

---

# 1        General Function Description

---

---

## 1.1     Features of Counter Module IP 242A/242B

---

■ Five 16–bit counters for general purpose operation with counter input/counter output and gate control (counting up or counting down)

   • Counter inputs and gate inputs adjustable to 5 V or 24 V logic and adaptable to signal frequencies up to 480 kHz

   • Counter cascading up to 80 bits, decimal counting up to $10^{20}$

   • Nineteen different operation modes for each counter

      – Frequency acquisition with internal or external gate control
      – Pulse counting
      – Pulse counting with comparator function
      – Time measurement
      – Frequency scaling with programmable scaling factors
      – Time delaying

   • Variable gate control (e.g., start–stop via light barriers)

■ Two counters for acquisition of incremental encoder signals (counting up and down)

   • Conting and gate inputs for 5 V signals (RS422) with maximum signal frequencies of 500 kHz

   • Counting width: 24 bits for IP 242A
                     32 bits for IP 242B

■ Control via 16–bit microprocessor 80C186

■ Internal frequency generator with 1 mHz and various scalers

■ Data exchange with the S5 via dual port RAM (page frame addressing)

■ Free parameterization of special functions by the user via command lists

---

▨ Optimal process monitoring via integrated interrupt processing

▨ Easy handling and system interface via standard function blocks

▨ Parameterization data can be stored on EEPROM in the IP 242A/242B.

▨ IP 242B: Conditioning and storing of measured values

▨ IP 242B: Past values can be followed by entering the results in a measured value sequence (trace function).

▨ IP 242B: Implementation of calculation functions to expand functionality

**Applications**

The IP 242A/B module is designed for use with SIMATIC S5 systems and is used in the following programmable controllers:

- S5–115U      (CPU 941A/B to CPU 944A/B)
- S5–135U      (CPU 922 firmware version 9.0 and later/CPU 928A firmware version ... –3UA12 and later/CPU 928B)
- S5–150U/S   (IP 242A only)
- S5–155U      (CPU 946 and CPU 947)

The module operates with page frame addressing when used with any of the above programmable controllers.
(When the IP 242A module is used with programmable controllers S5–115U and S5–155U, linear addressing ($2^{10}$–byte address volumes) is also possible.

The modules are available under the following order numbers:

IP 242A: 6ES5 242–1AA32
IP 242B: 6ES5 242–1AA41

Counters 1 to 5, for which there are 19 operation modes (see section 9), are used either for counting up **or** counting down. Counters 6 and 7 are used to count up **and** down by processing the 90˚–displaced input pulses.

Parameterization is performed via the S5 bus.

## 1.2      Hardware Description

This overview shows the function blocks of the IP 242A/242B counter module.
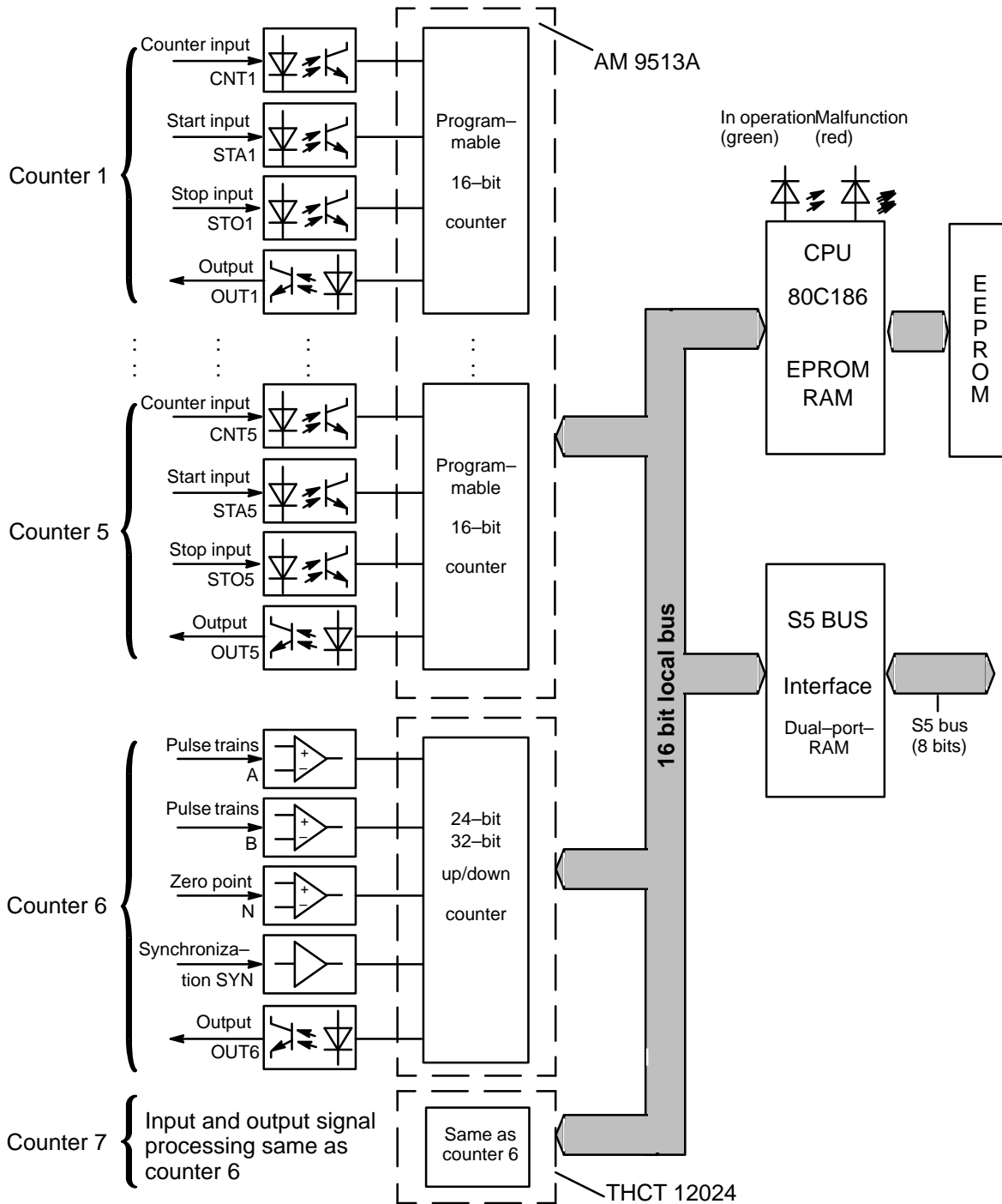
## 1.3    16–Bit Counter

**Block Circuit Diagram of Counter 1**
**(Counters 2 to 5 are analogous.)**

**Output, counter 1**                          **Inputs, counter 1**

Stop       Start      Counter input
STO1       STA1       CNT1

0 V$_{external}$   OUT1   24 V$_{external}$

Front panel

Gate control logic

Gate control register

1 MHz

F6

Internal
scaler

Gate 1           Counter input 1

Counter 1

Pre–
scaler

Master
mode
register

Output 1

Counter
mode
register

AM 9513A

16–bit local bus

Interrupt         Interrupt      16–bit local bus
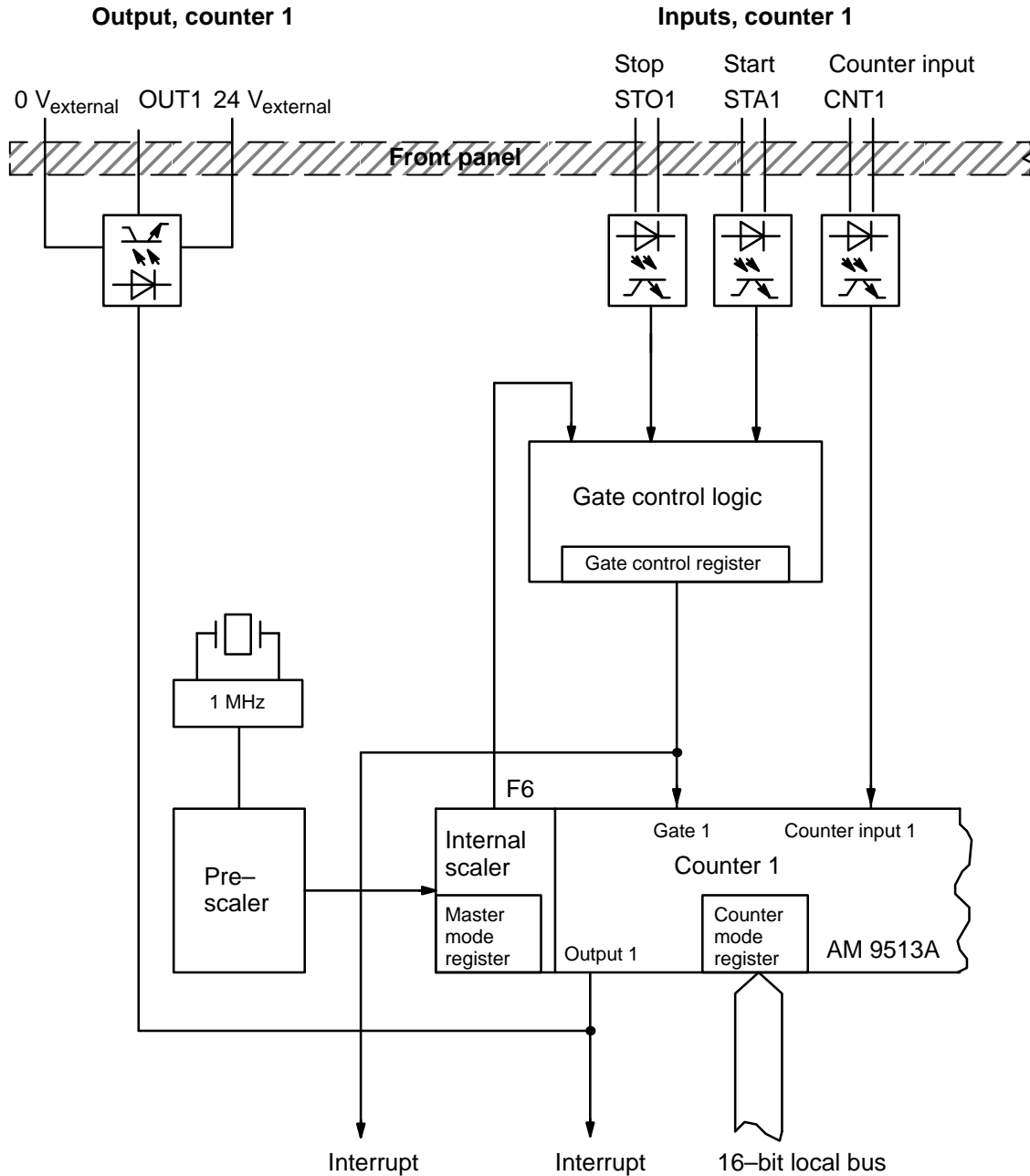
<table>
<tr><td>

## 1.4    Inputs of Counters 1 to 5

</td></tr>
</table>

Five inputs, which are isolated by optocouplers, are available.

Each counter has the following inputs:

- CNT  (counter input)
- STA  (start input)
- STO  (stop input)

The inputs are protected against polarity reversal.

Condition each input to your signal level by using a plug in jumper as shown below:



Counter clock     Open collector     Conditioning to 5 V level
pulse final stage



Counter clock     Open collector     Conditioning to 24 V level
pulse final stage

See section 2.4 for the positions of the plug–in jumpers on the module, their designation, and allocation.

## 1.5 24/32–Bit Up/Down Counter

Counter pulses from incremental encoders with 90˚–displaced pulse trains with RS 422 level are acquired by counters 6 and 7.

Counting width:       IP 242A      24 bits
                      IP 242B      32 bits

**Block Diagram of Counter 6
(Counter 7 is analogous.)**

Output, counter 6                                              Inputs, counter 6

0 V    OUT    24 V                    SYN   $\overline{N}$ N      $\overline{B}$ B    $\overline{A}$ A

Front panel

$V_{reference}$

&

Enable logic
for
counter reset

Counter
mode
register

Control
logic

Counter
reset

24/32–bit
up/down counter
THCT12024

Output 6

Intermedi-
ate stor-
age

8–bit local bus        Interrupt                    8–bit local bus

The inputs cannot be circuited with 24 V signals unless the "24 V asymmetric to 5 V (RS422) symmetric converter" is used ($\rightarrow$ section. 10.11).

## 1.6    Inputs of Counters 6 and 7

The inputs are designed for incremental encoders with two 90° displaced symmetrical output signals, zero pulse, and mechanical zero reset (synchronous pulse).

Counter channels 6 and 7 have the following inputs:

- A      Differential input
- B      Differential input
- N      Differential input, zero marking pulse
- SYN    Input for synchronous contact (preliminary contact)

An incremental encoder with RS 422 interface (5 V level) supplies input signals A, $\overline{A}$, B, $\overline{B}$, and N, $\overline{N}$. The suppressor circuit limits the input voltages to approximately 13 V (against 0 V). A preliminary contact (24 V level) provides synchronous signal SYN.

Inputs A, B, and N are designed as differential inputs as shown below:
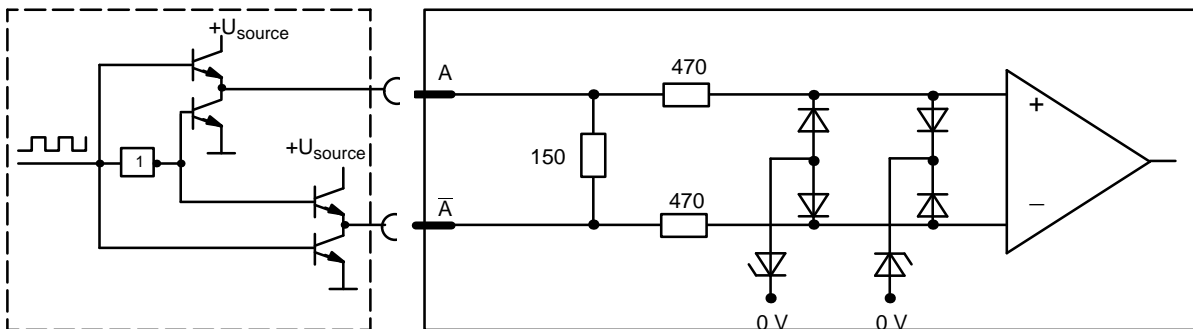


Counter clock pulse final stage
(RS422 driver)

Circuitry of the SYN Input:
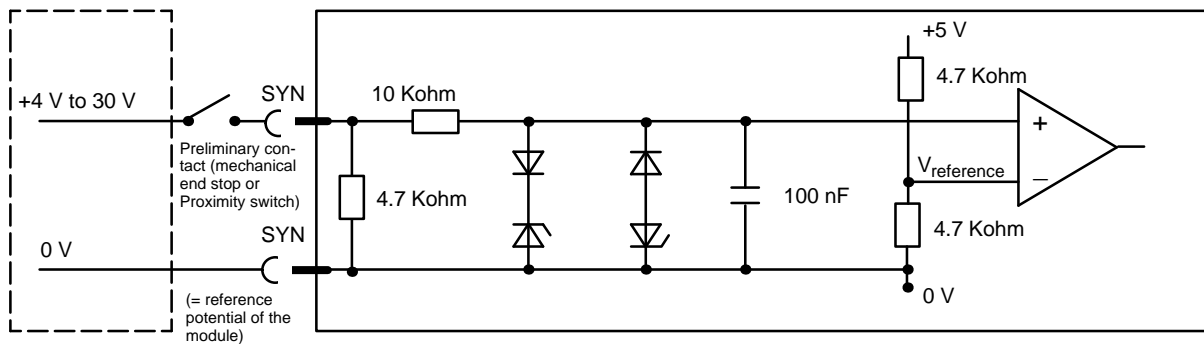
Pulse Diagram for Counters 6 and 7:
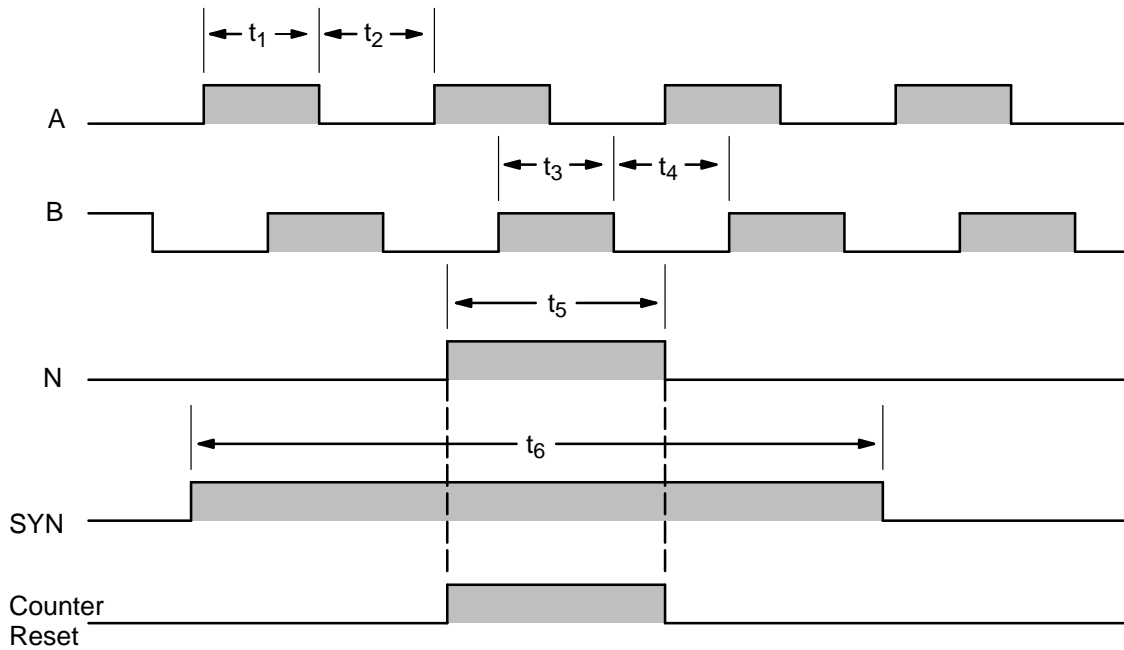


$t_1, t_2, t_3, t_4, t_5 \qquad \geq \quad 1 \text{ msec}$
$t_6 \qquad\qquad\qquad \geq \quad 3 \text{ msec}$

The applicable counter is reset when the zero marking pulse of the encoder coincides with the synchronous pulse. This function can be enabled separately for each counter channel in the counter mode register of counter channels 6 and 7. It is possible to trigger a group interrupt on the S5 central processing unit with the counter reset pulse ($\rightarrow$ section 3.3.4) and/or activate a command list.

Synchronization occurs at the falling edge of the zero mark pulse. The counter is reset to zero. Section 7.6 contains an example of synchronization with zero point shift.

## 1.7 Gate Control Logic

Four different types of gate control for counters 1 to 5 are available.

| Gate Mode | Gate Control Register Bit 9 (to 13) | Bit 1 (to 5) | Signal Generation | Type of Control | Signals Used | Explanation (See next page for examples) |
|---|---|---|---|---|---|---|
| 1 | 0 | 1 | Internal | Level | F6 | Generated by scaling from the internal frequency generator (1 mHz) (Clock pulse rate is parameterizable.) |
| 2 | 0 | 0 | External | Level or edge[1] | **only** STA | Signal **only** at the start input of the applicable counter |
| 3 | 1 | 0 | External | ⌐ | STA/STO | Signals at the start **and** stop input of the applicable counter |
| 4 | 1 | 1 | External | ⌐ | STA/STO | Signals at the start **and** stop input of the applicable counter |

1) The settings in CMR apply when edge control is used.

Enter the bit pattern, given in the second and third column, in the gate control register to select a certain gate mode. In the gate control register, two bits are allocated to each counter.

The below schematic drawing shows the main gate switchover logic.
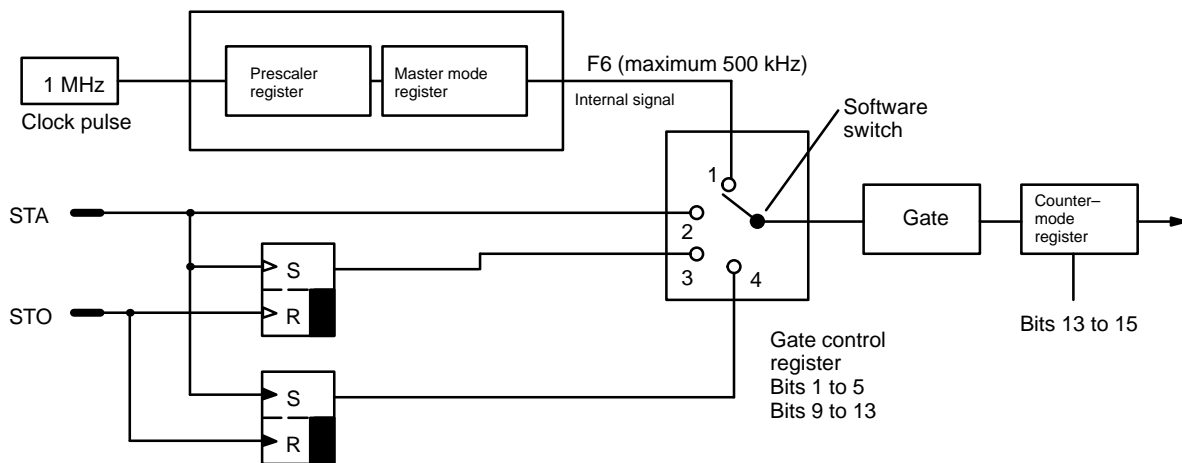


The prescaler, master mode, counter mode, and gate control registers must be allocated to select the gate control.

**Gate Mode 1:**

Frequency F6 is applied directly to the gate.



$t_1 \geq 1\ \mu sec$
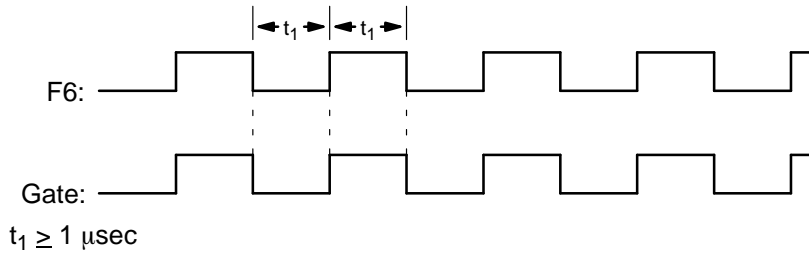
**Gate Mode 2:**

The level/edge at the start input (STA) is applied to the gate.



$t_1, t_2 \geq 1.04\ \mu sec$
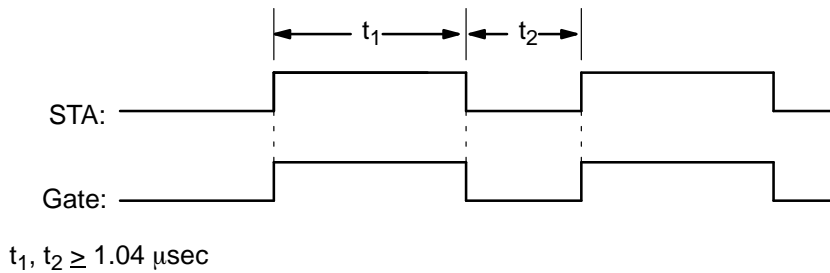
The appropriate settings in the counter mode register are required for edge–controlled operating modes.

**Gate Mode 3:**

The gate is opened with a rising edge at the start input and closed again with the next rising edge at the stop input.



$t_1, t_2, t_3, t_4 \geq 1.04\ \mu sec$

$t_1, t_3$ = Pulse width high from STA or STO
$t_2, t_4$ = Pulse width low from STA or STO

The start and stop signals must not occur simultaneously. There must be an interval of at least 1.04 $\mu sec$ between both rising edges.

**Gate Mode 4:**

The gate is opened with a falling edge at the start input and closed again with the next falling edge at the stop input.

STA:

STO:

Gate:

$t_1, t_2, t_3, t_4 \geq 1.04$ μsec

$t_1, t_3$ = Pulse width low from STA or STO
$t_2, t_4$ = Pulse width high from STA or STO

Start and stop signals must not occur simultaneously. There must be an interval of at least 1.04 μsec between both falling edges.

## 1.8 Outputs of All Counters

Connect an external supply voltage of 24 V for the output signals. This applies to counters 1 to 7 jointly. The outputs are metallically separated from the internal supply voltage by optocouplers.

All counters have the same output circuitry.
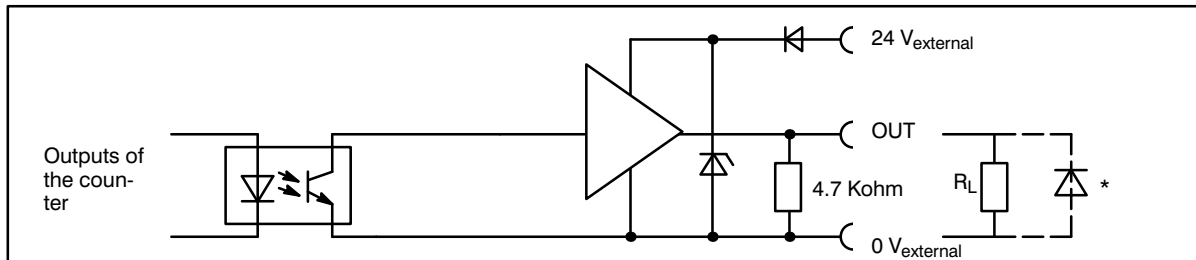


> **!**  *Use a free–wheel diode to protect outputs with inductive loads.  **!**

The active state of the output signal can be parameterized as follows:

- High pulse (24–V pulse)
- Low pulse (0–V pulse)
- Toggle function
  (alternating between 0–V and 24–V pulses with defined initial status)
- High or low pulse with parameterizable length (IP 242B only)

The desired signal form is selected in the counter mode register of the individual counters ($\rightarrow$ section 3.4.1).

An output signal is generated as soon as the counter reaches the value of the corresponding interrupt register ($\rightarrow$ section 3.4.4). In addition, the comparator must be enabled in the master mode register for counters 1 to 5 ($\rightarrow$ section 3.3.1).

When the IP 242B is used, the length of the pulse can be parameterized in the pulse duration register ($\rightarrow$ section 3.3.10).

The outputs must be specifically enabled with the FA command (enable outputs); if not, the output of the signals is suppressed.

The following table shows when the individual outputs activate.

**Output Function**

| Counter Setting | Counters 1 and 2 | Counters 3 to 5 | Counters 6 and 7 | | |
|---|---|---|---|---|---|
| Counting cycle zero<br>Counter terminal count | High pulse or<br>Low pulse<br>Toggle function | High pulse or<br>Low pulse<br>Toggle function | Illegal | **A** | **B** |
| Comparator function<br>Contents of counter =<br>comparison value | High pulse or<br>Low pulse | Not available | Not available | **A** | |
| | | High or low pulse<br>with parameteri–<br>zable length[2]) | High or low pulse<br>with parameteri–<br>zable length[2]) | | **B** |
| Comparator function<br>Contents of counter $\geq$<br>comparison value | Not available | High level[1]) | High level[1]) | **A** | |
| | | High level or low<br>level[1]) | High level or low<br>level[1]) | | **B** |

1) See section 1.9.

2) The parameterizable pulse length can be used, for example, to set the minimum pulse duration for switching a contactor. Previously, a separate counter had to be used.

**Overload Protection of the Outputs:**

In the event of thermal overload, all outputs are disabled. A thermal overload occurs when the total current of all outputs exceeds 700 mA.

In the event of a short circuit, only the affected output is disabled and switched to low level.

In both cases, an error message is entered in the error information register, and the error LED goes on.

> After correction of the short circuit, enable the disabled output again with the FA control word, (enable outputs) from the S5 central processing unit.
> Remember, however, that all switched on outputs are also briefly disabled by the control word (i.e., $\leq$ 30 $\mu$sec).
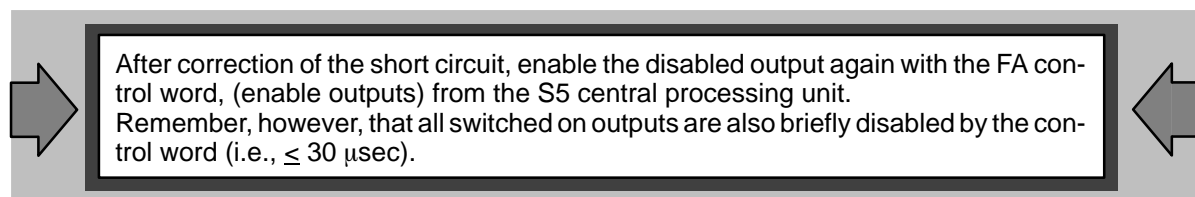
## 1.9 Comparator Function

When the comparator function is enabled, the current counter status is compared to the value stored in the interrupt register.

For counters 1 to 5, the comparator in the master mode register is enabled; for counters 6 and 7, the comparator in the respective counter mode register is enabled.

## 1.9.1 Counters 1 and 2 (Only for Actual Value = Interrupt Value)

Comparators 1 and 2 are hardware comparators. When the counter values equal the interrupt values, an output pulse is generated **for the period of time in which the values coincide.**

| Parameterization in the Counter Mode Register | Output Signal |
|---|---|
| bit 2 / bit 1 / bit 0 = 001<br>or = 010<br>or = 011 | ⎍ High pulse |
| bit 2 / bit 1 / bit 0 = 101 | ⎍ Low pulse |

An interrupt value of "0" is **illegal**.

If an interrupt value of "0" is entered in the interrupt register, a message is generated in the error information register when the counter is parameterized. (See section 6.1.22.)

**A** An interrupt value of "0" is also **illegal** if the comparator function in the master mode register has **not** been enabled.

When the comparator function is enabled, the outputs can only be activated by the comparator. For this reason, the outputs cannot be activated at counting cycle zero and at terminal count.

## 1.9.2   Counters 3 to 5          (for Actual Value $\geq$ Interrupt Value)

Comparators 3 to 5 are software comparators.

The contents of the counter are read out cyclically (approximately 10 msec) and compared with the interrupt value.

An interrupt value of "0" is **illegal**.

| A | B |

If an interrupt value of "0" is entered in the interrupt register, a message is generated in the error information register when the counter is parameterized. (See section 6.1.22.)

An interrupt value of "0" is also illegal if the comparator function in the master mode register has not been enabled.

| A |

> When the comparator function is enabled, only the comparator can activate the outputs. For this reason, the outputs cannot be activated at counting cycle zero and at terminal count. Disregard the assignment of bits 0 to 2 in the respective counter mode register.

> ⚠ The software comparator <u>cannot</u> be used with operation modes G, H, I, J, K, L, S, and V. ⚠

Output Signals for Counters 3 to 5

– The interrupt value must be at least 10 msec away from overflow/underflow.

– The output signals can be inverted for the IP 242B.

– A pulse with parameterizable length is available on the IP242B regardless of the direction of counting at the interrupt value.

L/H:     Counting value is reloaded from the load or hold register.
IV:      Interrupt value
TC:      Terminal count (overflow or underflow)

## 1.9.3    Counters 6 and 7

The contents of counters 6 and 7 are cyclically compared with the comparator values.

This comparison is always performed when control words, interrupt requests, and command lists are not being processed. The minimum scanning frequency is determined by the longest break in the processing of a control word, a string of control words (command list), or an interrupt triggered on the S5 central processor. See section 10.8 for processing times of the control words.

**Positive Interrupt Value:**
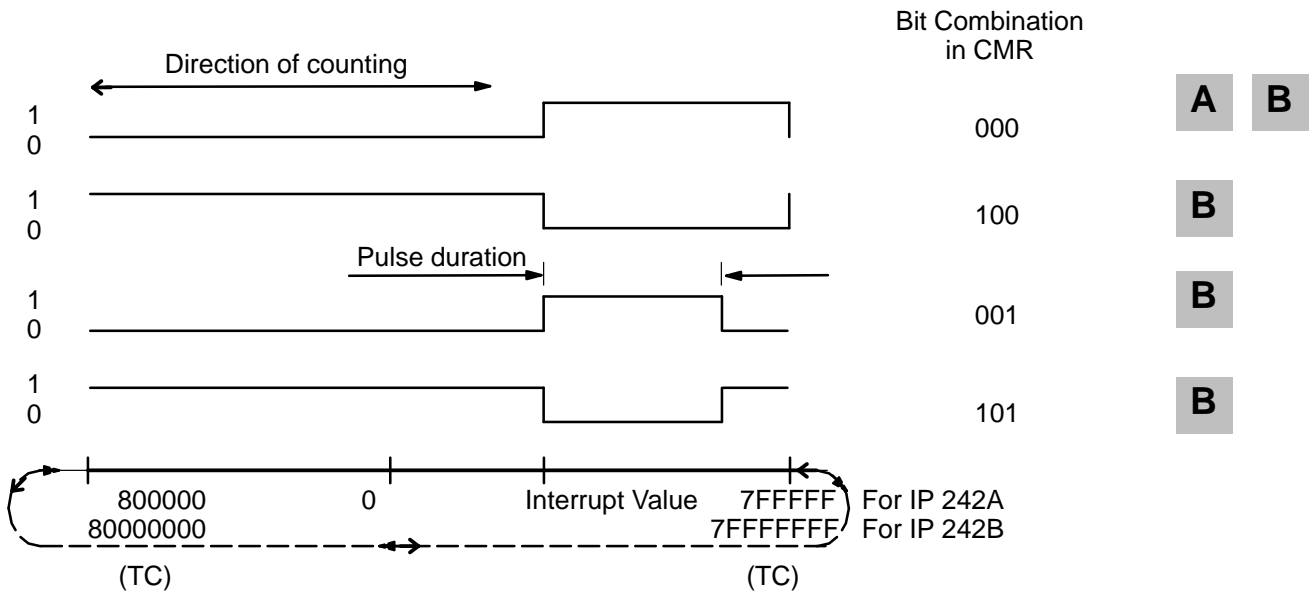
Bit combination
000        The output is set to "high" outside the interrupt value.
           Actual value $\geq$ interrupt value => output "high".
           The output remains on "low" up to the interrupt value
           in the range        $800000_H$ ($= -8\,388\,608$) for IP 242A
                                $80000000_H$ ($= -2\,147\,483\,648$) for IP 242B

           A **change in output signal** occurs at overflow/underflow, but no interrupt is
           generated (no command list can be activated).

100        The output is set to "high" below the interrupt value.
           Actual value < interrupt value => output "high".
           The output remains on "low" in the range from interrupt value to $7FFFFFFF_H$
           ($= 2\,147\,483\,647$).

           A **change in output signal** occurs at overflow/underflow, but no interrupt is
           generated (no command list can be activated).

001     The output gives a high pulse with parameterizable length at the interrupt value. Actual value = interrupt value => Output remains "high" for the parameterized pulse duration. The output remains on "low" for the remainder of the range.

        **No change in output signal** occurs at overflow/underflow and no interrupt is generated (no command list can be activated).

101     The output gives a low pulse with parameterizable length at the interrupt value. Actual value = interrupt value => Output remains "low" for the parameterized pulse duration. The output remains on "high" for the remainder of range.

        **No change in output signal** occurs at overflow/underflow and no interrupt is generated (no command list can be activated).

**Positive Interrupt Value (for Gate Time and Frequency Measurement):**



TC: Terminal count (overflow or underflow)
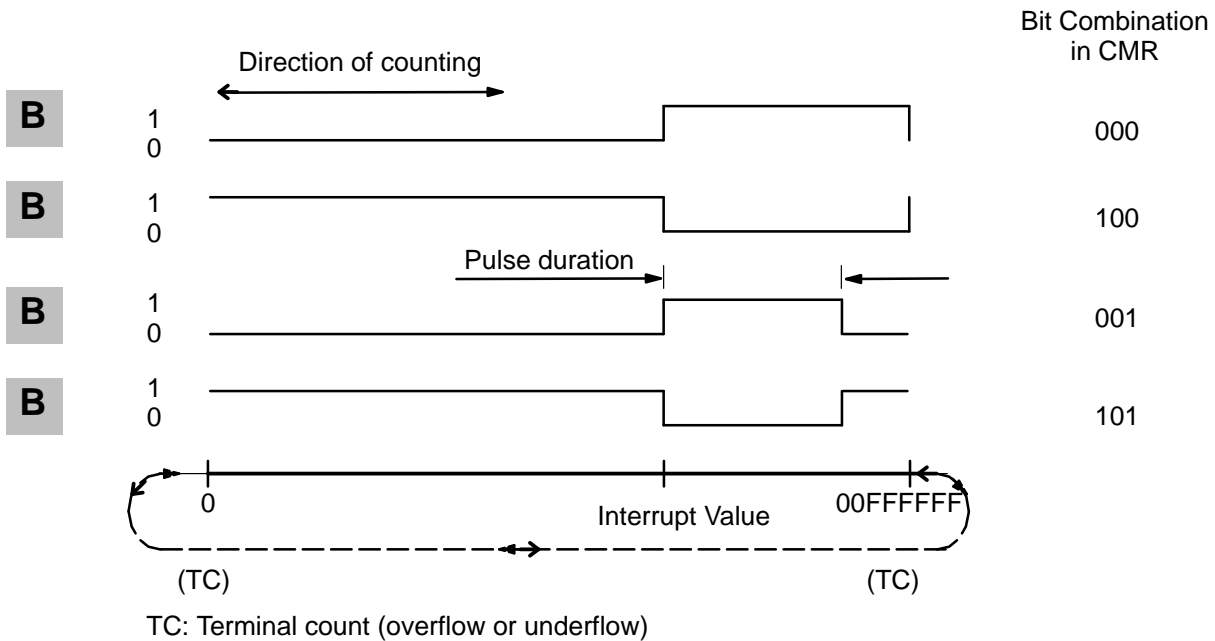
Only positive interrupt values are permitted for gate time and frequency measurements. The number range is limited from 0 to FFFFFF. The outputs provide the signals as for positive interrupt value (see previous page).

**Negative Interrupt Value:**
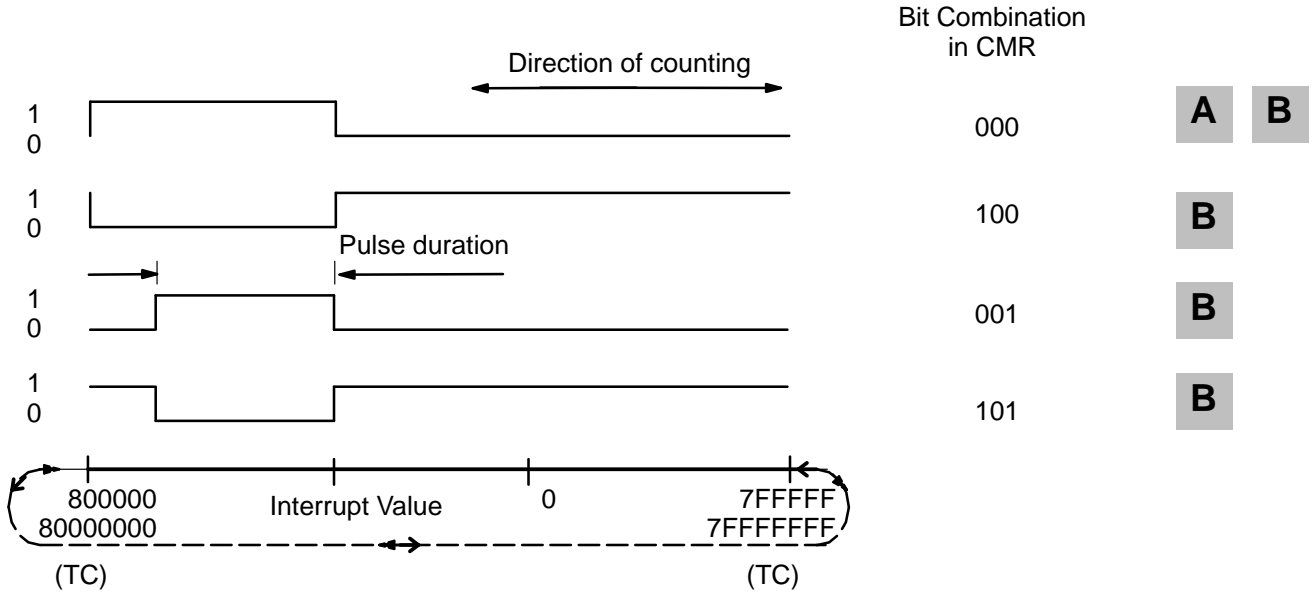


TC: Terminal count (overflow or underflow)

Bit combination

000    The output is set to "high" below the interrupt value.
Actual value $\leq$ interrupt value => output "high".
The output remains on "low" up to the interrupt value
in the range    $7FFFFF_H$ (= –8 388 607) for IP 242A
$7FFFFFFF_H$ (= –2 147 483 647) for IP 242B

A **change in output signal** occurs at overflow/underflow, but no interrupt is
generated (no command list can be activated).

100    The output is set to "high" above the interrupt value.
Actual value > interrupt value => output "high".
The output remains on "low" in the range from interrupt value to  $80000000_H$
(= 2 147 483 648).

A **change in output signal** occurs at overflow/underflow, but no interrupt is
generated (no command list can be activated).

001    The output gives a high pulse with parameterizable length at the interrupt value.
Actual value = interrupt value => Output remains "high" for the parameterized pulse
duration. The output remains on "low" for the remainder of the range.

**No change in output signal** occurs at overflow/underflow, and no interrupt is
generated (no command list can be activated).

101    The output gives a low pulse with parameterizable length at the interrupt value.
Actual value= interrupt value => Output remains "low" for the parameterized pulse
duration. The output remains on "high" for the remainder of the range.

**No change in output signal** occurs at overflow/underflow, and no interrupt is
generated (no command list can be activated).

## 1.10    Reference Frequency
## 1.10.1  Internal Clock Pulses for Counters 1 to 5

Clock pulses F1 to F6 are generated with various frequencies for time or frequency measuring.
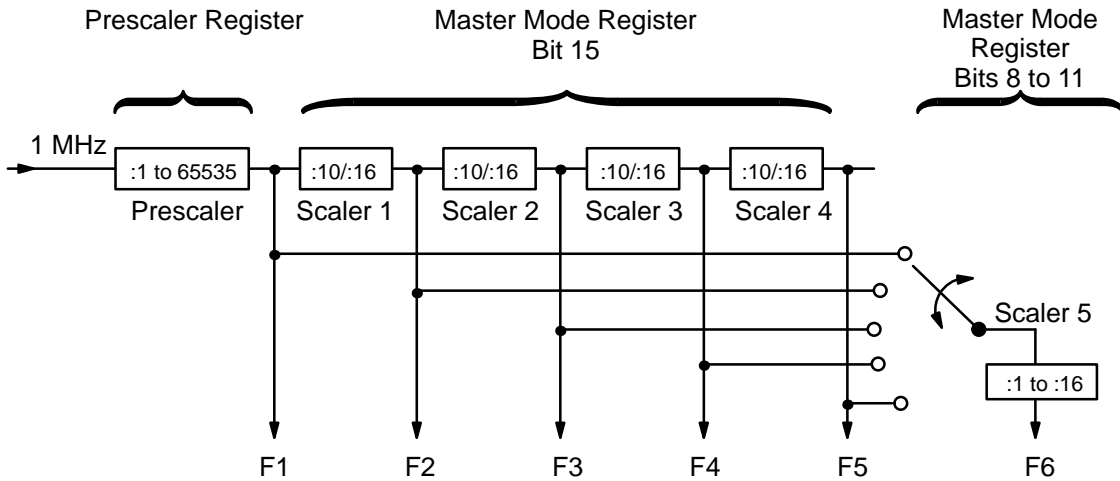


The basic clock pulse has a frequency of 1 mHz. This frequency is fed to a prescaler. Output signal F1 of the prescaler serves as the input clock pulse for scaler 1.

Further frequency scaling is then determined in the master mode register. The clock pulse is divided by four scalers. These scalers have a scaling factor of 10 (BCD scaling) or a scaling factor of 16 (binary scaling). The common scaling factor for scalers 1 to 4 is determined in bit 15 of the master mode register.

The output signals of scalers 1 to 4 are designated as F2, F3, F4, and F5. Together with frequency F1, these signals can be used as input or gate signals for counters 1 to 5.

Frequency F6 is derived from either frequency F1, F2, F3, F4, or F5. The determination of the frequency to be used as the basic clock pulse for frequency F6 takes place in bits 4 to 7 of the master mode register. This frequency is then conducted via scaler 5.

The scaling factor of scaler 5 can be varied from 1 to 16. The scaling factor is determined in bits 8 to 11 of the master mode register.

Frequency F6 can be applied to each gate of counters 1 to 5 via the gate control logic.

> **!** Because of the synchronization circuitry, F6 must not exceed 500 kHz
> (500 kHz $\triangleq$ 1 $\mu$sec gate time).

**The following formula is used to calculate the frequencies:**

$$F1 = \frac{1\,\text{MHz}}{\text{Prescaler Register}}$$

| Master Mode Register, Bit 15 = 0<br>Binary Scaling :16 | Master Mode Register, Bit 15 = 1<br>BCD Scaling :10 |
|---|---|
| $F2 = \dfrac{F1}{16}$ | $F2 = \dfrac{F1}{10}$ |
| $F3 = \dfrac{F1}{256}$ | $F3 = \dfrac{F1}{100}$ |
| $F4 = \dfrac{F1}{4096}$ | $F4 = \dfrac{F1}{1000}$ |
| $F5 = \dfrac{F1}{65536}$ | $F5 = \dfrac{F1}{10000}$ |

The following maximum scaling factors (F6) result:

Maximum scaling factor:

$$65535 \bullet 65536 \bullet 16 = 6.8 \bullet 10^{10}$$

Maximum cycle duration:

68718 sec

Maximum gate time:

34359 sec

---

## 1.10.2  Internal Reference for Counters 6 and 7                    IP 242B

The internal reference frequency of 10 MHz is available to counters 6 and 7 for time measurements with a resolution of up to 100 nsec.

**B**

## 1.11      Switching of Counters Among Each Other

Basically, counters can be operated separately.
However, more complex applications often require skilled combination of individual counters with different single functions.
Even simple cascading is an example of such a combination ($\rightarrow$ section 7.1).

The master mode and counter mode registers for counters 1 to 5 make it possible to implement such "switching without external wiring".

This allows, for example, a signal from one single encoder to be used by one counter to acquire the number of revolutions while the second counter uses this signal to make length measurements of parts passing by in connection with the light barriers connected to the gate inputs, etc., etc. (see also section 8.2.2).

There are almost no limits to your imagination.
Remember the following points, however, when implementing your applications:

1.          The scaling sequence is only present once.
            The settings in the MMR apply to all counters (1 to 5).
            Of course, you can also give up one counter as an additional scaler.

2.          The maximum output frequency of all counters (1 to 7) is 40 kHz.

Since there is no internal connection for counters 6 and 7 to counters 1 to 5, or the scaling sequence, circuiting of counters 1 to 5 with counters 6 and 7 always requires an adjustment of the level.
A 24 V asymmetrical signal to 5 V (RS422) symmetrical signal converter is available in two–channel design for installation on a top hat rail ($\rightarrow$ section10.11).

## 1.12      Command Lists and Measured Value Memory

Command lists allow you to store command sequences (i.e., the smallest of "programs") in the IP 242A/242B. The control words used here are the same as those used to address the module with the standard function block from the S5.

Section 6 gives you an overview of the command set and its uses.

These command lists can be triggered either via the gate inputs of counters 1 to 5, the counter outputs (i.e., a counter which has expired, for example), or via reset inputs 6 and 7.

Assuming the enable in the interrupt enable register ($\rightarrow$ section 3.3.4), you have the choice of simul–taneously forwarding this interrupt to the command list or to the S5.
The interrupt filter register of the IP 242B offers you a specific, individual means of influence.

Store the desired command sequence in the parameterization data block. The lists are activated when you transfer the data to the module with the "parameterize counter (global register)".

Additional command lists were installed for the IP 242B since storage space for the interrupt command lists is tight.
The size of these additional command lists can be specified in a separate directory and the lists are enabled there.
They are called with a "process command list" command from one of the interrupt command lists, or by the same command in the standard function block from the S5.

Interrupt command lists must be concluded with the "command list end" command (BE) when a list contains less than five commands. The additional command lists are managed in a separate directory.

What can be accomplished with such a command list?

Individual counters can be controlled separately.
But the greatest advantage is that you can use the transfer, calculation and comparison value functions together with the constants and/or process conditions which can be specified, the counter values, etc. to combine the seven counters into a more complex "circuit" which, most importantly, is independent of the S5 ($\rightarrow$ sections 6.2. and 1.13).

And, of course, you can also program "abbreviated" calculations with up to 5 commands in the interrupt command lists.

A Simple Example:

With the IP 242A, you could generate a difference of two counters directly on the module. With the IP 242B, you can now even acquire the difference of several counters on board, and use the comparison function together with the counters and their direct digital outputs to achieve very fast effects on the process.
You can also buffer the results of your calculations and links parallel to the new result registers in a measured value memory (up to 100 $*$ 2 DWs), and fetch all of them from the S5 when required ($\rightarrow$ sections 1.13 and 3.7.2).

This increased flexibility makes it impossible to list all capabilities here.
For this reason, sections 7 and 8 are only an introduction to the basics, and are primarily intended to arouse your interest in the many ways in which you can now solve technological tasks.

## 1.13    Calculation Functions (For IP 242B Only)

The IP 242B permits simple calculation operations to be performed directly on the module without using the S5 CPU. The required commands are grouped together in the interrupt and/or additional command lists.

Section 8 contains an example of how and when the calculation functions can be used.

```
– Load from hold 1      (Counter value)                        |    4    |
– Load constant 3                                              |    3    |
– MUL                                            FW            |    2    |
– Transfer load 2                                        | ACCU   1    |
– Transfer result 1
– Load/start  C2


                         Command list X
```
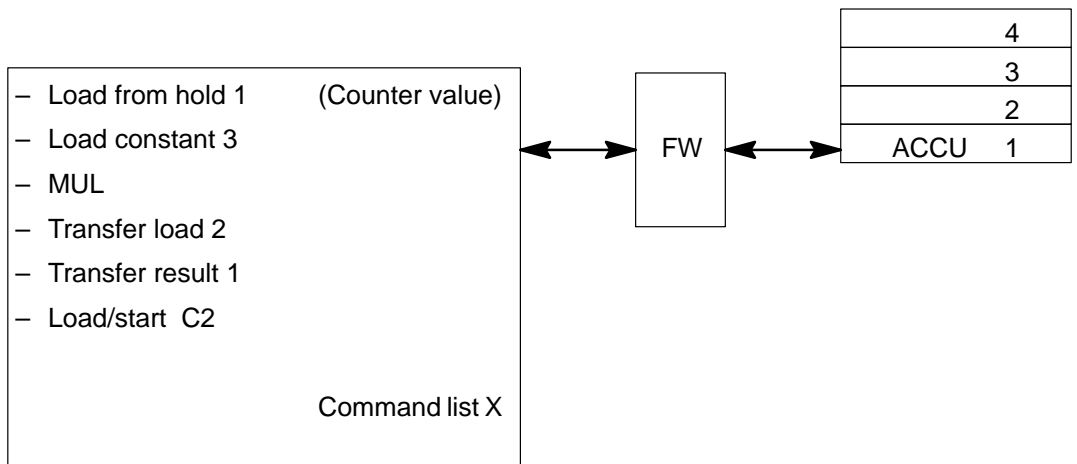
A calculation stack of 4 accumulators, all of which are updated for every load and calculation command, is used for the calculation functions.

A command list can be started via the following.

– Gate interrupt
– Output interrupt

Additional command lists can be started via the following.

– Internally in the command list with the "process command list" control word
– Externally via the control word register with the "process command list" control word

The accumulators must be appropriately loaded before the actual calculation operations are executed (→ section 6.2.4). This is done with the "load from register" or "exchange accumulators" transfer operations (→ section 6.2.3). After the calculation, the result is loaded in a result register with the "transfer in register" control word, for example. It can be accessed there with a "read" command.

The comparison operations can be used for conditional jumps in a command list (→ section 6.2.5).

# 2      Hardware Settings

## 2.1 Layout of the Setting Elements



2 LEDs for operation status
(section 5.2)

Conditioning of the levels for counter and gate inputs of counters 1 to 5
(section 2.4)

Frequency conditioning of counters 1 to 5
(section 2.5)

Setting of the basic address
(section 2.2)

Setting of the page frame number
(section 2.2)

Assignment of basic connector
(section 10.9)

Process interrupt via input byte IB 0
(section 2.3)

Section of an interrupt line
(section 2.3)

Assignment of front connectors X3 to X6
(see section 2.6)

Jumpers BR1 and BR2 must be open (for factory testing purposes).

## 2.2      Setting of the Module Address

**Basic Address**

The IP 242A occupies one page frame or a $2^{10}$–byte area of memory in the I/O area of the programmable controller's central processing unit.
The IP 242B occupies one page frame.
Use DIP switch S3 to set the basic address.

The module usually operates with page frame addressing.

**A** Linear addressing is only possible with programmable controllers S5–115U and S5–155U using the IP242A.

Set the basic address to F400H (status on delivery) when using page frame addressing.

S3.1                                                                        S3.8

| A15 | A14 | A13 | A12 | A11 | A10 | | KA |
|-----|-----|-----|-----|-----|-----|---|----|
| $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | | |

S3

ON

OFF

ON  : Page frame addressing
OFF : Linear addressing

Switch is always OFF.

Address bits A10 to A15
Basic address of the module
ON  = "1"
OFF = "0 "
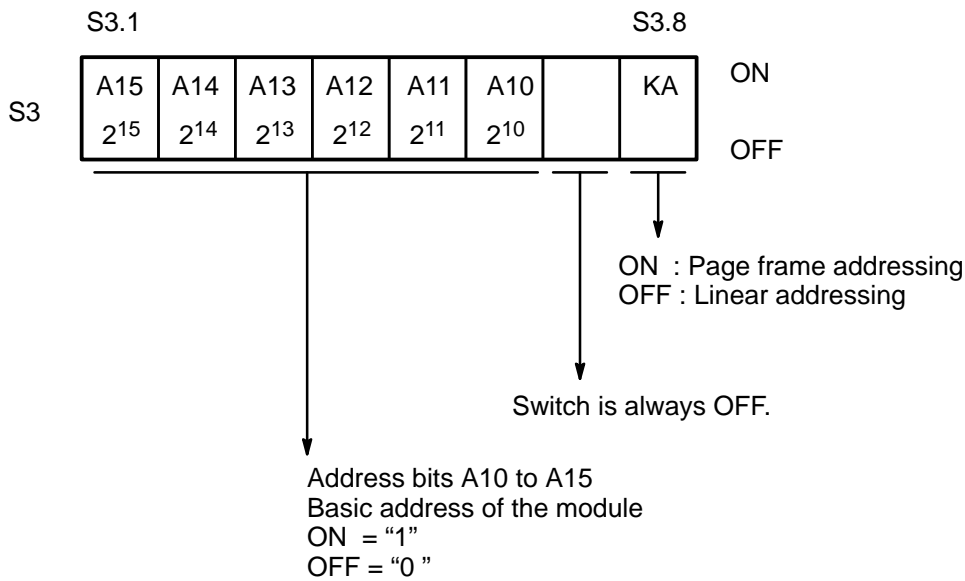
**B** The IP 242B uses only function blocks which support page frame addressing. Switch S3.8 is always ON.

**Example:**

The module is addressed linearly. Basic address is E000H.

S3.1                                                                        S3.8

S3

ON

OFF

E                    0
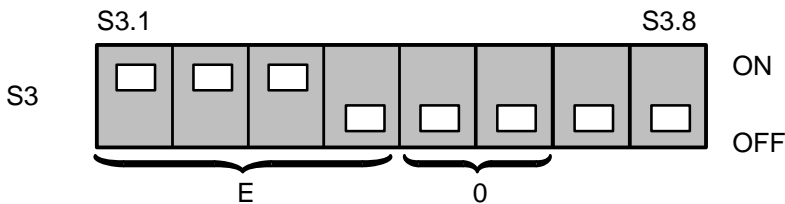
**Status on Delivery:**

Page frame addressing is selected and set to F400H.

S3.1                                                           S3.8

S3

ON

OFF
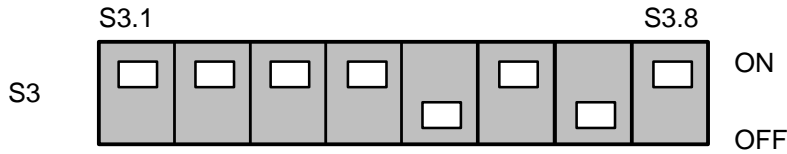
**Page Frame Number**

In addition to basic address F400H, the page frame number must also be set on the module for page frame addressing. The page frame number consists of 8 bits (bits 0 to 255) and is set with DIP switch S4.

S4.1                                                           S4.8

| S4 | KN7 | KN6 | KN5 | KN4 | KN3 | KN2 | KN1 | KN0 | ON |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
|    | $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ | OFF |
|    | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 | decimal |

ON  = "1"
OFF = "0"
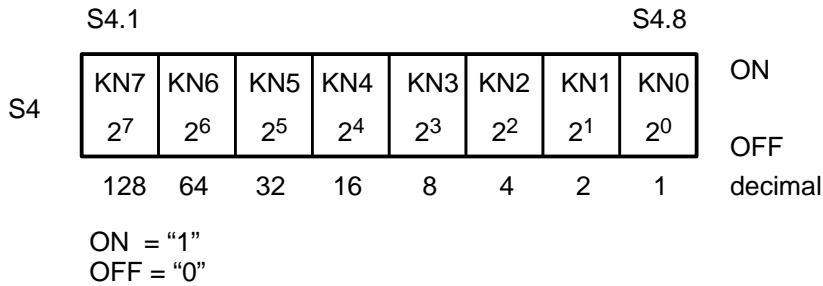
**Example:**

Page frame number = 70 (46H)

S4.1                                                           S4.8

S4

ON

OFF
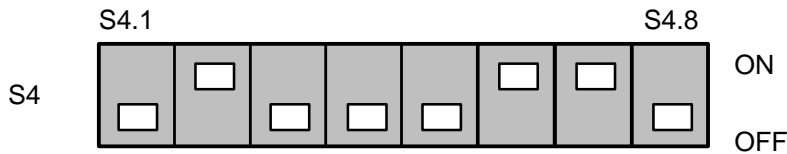
**Status on Delivery:**

Page frame number = 00H

S4.1                                                           S4.8

S4

ON

OFF
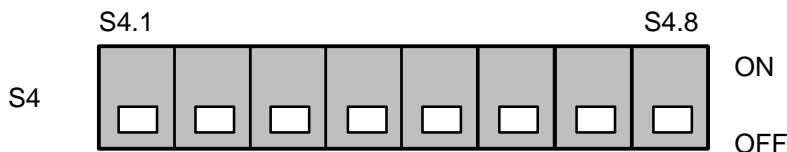
## 2.3      Interrupts and Process Interrupts

**Selecting an Interrupt Line**

Interrupt processing via interrupt lines or input byte IB 0 depends on the programmable controller (see section 4, "Interrupt Processing").

Set DIP switch S1 to select the S5 interrupt line which triggers interrupts on the S5 central processing unit.

Set only one switch to ON.



|     |     |                                        |
| --- | --- | -------------------------------------- |
| ON  | :   | Interrupt via corresponding line       |
| OFF | :   | No interrupt                           |
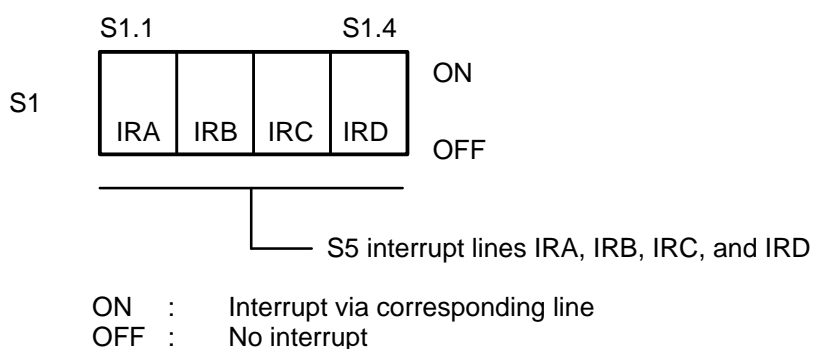
**Example:**                                        **Status on Delivery:**

Interrupt via interrupt line IRB
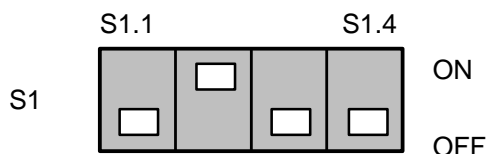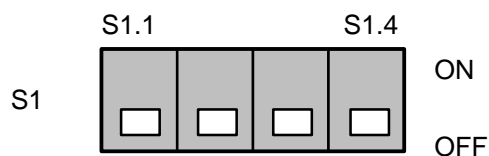


**Process Interrupt via Input Byte IB 0**

For page frame addressing and triggering of process interrupts via input byte IB 0, define the first module as master interrupt source and all other modules as slave interrupt sources. Processing of process interrupts is not possible (switch S2.10 = off) for linear addressing.

Set this configuration with DIP switch S2.

**Slave:**

For every module defined as a slave interrupt source, close only switches S2.9, S2.10, and one of switches S2.2 to S2.7 (no double allocation). The corresponding bit of input byte IB 0 is thus assigned to this module.

**Master:**

For the module defined as master interrupt source, close only switches to which no slave interrupt sources are assigned (i.e., do not use the corresponding bits of input byte IB 0 in your S5 system).

IP 242A/242B Equipment Manual
                                                     © Siemens AG 1993, Order No: 6ES5 998–0KM21

S2.1                                                                 S2.10

| EMA | SA1 | SA2 | SA3 | SA4 | SA5 | SA6 | SA7 | MSA | PB0 | ON |

S2                                                                              OFF

**Process interrupt via input byte IB 0**
ON  : enabled
OFF : disabled

**Slave/master interrupt source**
ON  : slave interrupt source
OFF : master interrupt source

**Slave process interrupt**
ON
 – On master: those bits which are not allocated to slaves
 – On slave:   one bit per slave (no double allocation)
OFF
 All other switches

**Master process interrupt**
ON  : enabled
OFF : disabled

> Do not use input/output modules with input bytes IB 0 and IB 1 when DIP switch S2.10 is ON.

**Status on Delivery:**

S2.1                                                                 S2.10

S2  [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ]   ON / OFF
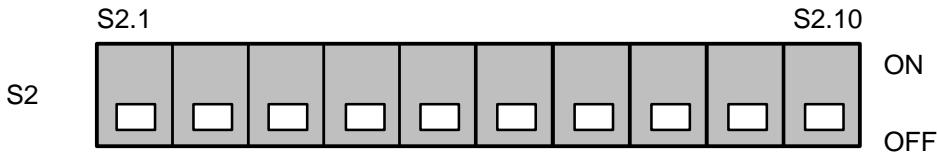
The following table shows the switch positions for three counter modules which are to operate as master, slave 1, and slave 6:

x = ON        – = OFF

| IB 0 | DIP Switch | Type of Interrupt | Master | Slave 1 | Slave 6 |
|------|-----------|-------------------|--------|---------|---------|
| Bit 0 | S2.1 | Master interrupt | x | – | – |
| Bit 1 | S2.2 | Slave interrupt | – | x | – |
| Bit 2 | S2.3 | No interrupt | x | – | – |
| Bit 3 | S2.4 | No interrupt | x | – | – |
| Bit 4 | S2.5 | No interrupt | x | – | – |
| Bit 5 | S2.6 | No interrupt | x | – | – |
| Bit 6 | S2.7 | Slave interrupt | – | – | x |
| Bit 7 | S2.8 | No interrupt | x | – | – |
|  | S2.9 |  | – | x | x |
|  | S2.10 |  | x | x | x |

**Example 1:**

The group interrupt is triggered via input byte IB 0. The IP 242A/242B is the master interrupt source. The programmable controller uses 3 additional interrupts.

S2.1                                                                                     S2.10

S2

ON

OFF

Process interrupt via
IB 0

Module is the master interrupt
source.

These bits are not used by any I/O module to trig-
ger interrupts. These bits are allocated with "0"
on the master interrupt module.

These bits are used by other I/O modules to trigger interrupts. Do not
allocate these bits on the master interrupt module.

The master process interrupt is enabled.

**Example 2:**

The IP 242A/242B triggers process interrupts via input byte IB 0 (SA1). The module is a slave interrupt source.

S2.1                                                                                     S2.10

S2

ON

OFF

Process interrupt via
input byte IB 0

The module is slave
interrupt source.

These bits can be used by additional I/O modules to trigger slave
interrupts.

Slave interrupt via bit 1

Master process interrupt is disabled.
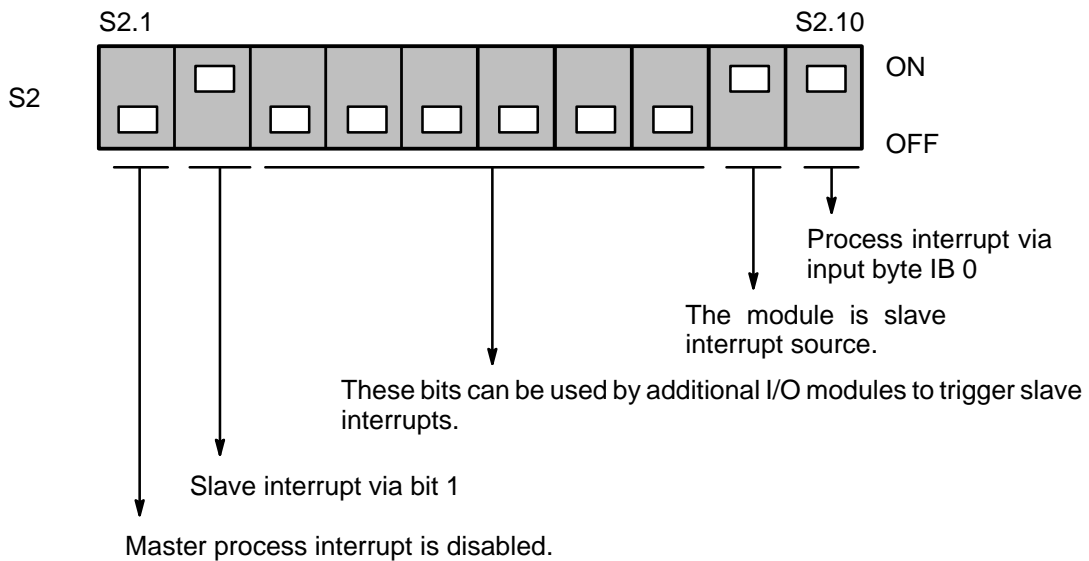
## 2.4    Level Conditioning of Counter Inputs 1 to 5

Use plug in jumpers BR3 to BR17 to set each of the inputs to 5 V level or to 24 V level.

The following section of the printed circuit board shows the location and designation of the plug in jumpers.
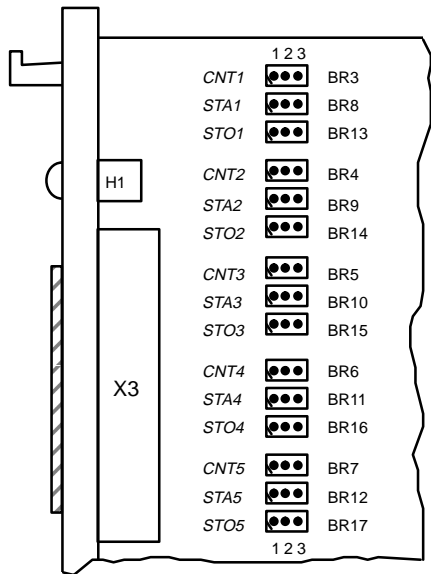
CNT n  $\triangleq$  counter input number n
STA n  $\triangleq$  start input number n
STO n  $\triangleq$  stop input number n
    n  =  1 to 5

| Jumper | Jumper Allocation | Input Level |
|--------|-------------------|-------------|
| BR$_X$ | 1 ——2   3 | 5 V |
| BR$_X$ | 1   2——3 | 24 V |
| BR$_X$ | 1   2   3 | 24 V |

x = 3 to 17

**Example:**

Insert jumpers BR5, BR10, and BR15 in position 1–2 to condition the count, start, and stop inputs of counter 3 to a 5–V level.

**Status on Delivery:**

Jumpers BR3 to BR17 are inserted in position 2–3 (24 V).

## 2.5 Frequency Conditioning for Counter Frequencies of Counters 1 to 5

If there is a danger of interference signals occurring despite external shielding, use capacitors C8 to C22 to reduce interference susceptibility.

The count, start, and stop inputs are then conditioned to the maximum counter frequency.

High frequency interference pulses are thus blanked out, particularly when processing relatively low frequencies.

The below drawing shows the position of one of these capacitors in the input circuitry.

Use the following table to determine what size capacitors to select. The specified values are intended as guide lines. Other values are also possible.



| Maximum Counting Frequency | C8 to C22 |
|---|---|
| Approximately 480 kHz | open |
| Approximately 200 kHz | 100 pF |
| Approximately 20 kHz | 1 nF |
| Approximately 2 kHz | 10 nF |
| Approximately 200 Hz | 100 nF |
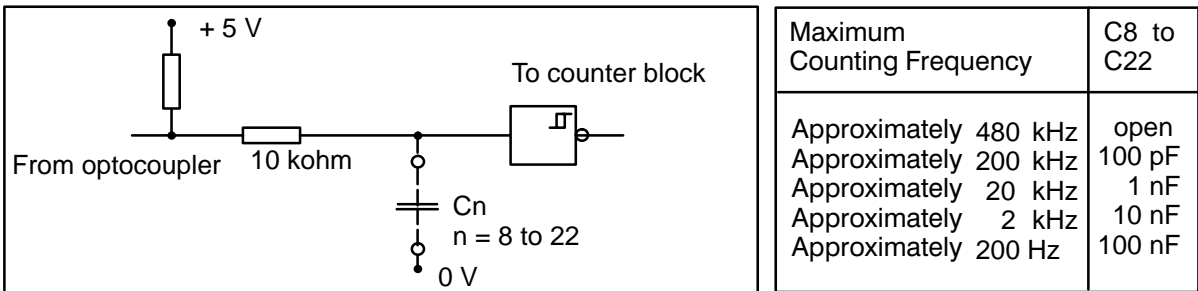
Now solder the selected capacitors in the corresponding slots.



CNT n ≙ counter input number n
STA n ≙ start input number n
STO n ≙ stop input number n
    n = 1 to 5
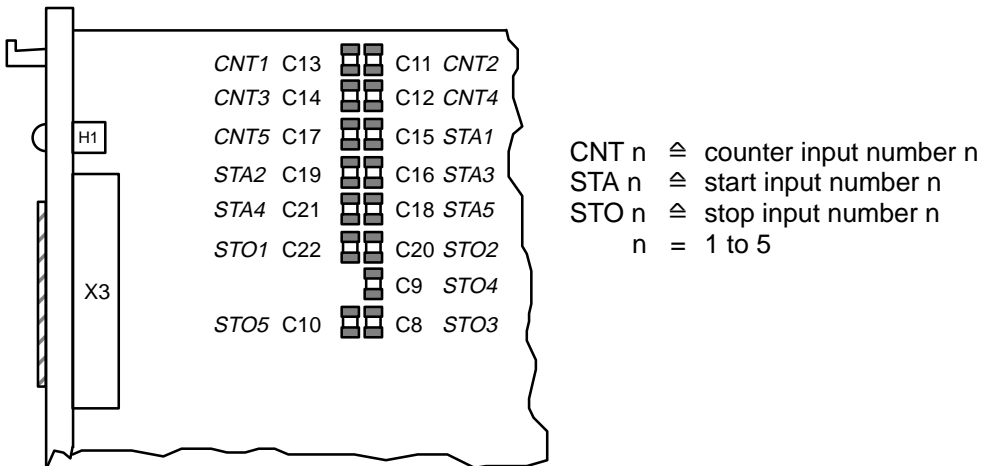
**Status on Delivery:**

**No** capacitors are soldered in when your module is delivered from the factory (i.e., counting frequency up to 480 kHz).

> ⚠ Please be careful not to get soldering tin and wire clippings on the printed circuit board while soldering. This can cause short circuits which not only affect the functions but also destroy your module. ⚠

Use 5–mm capacitors.

---

| | 2.6 | Front Panel and Front Connectors |
|---|---|---|

The illustration shows the front panel with labelling and connectors.

RUN  F
(operating)

RUN LED (green) / F LED (red)
  (operating)      (malfunction)

1°

37

IN 1–5

X3: 37–way sub D connector (pins)
    Counter and gate inputs (start, stop) of counters 1 to 5

1°

9

IN 6

X4: 9–way sub D connector (pins)
    Inputs for counter 6
    (incremental encoder)

1°

9

IN 7

X5: 9–way sub D connector (pins)
    Inputs for counter 7
    (incremental encoder)

15

1

OUT 1–7

X6: 15–way sub D connector (sockets)
    Outputs for counters 1 to 7

**1. Allocation of Front Connector X3:**

Start, Stop, and Count Inputs of Counters 1 to 5:

| Pin | Signal |     | Pin | Signal |
|-----|--------|-----|-----|--------|
|     |        |     | 1 | CNT1 |
| 20 | CNT1$_\perp$ |  | 2 | STA1 |
| 21 | STA1$_\perp$ |  | 3 | STO1 |
| 22 | STO1$_\perp$ |  | 4 | |
| 23 | CNT2$_\perp$ |  | 5 | CNT2 |
| 24 | STA2$_\perp$ |  | 6 | STA2 |
| 25 | STO2$_\perp$ |  | 7 | STO2 |
| 26 | |  | 8 | |
| 27 | CNT3$_\perp$ |  | 9 | CNT3 |
| 28 | STA3$_\perp$ |  | 10 | STA3 |
| 29 | STO3$_\perp$ |  | 11 | STO3 |
| 30 | |  | 12 | |
| 31 | CNT4$_\perp$ |  | 13 | CNT4 |
| 32 | STA4$_\perp$ |  | 14 | STA4 |
| 33 | STO4$_\perp$ |  | 15 | STO4 |
| 34 | |  | 16 | |
| 35 | CNT5$_\perp$ |  | 17 | CNT5 |
| 36 | STA5$_\perp$ |  | 18 | STA5 |
| 37 | STO5$_\perp$ |  | 19 | STO5 |
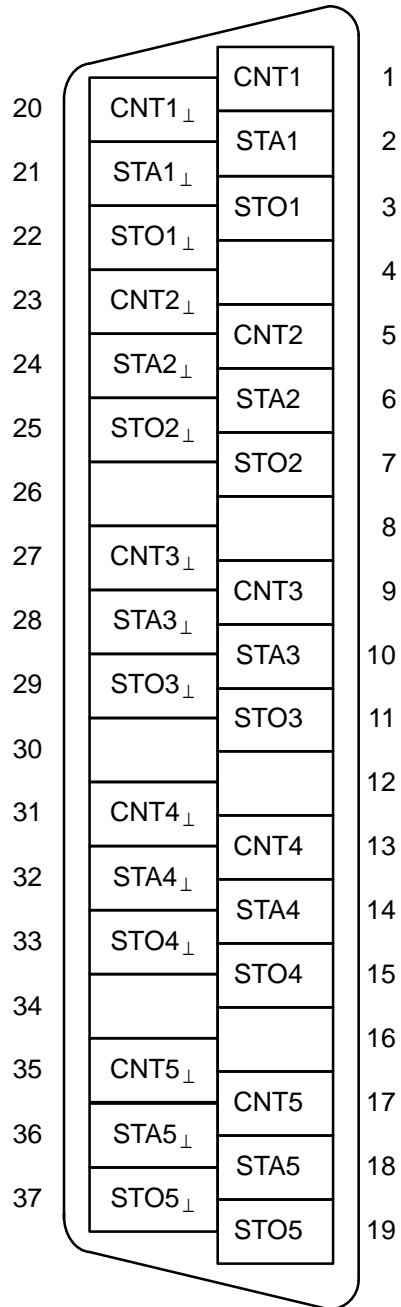
Signal Designation:     CNTn :   counter input, counter n
                        STAn :   start input,    counter n
                        STOn :   stop input,    counter n

                        xxxn$_\perp$ :   reference potential of the corresponding input

IP 242A/242B Equipment Manual
                                                                      © Siemens AG 1993, Order No: 6ES5 998–0KM21

## 2. Allocation of Front Connectors X4 and X5:

Inputs of Counters 6 and 7:

| | | |
|---|---|---|
| 6 | $\overline{A}n$ | An   1 |
| 7 | $\overline{B}n$ | Bn   2 |
| 8 | $\overline{N}n$ | Nn   3 |
| 9 | $SYNn_\perp$ | SYNn   4 |
| | | 5 |

Signal Designation:

| | | |
|---|---|---|
| An, $\overline{A}n$ | : | differential counter input, counter n |
| Bn, $\overline{B}n$ | : | differential counter input, counter n |
| Nn, $\overline{N}n$ | : | differential input, zero marking pulse |
| SYNn | : | input, synchronous pulse |
| $SYNn_\perp$ | : | reference potential, synchronous input |
| | | (= reference potential of the counter module) |
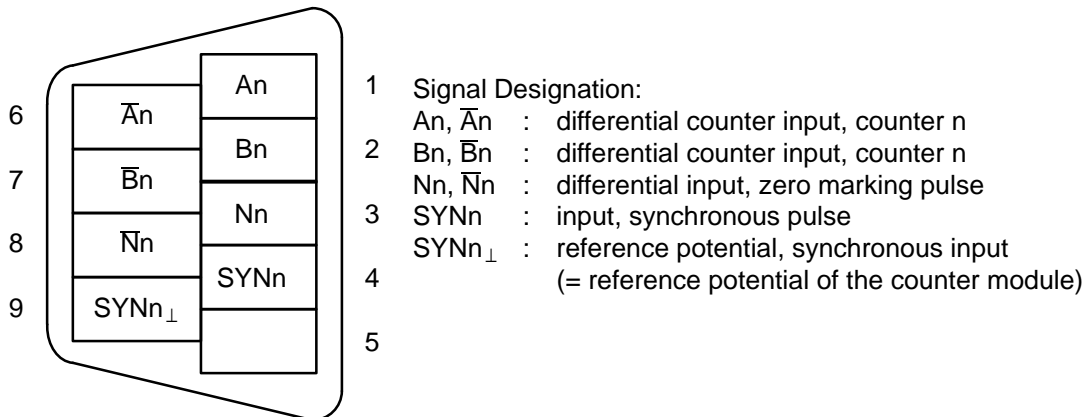
## 3. Allocation of the 15–Way Sub D Socket Connector X6:

Outputs of All Counters:

| | | |
|---|---|---|
| 15 | +24 $V_{external}$ |   8 |
| 14 | +24 $V_{external}$ | OUT7   7 |
| 13 | | OUT6   6 |
| 12 | | OUT5   5 |
| 11 | | OUT4   4 |
| 10 | 0 $V_{external}$ | OUT3   3 |
| 9 | 0 $V_{external}$ | OUT2   2 |
| | | OUT1   1 |

Signal Designation:

| | | |
|---|---|---|
| OUTn | : | output of counter n |
| +24 $V_{external}$ | : | external supply voltage |
| 0 $V_{external}$ | : | external reference potential for +24 $V_{external}$ |
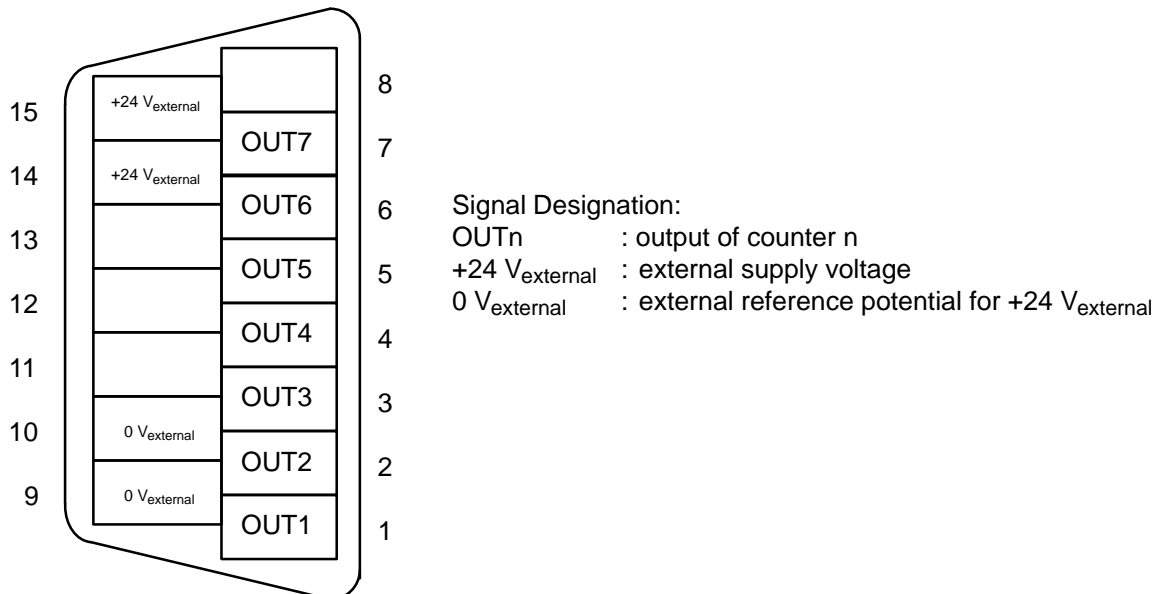
Follow the SIMATIC S5 configuration guidelines when connecting the front plug connectors to ensure smooth IP 242A/242B operation.
Since interferences can also occur on the external supply voltage lines:
       – Keep the lines short.
       – Use a filter for longer lines.

---

# 3    Software Settings (Registers)

## 3.1    Overview of the Registers

The following registers are used to control the counter module (e.g., determine the purpose and mode of the counters). Use the parameterization data block to set the registers ($\rightarrow$ section 11.6 or 14.6).
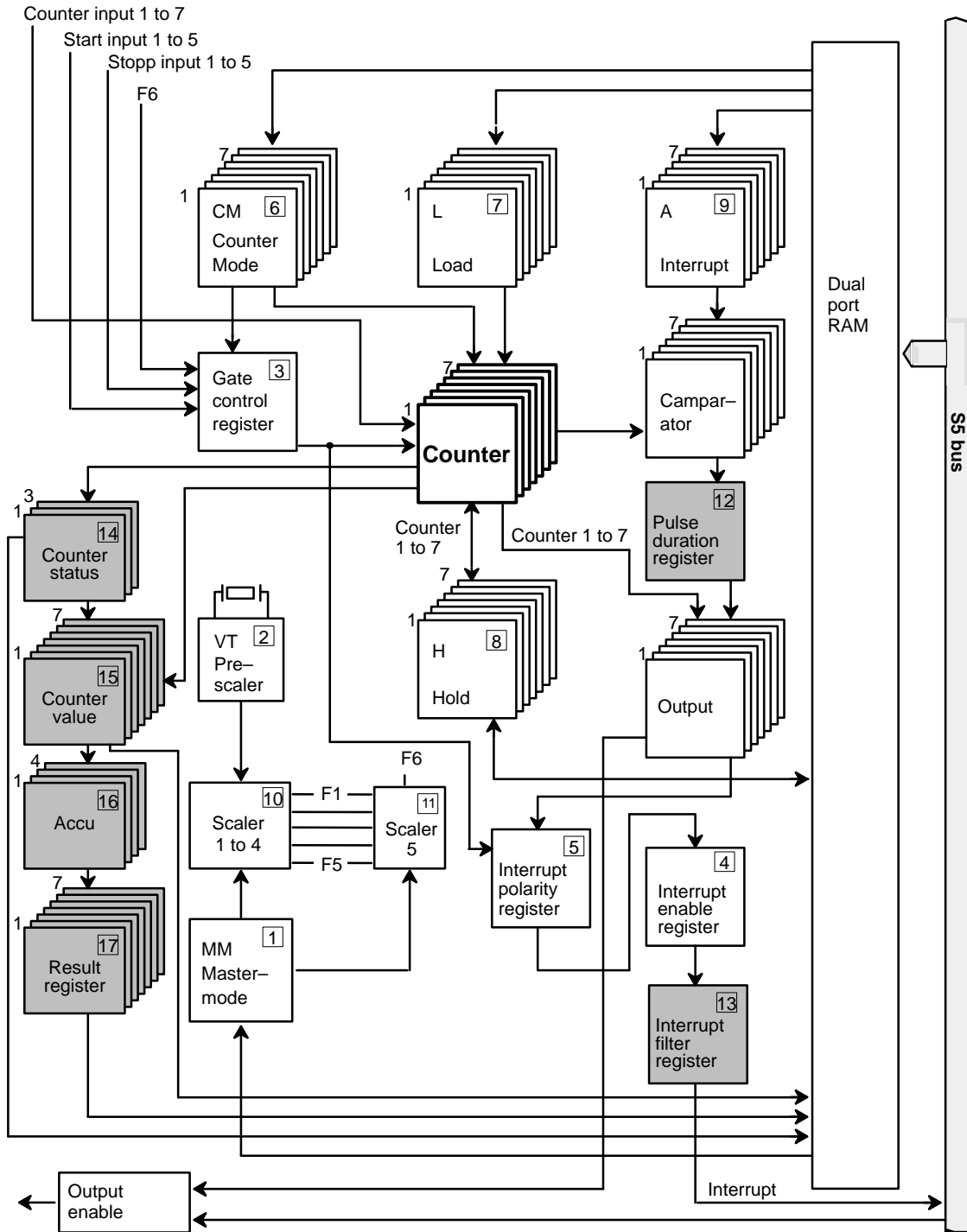


For IP 242B only

## 3.2      Definition of the Registers

| | |
|---|---|
| 1 [1] | Master mode register MM → Parameterize scalers, scaling factors, comparator enable, and pulse source here. (See section 3.3.1 for more details.) |

1 [1]  Master mode register MM → Parameterize scalers, scaling factors, comparator enable, and pulse source here. (See section 3.3.1 for more details.)

2 [1]  Prescaler register → Set the prescaling and scaling factors here. (See section 3.3.2 for more details.)

3 [1]  Gate control register → Set the gate mode here. (See section 3.3.3 for more details.)

4 [1]  Interrupt enable register → Set the interrupt sources which trigger interrupts here. (See section 3.3.4 for more details.)

5 [1]  Interrupt polarity register → Set the triggering edge here. (See section 3.3.5 for more details.)

6  Counter mode register CM → Parameterize operation mode, counter pulse source, and output signals of the counter here. (See section 3.4.1 for more details.)

7  Load register L → Enter the start value of the counter here. (See section 3.4.2 for more details.)

8  Hold register H → The counter uses this register for intermediate storage. (See section 3.4.3 for more details.)

9  Interrupt register → Enter an interrupt value (interrupt limit) here. (See section 3.4.4 for more details.)

10 [2]  Scalers 1 to 4 → Frequencies F2 to F5 are generated here. The scaling factor is determined in the master mode register. (See sections 1.10 and 3.3.1 for more details.)

11 [2]  Scaler 5 → Frequency F6 is generated here. The input frequency and the scaling factor are determined in the master mode register. (See sections 1.10 and 3.3.1 for more details.)

1) 2) See next page for explanation

| 12 | 1) Pulse duration register | → | The pulse duration for software comparator for counters 3 to 7 is parameterized here. (See section 3.3.10 for more details.) |

| 13 | 1) Interrupt filter register | → | The interrupts from the module to the S5 can be specifically enabled here. (See section 3.3.6 for more details.) |

| 14 | Counter status register | → | The current module status is entered here for each time the function block is called. This status can be read by the S5. (See section 3.8.1 for more details.) |

| 15 | Counter values | → | The IP 242B enters the counter values cyclically here. (See section 3.4.5 for more details.) |

| 16 | Accu | → | The calculation functions are implemented with the accu. (See section 6.2.3 for more details.) |

| 17 | Result register | → | The results of the calculation functions are stored here. (See section 3.5.1 for more details.) |

For IP 242B only

> 1) These registers are global registers (i.e., the registers are only present once on the module, and the settings in these registers apply to **all** counters).

2) Scalers 1 to 4 and scaler 5 are also only present once. Here, however, a certain amount of "subassignment" (adaptation) can be achieved for the individual counters by varying the use of the output frequencies.

## 3.3 Description of the Global Registers
## 3.3.1 Master Mode Register (MMR)

**IP 242A/242B**

**A** **B** The master mode register is located in data word DW4 of the parameterization data block.

The master mode register controls the scalers and counters 1 to 5.

The relationship between frequencies F1 to F6 is explained in section 1.10.1.

The command "parameterize counter" (global register) causes the contents of this register to be transferred to the module.

**F6 Scaling factor, scaler 5**
0001 = : 1
0010 = : 2
0011 = : 3
0100 = : 4
0101 = : 5
0110 = : 6
0111 = : 7
1000 = : 8
1001 = : 9
1010 = :10
1011 = :11
1100 = :12
1101 = :13
1110 = :14
1111 = :15
0000 = :16

**F6 Pulse source, scaler 5**
0000 = F1
0001 = Counter input 1
0010 = Counter input 2
0011 = Counter input 3
0100 = Counter input 4
0101 = Counter input 5
0110 = Gate 1
0111 = Gate 2
1000 = Gate 3
1001 = Gate 4
1010 = Gate 5
1011 = F1
1100 = F2
1101 = F3
1110 = F4
1111 = F5

Bit 15                                                                          Bit 0

MM | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

**Comparator** 3   **Comparator** 2
**Comparator** 4   **Comparator** 1
**Comparator** 5

**Comparators**
0 = off
1 = on

Allocate with 00

**Scaler mode, scalers 1 to 4**
0 = Binary scaling
1 = BCD scaling

### 3.3.2  Prescaler Register (VTR)                                        IP 242A/242B

The prescaler register is located in data word DW5 of the parameterization data block.

$\boxed{A}$ $\boxed{B}$

The prescaler is a 16–bit scaler which has a 1–mHz frequency as source and supplies frequency F1 at the output.

The prescaler scales the frequency which is then used by the counter block as the input signal for determination of the internal reference times.

The gate opening time, for example, can thus be determined for measuring frequencies or times. (See section 1.10.1 "Internal Clock Pulses" and section 7 "Special Functions" for more details.)

The command "parameterize counter" (global register) causes the prescaler register to be transferred to the module.

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
VT
| $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^9$ | $2^8$ | $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |

Scaling factor
FFFFH = ÷ 5535
to
0001H = ÷ 1
0000H = ÷ 1
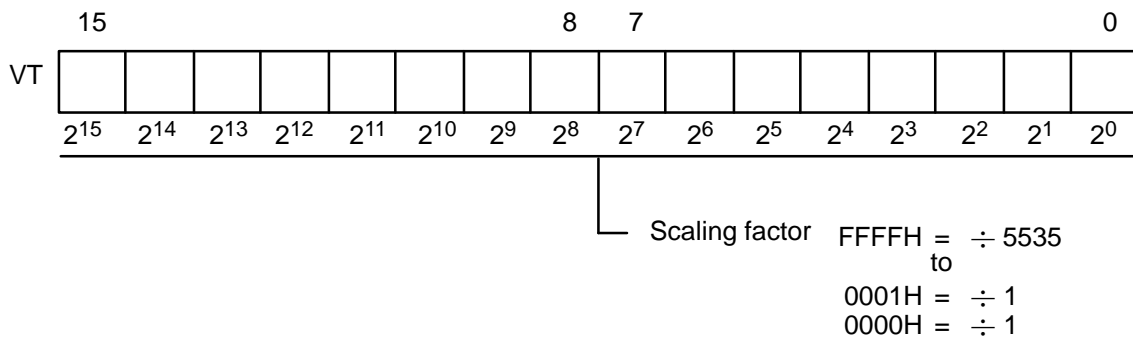
For even scaling factors, the pulse interval ratio of F1 is exactly 1:1.
For odd scaling factors, the pulse is always 1 μsec longer than the interval.
For scaling factors 0 and 1, the output frequency is F1 = 1 MHz.

**A**  **B**

---

### 3.3.3  Gate Control Register (TSR)

**IP 242A/242B**

---

The gate control register is located in data word DW6 of the parameterization data block.

Through the gate control logic, internal frequency F6 can be used as the gate enable for counters 1 to 5. Single frequency measurement can be performed in this manner.

Separate start and stop inputs are provided for counters 1 to 5 to allow start and stop operation with separate encoders.

The gate control logic makes it possible to switch from start and stop operation with separate inputs to level control at the start inputs. (See section 1.7.)

The command "parameterize counter" (global register) causes the gate control register to be transferred to the module.

```
      15                            8   7                      0
TS  | 0 | 0 | S5 | S4 | S3 | S2 | S1 | 0 | 0 | 0 | T5 | T4 | T3 | T2 | T1 | 0 |
```

| Gate Mode | Gate Control Register Sn \| Tn | | Signal Generation | Type of Control | Signals Used | Explanation |
|-----------|----|----|-----------|-----------|-----------|-------------|
| 1 | 0 | 1 | Internal | Level | F6 | 1–mHz signal generated by scaling from the internal frequency generator (Clock pulse rate is parameterizable.) |
| 2 | 0 | 0 | External | Level or edge [1] | STA **only** | Signal **only** at the start input of the applicable counter |
| 3 | 1 | 0 | External | ⌐ | STA/STO | Signals at the start/stop input of the applicable counter |
| 4 | 1 | 1 | External | ⌐_ | STA/STO | Signals at the start/stop input of the applicable counter |

n = 1 to 5

1) The settings in the CMR apply when edge control is used.

## 3.3.4   Interrupt Enable Register (IFR)                                IP 242A/242B

The interrupt enable register is located in data word DW7 of the parameterization data block.     **A**   **B**

The following internal interrupt sources are available:

– Counter outputs    1 to 7
– Counter gates      1 to 5
– Reset inputs       6 to 7
– Module error messages

The internal interrupt is enabled when the bit is set (= 1) (i.e., a corresponding command list is executed and the appropriate bit set in the interrupt information register).

An entry in the interrupt information register is made on the IP 242A under the following conditions.     **A**
– The SA bit is set (for all sources).
– The interrupt to the programmable controller is enabled.
  (→ section 3.8.2)

An entry is made in the IIR on the IP 242B depending on the status of the SA bit.     **B**
– SA bit = 1          Entries made as on the IP 242A
– SA bit= 0           The interrupt filter register allows specific selection of the sources for an interrupt
                      to the S5 (→ section 3.3.6).

If the interrupt or process interrupt is also enabled (SA bit = 1), a process interrupt is also transmitted to the     **A**   **B**
S5 for **each** counter interrupt.

If the triggering of an interrupt is desired when an error message occurs, set the ER bit to "1". Interrupts are still triggered here even if the SA bit is set to "0".

The command "parameterize counter" (global register) or the command "mask interrupt" IM causes the interrupt enable register to be transferred to the module.

```
   15                           8  7                            0
IF │A7│A6│A5│A4│A3│A2│A1│SA│R7│R6│T5│T4│T3│T2│T1│ER│
```

- Interrupt enable at error message
- Gate inputs 1 to 5
- Counter reset 6, 7
- Enable for interrupt or process interrupt to the programmable controller

  SA = 0  Interrupt filter register can be used.     **B**
  SA = 1  Interrupt filter register cannot be used.
            (→ section 3.3.6)

- Outputs 1 to 7
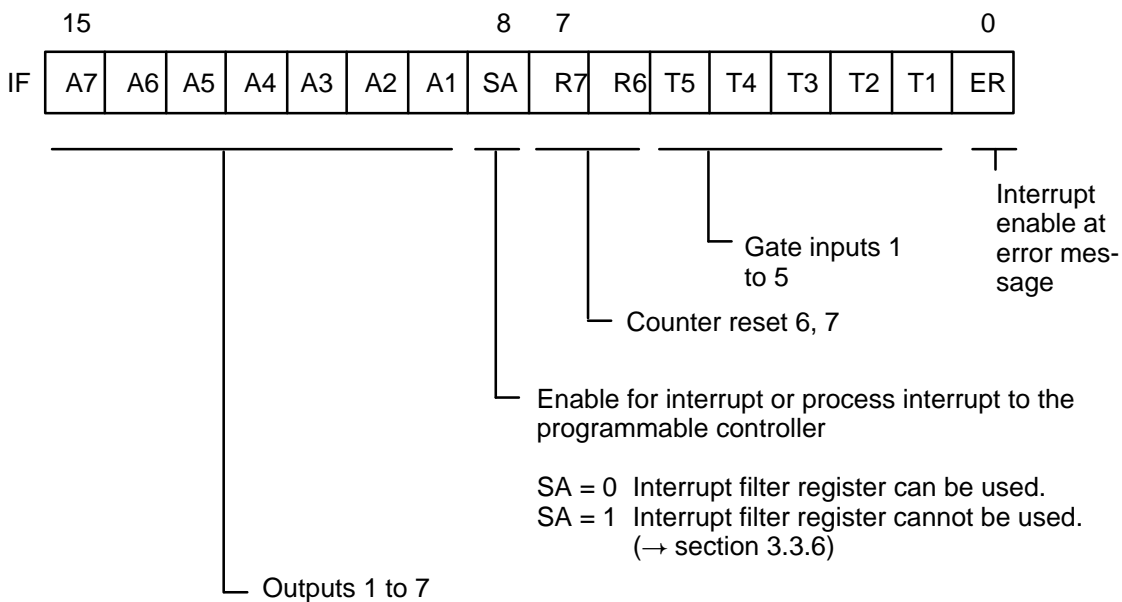
### 3.3.5    Interrupt Polarity Register (IPR)

**IP 242A/242B**

**A** **B** The interrupt polarity register is located in data word DW8 of the parameterization data block.

The edge with which an interrupt or process interrupt is triggered is set with the interrupt polarity register.

The following applies in general:

– T1  to  T5 =    0: Interrupt or process interrupt is triggered at the rising edge at the counter gate (T).
– T1  to  T5 =    1: Interrupt or process interrupt is triggered at the falling edge at the counter gate (T).

The following applies when the comparator function is not enabled:

– Q1 to  Q5 =    0: Interrupt or process interrupt is triggered at the rising edge of the counter output (Q).

– Q1 to  Q5 =    1: Interrupt or process interrupt is triggered at the falling edge of the counter output (Q).

The following applies when the hardware comparator function (counters 1 and 2) is enabled:

– Q1  and Q2 =    0: Interrupt or process interrupt is triggered at the rising edge of the counter output (Q).

– Q1  and Q2 =    1: Interrupt or process interrupt is triggered at the falling edge of the counter output (Q).

Remember that, for hardware comparators, the output is active only when the counter value coincides with the interrupt value.

When the **software comparator** (counters 3 to 5) is **enabled**, an interrupt or process interrupt is triggered at the comparison of counter status with the interrupt register as shown in the following table:

| Q3 to Q5 | Direction of Counting | Interrupt at |
|----------|-----------------------|--------------|
| "0" | Counting up | Rising edge, counter output |
| "1" | Counting up | Falling edge, counter output (The counter is simultaneously reloaded with the contents of the load register.) |
| "0" | Counting down | Rising edge, counter output (The counter is simultaneously reloaded with the contents of the load register.) |
| "1" | Counting down | Falling edge, counter output |

The behavior of the counter outputs when the comparator function is enabled is explained in section 1.9. The command "parameterize counter" (global register) or the command "mask interrupt" IM causes the interrupt polarity register to be transferred to the module.

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IP | 0 | 0 | A5 | A4 | A3 | A2 | A1 | 0 | 0 | 0 | T5 | T4 | T3 | T2 | T1 | 0 |

└── Outputs 1 to 5              └── Gate inputs 1 to 5

## Overview of the Interrupt Sources

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Output, counter | 1 | Q1 | 1 | Q1 | SA=1 / SA=0 | *) | Q1 | |
| | 2 | Q2 | 1 | Q2 | | *) | Q2 | |
| | 3 | Q3 | 1 | Q3 | | *) | Q3 | |
| | 4 | Q4 | 1 | Q4 | | *) | Q4 | |
| | 5 | Q5 | 1 | Q5 | | *) | Q5 | |
| | 6 | Q6 | | | | *) | Q6 | |
| | 7 | Q7 | | | | *) | Q7 | |
| Gate, counter | 1 | T1 | 1 | T1 | | *) | T1 | |
| | 2 | T2 | 1 | T2 | | *) | T2 | |
| | 3 | T3 | 1 | T3 | | *) | T3 | |
| | 4 | T4 | 1 | T4 | | *) | T4 | |
| | 5 | T5 | 1 | T5 | | *) | T5 | |
| Reset, counter | 6 | R6 | | | | *) | R6 | |
| Reset, counter | 7 | R7 | | | SA=1 | *) | R7 | SA=0 |

≥1 → Interrupt to the S5

*) Trigger command list

Error message
Entry in the error
information register — ER

IFR          IPR          AFR

## 3.3.6    Interrupt Filter Register (AFR)

**IP 242B**

**B** The interrupt filter register is located in data word DW14 of the parameterization data block.

Interrupt filter register AFR is used for a specific "forwarding" of interrupts on the module to the programmable controller. **An interrupt to the S5 is not triggered for events unless the corresponding bit in the IFR and the AFR is set to "1".**

When an interrupt is only enabled in the IFR and not in the AFR, an entry is not made in the IIR and an interrupt is not triggered although a command list on the module is executed.

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AFR | A7 | A6 | A5 | A4 | A3 | A2 | A1 | 0 | R7 | R6 | T5 | T4 | T3 | T2 | T1 | 0 |

          Interrupt for gate interrupts 1 to 5

          Interrupt for interrupt at external reset for counters 6 and 7

Interrupt for output interrupts 1 to 7

0: Disabled
1: Enabled

- The interrupt filter register takes effect when the SA bit in the IFR = "0" (i.e., group interrupt disabled).

- **All** interrupts bypass the AFR when the SA bit in the IFR = "1". In case of interrupts, the IP 242B behaves as the IP 242A when the AFR is disabled.

## 3.3.7 Difference Register (DR)

**IP 242A**

The difference register is located in data double word DD9 of the parameterization data block.

**A**

The module uses the command "difference generation" with the appropriate counter bits to generate the difference between the states of the specified counters. The difference is then entered in the parameterization data block.

When "BCD counting mode" is selected, a BCD offset is not performed for the difference register (i.e., the use of this mode is only practical in "binary counting mode".

Bit 31                           Bit 16

$2^{23}$   $2^{22}$   $2^{21}$   $2^{20}$   $2^{19}$   $2^{18}$   $2^{17}$   $2^{16}$

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sign | | | | | | | | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

DD9

$2^{15}$   $2^{14}$   $2^{13}$   $2^{12}$   $2^{11}$   $2^{10}$   $2^{9}$   $2^{8}$   $2^{7}$   $2^{6}$   $2^{5}$   $2^{4}$   $2^{3}$   $2^{2}$   $2^{1}$   $2^{0}$

Bit 15                           Bit 0

This register is no longer required on the IP 242B since the calculation functions and result register have been added.

**B**

## 3.3.8 Version Number Register (VNR)

**IP 242A/242B**

The version number register is located in data word DW11 of the parameterization data block.

**A**   **B**

The firmware status of the module is entered in KY format in the version number register.

The command "read module" (global register) causes the firmware status of the module to be taken over in the data block.

Example: Firmware Status V1.1

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

DW11

                   1                                1

### 3.3.9    FB Version Identifier (FBV)

IP 242B

**B**  The FB version identifier is located in data word DW12 of the parameterization data block.

The standard function block enters an identifier for its own version in the FBV. The current function block (FB 183 and FB 184) has the version "1".

The IP 242B clears this register during startup and scans it each time a control word is processed. The control word is ignored when the identifier is not equal to "1". An error message is not entered since an incorrect function block would not be able to evaluate it anyway.

### 3.3.10   Pulse Duration Register (PDR)

IP 242B

**B**  The pulse duration register is located in data word DW15 of the parameterization data block.

The duration of the output pulse is specified in KT format in this register for software comparators for counters 3 to 7. The set time applies to **all** channels for which the "constant pulse duration" output signal form is parameterized ($\rightarrow$ section 1.8).

Value range:

| | |
|---|---|
| 000.0 ⎫ 001.0 ⎭ | 0 to 10 msec minimum value |
| 002.0 | 10 to 20 msec |
| . . . . | |
| 001.1 | 99 to 100 msec |
| . . . . | |
| 999.5 | 999 000 000 msec maximum value |

The output signal form is set in the counter mode register of the applicable counter. The "constant pulse duration" output signal form cannot be used in operating modes G, H, I, J, K, L, S, V and X of counters 3 to 5.

## 3.4    Description of the Counter Registers
### 3.4.1   Counter Mode Register (CMR)

**IP 242A/242B**

There is one counter mode register for every counter. Register assignment for counters 1 to 5 differs from register assignment for counters 6 and 7.

**A**  **B**

The command "parameterize counter" (global register) causes the counter mode register to be transferred to the module.

**Counter Mode Register for Counters 1 to 5**

CMR1: DW16 of the parameterization data block
CMR2:  DW30 of the parameterization data block
CMR3:  DW44 of the parameterization data block
CMR4:  DW58 of the parameterization data block
CMR5:  DW72 of the parameterization data block

The counter mode register controls the operation mode (section 9) for the applicable counter. Control of the operation mode includes the following items:

        – Counting mode
        – Counting pulse source
        – Gate selection
        – Direction of counting
        – Active counting pulse edge
        – Output signal

**Counting pulse source**

| Counter output | n–1 | = | 0000 |
|---|---|---|---|
| Counter input | 1 | = | 0001 |
| Counter input | 2 | = | 0010 |
| Counter input | 3 | = | 0011 |
| Counter input | 4 | = | 0100 |
| Counter input | 5 | = | 0101 |
| Gate | 1 | = | 0110 |
| Gate | 2 | = | 0111 |
| Gate | 3 | = | 1000 |
| Gate | 4 | = | 1001 |
| Gate | 5 | = | 1010 |
| Frequency | F1 | = | 1011 |
| Frequency | F2 | = | 1100 |
| Frequency | F3 | = | 1101 |
| Frequency | F4 | = | 1110 |
| Frequency | F5 | = | 1111 |

See section 9.1 for **operation modes**.

**Counting mode**
0 = Binary
1 = BCD

**Counting direction**
1 = Up
0 = Down

| CM | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

**Counting pulse edge**
0 = Rising
1 = Falling

**Output signal form** [2]

**Gate selection**

| 000 | Without gate control |
|---|---|
| 001 | Counter output n–1, high active (internal link) |
| 010 | High level, gate n+1 [1] |
| 011 | High level, gate n–1 [1] |
| 100 | High level, gate n [1] |
| 101 | Low level, gate n |
| 110 | Rising edge, gate n |
| 111 | Falling edge, gate n |

**IP 242A, IP 242B** (Counters 1 to 5 without comparator when counter has expired)

| 000 | Disabled (low level) |
|---|---|
| 001 | Pulse, active high |
| 010 | Square wave, start low |
| 011 | Square wave, start high |
| 100 | Illegal |
| 101 | Pulse, active low |
| 110 | Illegal |
| 111 | Illegal |

**IP 242A, IP 242B** (Counters 1 and 2 for enabled hardware comparator)

| 000 | Disabled (low level) |
|---|---|
| 001 | Pulse, active high |
| 010 | Pulse, active high |
| 011 | Pulse, active high |
| 100 | Illegal |
| 101 | Pulse, active low |
| 110 | Illegal |
| 111 | Illegal |

**IP 242A** (Counters 3 to 5 for enabled software comparator)

| 000 | High for counter value $\geq$ interrupt value |
|---|---|
| 001 | High for counter value $\geq$ interrupt value |
| 010 | High for counter value $\geq$ interrupt value |
| 011 | High for counter value $\geq$ interrupt value |
| 100 | Illegal |
| 101 | High for counter value $\geq$ interrupt value |
| 110 | Illegal |
| 111 | Illegal |

**IP 242B** (Counters 3 to 5 for enabled software comparator)

| 000 | High for counter value $\geq$ interrupt value |
|---|---|
| 001 | Pulse, active high for interrupt value (Length can be parameterized in PDR.) |
| 010 | Illegal |
| 011 | Illegal |
| 100 | High for counter value $\geq$ interrupt value |
| 101 | Pulse, active high for interrupt value (Length can be parameterized in PDR.) |
| 110 | Illegal |
| 111 | Illegal |

[1] Gate modes 3 and 4 use only these settings. (See section 1.7.)
[2] See section 1.9 if comparator function is enabled.

**Counter Mode Register for Counters 6 and 7**

CMR 6: DW86 of the parameterization data block
CMR 7: DW103 of the parameterization data block

The counter mode register controls the following items for the applicable counter:

    – Reset input (counter reset)
    – Operation mode
    – Output signal



**Output signal form** [1]

**IP 242A/242B**

| 000 | High for counter value $\geq$ interrupt value |
| 001 | Pulse, active high for interrupt value |

**IP 242B**
(Depending on output signal bit 4)

| 010 | (Length can be parameterized in PDR.) |
| 011 | Illegal |
| 100 | Illegal |
| 101 | High for counter value $\geq$ interrupt value |
| 110 | Pulse, active low for interrupt value |
| 111 | (Length can be parameterized in PDR.) |
|     | Illegal |
|     | Illegal |

15        8  7        0

CM | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | 0 | 0 | 0 |

**Counter reset**
0 = disable
1 = Enable

**Output signal**
0 = Disabled (low level)
1 = Active (see section 1.9 for comparator function)

**Operation mode (see section 9.2)**

000  Illegal
001  Single–edge evaluation with rising edge at input A
    Input B determines the direction of counting as follows:
        High level = counting down
        Low level = counting up
010  Single–edge evaluation with rising edge at input B
    Input A determines the direction of counting as follows:
        High level = counting up
        Low level = counting down
011  Dual–edge evaluation at input A
100  Dual–edge evaluation at input B
101  Quadruple–edge evaluation at all edges
110  Gate time measurement
111  Frequency measurement    } IP 242B only

> ⚠ 1) The output signals are inverted when negative interrupt values are specified
> ($\rightarrow$ section 1.9.3).

## 3.4.2  Load Register (LR)

**IP 242A/242B**

**A  B**

LR1: DW17 of the parameterization data block
LR2: DW31 of the parameterization data block
LR3: DW45 of the parameterization data block
LR4: DW59 of the parameterization data block
LR5: DW73 of the parameterization data block
LR6: DW87 and DW88 of the parameterization data block
LR7: DW104 and DW105 of the parameterization data block

The previous counter values (load values) are stored in the load register. There is one load register for each counter.

Sixteen–bit load registers are provided for counters 1 to 5. Thirty–two–bit registers are provided for counters 6 and 7.

The commands "load counter" (for counters 1 to 5 only when the IP 242A is used), "load and start counter", or "parameterize counter" cause the contents of the load register to be transferred to the module.

**A  B**

**Load Register for Counters 1 to 5**

Bit 15 ... Bit 0

| L | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^9$ | $2^8$ | $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
| | 32768 | 16384 | 8192 | 4096 | 2048 | 1024 | 512 | 256 | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

**A**

**Load Register for Counters 6 and 7 (IP 242A)**

Bit 31 ... Bit 16

| | | | | | | | | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sign | | | | | | | | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |

| L | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^9$ | $2^8$ | $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |

Bit 15 ... Bit 0

**B**

**Load Register for Counters 6 and 7 (IP 242B)**

Bit 31 ... Bit 16

| | $2^{31}$ | $2^{30}$ | $2^{29}$ | $2^{28}$ | $2^{27}$ | $2^{26}$ | $2^{25}$ | $2^{24}$ | $2^{23}$ | $2^{22}$ | $2^{21}$ | $2^{20}$ | $2^{19}$ | $2^{18}$ | $2^{17}$ | $2^{16}$ |
|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |

| L | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| | $2^{15}$ | $2^{14}$ | $2^{13}$ | $2^{12}$ | $2^{11}$ | $2^{10}$ | $2^9$ | $2^8$ | $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |

Bit 15 ... Bit 0

**Control Word Accesses to the Load and Hold Registers**

**Counters 1 to 5**

**Dual Port RAM CPR**

**Parameterization Data Block**

PA, PO,
LD, LS,
terminal
count (TC)

Counter register in AM 9513

PA, PO
LV, LD, LS

**Load register**

PA, PO
LV, LD, LS

(BS or RS)

**Load register**

Internal load register

$TC^1$

PA, PO
LV, LD, LS

(BS or RS)

PA, PO
$(LV, LD, LS)^1$

**Internal hold register**

**Hold register**

LE, SL, CO

**Hold register**

A

SV,
LE,
SL,
active gate edge [2]

LE,
SL,
CO

BL   A

B

**Counter values, cyclic ZSZ**

**Counter values, cyclic ZSZ**

For every FB call

**Control words for command lists + FB accesses**

**Control words for FB accesses only**

Paths not identified as **A** or **B** apply to both the IP 242A and IP 242! Remember that individual commands such as PO or LV are only available with the IP 242B.

For the IP 242B, RL fetches all registers from the dual port RAM to the parameterization data block.

All abbreviations identify control words (exception: TC, → section 9.1.2).

1) For operating modes G, H, I, J, K, L, S and V

2) For operating modes N, O, Q, R and X

### 3.4.3    Hold Register (HR)

**IP 242A/242B**

HR1: DW18 of the parameterization data block
HR2: DW32 of the parameterization data block
HR3: DW46 of the parameterization data block
HR4: DW60 of the parameterization data block
HR5: DW74 of the parameterization data block
HR6: DW89 and DW90 of the parameterization data block
HR7: DW106 and DW107 of the parameterization data block

The hold registers have the same format as the load registers.

**A** On the IP242A, the counter value is stored in the hold registers via control words "stop and read counter" (counters 1 to 5 only), "copy counter " (counters 1 to 5 only), and "read counter".

**B** On the IP 242B, the counter value is entered in the **counter value register** with control words SL, CO and LE.

The command "parameterize counter (PA/PO)" causes the contents of the hold registers to be transferred to the module.

The **hold register** and the load register are used alternately **as load registers** in operation modes G, H, I, J, K ,L, S, and V for counters 1 to 5. In these operation modes, the hold register must be preassigned before parameterization of the counter. In addition, the hold register is transferred to the module for LD, LS and LV.

An **internal hold register** is used **as intermediate storage** for operation modes N, O, Q, R, and X. In this case, the command "copy counter" can be used to transfer the contents of this register to the hold register of the dual port RAM (IP 242A) or counter value register (IP 242B).

## 3.4.4   Interrupt Register (AR)

IP 242A/242B

AR1: DW19 of the parameterization data block
AR2: DW33 of the parameterization data block
AR3: DW47 of the parameterization data block
AR4: DW61 of the parameterization data block
AR5: DW75 of the parameterization data block
AR6: DW91 and DW92 of the parameterization data block
AR7: DW108 and DW109 of the parameterization data block

**A**   **B**

The interrupt registers have the same format as the load registers.

When the comparator function is enabled, the value in the interrupt register is compared to the counter. The output is set and, if enabled in IFR, an interrupt is triggered if these two values are equal. (See also section 1.9.)

**Interrupt Register for Counters 1 to 5**

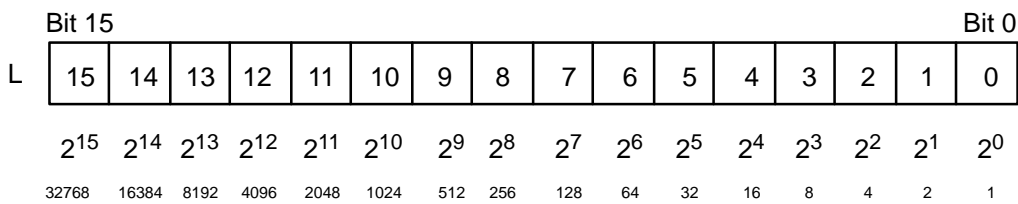An interrupt value of "0" is illegal. If an interrupt value of "0" is entered in the interrupt register, a message is generated in the error information register when the counter is parameterized. (See section 6.1.22.) An interrupt value of "0" is also illegal if the comparator function in the master mode register has not been enabled.

**A**   **B**

**Interrupt Register for Counters 6 and 7**

An interrupt value of "0" is permitted for counters 6 and 7. This interrupt value is counted as belonging to the positive axis.

**A**   **B**

## 3.4.5   Cyclic Counter Values (ZSZ)

IP 242B

ZSZ1: DW123 of the parameterization data block
ZSZ2: DW124 of the parameterization data block
ZSZ3: DW125 of the parameterization data block
ZSZ4: DW126 of the parameterization data block
ZSZ5: DW127 of the parameterization data block
ZSZ6: DW128 and DW129 in the parameterization data block
ZSZ7: DW130 and DW131 in the parameterization data block

**B**

The CPU on the IP 242B reads the counter values of counters 1 to 7 cyclically. The counter values of the last measuring cycle are entered automatically for every control word (even when the control word is invalid).

For counters 1 to 5, the value specified in the hold register during the last parameterization is entered for operation modes G, H, I, J, K, L, S and V, and the counter value at the last active gate edge for operation modes N, O, Q, R and X.

## 3.5 Description of the Registers for the Calculation Functions
## 3.5.1 Result Register (ERG)

**IP 242B**

**B** The result registers are located in the following data words of the parameterization data block.

ERG1: DW132 and DW133
ERG2: DW134 and DW135
ERG3: DW136 and DW137
ERG4: DW138 and DW139
ERG5: DW140 and DW141
ERG6: DW142 and DW143
ERG7: DW144 and DW145

The results of the calculation functions are stored in these registers. The length is 32 bits. The calculation results can be assigned to any result registers (can be selected as desired with the transfer command).

If the measured value memory for a calculation function is enabled in the directory (DM), a transfer is also automatically performed to the measured value memory when the result is transferred to the result register.

The result registers are automatically stored in the dual port RAM (just as the cyclic counter values and the counter status registers) each time an FB is called.

## 3.5.2 Constant Register (KON)

**IP 242B**

**B** The constant registers are located in the following data words in the parameterization data block.

KON0: DW146 and DW147
KON1: DW148 and DW149
KON2: DW150 and DW151
  ⋮
KON14: DW174 and DW175
KON15: DW176 and DW177

Constants can be stored in these registers for the calculations. The length is 32 bits.

The constant registers are only present once on the module as a block. They are handled (as a block) like the global registers.

The constant registers are automatically accepted with "parameterize counters" (PA). The "write constant registers" (KS) command can be used to condition the constants separately.

---

## 3.6 Description of the Registers for Additional Command Lists
## 3.6.1 Directory of the Additional Command Lists (DZB)

**IP 242B**

---

DZB: The directory of the additional command lists is located in DW178 to DW185 of the parameterization data block.

**B**

The 61 data words available for the additional command lists are used to specify whether or not an additional command list is to begin, and, if so, at which data word. The length of the list is determined by the difference to the starting address of the next list. The data have KY format and the lefthand number is ignored.

A list is disabled by entering KY 0,0 and enabled by entering the starting address. The length of the list is determined automatically by the difference to the starting address of the next enabled list.

When the addresses are invalid, one of the error messages KH51 to KH57 is output during parameteri–zation ($\rightarrow$ section 3.8.3) and parameterization is terminated.

**Setup of the Directory**

**Start address**
The DW number of the first word of a list is entered here. List 1 always begins at DW 186.

XX: Disregard

**Explanation of the Example**

| | | | | List Size | Start of List | Last DW |
|---|---|---|---|---|---|---|
| DW178 | XX | 186 | List 1: | 10 DW | DW186 | DW 195 |
| DW179 | XX | 196 | List 2: | 28 DW | DW196 | DW 223 |
| DW180 | XX | 224 | List 3: | 6 DW | DW224 | DW 229 |
| DW181 | XX | 00 | List 4: | 0 DW | None | – |
| DW182 | XX | 230 | List 5: | 6 DW | DW230 | DW 235 |
| DW183 | XX | 236 | List 6: | 4 DW | DW236 | DW 239 |
| DW184 | XX | 00 | List 7: | 0 DW | None | – |
| DW185 | 00 | 240 | First free data word: DW240 here | | | |

**Error number**
The firmware stores the error number of an erroneous directory entry here.

> The data word number corresponding to the first free data word after all additional command lists must be entered in DW185 (DW247 maximum).

**B**

## 3.6.2    Additional Command Lists (ZB)

**IP 242B**

**B** ZB: The additional command lists are located in DW186 to DW246 of the parameterization data block.

Up to seven lists of variable lengths are available. The commands are stored consecutively in the lists without breaks. Since the additional command lists can be assigned to the interrupt command lists as desired, an additional command list does not have to be assigned to each interrupt command list.

The structure of the lists is stored in the "directory for additional command lists". Unassigned lists must be disabled with the address entry KY 0,0.

"PA (global register)" is used to transfer the ZB to the module. The "process command list" control word is used to process the additional command lists.

## 3.7      Description of the Registers for Measured Values
## 3.7.1    Directory of the Measured Value Memory (DM)

**IP 242B**

**B** DM: The directory of the measured value memory is located in DW247 to DW254 of the parameterization data block.

The total of 100 x 2 data words available for the measured value series are used to specify the data word (DW) at which the block is to begin. Assignment of the result registers to the measured value series is fixed ($\rightarrow$ section 3.7.2).

"Rising" DW addresses must be entered and separation of the measured value series is not permitted.

A measured value block is disabled by entering KY 0,0, and enabled by entering the starting address. The length of the block is automatically determined by the difference to the start address of the next enabled block. The firmware checks the entered values for validity during parameterization of the global registers. When an error is detected, parameterization is terminated and one of the error messages (KH41 to KH46) is output ($\rightarrow$ section 3.8.3).

The values of the filling state indicator present in the parameterization data block are ignored during parameterization ("parameterize counter PA"). In addition, **all** measured value memories and the filling state indicators are reset on the module.

When "parameterize without command lists" (PO) is used, directory and measured value memory remain unchanged.

**Setup of the Directory**

**Filling state indicator**
The firmware indicates the next free space after the last acquired measured values (255 = overflow of the block). Entries made during parameterization are ignored.

**Start address**
The DW number of the first word of the block is entered here (e.g., DW30 for 2nd block).
Block 1 always starts at DW16

Example:

**Explanation of the Example**

| | | | Block Size | Start of Block |
|---|---|---|---|---|
| DW247 | 16 | 16 | Block 1: 7 x 2 DW | Start at DW16 |
| DW248 | 34 | 30 | Block 2: 5 x 2 DW | Start at DW30 |
| DW249 | 60 | 40 | Block 3: 10 x 2 DW | Start at DW40. Block is full! |
| DW250 | 00 | 00 | Block 4: 0 x 2 DW | No start |
| DW251 | 255 | 60 | Block 5: 10 x 2 DW | Start at DW60<br>Overflow indicated (filling state indicator = 255). |
| DW252 | 96 | 80 | Block 6: 15 x 2 DW | Start at DW80 |
| DW253 | 00 | 00 | Block 7: 0 x 2 DW | No start |
| DW254 | 00 | 110 | | Block end; DW110 here<br>(Corresponds to the first free data word after all measured value blocks) |

**Error number**
The firmware stores the error number of an erroneous directory entry here.

The filling state indicator is updated for all blocks when the measured values are read.

DW255 contains the indicator to the measured value data block ($\rightarrow$ section 14.6.1).
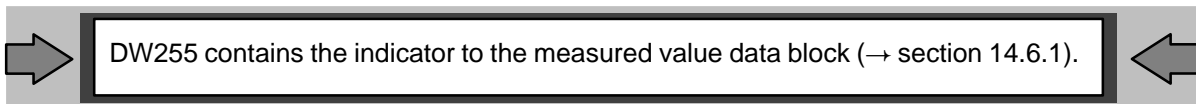
### 3.7.2 Measured Value Memory (M)

**IP 242B**

**B** The measured value memory is located in DW16 to DW215 of the measured value data block.

The structure and size of the individual measured value memories are specified to your requirements when you parameterize the global registers in the directory for measured value memories. Each individual measured value occupies 2 data words.

The firmware manages the blocks on a "first–in, first–out" basis (FIFO). Starting with the lowest address of the block, the contents of the corresponding result registers are entered. When a block is full, this is reported in the counter status register (ZSR) by setting the corresponding bits "M1" to "M7". If the control word "read and reset measured values" (MR) is executed, the corresponding status bit in ZSR and the measured value series are deleted and the filling state indicator is reset.

The oldest values are lost if new measured values are entered before "MR" was executed.

Thus, the entry of the measured values (results) in a measured value series also allows you to follow the past values of a result register (trace function). When malfunctions occur, the causes can be easily determined by using the trace function.

The measured value series are permanently allocated to the result registers.

| Result register 1 | $\rightarrow$ | Measured value series 1 |
| Result register 2 | $\rightarrow$ | Measured value series 2 |
| Result register 3 | $\rightarrow$ | Measured value series 3 |
| Result register 4 | $\rightarrow$ | Measured value series 4 |
| Result register 5 | $\rightarrow$ | Measured value series 5 |
| Result register 6 | $\rightarrow$ | Measured value series 6 |
| Result register 7 | $\rightarrow$ | Measured value series 7 |

If the "read measured values" (ML) function is executed, the status bits, filling state indicator and measured value series remain unchanged.

## 3.8 Description of the Information Registers
## 3.8.1 Counter Status Register (ZSR)

**IP 242B**

The counter status register are used for reporting module states to the S5 CPU. They can be read by the PLC each time a control word is accessed, and are updated each time a function block is called.

**B**

ZSR1: DW120 in the parameterization data block

| 15 | | | | | | | | 8 | 7 | | | | | | | 0 |
|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | DS3 | DS2 | DS1 | DS0 | ND7 | ND6 | KL5 | KL4 | KL3 | KL2 | KL1 | 0 |

Operating mode of counters 1 to 5 does not allow cyclic reading.

Zero crossing counters 6 and 7
The bit is set the first time the function block is called after zero crossing by counter 6 or 7. It is then reset again afterwards.

Currently selected data record in the EEPROM (data record number)
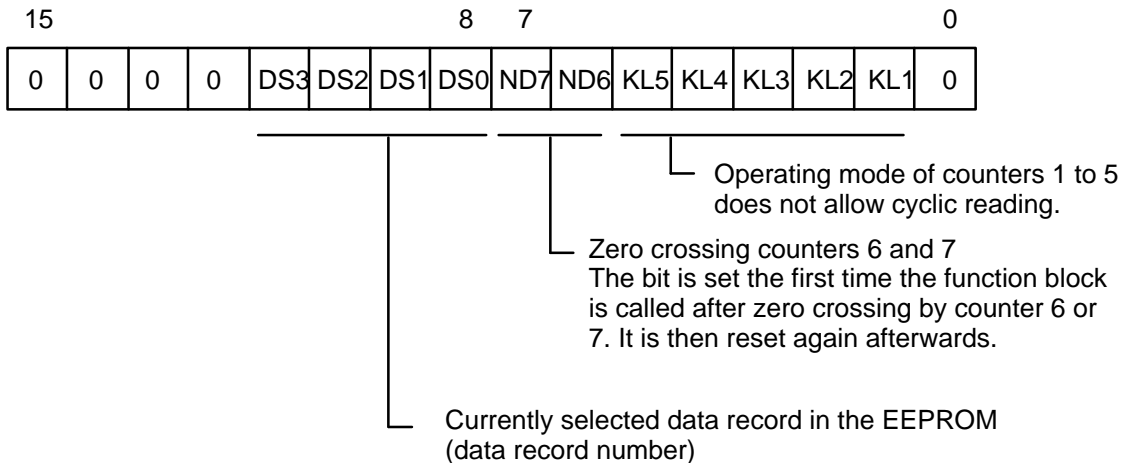
ZSR2: DW121 in the parameterization data block

| 15 | | | | | | | 8 | 7 | | | | | | | | 0 |
|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BL7 | BL6 | BL5 | BL4 | BL3 | BL2 | BL1 | BLD | M7 | M6 | M5 | M4 | M3 | M2 | M1 | PA |

Module is parameterized.
The bit is set as soon as al least one counter and the global registers are correctly parameterized. The bit is deleted after startup.

Measured value memories 7 to 1 are full.
The bit is set as soon as a measured value memory is completely filled. It is reset after the "MR", "PA", "PZ", "GR" or "TF" commands.

Directly triggered command list is being executed.
The bit remains set as long as a command list triggered by the S5 with the "process command list" (BB) control word is being executed.

Command list (7 to 1) is being executed.
The bit remains set for the corresponding counter as long as the gate or counter interrupt command list of the corresponding counter is being processed with all "subcommand lists" triggered by the "process command list" (BB) control word. When command list processing is caught in a continuous loop due to incorrect parameterization, this is indicated by a continuously set BL bit.

ZSR3: DW122 in the parameterization data block

| 15 | | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ZL7 | ZL6 | ZL5 | ZL4 | ZL3 | ZL2 | ZL1 | 0 |
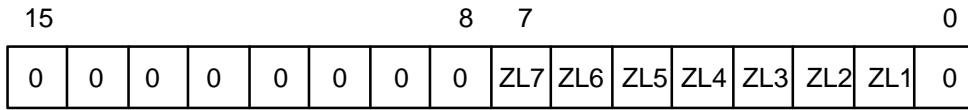
Counter (7 to 1) is running.
The bit is set for the corresponding counter when a change
has occurred in the counter value from one S5 access to the
next during internal, cyclic reading of the counter values. The
bit is reset after the access. In operating modes G, H, I, J, K,
L, S and V, the bit is already set after a "start counter" or
"load and start counter", and reset after a "stop counter" or
"stop and read counter".

## 3.8.2    Interrupt Information Register (IIR)

IP 242A/242B

**A** The command "BEF = IN" causes the interrupt information register to be read on the IP 242A out of the
dual port RAM and stored in parameter IIR of the standard function blocks (FB 178/179/182).

**B** The interrupt information register is read on the IP 242B after FB 184 is called, and stored in the IIR param-
eter.

The processing of an interrupt is reported to the S5 central processing unit by triggering an interrupt or
process interrupt if the SA bit in the interrupt enable register is set.

The interrupt information register supplies information about all interrupts which occurred since the last
scan.

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| II | A7 | A6 | A5 | A4 | A3 | A2 | A1 | SA | R7 | R6 | T5 | T4 | T3 | T2 | T1 | 0 |

Gate inputs 1 to 5

Reset inputs 6 and 7

Group interrupt bit

Outputs 1 to 7
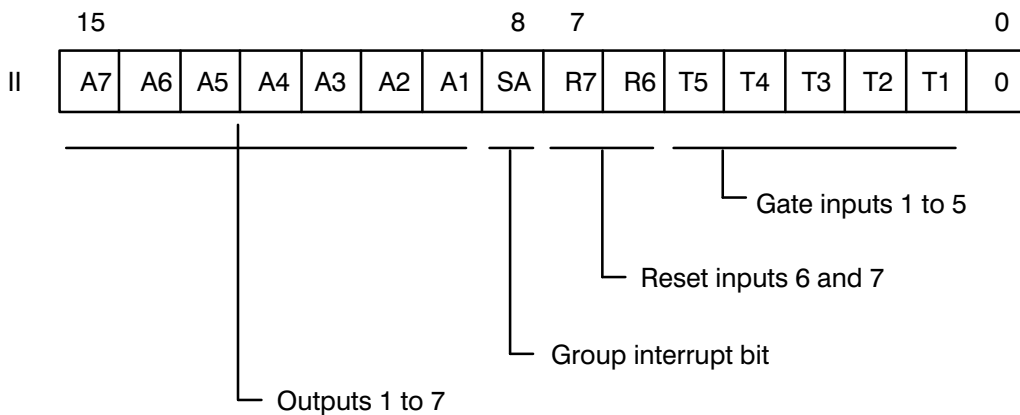
The group interrupt bit is also set when at least one interrupt from the gate, counter reset , or output oc-
curs.

During processing of the interrupt, the interrupt request of the counter module is reset after evaluation of
the interrupt information register.

## 3.8.3 Error Information Register (ERR)

IP 242A/242B

The error information register in the dual port RAM is read for every command and stored in the ERR parameter of the standard function block.

A number is stored in the ERR parameter for the respective error which occurred last. A value of zero at the ERR parameter indicates that no error has occurred.

IP 242A: The signal status of **bit 3** of parameter **MELD** is set to "**1**" when an error is determined (parameter ERR >< zero).

**A**

IP 242B: The signal status of **bit 5** of parameter **MELD** is set to "**1**" when an error is determined (parameter ERR >< zero).

**B**

The assignment of parameter ERR occurs at two different times: before writing a control word and after writing a control word.

When parameter ERR is assigned **after writing** a control word, this is usually due to an error which occurred during the writing of the control word. Processing of the control word in which the error occurred is interrupted.

When parameter ERR is assigned **before writing** the control word, the setting of an error number is usually caused by a module malfunction (e.g., short circuit at the outputs). The control word is not transferred.

List of Possible Errors:

| No. | Type of Error | Short Description |
|------|----------------|-------------------|
| KH01 | Control word error | Undefined control word in control word register |
| KH02 | Control word error | Illegal control word for counter 6/7 |
| KH03 | Selection error | No counter selected for control word |
| KH04 | Interrupt error | Interrupt register was loaded with "0" |
| KH05 | RAM error | Error during test function, RAM |
| KH06 | Dual port RAM error | Error during test function, dual port RAM |
| KH07 | EPROM error | Error during test function, EPROM |
| KH08 | EEPROM error | Error during test function, EEPROM |
| KH09 | Counter error | Error during test function, AM 9513A counter block |
| KH0A | Short circuit | Output driver overloaded and disabled |
| KH0B | Module error | Faulty module operation |
| KH0C | Module error | Faulty module operation (internal watchdog error) |
| KH0D | Interrupt overflow | Interrupt frequency is too high. (Module is no longer able to process the interrupts.) |

**A** **B**

|  | No. | Type of Error | Short Description |
|---|---|---|---|
| **B** | KH10 | | Unidentifiable command in the interrupt or additional command list. (The number of the erroneous data word is stored in the error address list command list FAB.) |
| **A** | KH11 | Gate 1 | Unidentifiable command in command list, gate 1 |
| | KH12 | Gate 2 | Unidentifiable command in command list, gate 2 |
| | KH13 | Gate 3 | Unidentifiable command in command list, gate 3 |
| | KH14 | Gate 4 | Unidentifiable command in command list, gate 4 |
| | KH15 | Gate 5 | Unidentifiable command in command list, gate 5 |
| | KH16 | Reset 6 | Unidentifiable command in command list, reset 6 |
| | KH17 | Reset 7 | Unidentifiable command in command list, reset 7 |
| | KH18 | Output 1 | Unidentifiable command in command list, output 1 |
| | KH19 | Output 2 | Unidentifiable command in command list, output 2 |
| | KH1A | Output 3 | Unidentifiable command in command list, output 3 |
| | KH1B | Output 4 | Unidentifiable command in command list, output 4 |
| | KH1C | Output 5 | Unidentifiable command in command list, output 5 |
| | KH1D | Output 6 | Unidentifiable command in command list, output 6 |
| | KH1E | Output 7 | Unidentifiable command in command list, output 7 |
| **A** **B** | KH21 | Counter mode 1 | Error in counter mode register 1 |
| | KH22 | Counter mode 2 | Error in counter mode register 2 |
| | KH23 | Counter mode 3 | Error in counter mode register 3 |
| | KH24 | Counter mode 4 | Error in counter mode register 4 |
| | KH25 | Counter mode 5 | Error in counter mode register 5 |
| | KH26 | Counter mode 6 | Error in counter mode register 6 |
| | KH27 | Counter mode 7 | Error in counter mode register 7 |
| **B** | KH30 | | Calculation overflow ($> 2^{31}-1$) |
| | KH31 | | Calculation underflow ($< 2^{31}$) |
| | KH32 | | Division by 0 |
| | KH33 | | Transfer value too large for counters 1 to 5 ($> 2^{16}-1$) |
| | KH34 | | Transfer value negative for counters 1 to 5 ($< 0$) |
| | KH35 | | Load from unspecified register |
| | KH36 | | Transfer to unspecified register |
| | KH37 | | Jump is not within a command list |
| | KH41 | | Directory for measured value series 1 invalid |
| | KH42 | | Directory for measured value series 2 invalid |
| | KH43 | | Directory for measured value series 3 invalid |
| | KH44 | | Directory for measured value series 4 invalid |
| | KH45 | | Directory for measured value series 5 invalid |
| | KH46 | | Directory for measured value series 6 invalid |
| | KH47 | | Directory for measured value series 7 invalid |
| | KH50 | | EEPROM data record undefined/data record invalid |
| | KH51 | | Directory for additional command list 1 invalid |
| | KH52 | | Directory for additional command list 2 invalid |
| | KH53 | | Directory for additional command list 3 invalid |
| | KH54 | | Directory for additional command list 4 invalid |
| | KH55 | | Directory for additional command list 5 invalid |
| | KH56 | | Directory for additional command list 6 invalid |
| | KH57 | | Directory for additional command list 7 invalid |

Error numbers KH01 to KH04 and from KH10 on indicate a parameterization error.

## 3.8.4    Error Address Command List (FAB)

**IP 242B**

The additional command list is terminated for all run time errors in command list processing, and for erroneous control words in the command list during parameterization. The number of the erroneous data word is stored in the error address command list (FAB). The FAB is read each time a function block is called, and stored in the ERR parameter of the standard function block.

**B**

The ERR parameter ($\rightarrow$ section 3.8.3) has been expanded on the IP 242B from byte to word format for the FAB.

```
  15                          8  7                          0
 ┌────────────────────────────┬────────────────────────────┐
 │            FAB             │            ERR             │
 └────────────────────────────┴────────────────────────────┘
```

---
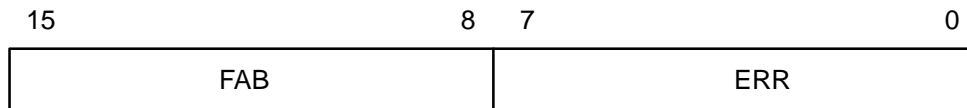
## 3.9    Basic Settings

---

**A** **B** The command "take over basic settings" GR causes the settings listed here to be taken over from the EPROM in the respective registers and transferred to the parameterization data block as follows:

– Counters 1 to 5

| | |
|---|---|
| Operation mode D: | Frequency generator without hardware gate control |
| Output signal: | Square wave, start high |
| Counting mode: | Binary |
| Prescaler register: | 5 |
| Counter input clock pulse: | 200 kHz |
| Load register: | Counter 1: +100 |
| | Counter 2: +200 |
| | Counter 3: +300 |
| | Counter 4: +400 |
| | Counter 5: +500 |

Resulting output frequencies:

    Counter 1: 1.0  kHz
    Counter 2: 0.5  kHz
    Counter 3: 0.33 kHz
    Counter 4: 0.25 kHz
    Counter 5: 0.2  kHz

| | |
|---|---|
| Interrupt  register: | Counter 1: +1 |
| | Counter 2: +1 |
| | Counter 3: +1 |
| | Counter 4: +1 |
| | Counter 5: +1 |

The comparators are disabled.

– Counters 6 and 7

| | |
|---|---|
| Operation mode: | Quadrature mode, quadruple–edge |
| Load register: | +100 |
| Interrupt register: | +1000 |

**Interrupt Enable Register:**

– Counter interrupts:                           Disabled
– Gate interrupts:                              Disabled
– Reset interrupts:                             Disabled
– Group interrupt, S5 counter events:          Disabled     (SA bit = 0)
– Group interrupt, S5 error messages:          Disabled     (ER bit = 0)

**Interrupt Polarity Register:**

Contents: 000H, interrupt at rising edge

**Interrupt Filter Register (only IP 242B):**

Empty (0000H)

**B**

**Interrupt Command Lists:**

Empty (0000H)

**Gate Control Register:**

Contents:             0000H

Gate signal:          Level control at start input

Gate signal selection:   External gate input

**Additional Command Lists (only IP 242B):**

Empty (0000H)

**B**

**Result Register (only IP 242B):**

Contents: 0000H

**B**

**Constant Register (only IP 242B):**

Contents: 0000H

**B**

**Measured Value Memory (only IP 242B):**

Empty (0000H)

**B**

> After activation of the basic settings, the output drivers are still disabled, and the counters have not been started. The command "enable the outputs" enables the output drivers. The command "load and start counters" activates the counters.

# 4    Guidelines for Interrupt Processing

## 4.1 What Is an Interrupt ?

Interrupts on the S5 CPU are triggered with the IP 242A/242B module.

**Explanation of Terms:**

An interrupt is the triggering of a break in a program on the S5 central processing unit by an I/O module. Interrupts are triggered on programmable controllers by input byte IB 0 (process interrupts) or by interrupt lines (interrupts).

| Programmable Controller | Interrupt Processing via: | Interrupt Points |
|---|---|---|
| S5–115U | Interrupt lines | Statement limits |
| S5–135U | Interrupt lines | Block limits; in data block DXO, adjustable to statement limits |
| S5–150U/S | Input byte IB 0 | Block limits |
| S5–155U     150U mode | Input byte IB 0 | Block limits |
| S5–155U     155U mode | Interrupt lines | Block limits; in data block DXO, adjustable to statement limits |

## 4.2 Interrupts via Interrupt Lines

An interrupt by an interrupt–generating module is fed directly to the central processor by a separate interrupt line.

Programmable controllers S5–115U, S5–135U, and S5–155U in S5–155U mode allow interrupt pro–cessing via interrupt lines, but only in authorized slots.

Be sure that the appropriate interrupt line is present at the slot which you have selected. (See equipment manual for your programmable controller.) The IP 242A/242B triggers interrupts via four interrupt lines (IRA, IRB, IRC, and IRD). However, always use only **one** interrupt line per module.

**A** **B**  **How Do Interrupts Reach the Interrupt Line?**

Use DIP switch S1 to select an interrupt line for interrupt processing.

Completely open switch S2. (See section 2.3.)
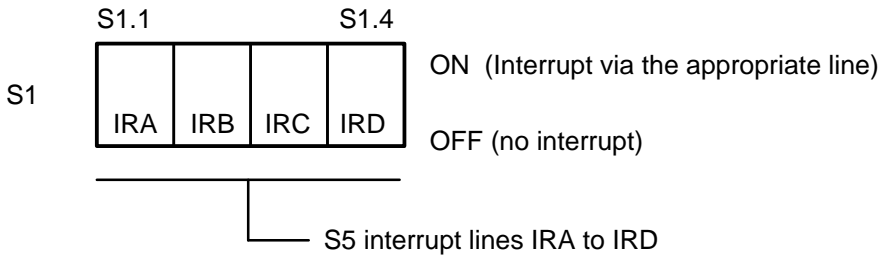


- **Setting for IP 242A/242B without Interrupt Triggering**

  DIP switch S1: open all switches

- **Setting for One IP 242A/242B with Interrupt Triggering via Interrupt Line**

  Depending on the interrupt line used, close appropriate DIP switch S1.n.

  Open the switches for the unused lines!

- **Setting for Several IP 242A/242B Modules with Interrupt Triggering via Interrupt Line**

  The same interrupt line can be set for several modules. The module which triggered the interrupt is then determined in your interrupt program.

## 4.3      Interrupts via Intput Byte IB 0

An interrupt is fed directly to the central processor by one bit of input byte IB 0.

Programmable controllers S5–150U/S and S5–155U (in S5–150U mode) process interrupts only by input byte IB 0.

The IP 242A/242B can be set to all 8 bits of input byte IB 0. Only one bit can be used per module. A maximum of eight interrupt–generating modules can be operated in this way in one programmable controller.

**How do interrupts reach the interrupt line through input byte IB 0?**

Use switch S2 to select a bit for interrupt processing. (See section 2.3.) Open switch S1 completely.

| A | B |

IB 0

| Bit 0 | Bit 1 | Bit 2 | Bit 3 | Bit 4 | Bit 5 | Bit 6 | Bit 7 |
|-------|-------|-------|-------|-------|-------|-------|-------|

S2.1                                                           S2.10     ON

S2

| EMA | SA1 | SA2 | SA3 | SA4 | SA5 | SA6 | SA7 | MSA | PB0 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|

OFF

**Process     interrupt
via input byte IB 0**
ON  : Enabled
OFF: Disabled

**Slave/master interrupt source**
ON  : Slave interrupt source
OFF : Master interrupt source

**Slave process interrupt**
ON
   – For master: those bits which are not allocated to slaves
   – For slaves: one bit for each slave (no double allocation)
OFF
   – All other switches

**Master process interrupt**
ON  : Enabled
OFF : Disabled

**A**    **B**    • **Setting for IP 242A/242B without Triggering of a Process Interrupt**

DIP switch S2: open all switches.

• **Setting for One IP 242A/242B with Triggering of a Process Interrupt**

If only one interrupt–generating module is used, set this module to bit 0.

| | | |
|---|---|---|
| S2.1 | ON | Setting a bit 0 |
| S2.2<br>S2.3<br>S2.4<br>S2.5<br>S2.6<br>S2.7<br>S2.8 | ON | These switches must be closed (defined bus termination)<br>since no additional modules are set to bits 1 to 7 |
| S2.9 | OFF | Module is set to bit 0 (master). |
| S2.10 | ON | Process interrupt processing is enabled. |

• **Setting for Several IP 242A/242B Modules with Triggering of a Process Interrupt**

A maximum of eight interrupt–generating modules can be operated in one programmable controller. Set each module to **one** bit of input byte IB 0.

One module must be set to bit 0 (master). The other modules can be assigned to bits 1 to 7 as desired.

➡ **Install all modules in the same central controller or expansion controller!** ⬅

➡ Do not operate input/output modules on input bytes IB 0 and IB 1 when DIP switch S2.10 is ON. ⬅

IP 242A/242B Equipment Manual

**Master:**

| | | |
|---|---|---|
| S2.1 | ON | Setting a bit 0 |

S2.2
S2.3
S2.4
S2.5     ON/OFF     ON:  Those bits which are not allocated to slaves
S2.6
S2.7                OFF: Those bits which are allocated to slaves
S2.8

| | | |
|---|---|---|
| S2.9 | OFF | Module is set to bit 0. |
| S2.10 | ON | Enable processing of process interrupts. |

**Slave:**

| | | |
|---|---|---|
| S2.1 | OFF | Bit 0 is allocated to the master module. |

S2.2
S2.3
S2.4     ON/OFF     ON:  That bit which is allocated to this module
S2.5                     (only one switch)
S2.6                OFF: Those bits which are not allocated to this module
S2.7                     (the remaining 6 switches)
S2.8

| | | |
|---|---|---|
| S2.9 | ON | Module is set to one of bits 1 to 7. |
| S2.10 | ON | Enable processing of process interrupts |

The following table shows an example.

Three counter modules are used here, one as master, one as slave 1, and one as slave 6.

x = ON          − = OFF

| IB 0 | DIP Switch | Type of Interrupt | Master | Slave 1 | Slave 6 |
|---|---|---|---|---|---|
| Bit 0 | 2.1 | Master interrupt | x | − | − |
| Bit 1 | 2.2 | Slave interrupt | − | x | − |
| Bit 2 | 2.3 | No interrupt | x | − | − |
| Bit 3 | 2.4 | No interrupt | x | − | − |
| Bit 4 | 2.5 | No interrupt | x | − | − |
| Bit 5 | 2.6 | No interrupt | x | − | − |
| Bit 6 | 2.7 | Slave interrupt | − | − | x |
| Bit 7 | 2.8 | No interrupt | x | − | − |
| | 2.9 | | − | x | x |
| | 2.10 | | x | x | x |

## 4.4       Interrupts via Interrupt Lines and Evaluation via Input Byte IB 0

**A**   **B**    For quick scanning of the modules which triggered the interrupt when several modules are used, you can also evaluate input byte IB 0 parallel to the interrupt line.

**One** interrupt line is selected to trigger a program interrupt in the central processor, as described in section 4.2.

Set input byte IB 0 (one module at bit 0, the others at bits 1 to 7), as described in section 4.3.

The module which triggered the interrupt is determined in the interrupt program by evaluating input byte IB 0 (or I/O byte PY0)[1]. Evaluation of periphery byte PY0 is recommended since this byte in continuously updated whereas input byte IB0 is only updated cyclically. The signal status of the appropriate bit is then "1". This module can now be specifically addressed.

> Do not use input/output modules with input bytes IB 0 and IB 1 when DIP switch S2.10 is ON.

---
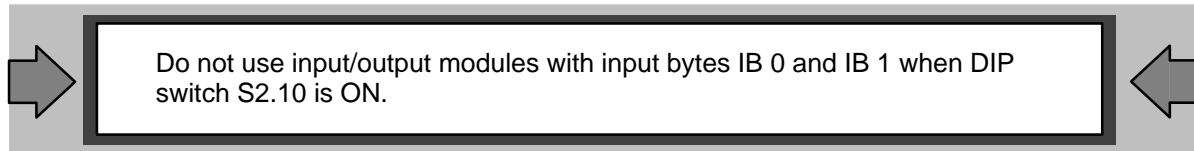
1   Evaluation of periphery byte PY0 is recommended since this byte is continuously updated whereas input byte IB0 is only updated cyclically.

---

## 4.5      Reaction Times for Interrupts

The reaction time of an interrupt is made up of the reaction time of the module and the reaction time in the programmable controller.

A time advantage can be achieved despite the slower reaction to the S5 by using additional command lists on the IP 242B. Calculations are performed by the PLC CPU when the IP 242A is used. This requires a CPU access to the module for every single operation. Access time is usually more than 1000 $\mu$sec ($\rightarrow$ section 11.7/14.7).

The Reaction Time of the Module:

|  | **A** | **B** |
|---|---|---|

- **One enabled interrupt source**

  If there are no command lists present and no control words to be executed, the reaction time is

  |  | | |
  |---|---|---|
  | Maximum | 20 $\mu$sec | 250 $\mu$sec |

  or if the comparator function for counters 6 and 7 is activated

  |  | | |
  |---|---|---|
  | Maximum | 70 $\mu$sec | 250 $\mu$sec |

- **Several enabled interrupt sources**

  The interrupts are executed in the order of their priority.

  The reaction time for the highest priority interrupt is

  |  | | |
  |---|---|---|
  | Maximum | 20 $\mu$sec | 250 $\mu$sec |

  The reaction time for the lowest priority interrupt is

  |  | | |
  |---|---|---|
  | Maximum | n • 50 $\mu$sec | n • 250 $\mu$sec |

  (n = the number of interrupts which occur at the same time).

  This reaction time can be increased by 50 $\mu$sec if the comparator function for counters 6 and 7 is enabled.

  The above specified times assume that there are no command lists to be executed for any of the enabled interrupt sources. When a control word is to be executed at the same time, the processing time of the control word (see section 10.8) must be added to the time specified above.

- **Several interrupt sources with command lists**

  If several interrupts occur at the same time, the interrupt with the highest priority is triggered first. (See list on next page.) If a command list is present for this interrupt, the command list is executed before the next interrupt is triggered. If a command list is present for this interrupt too, it is also executed first before another interrupt is triggered. The processing time of a command list is made up of the processing times of the control words in the command list.

Priority List of the Interrupt Sources

| | | |
|---|---|---|
| Highest priority: | Output  1 | |
| | Output  2 | |
| | Output  3 | } when comparator |
| | Output  4 | function is disabled |
| | Output  5 | |
| | Gate 1 | |
| | Gate 2 | |
| | Gate 3 | |
| | Gate 4 | |
| | Gate 5 | |
| | Reset 6 | |
| | Reset 7 | |
| | Output  3 | } when comparator |
| | Output  4 | function is enabled |
| | Output  5 | |
| | Output  6 | |
| Lowest priority: | Output  7 | |

The Reaction Time of the Programmable Controller:

- **For interrupt processing at block limits**

  The maximum reaction time here is:

  – the processing time of the longest program without block calls or block limits
  – the longest interrupt program (Interrupt programs cannot be interrupted.)

  Add this time to the processing time of your own interrupt program.

- **For interrupt processing at statement limits**

  The maximum reaction time here is:

  – the processing time of the longest block transfer or
  – the time spent in the operating system of the programmable controller or special function.

  Add this time to the processing time of your own interrupt program.

  The following applies to both types of interrupt processing.
  Add the interrupt disable time for interrupts to the reaction time (e.g., with the "AS" and "AF" statements).

## 4.6      Sources of Interrupts

The interrupt sources are as follows:

**A**   **B**

    – Counter outputs     1 to 7
    – Counter gates       1 to 5
    – Reset inputs        6 and 7
    – Module error messages

Each interrupt source must be enabled in the interrupt enable register. In addition, the SA bit in this regis-
ter must be set to trigger an interrupt. If a module error message is used as the source of an interrupt, the
ER bit in this register must be set (see sections 3.3.4 and 3.3.5).

All interrupts are disabled after the start up of the module.

**Only for IP 242B:**
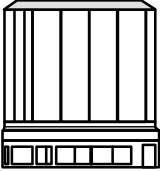
**B**

The AFR interrupt filter register ($\rightarrow$ section 3.3.6) allows you to process interrupts on the module with a
command list without an interrupt being triggered for the S5.

The interrupts are intermediately stored as long as the BASP signal is active. Intermediate storage
continues until the module detects that the BASP signal is no longer active.

## 4.7 The Best Way to Proceed

There are several preparations to make if you plan to use one or more counter modules with interrupt processing. The following overview provides helpful hints.
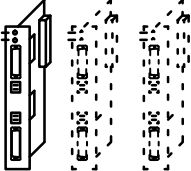
? The first thing to find out is whether your programmable controller evaluates interrupts
via interrupt lines or
via input byte IB 0.
Look this up in the table in section 4.1.

```
L    EW   100
T    DW   20
JU   PB   17
A    I    7.5
A    I    6.3
=    Q    5.1
```

? The table in section 4.1 also shows you whether an interrupt interrupts your user program at
block limits or at statement limits.

? Decide how many interrupt–generating modules you want to use.
Check to see where to insert the modules.
($\rightarrow$ Section 10.12)
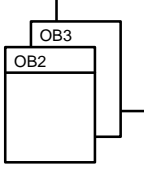
? Section 2.3 shows you how to set the switches on an IP 242A/242B counter module.
Depending on your programmable controller, see sections 4.2, 4.3, or 4.4 for details.

| 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |

| 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 |

? Parameterize the registers for interrupt processing described in section 3 (i.e., interrupt enable register and interrupt polarity register, and (for IP 242B) the interrupt fiter register) in accordance with your application.

OB3
OB2

? Now set up your own interrupt program in the appropriate interrupt organization blocks.

# 5 Putting the Counter Module into Operation

## 5.1      Guidelines for Putting into Operation

Before putting your counter module into operation, make hardware (points 1 to 5) and software (points 6 to 10) settings.

Standard function blocks are included with your counter module. The corresponding floppy disk also contains an example which includes all blocks necessary for a real program ($\rightarrow$ section 12 or 15).
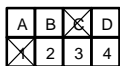
1. After unpacking, check the following:
   Release status of the module
   (See front panel.)
   Firmware version (See EPROMs.)

2. Set the following addresses on the module:
   Basic address (DIP switch S3)
   Page frame number (DIP switch S4)      Section 2.2
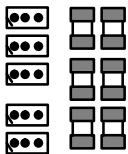
3. Adjust the following inputs on the module:
   Input level (BR3 to BR17)      Section 2.4
   Input frequency (C8 to C22)      Section 2.5
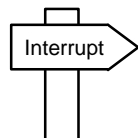
4. Set the following on the module for interrupt processing:
   Interrupt (DIP switch S1) or
   Process interrupt (DIP switch S2)      Section 2.3
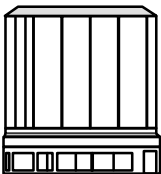
5. Now insert the module in the correct slot after turning off the programmable controller.
   (When interrupts are processed via interrupt lines, only certain specific slots may be used, $\rightarrow$ section 10.12.)

**A**
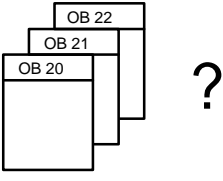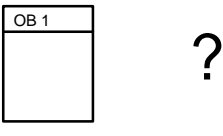
6. Set up the parameterization data block.
   Section 11.6
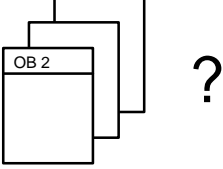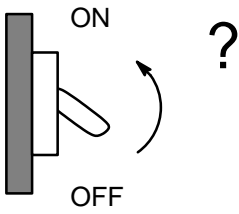
**B**

Set up the parameterization data block and the measured value data block.
Section 14.6

| | **A** | **B** |
|---|---|---|
| OB 22<br>OB 21<br>OB 20<br><br>**?** | 7. Program the start–up organization blocks (OB 20 to OB 22).<br>• Save flag bytes FY 200 to FY 255 (only for a warm restart).<br>• Call function block FB 178 (PER:ZSTK) or FB 179 (PER:ZSTL) with the "BEF = PA" command (parameterize counter).<br>• Call function block FB 178 or FB 179 with the "BEF = FA" command (enable the outputs).<br>• Load flag bytes FY 200 to FY255 (only for warm restart).<br><br>Section 11 | Program the start–up organization blocks (OB 20 to OB 22).<br>• Save flag bytes FY 200 to FY 255 (only for a warm restart).<br>• Call function block FB 183 (ZYK:242B) with the "BEF = PA" command (parameterize counter).<br>• Call function block FB 183 with the "BEF = FA" command (enable the outputs).<br>• Load flag bytes FY 200 to FY255 (only for warm restart).<br><br>Section 14 |
| OB 1<br><br>**?** | 8. Now set up your own cyclical program in organization block OB 1! During the cyclical program, the IP 242A is controlled by function block FB 178 (PER:ZSTK) or FB 179 (PER:ZSTL) in accordance with your application). | Now set up your own cyclical program in organization block OB 1! During the cyclical program, the IP 242B is controlled by function block FB 183 (ZYK:242B) in accordance with your application. |
| OB 2<br><br>**?** | 9. If necessary, set up your interrupt program in the appropriate interrupt organization block. | |
| ON<br><br>**?**<br><br>OFF | 10. When you turn on your programmable controller, follow the sequence recommended here.<br>1. Operation mode switch on the central processor must be in STOP position<br>2. Turn on power for the programmable controller. (If turning on for the first time, perform an "overall reset" of the program memory.)<br>3. Transfer blocks from the programmer to the programmable controller.<br>4. Start up the central processor with a "cold" start. | |

---

! We draw your attention to the fact that the IP 242A/242B module can be damaged by unqualified handling, operation, or incorrect circuitry. This can cause extensive damage to your system. We make the assumption that the module will be handled only by qualified personnel who are also familiar with ESD protection regulations. !

## 5.2    Start–Up Behavior of the Counter Module/Meaning of the LEDs

All test functions are performed during the start–up phase. The red LED lights up to indicate this. The green LED starts to flash when the test functions are satisfactorily completed (after approximately 0.5 sec). This also indicates that the module still does not have valid parameterization.

Only after error–free processing on

– The IP 242A with standard function block FB 178/179 or

– The IP 242B with standard function block FB 183

and the "parameterize counter", "rewrite parameter" or "accept basic setting" commands does the green LED go on continuously.

Interrupts are not recorded or triggered as long as the BASP signal is present.

| Indicator | | Meaning |
|---|---|---|
| RUN LED (Operation) | F LED (Error) | |
| OFF | OFF | – S5 central processor not ready |
| OFF | ON | – Module start–up <br> – Module error  (e.g.,   overload short circuit <br> bad test function <br> watchdog error |
| FLASHING | OFF | Start–up successful. Module still not parameterized. |
| ON | OFF | Module is parameterized and functioning properly. |

## 5.3    Switching the Counters with Each Other

Remember the following when switching the counters with each other ($\rightarrow$ section. 1.11).

1. The output level of the first counter must be conditioned as the input level of the next counter.

   – **Counters 1 to 5** are switched internally. Only the settings in the master mode register and the corresponding counter mode registers are required.

   – **Counters 1 to 5** can **only** be switched **externally to counters 6 and 7**. Use the "24–V, asymmetrical to 5 V (RS422) symmetrical converter" ($\rightarrow$ section. 10.11).

2. The maximum output frequency is always $\leq$ 40 kHz.

# 6 Control Words

## 6.1       General Module Functions
## 6.1.1    Control Word Format

The control word triggers the corresponding function on the module. There are three ways to do this.

- Direct parameterization using parameters BEF and PAR on the standard function block
  (see section 11 or 14)
- Indirect parameterization using parameter STEU
- Command lists during interrupt processing

You will find the bit pattern for parameter STEU and the command lists **in this section**. The **functions which the module executes** when it receives the control word are described here.

Section 12.8.2 or 15.8.2 describes what the function block does when the control word is specified with parameter STEU.

```
 15                                        8   7                                  0
 ┌──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┐
 │  │  │  │  │  │  │  │  │C7│C6│C5│C4│C3│C2│C1│GL│
 └──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┘
```

Command code               Selection of counter     Selection of
                           for counter n            global registers

The command is executed only for
the counters and global registers
which are selected with "1".

Control words in command lists only work in both directions between the internal module registers and the dual port RAM.

Control words of the function block (from the S5) work in both directions between parameterization data block and module/dual port RAM.

IP              DPR                    DB

Command list    From FB

Details on the individual commands → section 3.4.2

| Ab-breviation (BEF) | Control word | Meaning | Legal for IP242A | | IP242B | | Permissible Parameters |
|---|---|---|---|---|---|---|---|
| | | | For S5 | For Command Lists | For S5 | For Command Lists | |
| BE | KH0000 | End of command lists | – | • | – | • | None |
| RB | KH0100 | Reset module | • | – | • | – | **No** selection of a counter or the global registers |
| SA | KH0200 | Disable the outputs | • | • | • | • | |
| FA | KH0300 | Enable the outputs | • | • | • | • | |
| IM | KH0400 | Mask interrupt | • | • | • | • | |
| KS | KH0500 | Write constant register | – | – | • | – | |
| ZA | KH0600 | Update counter values | – | – | • | – | |
| ST | KH11xx | Start counter | • | • | – | – | Counters 1 to 5 |
| LD | KH12xx | Load counter | • | • | – | – | |
| SP | KH13xx | Stop counter | • | • | – | – | |
| SL | KH14xx | Stop and read counter | • | • | – | – | |
| SZ | KH15xx | Step counter | • | • | • | • | |
| SV | KH16xx | Save counter | • | • | • | • | |
| CO | KH17xx | Copy counter | • | • | • | • | |
| LV | KH18xx | Prepare for load | – | – | • | • | |
| LS | KH31xx | Load and start counter | • | • | • | • | Counters 1 to 7 |
| LE | KH32xx | Read counter | • | • | • | • | |
| RZ | KH33xx | Reset counter | • | • | • | • | |
| AW | KH34xx | Accept interrupt value | • | • | • | • | |
| DF | KH35xx | Generate difference | • | • | – | – | |
| ST | KH36xx | Start counter | – | – | • | • | |
| LD | KH37xx | Load counter | – | – | • | • | |
| SP | KH38xx | Stop counter | – | – | • | • | |
| SL | KH39xx | Stop and read counter | – | – | • | • | |
| PA | KH71xx | Parameterize counter | • | – | • | – | Counters 1 to 7 and global registers |
| PS | KH72xx | Store parameter | • | – | • | – | |
| PZ | KH73xx | Rewrite parameter | • | – | • | – | |
| GR | KH74xx | Accept basic setting | • | – | • | – | |
| PO | KH75xx | Parameterize counter (Without command list) | – | – | • | • | |
| RL | KH76xx | Read register | – | – | • | – | |
| RS | KH77xx | Write register | – | – | • | – | |
| TF | KH81xx | Execute test function | • | – | • | – | A test function must be selected. |
| BB | KH82xx | Process command list | – | – | • | • | Additional command lists 1 to 7 |
| ML | KH83xx | Read measured value series | – | – | • | – | Measured value series 1 to 7 and directory of the measured value memory |
| MR | KH84xx | Read and reset measured value series | – | – | • | – | |
| BL | KHF1xx | Read module | • | – | – | – | Counters 1 to 7 and the global registers; (only for standard function block, not for interrupt command lists) |
| BS | KHF2xx | Write module | • | – | – | – | |

• Control word permitted on the module

– Control not permitted on the module

xx Selection for counter and global register

## 6.1.2    End of Command List (BE)

IP 242A/242B

Control word:  KH = 0000

`A`  `B`

Description:
The end of the interrupt command list has been reached.  Additional commands are not permitted.  Commands entered after "BE" will be ignored.

The control word is always required to conclude an interrupt command list (it has the same effect as a BE block end of the S5).

Exception only in the following cases:

– The full length of the interrupt command list is used (5 DWs).
– Not in the additional command lists (The list end is specified there in the directory.)

**Example:**

```
15                        0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Control word: KH = 0000 Command list end

## 6.1.3    Resetting Modules (RB)

IP 242A/242B

Control word:  KH = 0100

Description:
The module is set to the status which it has after supply voltage is turned on (i.e., master reset).

The following test functions are performed, and the red LED goes on:
– RAM test
– Dual port RAM test
– EPROM test
– EEPROM test
– AM 9513A test (counters 1 to  5)

After the control word is executed, the counters are not parameterized and the counter outputs are turned off.

`A`  `B`

On the IP 242B the counter values, the counter status registers and the result registers are set to zero.

`B`

The read LED goes off and the green LED starts to flash if the test is free of errors. The module must then be parameterized with PA, PZ or GR.

`A`  `B`

**Example:**

```
15                        0
 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
```

Control word: KH = 0100  Reset the module

## 6.1.4     Disabling Outputs (SA)                                    IP 242A/242B

**A** **B**  Control word: KH = 0200

Description:
All counter outputs are turned off (4.7 kΩ against 0 V$_{external}$).

**Example:**

```
15                         0
0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
```
Control word: KH = 0200  disable the outputs


## 6.1.5     Enabling Outputs (FA)                                     IP 242A/242B

**A** **B**  Control word:  KH = 0300

Description:
All counter outputs are enabled.

When a counter output is automatically turned off because of a short circuit, you must enable the output again with this control word.

**Example:**

```
15                          0
0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0
```
Control word: KH = 0300 Enable the outputs


## 6.1.6     Masking Interrupts (IM)                                   IP 242A/242B

Control word:  KH = 0400

Description:
**A** **B**  The interrupt enable register and the interrupt polarity register are taken over from the dual port RAM. Interrupt processing is then performed as specified in these registers.

**B**  On the IP 242B module, the interrupt filter register also goes into effect when the SA bit in the interrupt enable register is "0".

**Example:**

```
15                          0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
```
Control word: KH = 0400  Mask interrupt

### 6.1.7    Write Constant Register (KS)

IP 242B

**B**

Control word: KH = 0500

Description:
The 16 constant registers for the calculation function are transferred again (e.g., readjustment of system parameters).  The new constants go into effect for calculation the next time the command lists are processed.

**Example:**

```
15                      0
 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0
```
Control word: KH = 0500  Write constant register


### 6.1.8    Update Counter Values (ZA)

IP 242B

**B**

Control word:  KH = 0600

Description:
The following registers are updated.

- Counter values, cyclic (ZSZ)

- Counter status register (ZSR)

- Result register (ERG)

In contrast to the "Read counter" (LE) control word, the values of the hold registers specified during parameterization are entered as the counter values for operating modes G, H, I, J, K, L, S and V. The counter value at the last active gate edge is entered for operating mode X.

**Example:**

```
15                      0
 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0
```
Control word: KH = 0600  Update counter values

---

### 6.1.9    Starting Counters (ST)                                  **IP 242A/242B**

---

**A** Control word:  KH = 11xx

This control word can be used only for counters 1 to 5.

Description:
Control word "start counters" enables the selected counters.

Depending on the operation mode, a gate enable must also be performed before the counting pulses can be counted.

Before starting a counter, you should parameterize and load it first.

When no counters are selected, error message KH03 is entered in the error information register. Error message KH02 is given if counters 6 and 7 are selected.

**Example:**

| 15 | | | | | | | | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | **1** | 0 | 0 | 0 | **1** | 0 | 0 | **1** | **1** | **1** | 0 | 0 | 0 |

Control word: KH = 1138  Start counters 3, 4, and 5

**B** Control word:  KH = 36xx

This control word can be used for all counters.

Description:
"Start counter" enables the selected counters.

Depending on the operating mode, a gate enable has to be performed so that all counting pulses will be counted.

A counter ought to be parameterized and loaded before it is started.

Error message KH03 is entered in the error information register when no counter is selected.

Example:

| 15 | | | | | | | | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | **1** | **1** | 0 | **1** | **1** | 0 | **1** | 0 | 0 | **1** | **1** | 0 | 0 | 0 |

Control word:  KH = 3698  Start counters 3, 4, 7

## 6.1.10  Loading Counter (LD)

**IP 242A/242B**

Control word:  KH = 12xx

**A**

This control word can be used only for counters 1 to 5.

Description:
Control word "load counters" loads the selected counters with the counter values stored in the load registers of the dual port RAM.

When no counters are selected, error message KH03 is entered in the error information register. Error message KH02 is given if counters 6 and 7 are selected.

**Example:**

```
15                           0
┌─────────────────────────────────┐
│ 0 0 0 1 0 0 1 0 │ 0 0 1 0 0 0 1 0 │
└─────────────────────────────────┘
```

Control word: KH =1222 Load counters 1 and 5

Control word:  KH = 37xx

**B**

This control word can be used for all counters.

Description:
"Load counter" loads the selected counters with the counting values stored in the load registers of the dual port RAM.

Error message KH03 is entered in the error information register when no counter is selected.

**Example:**

```
15                           0
┌─────────────────────────────────┐
│ 0 0 1 1 0 1 1 1 │ 0 1 0 0 0 0 1 0 │
└─────────────────────────────────┘
```

Control word: KH = 3742 Load counters 1 and 6

## 6.1.11  Stopping Counters (SP)

**IP 242A/242B**

**A** Control word:  KH = 13xx

This control word can be used only for counters 1 to 5.

Description:
Control word "stop counters" disables the selected counters.  Available counting pulses are then ignored. The last counter status is retained but not transferred to the hold register.

When no counters are selected, error message KH03 is entered in the error information register. Error message KH02 is given if counters 6 and 7 are selected.

**Example:**

```
15                             0
 0 0 0 1 0 0 1 1 | 0 0 0 1 1 0 0 0
```

Control word: KH = 1318 Stop counters 3 and 4

**B** Control word:  KH = 38xx

This control word can be used for all counters.

Description:
"Stop counter" disables the selected counters.  Counting pulses are then ignored.  The last counter value is retained.

Error message KH03 is entered in the error information register when no counter is selected.

**Example:**

```
15                             0
 0 0 1 1 1 0 0 0 | 1 0 0 1 0 1 0 0
```

Control word: KH = 3894 Stop counters 2, 4 and 7

## 6.1.12  Stopping and Reading Counters (SL)                    IP 242A/242B

Control word:  KH = 14xx                                                                          **A**

This control word can be used only for counters 1 to 5.

Description:
Control word "stop and read counters" performs the "stop counter" and "read counter" functions simultaneously for the selected counters.

The selected counters are stopped. Available counting pulses are then ignored. The last counter status is transferred to the corresponding hold register in the dual port RAM.

When no counters are selected, error message KH03 is entered in the error information register. When counters 6 and 7 are selected, error message KH02 is given.

**Example:**

```
15                          0
 0 0 0 1 0 1 0 0  0 0 0 0 1 0 1 0
```
Control word: KH = 140A Stop and read counters 1 and 3

Control word:  KH = 39xx                                                                          **B**

This control word can be used for all counters.

Description:
"Stop and read counter" executes the "stop counter" and "read counter" functions simultaneously for the selected counters.

The selected counters are stopped.  Counting pulses are then ignored.  The last counter value is transferred to the corresponding counter value register in the dual port RAM.

Error message KH03 is entered in the error information register when no counter is selected.

**Example:**

```
15                          0
 0 0 1 1 1 0 0 1  0 1 0 0 1 0 1 0
```
Control word: KH = 394A Stop and read counters 1, 3 and 6

---

### 6.1.13  Stepping Counters (SZ)

**IP 242A/242B**

**A** **B**  Control word:  KH = 15xx

This control word can be used only for counters 1 to 5.

Description:
Control word "step counters" steps the status of the selected counters by 1 in the direction specified by the parameterization. This is also done when the counter is stopped.

When no counters are selected, error message KH03 is entered in the error information register. When counters 6 and 7 are selected, error message KH02 is given.

**Example:**

```
15                        0
 0 0 0 1 0 1 0 1 | 0 0 0 1 1 0 0 0
```

Control word: KH = 1518 Step counters 3 and 4

---

### 6.1.14  Saving Counters (SV)

**IP 242A/242B**

**A** **B**  Control word:  KH = 16xx

This control word can be used only for counters 1 to 5.

Description:
Control word "save counters" saves the present counter values of the selected counters in the internal hold registers of the counter block. The contents can be read out with control word "copy". The counter is not stopped during this procedure.

When no counters are selected, error message KH03 is entered in the error information register. When counters 6 and 7 are selected, error message KH02 is given.

**Example:**

```
15                        0
 0 0 0 1 0 1 1 0 | 0 0 1 0 0 0 1 0
```

Control word: KH = 1622 Save counters 1 and 5

## 6.1.15  Copying Counters (CO)                          **IP 242A/242B**

Control word:  KH = 17xx

This control word can be used only for counters 1 to 5.

Description:
Control word "copy counters" copies the contents of the internal hold registers of the counter block to the **A**
corresponding hold registers of the dual port RAM. The counter is not stopped during this procedure. The
command applies to all counters selected.

For operation modes N, O, Q, R, and X, the counter status which was transmitted to the internal hold
register with the active gate edge is also transferred to the dual port RAM.

"Copy counter" copies the contents of the internal hold registers of the counting block to the **counter** **B**
**value registers** in the dual port RAM. It does not stop the counter.  The command applies to all selected
counters.

In operating modes N, O, Q, R and X, this can cause the counter value, which was transferred to the inter-
nal hold registers with the active gate edge, to be transferred to both the dual port RAM and the selected
counter value register.

The "save counter" control word which was called, for example, by a command list of another counter
causes the counter value which was current at the time of the command to be stored intermediately in the
internal hold register.  The counter value can then be fetched with "copy" at the time of the "save" com-
mand.

When no counters are selected, error message KH03 is entered in the error information register. When **A**　**B**
counters 6 and 7 are selected, error message KH02 is given.

**Example:**

| 15 | | | | | | | | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | **1** | 0 | **1** | **1** | **1** | 0 | 0 | **1** | 0 | **1** | 0 | **1** | 0 |

Control word:  KH = 172A Copy counters 1, 3 and 5

---

### 6.1.16 Prepare Loading (LV)

IP 242B

**B**

Control word:  KH = 18xx

This control word can only be used for counters 1 to 5.

Description:
"Prepare load" causes the contents of the load registers to be copied to the internal load registers for the selected counters.  This does not cause the counter to be loaded.  The new load values are accepted with LD, LS or automatically when the counter expires.  This can cause frequency ramps to be generated although pulse gaps do not occur during loading.

In addition, LV transfers the contents of the hold registers from the parameterization data block to the dual port RAM.  For operating modes G, H, I, J, K, L, S and V, the internal hold registers are also loaded.  The hold value is accepted when the counter expires.

Error message KH03 is entered in the error information register when no counter is selected.  Error message KH02 is used when counters 6 and 7 are selected.

**Example:**

```
15                          0
 0 0 0 1 1 0 0 0  0 0 1 0 0 1 1 0
```

Control word:    KH = 1826   Prepare loading for
                             counters 1, 2 and 5

---

### 6.1.17 Loading and Starting Counters (LS)

IP 242A/242B

**A**  **B**

Control word:  KH = 31xx

This control word can be used for all counters

Description:
Control word "load and start counters" simultaneously performs the "load counters" and "start counters" functions for the selected counters.

Depending on the operation mode, a gate enable must also be performed before the counting pulse can be counted.

When no counters are selected, error message KH03 is entered in the error information register.

**Example:**

```
15                          0
 0 0 1 1 0 0 0 1  0 1 0 0 0 0 1 0
```

Control word: KH = 3142  Load and start counters 1 and 6

## 6.1.18  Reading Counters (LE)
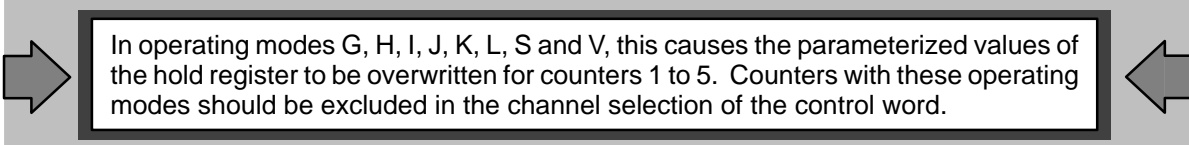
**IP 242A/242B**

Control word:  KH = 32xx

This control word can be used for all counters.

Description:
Control word "read counters" reads the actual counter values of the selected counters and stores the values in the corresponding **hold registers** in the dual port RAM.

**A**

> In operating modes G, H, I, J, K, L, S and V, this causes the parameterized values of the hold register to be overwritten for counters 1 to 5.  Counters with these operating modes should be excluded in the channel selection of the control word.

"Read counter" reads the current counter values of the selected counters, and stores the values in the corresponding **counter value registers** in the dual port RAM

**B**

The counter is not stopped during this procedure.

**A**   **B**

When no counters are selected, error message KH03 is entered in the error information register.

**Example:**

```
15                          0
 0 0 1 1 0 0 1 0 1 1 1 1 1 1 1 0
```

Control word:  KH = 32FE  Read all counters

**6.1.19  Resetting Counters (RZ)**                                       **IP 242A/242B**

**A** **B**  Control word:  KH = 33xx

This control word can be used for all counters.

Description:
Control word "reset counters" resets the selected counters to 0.

When control word "reset counters" is used for counters 6 and 7, a **group interrupt** is **not** triggered in contrast to a reset at the SYN input.

When no counters are selected, error message KH03 is entered in the error information register.

**Example:**

```
15                        0
 0 0 1 1 0 0 1 1 1 1 0 1 0 0 0 0
```
Control word:  KH = 33D0  Reset counters 4, 6 and 7

## 6.1.20  Taking Over Interrupt Values (AW)                    **IP 242A/242B**

Control word:  KH = 34xx

**A   B**

This control word can be used for all counters.

Description:
Control word "take over interrupt values" takes over the contents of the interrupt registers of the selected counters from the dual port RAM.

When the comparator function is enabled for counters 1 to 5, the counter output is activated when the counting value reaches the interrupt value (see section 1.9).

The counter output is activated for counters 6 and 7 when the counting value reaches the interrupt value.

When  the comparator function is not enabled for counters 1 to 5, disregard the contents of the interrupt registers. The interrupt register must not, however, be zero for counters 1 to 5. The counter output is then activated at counting cycle zero.

When no counters are selected, error message KH03 is entered in the error information register.

**Example:**

```
15                              0
 0 0 1 1 0 1 0 0 1 0 0 1 0 0 1 0
```
Control word:  KH = 3492  Take over interrupt values 1, 4 and 7

## 6.1.21  Generating Differences (DF)

**IP 242A**

**A** Control word:  KH = 35xx

This control word can be used for all counters.

Description:
Control word "generate difference" subtracts the counter status of the counter with the higher number from the counter status of the counter with the lower number. The result is stored in the difference register in the dual port RAM.

For generation of the difference, the counter values are defined as follows:

Counters 1 to 5    from  0000 0000
                     to   0000 FFFF


            6 and 7 from  FF80 0000
                     to   007F FFFF


  xxxx  are supplemented by the counter module during difference generation.

If more than two counters are selected, only the counters with the lowest numbers are used. All other selected counters are ignored.

**Example:**

```
15                      0
 0 0 1 1 0 1 0 1 | 0 0 1 0 1 0 0 0
```

Control word:  KH = 3528  Generate difference
                          (counter status of counter 3)
                          (counter status of counter 5)

**B** The "generate difference" control word does not exist for the IP 242B.  The calculation functions can be substituted here.  The difference register for the command also does not exist.

---

### 6.1.22  Parameterizing Counters (PA)                           IP 242A/242B

---

Control word:  KH = 71xx

This control word can be used for all counters and global registers (not in command lists).

Description:
Control word "parameterize counters" takes over the following data from the dual port RAM:

|  |  |  |  |
|---|---|---|---|
| – For counters 1 to 7: | Counter mode register<br>Load register<br>Hold register<br>Interrupt register<br>Interrupt command lists | **A** | **B** |

|  |  |  |  |
|---|---|---|---|
| – For the global registers: | Master mode register<br>Prescaler register<br>Gate control register<br>Interrupt enable register<br>Interrupt polarity register | **A** | **B** |

|  |  |  |
|---|---|---|
| Interrupt filter register<br>Pulse duration register<br>Constant register<br>Directory of the additional command lists<br>Additional command lists<br>Directory of the measured value memory | Additional<br>for IP 242B | **B** |

When no counters are selected, error message KH03 is entered in the error information register.

> After parameterization, the parameterized counters for counters 1 to 5 are stopped. They must be started again with control words "start counters" or "load and start counters".

**Example:**
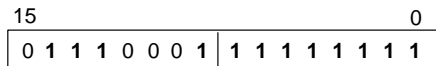
```
15                        0
0 1 1 1 0 0 0 1 | 1 1 1 1 1 1 1 1
```
Control word: KH = 71FF   Parameterize all counters with take over of the global registers

The green RUN LED continues to flash until at least one counter and the global registers are parameterized correctly.   **A**  **B**

The RUN LED stats to flash again if this condition ceases to be met during operation (e. g., due to incorrect parameterization).   **B**

## 6.1.23  Storing Parameters (PS)

IP 242A/242B

Control word:  KH = 72xx

This control word can be used for all counters and global registers (not in command lists).

Description:
The parameters stored in the dual port RAM are backed up by a non volable EEPROM on the module.

**A** **Only one data record** can be stored on the **IP 242A**.

**B** **Up to eight data records** can be stored on the **IP 242B**.  This requires that, during parameterization of the IP 242B, the S5 specify which data record is to be stored.  The selection is provided via a data record number (0 to 7) in DW13 of the parameterization data block

You can rewrite the parameters to the dual port RAM any time you wish by using the "rewrite parameter" function.

**A** **B** When no counter is selected, error message KH03 is entered in the error information register.

**B** Error message KH50 is output when the data record number is invalid (not 0 to 7).

**Example:**

```
15                          0
 0 1 1 1 0 0 1 0 0 1 0 0 1 0 1 1
```
Control word:  KH = 724B   Store parameters, counters 1, 3 and 6 and global register

Data word DW 255 with the indicator to the measured value data block is **not** stored.

---

## 6.1.24  Rewriting Parameters (PZ)                           IP 242A/242B

Control word:  KH = 73xx

This control word can be used for all counters and global registers (not in command lists).

Description:
The data for the selected counters which is stored on the EEPROM is rewritten to the dual port RAM and the "parameterize counters" command is performed for the selected counters. This requires that the "store parameter" command be executed at least once beforehand.  **A**

On the IP 242B, DW13 of the parameterization data block must contain the data record (0 to 7) from which the parameters are to be rewritten.  The indicator to the measured value data block cannot be written back.  **B**

When no counters are selected, error message KH03 is entered in the error information register.  **A**  **B**

Error message KH50 is output when the data record number is invalid or a data record which not yet been stored is to be rewritten.  **B**

**Example:**

```
15                        0
0 1 1 1 0 0 1 1 | 1 0 0 1 0 1 0 0
```

Control word:  KH = 7394   Take over parameterization counters 2, 4 and 7 without global register

---

## 6.1.25  Taking Over Basic Settings (GR)                      IP 242A/242B

Control word:  KH = 74xx

This control word can be used for all counters and global registers (not  in command lists).

Description:
The specified basic setting (section 3.9) is transferred to the selected counters. Control word "parameterize counters" is performed for the selected counters.  **A**  **B**

In addition, on the IP 242B, all additional command lists, the constants, the measured value directory and the measured value memory are specifically set to zero.  **B**

When no counters are selected, error message KH03 is entered in the error information register.

**Example:**

```
15                          0
0 1 1 1 0 1 0 0 | 0 0 0 0 1 1 1 1
```

Control word:  KH = 740F   Transfer basic setting, counters 1, 2 and 3 and global register

---

## 6.1.26  Parameterize Counter (Without Command List) (PO)

**IP 242B**

**B** Control word:  KH = 75xx

This control word can be used for all counters and global registers (also in the command lists).

Description:
"Parameterize counter (without command list)" accepts the following from the dual port RAM.

|   |   |
|---|---|
| – For counters 1 to 7: | Counter mode register<br>Load, hold and alarm registers |
| – For the global registers: | Master mode register<br>Prescaler register<br>Gate control register<br>Interrupt enable register<br>Interrupt polarity register<br>Interrupt filter register<br>Pulse duration register |

In contrast to "parameterize counter", the interrupt command lists, additional command lists, and the measured value directory are not transferred with "parameterize counter (without command list)".  This makes reparameterization during operation faster (< 1 msec).

Error message KH03 is entered in the error information register when no counter is selected.

**Example:**

```
15                        0
 0 1 1 1 0 1 0 1 1 0 0 1 0 0 1 1
```
Control word:  KH = 7593   Parameterize counters 1, 4 and 7, and the global registers without command list

The RUN LED does not go on while control word PO is being processed.

## 6.1.27  Read Register (RL)

IP 242B

Control word:  KH = 76xx

**B**

This control word can be used for all counters and global registers.  It may **not** be used **in command lists**.

Description:
The contents of the data blocks in the parameterization data block are updated for the selected counters.

When the global registers are selected, the constant registers, the additional command lists and the measured value directory are also read out.

Error message KH03 is entered in the error information register when no counter is selected.

**Example:**

```
15                              0
 0 1 1 1 0 1 1 0 | 0 1 1 0 0 1 0 0
```
Control word:  KH = 7664   Read register for counters 2, 5 and 6

## 6.1.28  Write Register (RS)

IP 242B

Control word:  KH = 77xx

**B**

This control word can be used for all counters and global registers.  It may **not** be used **in command lists**.

Description:
The contents of the data blocks on the page frame are updated for the selected counters.

When the global registers are selected, the constant registers, the additional command lists and the measured value directory are also transferred.

Error message KH03 is entered in the error information register when no counter is selected.

**Example:**

```
15                                0
 0 1 1 1 0 1 1 1 | 1 0 0 1 1 0 0 1
```
Control word:  KH = 7799  Store register for counters 3, 4 and 7 and global registers

## 6.1.29 Executing Test Functions (TF)

**IP 242A/242B**

**A** **B** Control word: KH = 81xx

Description:
The test functions selected by bits 0 to 4 of the control word are executed. When an error is detected, the error bit in the interrupt information register is set (group interrupt) and the error is stored according to type in the error information register as follows:

KH05 RAM error
KH06 Dual port RAM error
KH07 EPROM error
KH08 EEPROM error
KH09 AM 9513A error

Command format:

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 0 | 0 | 0 | 0 | 0 | 0 | **1** | 0 | 0 | 0 | | | | | |

Counter block AM 9513A

EEPROM

EPROM

Dual port RAM

RAM
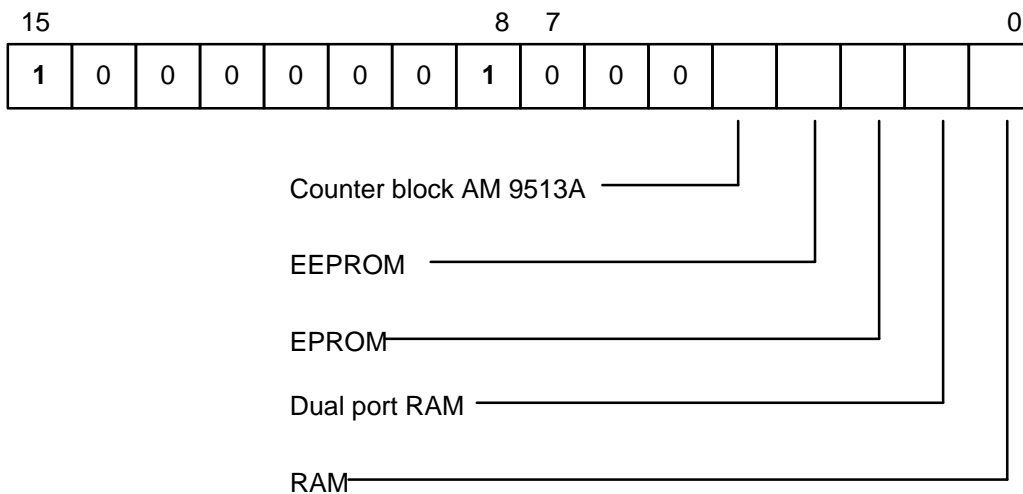
Bit = 1: Test is performed.

When no counters are selected, error message KH03 is entered in the error information register. Error message KH01 is given when bits 5, 6, or 7 are assigned.

**Example:**

| 15 | | | | | | | | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 0 | 0 | 0 | 0 | 0 | 0 | **1** | 0 | 0 | 0 | **1** | 0 | 0 | 0 | **1** |
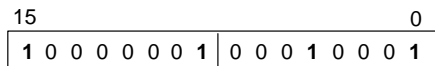
Control word:  KH = 8111   Test RAM and counter block AM 9513A

Notes:

The module is parameterized as follows during the counter test:

Operation mode D:   Frequency generator without hardware gate control

Gate control:        None
Load register:       65535
Counting direction:  Down
Output signal:       Low

The original counter values are lost. You should reparameterize after the test function is completed.

During the dual port RAM test, the module temporarily writes a bit pattern on every byte separately. The original values are then restored.

When an error is detected during execution of the test functions, the green LED goes out and the red LED goes on.

The control word may **not** be used **in command lists**.

**B** 
### 6.1.30  Process Command List (BB)

**IP 242B**

Control word:  KH = 82xx

This control word can be used to call all additional command lists.

Command format:

| 15 | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | L7 | L6 | L5 | L4 | L3 | L2 | L1 | 0 |

Command code                          List selection
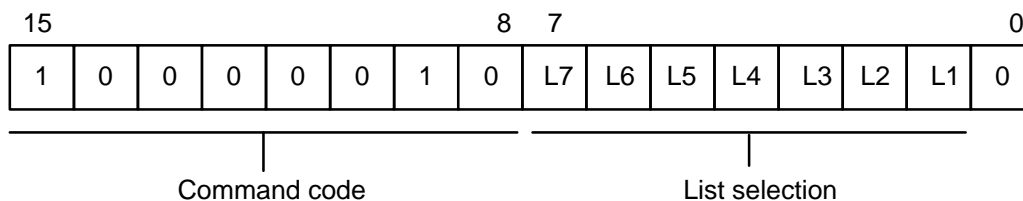
Description:
Processing of the additional command lists is the same as the execution of a subprogram.  When several lists are selected, the lists are processed in ascending order of the list numbers.

The additional command lists are reloaded with the selected global registers with the "parameterize counter" command and converted.  Required current data (e.g., load values) must be transferred to the module beforehand (e. g., with the "write register" control word).

Error message KH03 is entered in the error information register when no list is selected or bit 0 is set.

**Example:**

| 15 | | | | | | | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 0 | 0 | 0 | 0 | 0 | **1** | 0 | 0 | 0 | **1** | 0 | 0 | **1** | 0 | 0 |
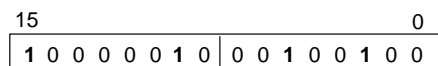
Control word:  KH = 8224  Process command lists 2 and 5

## 6.1.31  Read Measured Value Series (ML)

**IP 242B**

Control word:  KH = 83xx

**B**

This control word can be used for all measured value series and the directory of the measured value me-
mory (not in command lists).

Command format:

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | M7 | M6 | M5 | M4 | M3 | M2 | M1 | MD |

Command code

Selection for
measured value
series

Selection for mea-
sured value direc-
tory

Description:
The internal measured value memory for the selected measured value series is copied to the dual port
RAM, and the directory for all measured value blocks is updated.

The "measured value memory full" status in the counter status register remains unchanged.

The measured value memory is **not** deleted after it is read, and the filling state indicator in the directory is
not changed.

Only the measured value directory is read if **only** bit 0 is selected.  When any other selection is made, the
measured value directory is also read but not changed.

When no directory is set up for the selected measured value block, processing of the control word is termi-
nated, and an error message (from KH41 to KH47, depending on the erroneous block) is output.

Error message KH03 is entered in the error information register when no measured value series is se-
lected.

**Example:**

| 15 | | | | | | | | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 0 | 0 | 0 | 0 | 0 | **1** | **1** | **1** | 0 | 0 | **1** | **1** | 0 | 0 | 0 |

Control word:  KH = 8398   Read measured values from
measured value series 3, 4
and 7

### 6.1.32  Read and Reset Measured Values Series (MR)

**IP 242B**

**B**  Control word:  KH = 84xx

This control word can be used for all measured value series and the directory of the measured value memory (not in command lists).

Command format:

| 15 | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | M7 | M6 | M5 | M4 | M3 | M2 | M1 | MD |

Command code

Selection for measured value series

Selection for measured value directory
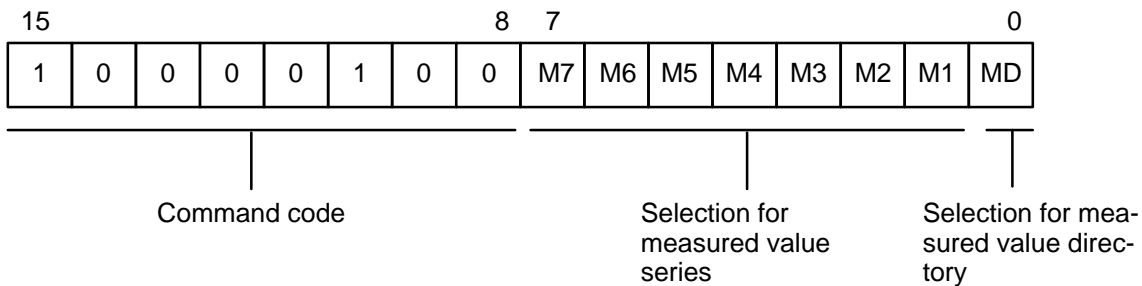
Description:
The internal measured value memory for the selected measured value series is copied to the dual port RAM, and the directory for all measured value blocks is updated.

The "measured value memory full" status is deleted for the selected measured value series in the counter status register.

The selected measured value series are deleted after they have been read, and the filling state indicator in the directory is placed at the beginning of the respective measured value block.  The data are not updated until the next read access occurs.

If **only** bit 0 is selected, only the measured value directory is read but **not** reset.  When any other selection is made, the measured value directory is also read and reset.

When no directory is set up for the selected measured value block, processing of the control word is terminated, and an error message (from KH41 to KH47, depending on the erroneous block) is output.

Error message KH03 is entered in the error information register when no measured value series is entered.

**Example:**

| 15 | | | | | | | | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 0 | 0 | 0 | 0 | **1** | 0 | 0 | 0 | **1** | **1** | 0 | 0 | **1** | 0 | 0 |
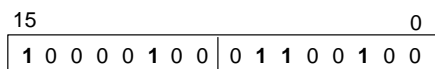
Control word:  KH = 8464   Read measured values from measured value series 2, 5 and 6, and reset the corresponding measured value memories

## 6.2 Calculation Functions
## 6.2.1 Overview

**IP 242B**

The calculation functions can be used in the interrupt and/or additional command lists.

**B**

A calculation function can be started by the following.

- Gate interrupt
- Output interrupt
- "Process command list" control word internally in command list
- "Process command list" control word externally via control word register.

The calculation functions are implemented with a calculation stack of four calculation registers (accumulators) which are all updated accordingly each time a load or calculation command occurs. Each accumulator consists of 32 bits.

The accumulators are accessed with a load command from the result, load hold, interrupt, constant and counter state registers. The calculation results are transferred again to the corresponding reulst register (measured value series).

Flags 0 to 15 are handled in the S5 is the usual manner (except internally in the module). See section 8.7 for an example of using the flags in the command lists.

During the command run time, the calculation results of all calculation functions are checked for validity. In case of an error, processing of the list is terminated, an error message is output, and the number of the erroneous data word is stored in the FAB byte of the ERR parameter ($\rightarrow$ section 3.8.4).

The calculations function are divided into three groups.

- Transfer operations ($\rightarrow$ section 6.2.3)
- Calculation operations ($\rightarrow$ section 6.2.4)
- Comparison operations ($\rightarrow$ section 6.2.5)

## 6.2.2    Control Word Format

**IP 242B**

**B** The control word triggers the corresponding function on the module.  The control words required for a calculation operation must be grouped together in a command list in the order in which they will be executed.

```
15                          8  7                    0
┌──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┬──┐
│  │  │  │  │  │  │  │  │  │  │  │  │  │  │  │  │
└──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┴──┘
```

Command code                   Register          Register number
                                 type                   z
                                  y
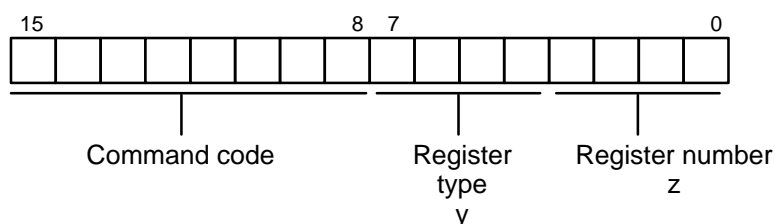
| Abbre–viation | Control word | Meaning | Permissible Parameters |
|---|---|---|---|
| L<br>T | KH90yz<br>KH91yz | Load from register<br>Transfer to register | Register type, register number |
| TAK | KH9200 | Exchange accumulators | – |
| ADD<br>SUB<br>MUL<br>DIV<br>KZD<br>ABS | KH93yz<br>KH94yz<br>KH95yz<br>KH96yz<br>KH97yz<br>KH98yz | Addition<br>Subtraction<br>Multiplication<br>Division (whole numbers)<br>Dual complement generation<br>Absolute value generation | Register type, register number |
| SPR | KHA0ww | Jump, relative | Jump length |
| >D<br>=D<br><D<br>$\geq$D<br>$\leq$D<br>><D<br>FIN<br>FAUS | KHA1yz<br>KHA2yz<br>KHA3yz<br>KHA4yz<br>KHA5yz<br>KHA6yz<br>KHA7yz<br>KHA8yz | Compare for "greater than"<br>Compare for "equals"<br>Compare for "less than"<br>Compare for "greater than/equal to"<br>Compare for "less than/equal to"<br>Compare for "unequal"<br>Compare for "within a window"<br>Compare for "outside a window" | Register type, register number |

ww: → section 6.2.5.1

### 6.2.2.1 Register Type, Register Number

**IP 242B**

| Register | Length | | Register Type y | Register Number z |
|---|---|---|---|---|
| Load | Counters 1 to 5: | 16 bits | 0 | 1 to 7 |
| | Counters 6 and 7: | 32 bits | | |
| Hold (in dual port RAM) | Counters 1 to 5: | 16 bits | 1 | 1 to 7 |
| | Counters 6 and 7: | 32 bits | | |
| Hold (internally) | Counters 1 to 5: | 16 bits | 2 | 1 to 7 |
| | Counters 6 and 7: | 32 bits | | |
| Interrupt | Counters 1 to 5: | 16 bits | 3 | 1 to 7 |
| | Counters 6 and 7 : | 32 bits | | |
| Result | | 32 bits | 4 | 1 to 7 |
| Constants | | 32 bits | 5 | 0 to F |
| Counter value | Counters 1 to 5: | 16 bits | 6 | 1 to 7 |
| | Counters 6 and 7: | 32 bits | | |
| Accumulator | | 32 bits | 7 | 1 to 4 |
| Flag | | 32 bits | 8 | 0 to F |

The register can contain the following number formats.

16–bit length: $0$ to $+(2^{16}-1)$

32–bit length: $-2^{31}$ to $+(2^{31}-1)$.

**B** **6.2.3 Transfer Operations**

**IP 242B**

Before calculation operations can be performed, the calculation registers (accumulators) must be loaded. After the calculation, the calculation results must be stored in registers or flags. The transfer functions are used for this purpose.

The data transfer is only practical when used from or to exactly one register. The corresponding selection is made in the righthand byte of the control word ($\rightarrow$ 6.2.2).

The **contents of the accumulators** do not change from one call of the command list to the next (i. e., if calculation functions are running in more than one command list, the contents of the accumulators before calculation from list 1 may be equal to the contents after calculation from list 2. If calculated values from one call of a calculation are to be forwarded to the next, the values must be stored in registers or flags at the end of the first call.

The **flags** are used as global intermediate storage for results to preset the accumulators with defined values from the previous calculations at the beginning of a calculation (since the contents of the accumulators are undefined at the beginning of a command list).

**6.2.3.1 Load from Register (L)**

**IP 242B**

**B** Control word: KH = 90yz

This control word can be used for all register types and numbers included in section 6.2.2.1.

Description:
Accumulator 1 is loaded with the contents of the specified register. The contents of accumulator 4 are lost. The "new" contents of accumulators 2 to 4 are the "old" contents of accumulators 1 to 3.

If accumulator 1 is loaded with the contents of a register for counters 1 to 5, the register contents are interpreted as a positive number in the range of 0 to 65535. Bits 31 to 16 in accumulator 1 are then "zero".

> A "read counter" is automatically executed during the load operation from a counter value register. This load operation cannot be used for counters 1 to 5 in operating modes G, H, I, J, K, L, S, V and X.

When a nonexistent register is specified as the source, error message KH35 is generated during parameterization, and parameterization is terminated.

**Example:**

```
15                          0
1 0 0 1 0 0 0 0 | 0 1 1 0 0 1 1 1
```
Control:  KH = 9067      Load from counter value register 7
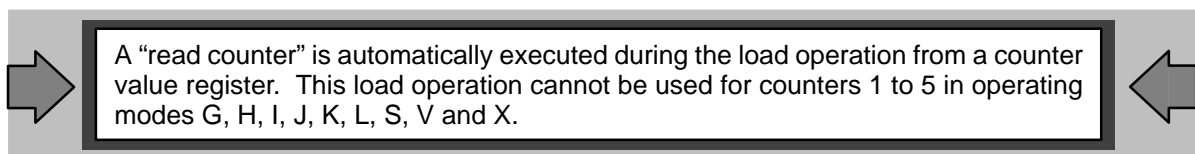
## 6.2.3.2  Transfer to Register (T)

IP 242B

Control word:  KH = 91yz

**B**

This control word can be used for all register types and numbers given in section 6.2.2.1..

Description:
The contents of accumulator 1 are transferred to the specified register and all accumulators remain unchanged.

During the transfer operation to the load register, the "prepare to load" command is also executed for the corresponding counter.  "Load counter" is executed for counters 6 and 7.

The "accept interrupt value" command is automatically executed during the transfer operation in the interrupt register.

A "load counter" is also executed during the transfer operation to the counter value registers.

When a nonexistent register or a constant register is specified as the destination, error message KH36 is generated during parameterization, and the parameterization is terminated.

If, during the command run time, the contents of accumulator 1 < 0 and load, hold, interrupt register or counter value of channels 1 to 5 are selected as the destination, error message KH34 is output, and processing of the list is terminated.

If, during the command run time, the contents of accumulator 1 > 65535 and load, hold, interrupt register or counter value of channels 1 to 5 are selected as the destination, error message KH33 is output, and processing of the list is terminated.

**Example:**

```
15                          0
 1 0 0 1 0 0 0 1 | 1 0 0 0 1 0 1 0
```
          Control word:  KH = 918A  Transfer to flag 10


## 6.2.3.3  Exchange Accumulators (TAK)

IP 242B

Control word:  KH = 9200

**B**

Description:
The contents of accumulator 1 and accumulator 2 are exchanged.
Accumulator 3 and accumulator 4 remain unchanged.

**Example:**

```
15                          0
 1 0 0 1 0 0 1 0 | 0 0 0 0 0 0 0 0
```
          Control word:  KH = 9200  Exchange accumulators

## 6.2.4    Calculation Operations

**IP 242B**

**B**  The following calculation operations can be executed with control words:

| Addition | Accu 2 + Accu 1  $\rightarrow$ <br>1) | Accu 1 |
|---|---|---|
| Subtraction | Accu 2 – Accu 1  $\rightarrow$ <br>1) | Accu 1 |
| Multiplication | Accu 2 * Accu 1 <br>Accu 1          1) | $\rightarrow$ |
| Division | Accu 2 / Accu 1 <br>Accu 1          1) | $\rightarrow$ |
| Dual complement generation | Accu 1 * (–1) <br>Accu 1          2) | $\rightarrow$ |
| Absolute value generation | \|Accu 1\|          $\rightarrow$ <br>2) | Accu 1 |

The contents of an accumulator can be a counting value, a result of a previous calculation or a constant. The accumulators have the following contents after a calculation operation.

- Result is always in accumulator 1.
- For operations identified with "1)"
    - Accumulator 2 (new) = accumulator 3 (old)
    - Accumulator 3 (new) = accumulator 4 (old)
    - Accumulator 4 is unchanged.

- For operations identified with "2)"
    - Accumulators 2 to 4 remain unchanged.

Although **exponentiation** is not available as a separate function, you can easily achieve this by including the appropriate load and multiplication in your command lists.

Example:• Squaring the contents of accumulator 1

Load from accumulator 1

MUL

Transfer to result register 7

- Third power of the contents of accumulator 1

Load from accumulator 1

Load from accumulator 1

MUL

MUL

Transfer to result register 6

The calculation and comparison operations can be used with **implicit load commands** to reduce the number of commands required in a list.  The register type and number in the low byte of the appropriate operation are filled with values not equal to KH = 00.  A load command to accumulator 1 from the register specified by the register type and number is performed before the calculation or comparison operation is executed.  In other words, the contents of accumulator 4 before the operation with the implicit load command are lost (the same as with load commands specified explicitly).

Below is an example of squaring to illustrate this shortcut.

| **Without Implicit Load Command** | | **With Implicit Load Command** | |
|---|---|---|---|
| Load from accu 1 | KH 9071 | MUL accu 1 | KH 9571 |
| MUL | KH 9500 | Transfer to result register 7 | KH 9147 |
| Transfer to result register 7 | KH 9147 | | |

**Extracting a root** is not available as a separate function. Use the "mathematical functions" S5 function blocks for this purpose.  These standard function blocks are available under order number 6ES5 848–xMT01 for PLCs S5–135U and S5–155U.

x = 8          For S5–DOS
x = 7          For MS–DOS, S5–DOS/MT

---

### 6.2.4.1 Addition (ADD)

**IP 242B**

---

**B** Control word: KH = 93yz

Description:
The operation executes the addition of two values. The contents of accumulator 2 are added to the contents of accumulator 1. The result is then available in accumulator 1.

| Function: | Accu 1 := Accu 2 + Accu 1 |
|---|---|
| Value range: | Arguments: $-2^{31}$ to $(+2^{31} -1)$ <br> Result: $-2^{31}$ to $(+2^{31} -1)$ |
| Error message: | Overflow: KH30 <br> Underflow: KH31 <br> The error address is specified in the FAB register. |
| Error reaction: | Termination of the command list. All other commands are not executed. |

**Example:**

```
15                      0
1 0 0 1 0 0 1 1 0 0 0 0 0 0 0 0
```
Control word:  KH = 9300  Add accu 1 and accu 2

---

### 6.2.4.2 Subtraction (SUB)

**IP 242B**

---

**B** Control word: KH = 94yz

Description:
The operation executes the subtraction of two values. The contents of accumulator 1 are subtracted from the contents of accumulator 2. The result is then available in accumulator 1.

| Function: | Accu 1 := Accu 2 − Accu 1 |
|---|---|
| Value range: | Arguments: $-2^{31}$ to $(+2^{31} -1)$ <br> Result: $-2^{31}$ to $(+2^{31} -1)$ |
| Error message: | Overflow: KH30 <br> Underflow: KH31 <br> The error address is specified in the FAB register. |
| Error reaction: | Termination of the command list. All other commands are not executed. |

**Example:**

```
15                      0
1 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0
```
Control word:  KH = 9400  Subtract accu 1 from accu 2

## 6.2.4.3  Multiplication (MUL)

IP 242B

Control word:  KH = 95yz

**B**

Description:
The operation executes the multiplication of two values.  The contents of accumulator 2 are multiplied by the contents of accumulator 1.  The result is available in accumulator 1.

| Function: | Accu 1 := Accu 2 x Accu 1 |
|---|---|
| Value range: | Arguments: $-2^{31}$ to $(+2^{31}-1)$<br>Result:        $-2^{31}$ to $(+2^{31}-1)$ |
| Error message: | Overflow:    KH30<br>Underflow:   KH31<br>The error address is specified in the FAB register. |
| Error reaction: | Termination of the command list.  All other commands are not executed. |

**Example:**

```
15                           0
1 0 0 1 0 1 0 1 | 0 0 0 0 0 0 0 0
```
Control word:  KH = 9500  Multiply accu 1 by accu 2

## 6.2.4.4  Division (DIV)

IP 242B

Control word:  KH = 96yz

**B**

Description:
The operation executes the division of two values.  The contents of accumulator 2 are divided by the contents of accumulator 1.  The result is available in accumulator 1.

| Function: | Accu 1 := Accu 2 / Accu 1 |
|---|---|
| Value range: | Arguments Accu 1: $-2^{31}$ to $(-1)$ and 1 to $(+2^{31}-1)$<br>Arguments Accu 2: $-2^{31}$ to $(+2^{31}-1)$<br>Result:                $-2^3$  to $(+2^{31}-1)$ |
| Error message: | Division by 0:        KH32<br>The error address is specified in the FAB register. |
| Error reaction: | Termination of the command list.  All other commands are not executed. |

**Example:**

```
15                           0
1 0 0 1 0 1 1 0 | 0 0 0 0 0 0 0 0
```
Control word:  KH = 9600  Divide accu 2 by accu 1

---

### 6.2.4.5  Generation of Dual Complement (KZD)

IP 242B

**B**  Control word:  KH = 97yz

Description:
The operation generates the dual complement of the contents of accumulator 1 (negation of the value). The result is available in accumulator 1.

| Function: | Accu 1 := − Accu 1 |
|---|---|
| Value range: | − $(2^{31} - 1)$ to $(+2^{31} - 1)$ |
| Error message: | Overflow for contents − $2^{31}$ in accu 1 before the negation (KH30) The error address is specified in the FAB register. |
| Error reaction: | Termination of the command list.  All other commands are not executed. |

**Example:**

```
15                       0
1 0 0 1 0 1 1 1 | 0 0 0 0 0 0 0 0
```
Control word:  KH = 9700  Generate dual complement of accu 1

---

### 6.2.4.6  Generation of the Absolute Value (ABS)

IP 242B

**B**  Control word:  KH = 98yz

Description:
The operation generates the absolute value of the contents of accumulator 1.  The result is available in accumulator 1.

| Function: | Accu 1 := \|Accu 1\| |
|---|---|
| Value range: | Argument:    − $(2^{31} - 1)$ to $(+2^{31} - 1)$ Result:        0 to $(+2^{31} - 1)$ |
| Error message: | Overflow for contents − $2^{31}$ in accu 1 before absolute value generation (KH30) The error address is specified in the FAB register. |
| Error reaction: | Termination of the command list.  All other commands are not executed. |

**Example:**

```
15                       0
1 0 0 1 1 0 0 0 | 0 0 0 0 0 0 0 0
```
Control word:  KH = 9800  Generate absolute value of accu 1

---

### 6.2.5    Comparison Operations

The following comparison operations can be executed with control words.

| Operation | Comparison result is "true" if ... |
|-----------|-----------------------------------|
| > D | Accu 2   >   Accu 1 |
| = D | Accu 2   =   Accu 1 |
| < D | Accu 2   <   Accu 1 |
| $\geq$ D | Accu 2   $\geq$   Accu 1 |
| $\leq$ D | Accu 2   $\leq$   Accu 1 |
| >< D | Accu 2   $\neq$   Accu 1 |
| FIN | Accu 2   $\leq$   Accu 1   $\leq$   Accu 3   or<br>Accu 3   $\leq$   Accu 1   $\leq$   Accu 2 |
| FAUS | Accu 1   <   Accu 2 or Accu 1   >   Accu 3   or<br>Accu 1   <   Accu 3 or Accu 1   >   Accu 2 |

The use of a comparison is only practical immediately prior to a jump command.  The results of the comparison determine whether the jump will be executed ($\rightarrow$ section 6.2.5.1).

The comparison result is **only current for a control word directly following a comparison**.  It is automatically set to "true" afterwards.  This changes a "conditional" jump to an "unconditional" jump.

The comparison operation compares the contents of accumulator 2 (e.g., constants) with the contents of accumulator 1.  Similar to the calculation operations, the control word can be provided with an implicit load command.  The comparison result is generated again for each comparison.

## 6.2.5.1  Relative Jump (SPR)

**IP 242B**

**B**

Control word:  KH = A0ww

Description:
If the comparison result is "true", a jump relative to the current address is executed.  Both forward and backward jumps are permitted.

> For this reason, be sure that the backward jumps in your command list do not cause continuous loops or loops with many passes since this can block the module.
> When the module is blocked for more than 100 msec, the HW watchdog resets the module and error message KH0C is output.
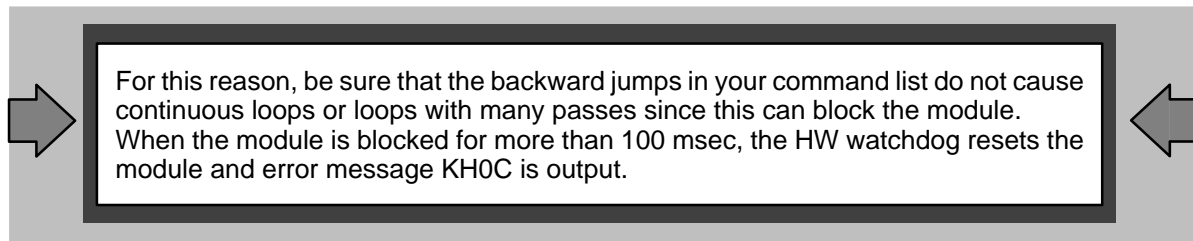
**No** indication is made as to whether or not the conditional jump was executed.

The less significant byte of the control word specifies the relative length of the jump in the range from −128 to +127 data words.

Determination of the Relative Jump Length for Conditional Jumps

| Control Word Sequence in the Command List | Jump length for Jump to Corresponding Control Word | Control Word "Conditional Jump to Corresponding Control Word" |
|---|---|---|
| Load from counter value 2 | −3 | A0FD |
| Load constant 7 | −2 | A0FE |
| >D? | −1 | A0FF |
| Conditional jump | 0 | A000 |
| Transfer to flag 7 | +1 | A001 |
| MUL | +2 | A002 |
| Transfer to result 1 | +3 | A003 |

(Jump lengths of 0 and +1 serve no useful purpose but are not blocked by the firmware.)

A jump to a position specified outside the current list is detected during parameterization, and the parameterization is terminated with error message KH37.

**Example:**

```
15                          0
1 0 1 0 0 0 0 0 0 0 0 0 0 1 0 1
```
Control word:  KH = A005  Relative jump 5 data words forward

## 6.2.5.2 Compare for "Greater Than" (> D)

IP 242B

Control word:  KH = A1yz

**B**

Description:
The comparison result is "true" when the contents of accumulator 2 > the contents of accumulator 1.

**Example:**

```
15                          0
 1 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0
```
Control word:  KH = A100  Compare whether accu 2 > accu 1

## 6.2.5.3 Compare for "Equal To" (= D)

IP 242B

Control word:  KH = A2yz

**B**

Description:
The comparison result is "true" when the contents of accumulator 2 = the contents of accumulator 1.

**Example:**

```
15                            0
 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0
```
Control word:  KH = A200  Compare whether accu 2 = accu 1

## 6.2.5.4 Compare for "Less Than" (< D)

IP 242B

Control word:  KH = A3yz

**B**

Description:
The comparison result is "true" when the contents of accumulator 2 < the contents of accumulator 1.

**Example:**

```
15                          0
 1 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0
```
Control word:  KH = A300  Compare whether accu 2 < accu 1

## 6.2.5.5 Compare for "Greater Than/Equal To" (≥ D)

IP 242B

**B** Control word: KH = A4yz

Description:
The comparison result is "true" when the contents of accumulator 2 ≥ the contents of accumulator 1.

**Example:**

```
15                        0
 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0
```
Control word: KH = A400  Compare whether accu 2 ≥ accu 1

## 6.2.5.6 Compare for "Less Than/Equal To" (≤ D)

IP 242B

**B** Control word: KH = A5yz

Description:
The comparison result is "true" when the contents of accumulator 2 ≤ the contents of accumulator 1.

**Example:**

```
15                        0
 1 0 1 0 0 1 0 1 0 0 0 0 0 0 0 0
```
Control word: KH = A500  Compare whether accu 2 ≤ accu 1

## 6.2.5.7 Compare for "Not Equal" (>< D)

IP 242B

**B** Control word: KH = A6yz

Description:
The comparison result is "true" when the contents of accumulator 2 ≠ the contents of accumulator 1.

**Example:**

```
15                        0
 1 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0
```
Control word: KH = A600  Compare whether accu 2 ≠ accu 1

## 6.2.5.8  Compare for "Within a Window" (FIN)

**IP 242B**

Control word:  KH = A7yz

**B**

Description:
The operation compares whether accumulator 1 is within the window limited by accumulator 2 and accumulator 3.
The comparison result is "true" if the following is valid.

$$\text{Accu } 2 \leq \text{Accu } 1 \leq \text{Accu } 3 \text{ or}$$

$$\text{Accu } 3 \leq \text{Accu } 1 \leq \text{Accu } 2$$

**Comparison result**
"False"

(Contents of accu 3 or accu 2)
**Upper limit value**

● X1

**Window**

● X2

"True"

● X3

● X4

**Lower limit value**
(Contents of accu 2 or accu 3)

"False"

(Xn: Contents of accu 1)

The comparison is "true" for X1, X2, X3 and "false" for X4.

**Example:**

| 15 | | | | | | | | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 0 | **1** | 0 | 0 | **1** | **1** | **1** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Control word:   KH = A700  Compare whether accu 1 is within
                           the window limited by accu 2
                           and accu 3.

## 6.2.5.9 Compare for "Outside a Window" (FAUS)

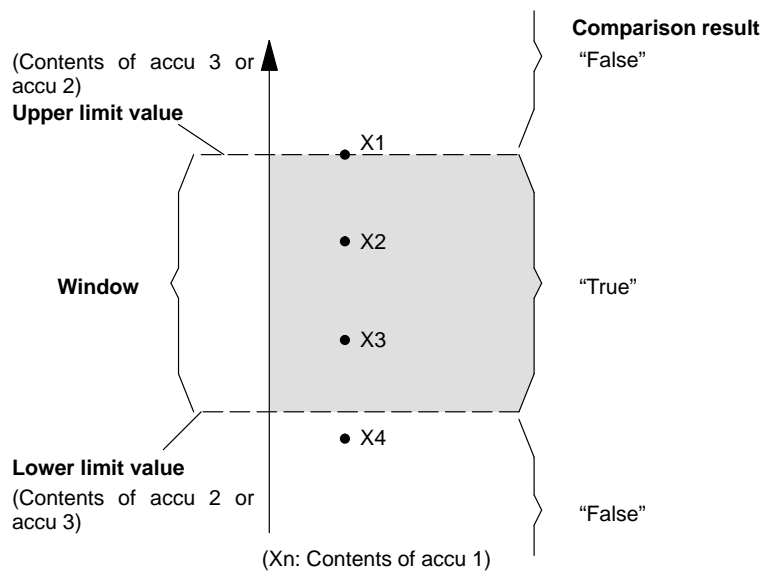**IP 242B**

**B** Control word: KH = A8yz

Description:
The operation compares whether accumulator 1 is outside the window limited by accumulator 2 and accumulator 3.
The comparison result is "true" if the following is valid.

Accu 1 < Accu 2 or Accu 1 > Accu 3 or

Accu 1 < Accu 3 or Accu 1 > Accu 2

.



(Xn: Contents of accu 1)

The comparison is "true" for X4 and "false" for X1, X2, X3.

**Example:**

| 15 | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|
| **1** 0 **1** 0 **1** 0 0 0 | 0 0 0 0 0 0 0 0 |

Control word:    KH = A800  Compare whether accu 1 is
outside the window limited by
accu 2 and accu 3.

# 7     Special Functions

## 7.1    Cascading (Counters 1 to 5)

Counter cascading is used to expand the counting area.

Only counters 1 to 5 can be cascaded. This provides a maximum counting area of $2^{80}-1$ for binary counting or $10^{20}-1$ for decimal counting.

You can cascade only counters which directly follow each other (i.e., counter output 1 to counter input 2, counter output 2 to counter input 3, and so on, up to counter output 5 to counter input 1).

**• Principle of Cascading**
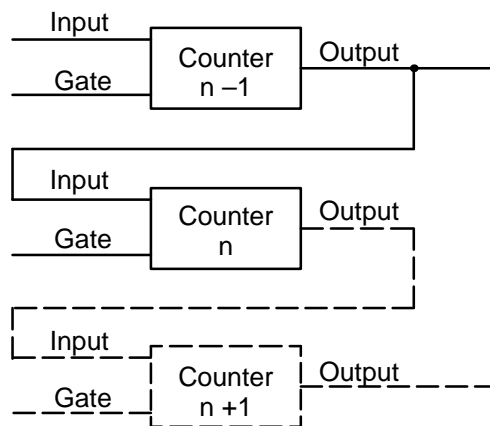


Specify counter output n–1 in bits 8 to 11 of the counter mode register as the counting pulse source. External circuitry is not required.

**• Special Case 1**
**A counter output is switched to the gate input of the counter directly following.**
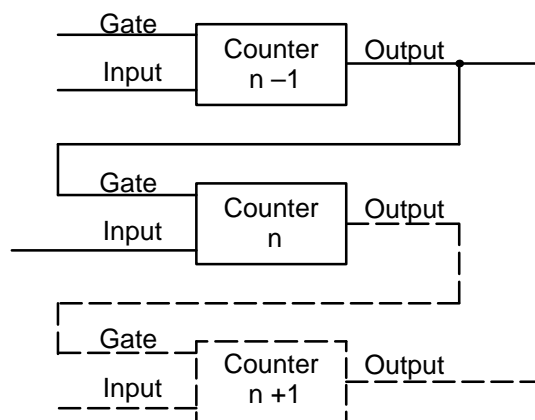


Specify counter output n –1 for a rising cascade in bits 13 to 15 of the counter mode register as the selected gate. Select gate n +1 instead of counter output n –1 to cascade the gates in falling sequence. External circuitry is not required.

- **Special Case 2**
  **Counters n and n −1 are cascaded. Counter n +1 is used here as the prescaler for the gate of the counter.**

```
        Gate
     ──────────┌──────────┐
                │ Counter  │ Output
        Input   │   n −1   │─────────●───────────
     ──────────└──────────┘

           Input
      ─────────┌──────────┐
                │ Counter  │ Output
           Gate │    n     │──────────────────
      ─────────└──────────┘

        Gate
     ──────────┌──────────┐
                │ Counter  │ Output
        Input   │   n +1   │────────────
     ──────────└──────────┘
```
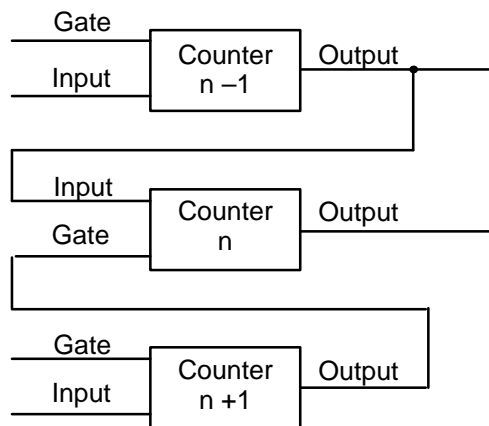
Specify counter output n −1 in bits 8 to 11 of the counter's counter mode register as the counting pulse source.

Select the high level, gate n +1 setting as the choice of gate (bits 13 to 15).

## 7.2     Command Lists for Interrupt Processing

For each of the 5 counters 1 to 5, you can define 5 commands to be executed at a gate interrupt and 5 commands to be executed at an interrupt from the counter output.

For each of the 2 counters 6 and 7, you can define 5 commands to be executed at a counter reset and 5 commands to be executed at an interrupt from a counter output.

Enter the commands for each counter in the parameterization data block ($\rightarrow$ section 11.6 or 14.6). Use the "parameterize counters" command to transfer these interrupt command lists to the module. A syntax check is performed at the same time.

Error numbers KH11 to KH1E are output (parameter ERR in the parameterization data block) for incorrect commands.   **A**

When the IP 242B is used, error number KH10 is output for erroneous commands. The FAB byte of the ERR parameter contains the number of the applicable data word.   **B**

All control words in section 6.1.1 can be used as commands in the command lists. However, several control words (e.g., "test function" and "take over basic setting") are of little use to you.   **A**

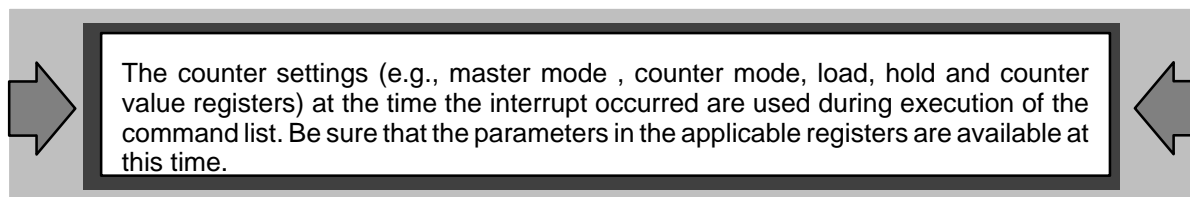The control words listed in table 6.1.1 can be used in the command lists for the IP 242B. All calculation functions are permitted in the interrupt command lists.   **B**

Enter the control words contiguously in the command list. Parameterize unassigned areas with KH = 0000 (i.e., when control word sequences have less than 5 control words, the control word sequence of the interrupt command list must be concluded with KH = 0000).

If several counter interrupts occur at the same time, the command lists are handled in the following order:

Highest priority:    Output  1
                     Output  2
                     Output  3 ⎫
                     Output  4 ⎬  For disabled
                     Output  5 ⎭  comparator function
                     Gate 1
                     Gate 2
                     Gate 3
                     Gate 4
                     Gate 5
                     Reset 6
                     Reset 7
                     Output  3 ⎫
                     Output  4 ⎬  For enabled
                     Output  5 ⎭  comparator function
                     Output  6
Lowest priority:     Output  7

**A**  Processing of a command list cannot be interrupted by any other counter interrupts or control words of the function block.

> The counter settings (e.g., master mode , counter mode, load, hold and counter value registers) at the time the interrupt occurred are used during execution of the command list. Be sure that the parameters in the applicable registers are available at this time.

IP 242A/242B Equipment Manual
                                    © Siemens AG 1993, Order No: 6ES5 998–0KM21

## 7.3    Time Measurement (Counters 1 to 5)

Timed processes can be acquired by the IP 242A/242B by using the following circuitry:

1. A reference frequency of F1 to F6 is generated and applied to the counter input of one counter.

2. The event to be measured is applied to the gate of the counter.

3. The measured value corresponds to the number of counting pulses counted while the gate is open.

Counter value
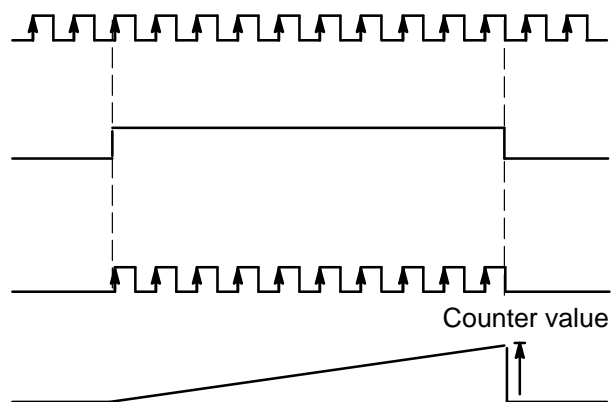
#### Performing Time Measurement

Operation modes E and N are particularly suited to time measurement.

#### Time Measurement with Operation Mode E

Counting pulses are counted only when the counter gate is active. For an inactive gate edge, parameterize a control word sequence as follows:

1. Transfer the counter value from the counter to the internal hold register with the "save counter" command.

2. Load the counters with the value KH = 0000 with the "load counter" command.

Time sequence:

t $\triangleq$ gate open

$\underline{\int}$ $\triangleq$ active gate edge

The counter value is transferred to the internal hold register at this time.
The counter is loaded with the contents of the load register (i.e., zero).
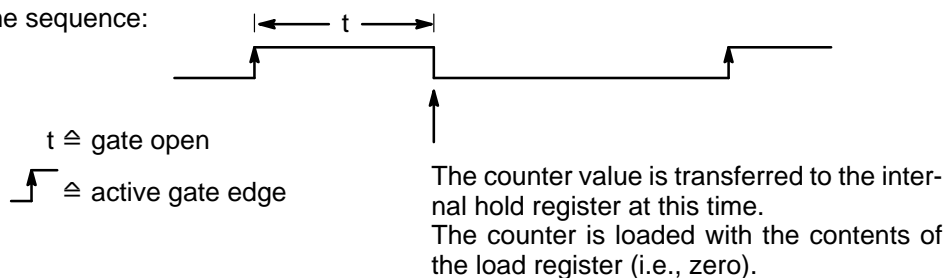
Use the "copy counter" command from the function block to transfer the counter status stored in the internal hold register to the parameterization data block.

## Assignment of the Registers

The prescaler register is assigned so that F1 equals 1 kHz. The number of counting pulses at the end of a measuring cycle, therefore, corresponds to the time in msec.
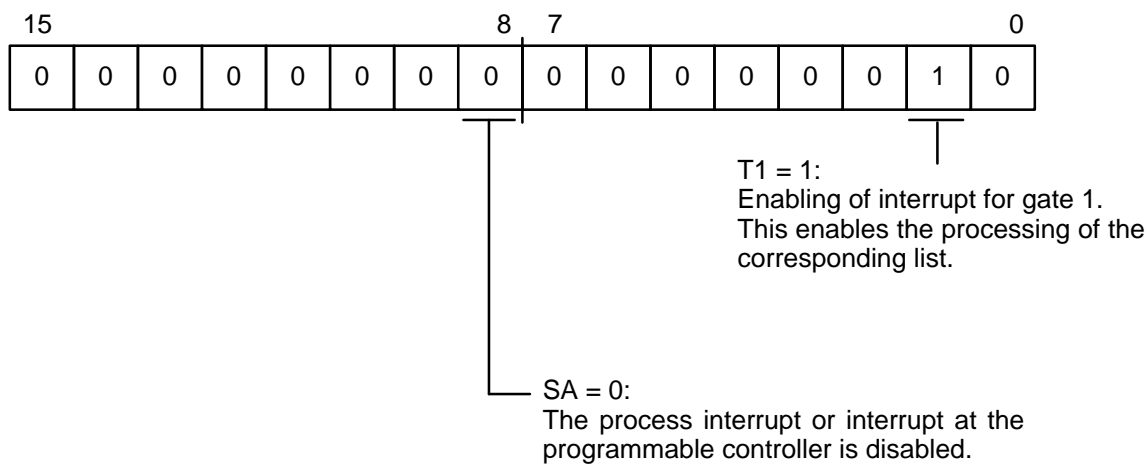
Master mode register:       MMR = KH0000

Prescaler register:         VTR = KF+1000 (scaling factor of 1000)

Gate control register:      TSR = KH0000

Interrupt enable register:  IFR = KH0002
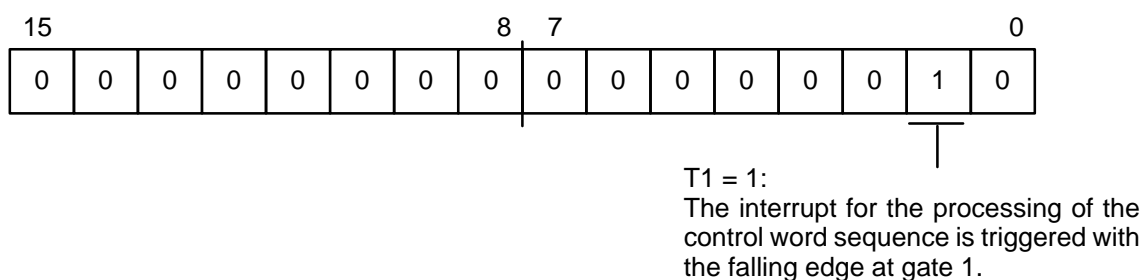
| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

T1 = 1:
Enabling of interrupt for gate 1.
This enables the processing of the corresponding list.

SA = 0:
The process interrupt or interrupt at the programmable controller is disabled.

Interrupt polarity register:  IPR = KH0002

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

T1 = 1:
The interrupt for the processing of the control word sequence is triggered with the falling edge at gate 1.

Counter mode register:     CMR = KH8B28

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |

Output signal switched off

Counting direction up

Binary counting mode

Operation mode E

Counting pulse source F1

Rising counting pulse edge

Gate selection, high level, gate 1
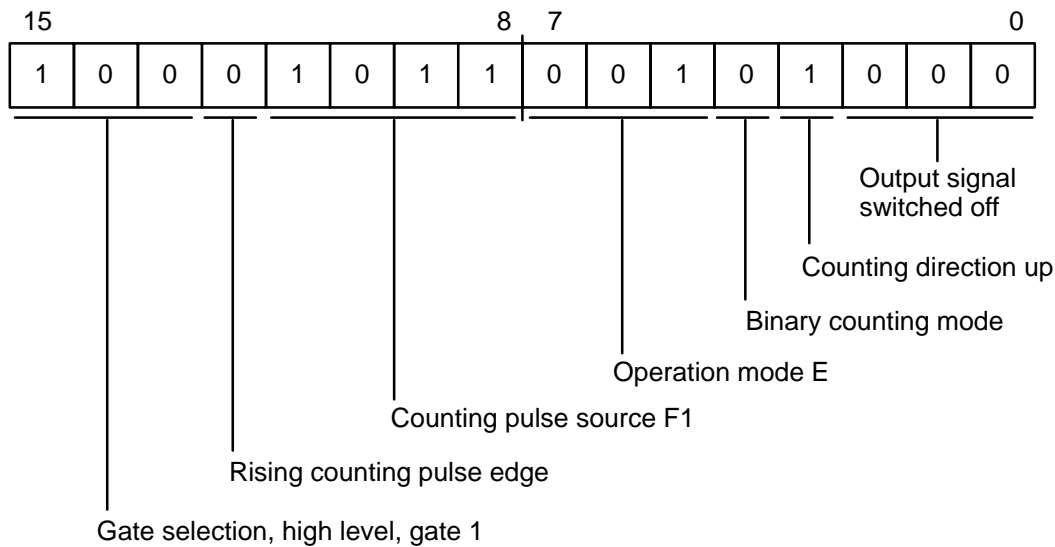
Load register:              LR   =  KH0000

Interrupt register:         AR   =  KHxxxx (any number except 0000)


**Control Word Sequence in Parameterization Data Block for Interrupt from Counter Gate**

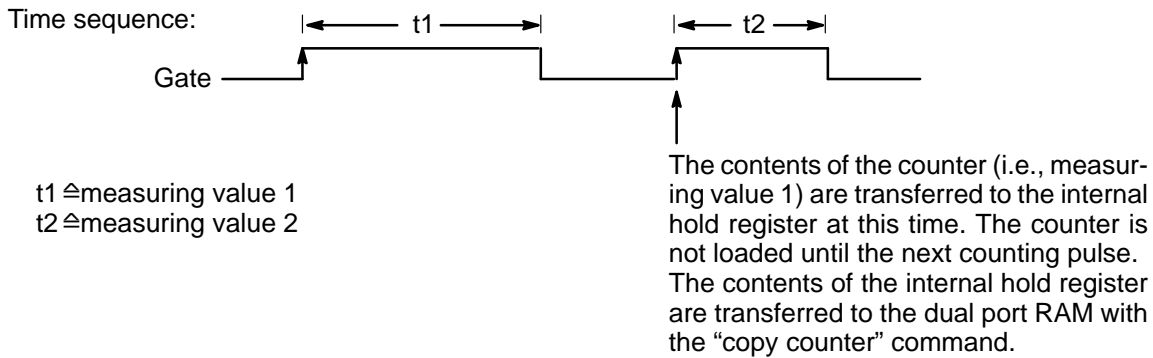| Command 1: | DW20 | = | KH1602 SV | Transfer counter value from the counter to the internal hold register ("save counter"). | |
|---|---|---|---|---|---|
| Command 2: | DW21 | = | KH1202 LD | The counter is loaded with the contents of the load register. The counter remains started. | **A** |
| Command 2: | DW21 | = | KH3702 LD | The counter is loaded with the contents of the load register. The counter remains started. | **B** |
| Command 4: | DW22 | = | KH0000 BE | End of the control word sequence | |

The "parameterize counters" command causes all parameters of the selected counter and the global registers to be taken over. "Start counter" starts the counter.

**Time Measurement with Operation Mode N**

The counting pulses are counted only when the gate is active. The contents of the counter are stored in the internal hold register after the active gate edge is applied. The counter is reloaded with the contents of the load register as soon as the first counting pulse occurs after the gate is opened. If a counting pulse does not occur (i.e., = time length zero), the counter value of the last measuring process is retained in the counter.

Proceed as follows when the **time length zero** is **not** to be **measured**:
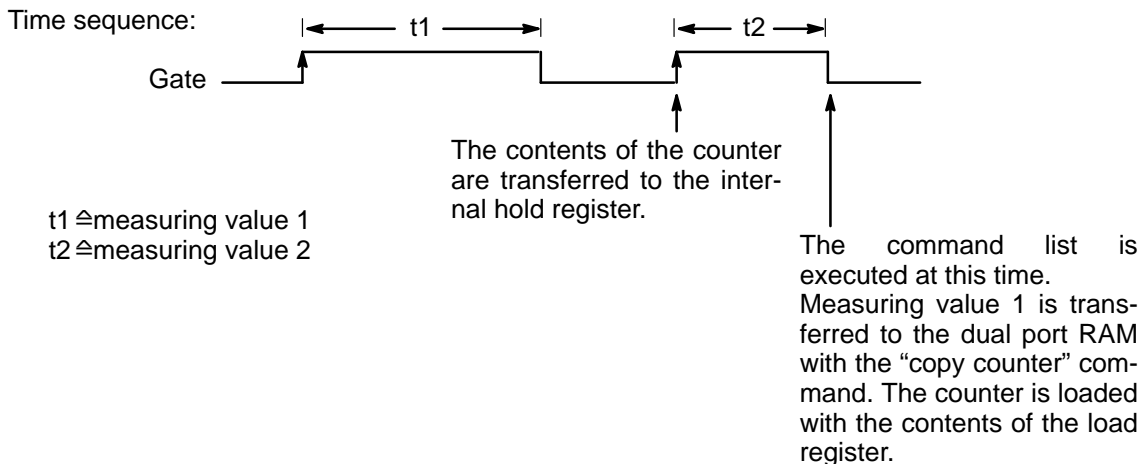
Parameterize control word "copy counter" for an active gate edge.

Time sequence:

Gate

t1 ≙ measuring value 1
t2 ≙ measuring value 2

The contents of the counter (i.e., measuring value 1) are transferred to the internal hold register at this time. The counter is not loaded until the next counting pulse. The contents of the internal hold register are transferred to the dual port RAM with the "copy counter" command.

Parameterize the following control words in the command list when the **time length zero** is to be **measured**:

1. Copy the internal hold register to the dual port RAM with the "copy" command.
2. Load the counter with the value KH = 0000.

The control word sequence is parameterized for an inactive gate edge.

Time sequence:

Gate

t1 ≙ measuring value 1
t2 ≙ measuring value 2

The contents of the counter are transferred to the internal hold register.

The command list is executed at this time. Measuring value 1 is transferred to the dual port RAM with the "copy counter" command. The counter is loaded with the contents of the load register.

Use the "read module" command for the IP 242A from the standard function block or the "read register" command for the IP 242B from the standard function block to transfer the data stored in the dual port RAM to the parameterization data block.

## 7.4    Frequency Measurement (Counters 1 to 5)

Frequencies can be measured by the IP 242A/242B by using the following circuitry:

1. Apply an internal clock pulse (e.g., 0.5 Hz) to the **gate** of a counter. The gate is now open for 1 sec.

2. Apply the frequency to be measured to the **input** of the same counter.

3. The counter value corresponds to the frequency in Hz.
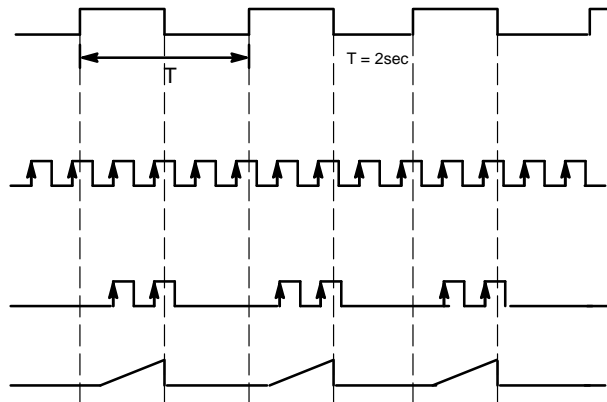
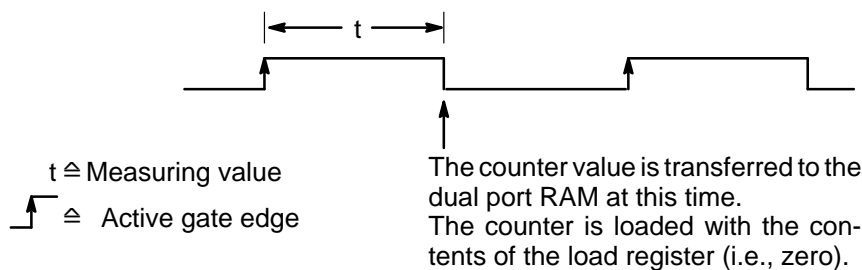**Performing Frequency Measurement:**

Operation modes E and N are particularly suited for frequency measurement.

**Frequency Measurement with Operation Mode E**

Counting pulses are counted only when the counter gate is active. For an inactive gate edge, parameterize a control word sequence as follows:
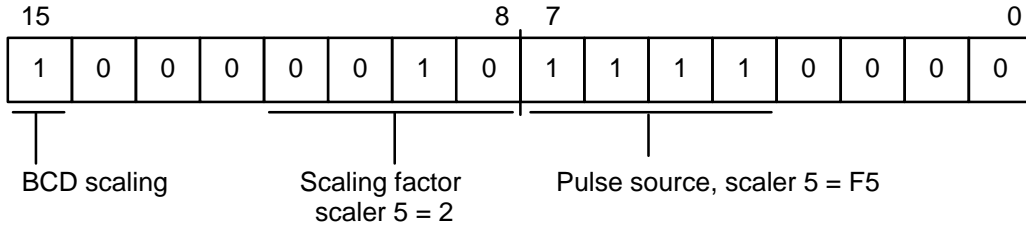
1. Transfer the counter value from the counter to the dual port RAM ("read counter").
2. Load the counter with the value KH = 0000.

Time sequence:

t ≙ Measuring value

⌐ ≙ Active gate edge

The counter value is transferred to the dual port RAM at this time.
The counter is loaded with the contents of the load register (i.e., zero).

**Assignment of the Registers:**

Master mode register:              MMR =   KH 82F0

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |

BCD scaling          Scaling factor          Pulse source, scaler 5 = F5
                     scaler 5 = 2

Prescaler register:              VTR   =   KF +100  (scaling factor 100)

Total scaling factor  = prescaler x scaler 1 x scaler 2 x scaler 3 x scaler 4 x scaler 5
                      =   100     x   10    x   10    x   10    x   10    x   2
                      =   $2 \times 10^6$

Gate open time is therefore 1sec.

Gate control register:              TSR   =   KH0002
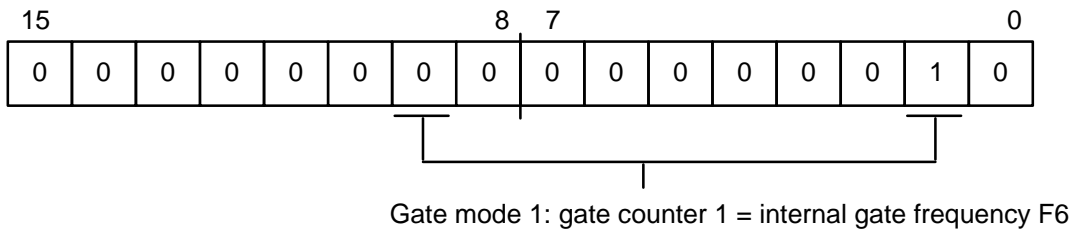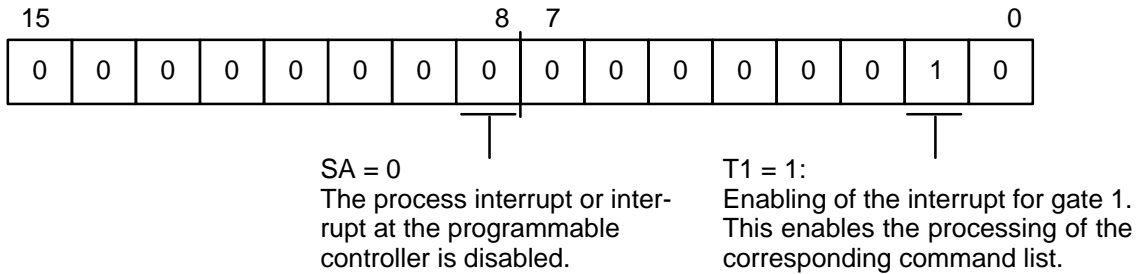
| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

Gate mode 1: gate counter 1 = internal gate frequency F6

Interrupt enable register:        IFR   =   KH0002

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

SA = 0                               T1 = 1:
The process interrupt or inter-      Enabling of the interrupt for gate 1.
rupt at the programmable             This enables the processing of the
controller is disabled.             corresponding command list.

Interrupt polarity register:          IPR =  KH0002

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
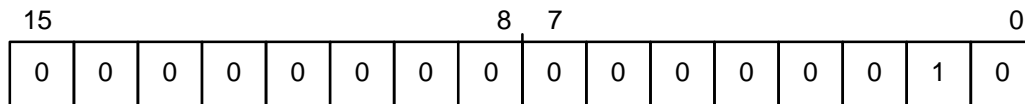
T1 = 1:
The interrupt for the processing of the
control word sequence is triggered with
the falling edge of gate 1.

Counter mode register:          CMR =  KH8128
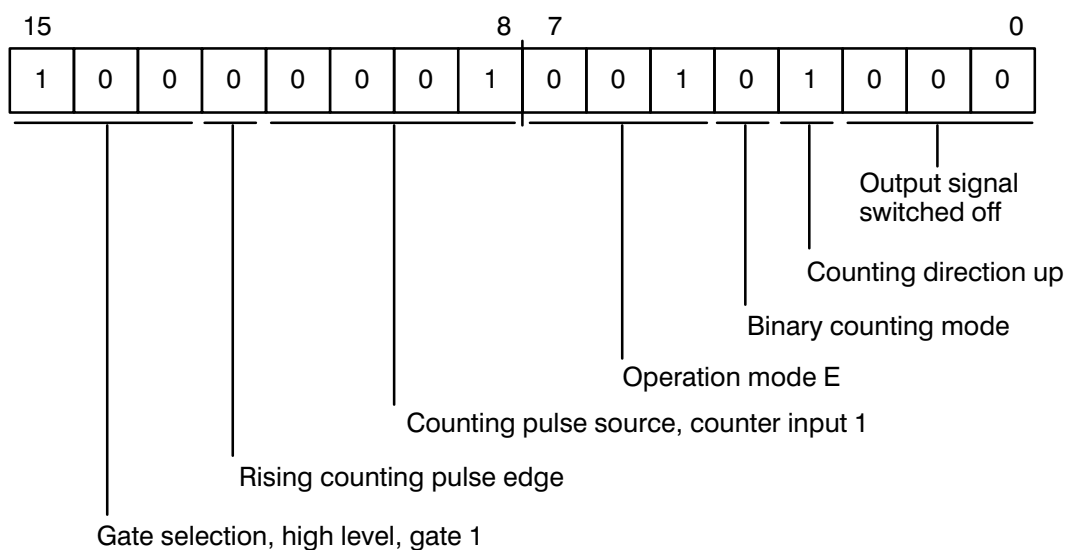
| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |

Output signal
switched off

Counting direction up

Binary counting mode

Operation mode E

Counting pulse source, counter input 1

Rising counting pulse edge

Gate selection, high level, gate 1

Load register:              LR  =  KH0000

Interrupt register:         AR  =  KHxxxx (any number except 0000)

**Control Word Sequence in Parameterization Data Block for Interrupt from Counter Gate**

Command 1 DW20 = KH3202      LE      Transfer counter value from the counter to the dual port
                                                            RAM ("read counter").

Command 2 DW21 = KH1202      LD      Counter 1 is loaded with the contents of the load register.  **A**
                                                            The counter remains started.

Command 2 DW21 = KH3702      LD      Counter 1 is loaded with the contents of the load register.  **B**
                                                            The counter remains started.

Command 3 DW22 = KH0000      BE      End of the control word sequence

The "parameterize counters" command causes all parameters of the selected counters and the global
registers to be taken over. "Start counter" starts the counter.
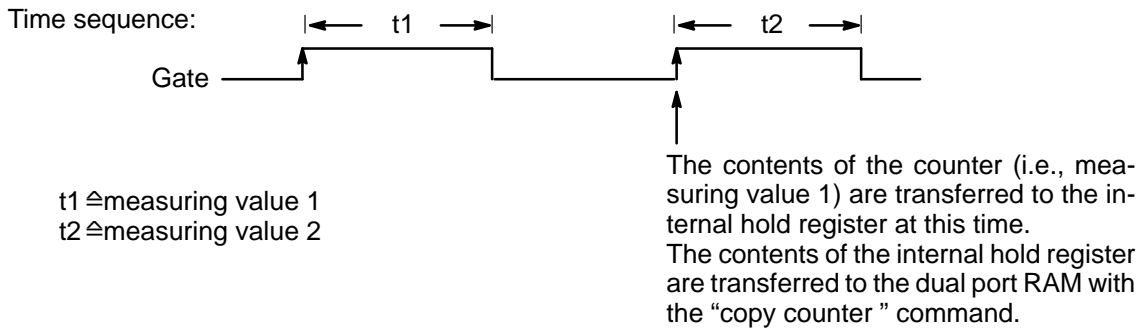
After the falling edge, you can use the "read module" command in the function block to transfer the value
from the dual port RAM to the parameterization data block.

**Frequency Measurement with Operation Mode N**

The counting pulses are counted only when the gate is active. The contents of the counter are loaded in the internal hold register when a gate edge is active. The counter is reloaded with the contents of the load register as soon as the first counting pulse occurs after the gate is opened. If a frequency of zero occurs (i.e., no pulses while the gate is active), the counting value of the last measuring process is retained in the counter.

Proceed as follows when **frequency zero** is **not** to be **measured**:
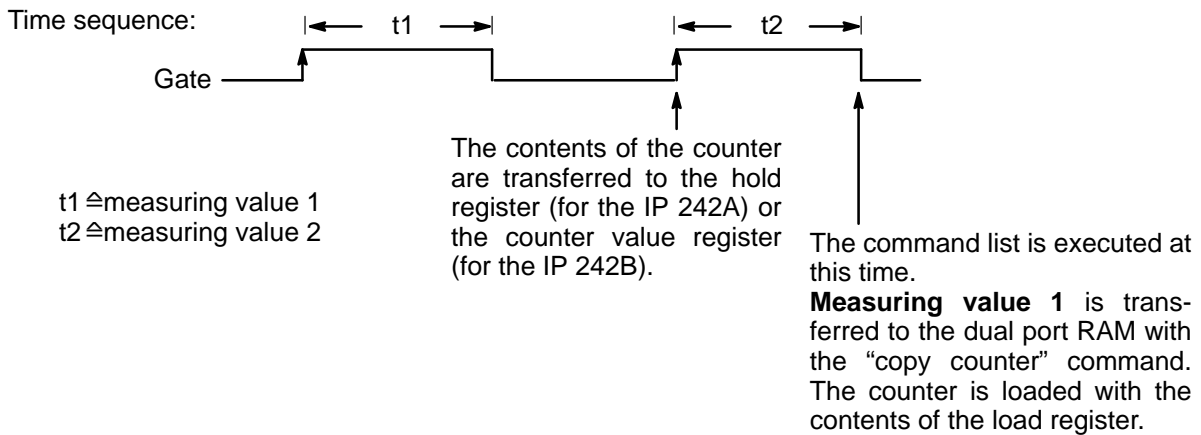
Parameterize control word "copy counter" for an active gate edge.

Time sequence:



t1 ≙ measuring value 1
t2 ≙ measuring value 2

The contents of the counter (i.e., measuring value 1) are transferred to the internal hold register at this time.
The contents of the internal hold register are transferred to the dual port RAM with the "copy counter " command.

Parameterize the following control words in the command list when **frequency zero** is to be **measured**:

1. Copy the internal hold register to the dual port RAM with the "copy" command.
2. Load the counter with the value KH = 0000.

The control word sequence is parameterized for an inactive gate edge.

Time sequence:



t1 ≙ measuring value 1
t2 ≙ measuring value 2

The contents of the counter are transferred to the hold register (for the IP 242A) or the counter value register (for the IP 242B).

The command list is executed at this time.
**Measuring value 1** is transferred to the dual port RAM with the "copy counter" command. The counter is loaded with the contents of the load register.

Use the "read module" command for the IP 242A from the standard function block or the "read register" command for the IP 242B from the standard function block to transfer the counter value stored in the dual port RAM to the parameterization data block.

## 7.5      Velocity Measurement with Light Barriers (Counters 1 to 5)

1. Velocity is measured with two light bar–
   riers. The signals are applied to the start
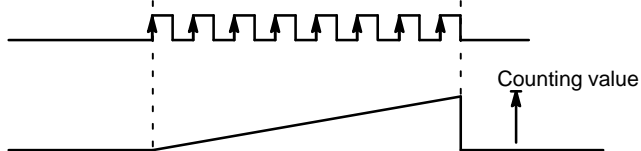   input (STA) and stop input (STO).
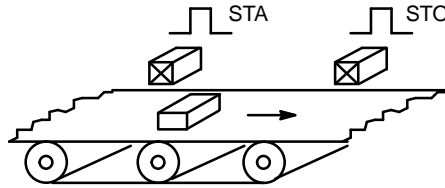
2. A counter gate is controlled by pulses
   STA and STO.
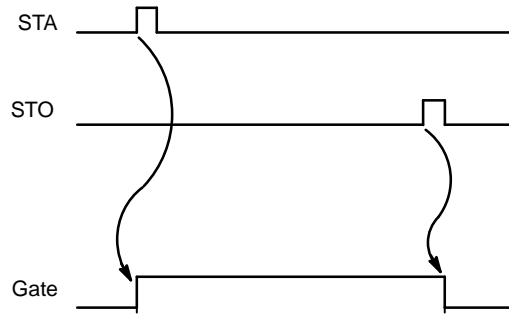
3. A reference frequency is applied to the
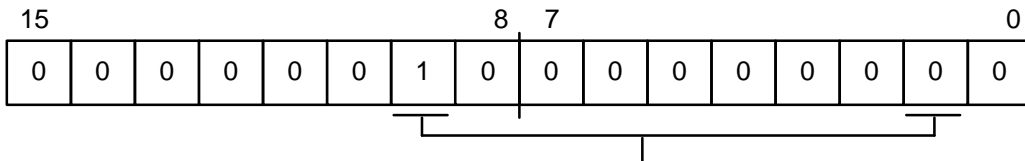   counting input of the counter.

4. The counting value represents the run-
   ning time of the work piece. The speed
   can be calculated if the distance be-
   tween the light barriers is known.

**Performing Velocity Measurement**

This example of velocity measurement is based on the time measurement described in section 7.3. The
determined time in msec is the running time of the work. Only the parameterization of the gate control
register is different. Parameterize it as follows:

TSR = KH0002

| 15 | | | | | | 8 | 7 | | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Gate mode 3:
The gate is opened with a rising edge at STA,
and closed with a rising edge at STO.

When the standard function block is used, the value of the internal hold register is transferred to the para-
meterization data block with the "copy counter" command.

Section 8.2.1 contains an example for the IP 242B using the calculation functions to measure speed.

## 7.6       Synchronization / Zero Point Shift (Counters 6 and 7)

The reference point is assigned a value other than zero with the zero point shift.

1. In this case, the "counter reset" signal cannot trigger the counters directly. You must, therefore, set bit 7 (i.e., counter reset) of the counter mode register to zero.

2. Write the value of the zero point shift in the load register.

3. Enter the following in the command list for the reset input of counter 6:

>       Command 1:   3140 "LS"   Load and start counter 6
>       Command 2:   0000 "BE"   End of the control word sequence

   Accordingly, enter the following in the command list for counter 7:

>       Command 1:   3180  "LS"  Load and start counter 7
>       Command 2:   0000  "BE"  End of the control word sequence

4. Enter a "1" in bit 6 (or bit 7) of the interrupt enable register to execute an available command list when the reference point is reached. In addition, enter a "1" in bit 8 (SA bit) to transmit an interrupt to the S5 CPU also.

5. The new parameters are transferred to the module with the "parameterize counter" (global register) command.
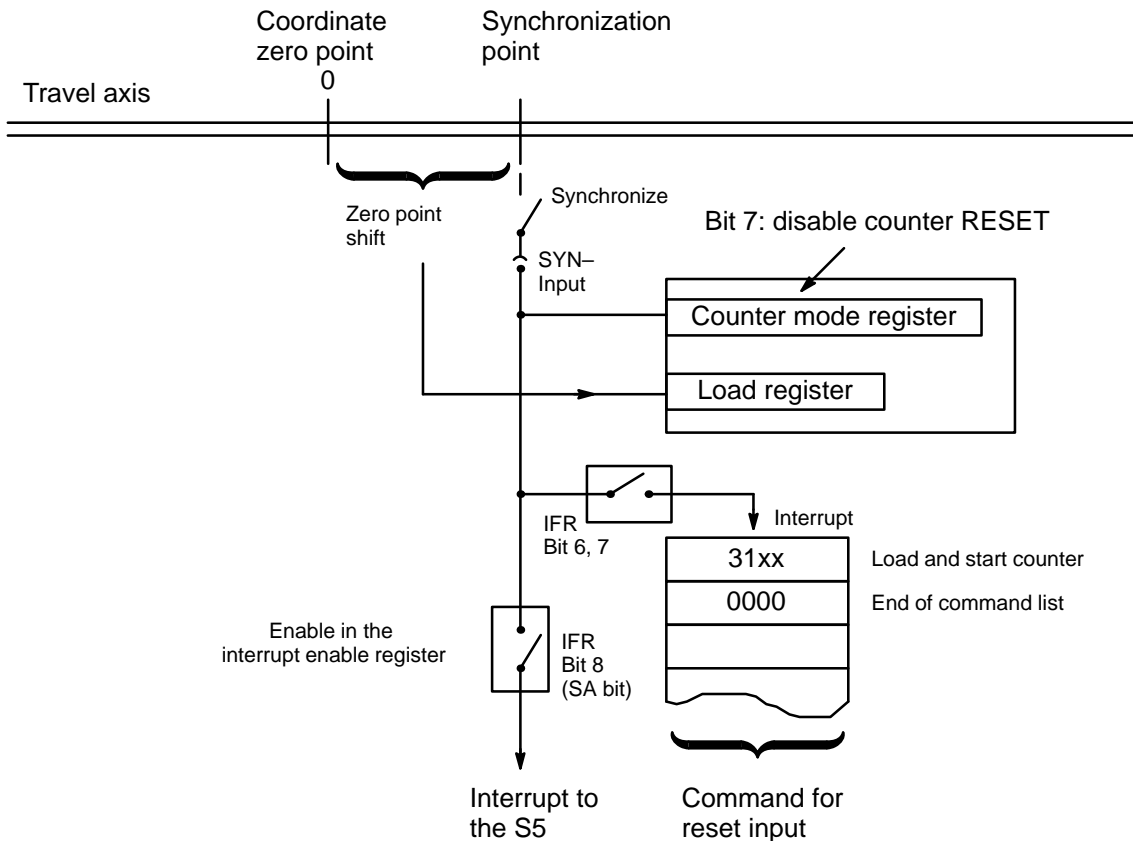
## 7.7 Read Counter Status Via Edge on External Input

An interrupt can be generated by a positive edge on a synchronization input, and the counter status can be "frozen" via "read counter".
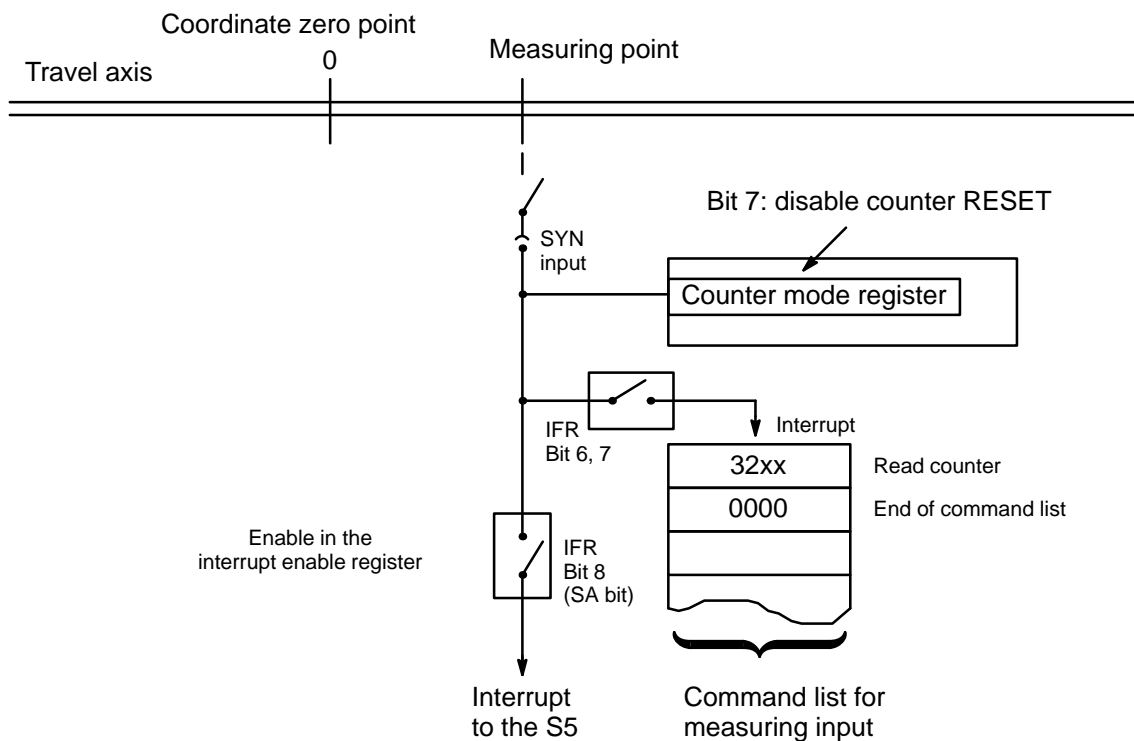
1. The external control signal is applied to the synchronization input.

2. The zero counting pulse input of the counter is specified: N to 5 V, $\overline{N}$ to mass.

3. In this case, the "reset counter" signal may not trigger the counters directly. Thus, you must set bit 7 (reset counter) of the counter mode register to zero.

4. Enter the following in the command list for the reset input of counter 6:

> Command 1:  3240  "LE"  Read counter 6
> Command 2:  0000  "BE"  End of control word sequence

Accordingly, enter the following in the command list for counter 7:

> Command 1:  3280  "LE"  Read counter 7
> Command 2:  0000  "BE"  End of control word sequence

5. Enter a "1" in bit 6 (or bit 7) of the interrupt enable register to execute an available command list when the measuring point is reached. In addition, enter a "1" in bit 8 (SA bit) to transmit an interrupt to the S5 CPU also.

6. The new parameters are transferred to the module with the "parameterize counter" (global register) command.

# 8    Calculation Functions

## 8.1    General

This section uses various examples to show you how to use the calculation functions on the IP 242B. The calculation functions can be used in the interrupt and/or additional command lists.

> All counters used in sections 8.2 to 8.6 retain their parameterization. In section 8.7, counter C1 is reparameterized.

The constant and result registers, and the additional command lists remain permanently assigned.

**Constant Registers:**

KON0     Distance between the light barriers on conveyor belt 1
KON1     Divident of the gearing factor (= 13)
KON2     Divisor of the gearing factor (= 8)
KON3     Distance between the light barriers on conveyor belt 2
KON4     Divisor for recalculation of fine dosing
KON5     Plaster density (dividend = 23)
KON6     Plaster density (divisor = 10)
KON7     Length value (= 1.20 m)
KON8     Length value (= 0.30 m)
KON9     Number of passes (= 8)
KON10    Value 1 }
KON11    Value 0 } For presetting of the internal "pass" counter

**Result Registers:**

ERG1     Speed of conveyor belt 1
ERG2     Length of a workpiece
ERG3     Length of a workpiece (via gearing factor)
ERG4     Speed of conveyor belt 2
ERG5     Weight (plaster)

Additional command lists require that the respective start address of a command list be entered in the directory of the additional command lists. The length of the list is determined by the difference to the start address of the next list ($\rightarrow$ section 3.6.1).

The **directory of the additional command lists** has the contents for our examples:

| | | | | List Length |
|---|---|---|---|---|
| DW178 | XX | 186 | List 1: | 4 DW |
| DW179 | XX | 190 | List 2: | 4 DW |
| DW180 | XX | 194 | List 3: | 5 DW |
| DW181 | XX | 199 | List 4: | 6 DW |
| DW182 | XX | 205 | List 5: | 6 DW |
| DW183 | XX | 211 | List 6: | 5 DW |
| DW184 | XX | 216 | List 7: | 11 DW |
| DW185 | 00 | 227 | First free data word: DW227 here |

**Start address**

**Error number**

XX: Disregard

## 8.2      Conversion of Counting Values to Physical Numbers

The speed and length of a workpiece on a conveyor belt, for example, can be measured with counter modules IP 242A and IP 242B.  Sections 7.5 and 7.3 contain suggested solutions (the calculation functions available on the IP 242B have not yet been included in the examples).  This section does, however, include the calculation functions.

### 8.2.1    Speed

Priniple of Implementation

1. The speed is measured with two light bar-
   riers.  The signals are applied to the start
   input (STA) and the stop input (STO).
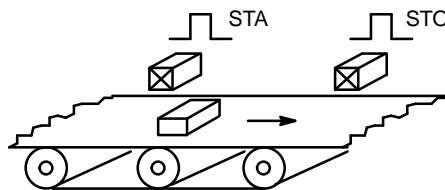
2. A counter gate is controlled with the STA
   and STO pulses.
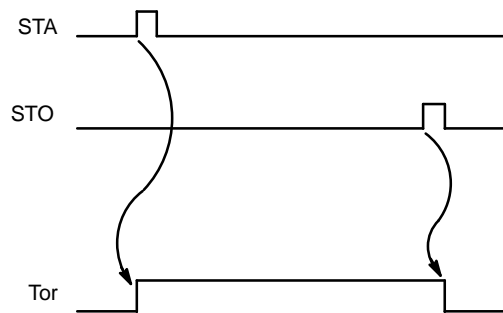
3. A reference frequency is applied to the
   counting input of the counter.

4. The counting value represents the run
   time of the workpiece.
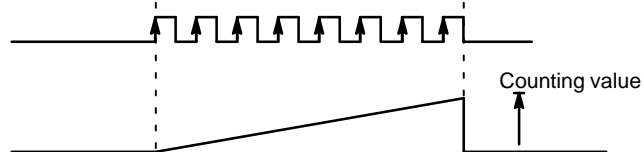
**Point in time for processing the command list at
gate interrupt (gate edge inactive)**
The counting value is transferred to the internal hold
register**.** The counter is loaded with the contents of
the load register.

**Calculation**

You are working with a command list for gate interrupts on an inactive gate edge. Include the "process command list" command in the command list to call an additional command list. This contains all rules of calculation. The additional command list is handled by the interrupt command list as a subprogram.

<u>Control Word Sequence in Parameterization Data Block for Interrupt from Counter Gate</u>

| | | | |
|---|---|---|---|
| Command 1 | DW20 = KH1602 | SV | Save Counter 1<br>Transfer the counting value from the counter to internal hold register 1 |
| Command 2 | DW21 = KH3702 | LD | Load counter 1<br>The counter is loaded with the contents of load register 1. The counter remains started. |
| Command 3 | DW22 = KH8202 | BB | Process command list 1 |
| Command 4 | DW23 = KH0000 | BE | End of the control word sequence |

<u>Control Word Sequence in Parameterization Data Block for Calculation of the Speed (Additional Command List 1):</u>

| | | | |
|---|---|---|---|
| Command 1 | DW186 = KH9050 | L | Load from constant register KON0<br>The registers must have been supplied with the corresponding values beforehand. In our example, this is the distance between the two light barriers in KON0. |
| Command 2 | DW187 = KH9021 | L | Load from internal hold register 1 (the run time) |
| Command 3 | DW188 = KH9600 | DIV | Divide accu 2 by accu 1 |
| Command 4 | DW189 = KH9141 | T | Transfer to result register ERG1 |
| Remember: | | | The "end of command list" command KH0000 is not required here (in contrast to the interrupt command list) because you have already specified the "fixed" length of the list (4 DWs in our example) in the directory of the additional command lists. |

The S5 can now read out the speed from result register 1.

## 8.2.2 Length

**Principle of Implementation**

A length measurement can be easily combined with a speed measurement described in section 8.2.1. This requires that an additional light barrier to control a counter gate be installed directly in front of the second light barrier for the speed measurement. The length measurement is thus implemented via a time measurement.

1. A frequency (reference) is generated (F1 to F6) and applied to the counting input of a counter.

2. The event to be measured is applied to the gate of the counter.

3. The measured value corresponds to the number of counting pulses counted when the gate is open.
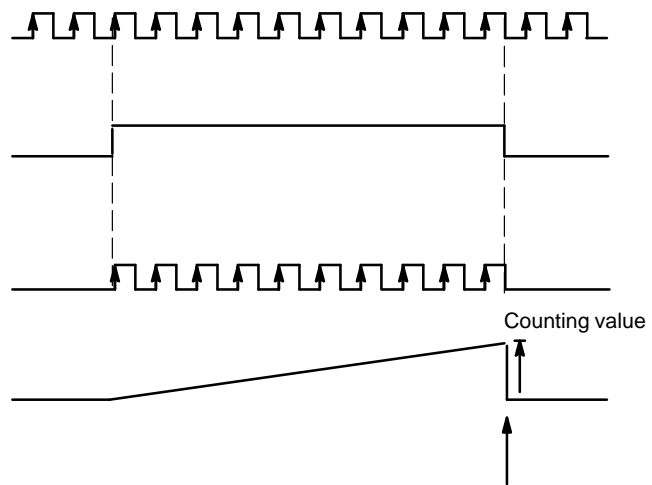
Counting value

**Point in time for processing the command list at gate interrupt (gate edge inactive)**
The counting value is transferred to the internal hold register. The counter is loaded with the contents of the load register.

**Calculation**

You are working with a command list for gate interrupts on an inactive gate edge. You include the "process command list" command in the command list to call an additional command list. This contains all rules of calculation. The additional command list is handled by the interrupt command list as a subprogram.

Control Word Sequence in Parameterization Data Block for Interrupt from Counter Gate

| Command 1 | DW34 = KH1604 | SV | Save counter 2<br>Transfer the counting value from the counter to internal hold register 2 |
| Command 2 | DW35 = KH3704 | LD | Load counter 2<br>The counter is loaded with the contents of load register 2. The counter remains started. |
| Command 3 | DW36 = KH8204 | BB | Process command list 2 |
| Command 4 | DW37 = KH0000 | BE | End of the control word sequence |

Control Word Sequence in Parameterization Data Block for Calculation of the Length (Additional Command List 2):

| Command 1 | DW190 = KH9041 | L | Load from constant register ERG1<br>(the previously calculated speed) |
| Command 2 | DW191 = KH9022 | L | Load from internal hold register 2<br>(the number of pulses $\triangleq$ time) |
| Command 3 | DW192 = KH9500 | MUL | Multiply accu 2 with accu 1 |
| Command 4 | DW193 = KH9142 | T | Transfer to result register ERG2 |

The S5 can now read out the length from result register 2.

---

### 8.3 Adjustment of the Counting Values to Physical Numbers Via Gearing Factor

**Use**

The calculation of the length is described in section 8.2.2. A gearing factor between the conveyor belt and drive shaft is not considered.

When gearing is used for the drive, the conveyor belt moves faster or slower depending on the gearing factor. Using the same distance between the light barriers as before, fewer or more pulses will be counted. The gearing factor is included in the additional command list for the calculation of numbers (example: 13 : 8, whereby 13 = dividend of the gearing factor and 8 = divisor of the gearing factor). After the calculation is concluded, the real number can be read out from the result register.

**Calculation**

Control Word Sequence in Parameterization Data Block for Interrupt from Counter Gate

| Command 1 | DW34 = KH1604 | SV | Save counter 2<br>Transfer the counting value from the counter to internal hold register 2 |
|-----------|---------------|----|----|
| Command 2 | DW35 = KH3704 | LD | Load counter 2<br>The counter is loaded with the contents of load register 2. The counter remains started. |
| Command 3 | DW36 = KH8208 | BB | Process command list 3 |
| Command 4 | DW37 = KH0000 | BE | End of the control word sequence |

Control Word Sequence in Parameterization Data Block for the Adjustment Via Gearing Factor (Additional Command List 3):

| Command 1 | DW194 = KH9041 | L | Load from result register ERG1 (the speed without gearing factor) |
|---|---|---|---|
| Command 2 | DW195 = KH9522 | MUL | Multiply with implicit load command (internal hold register 2) |
| Command 3 | DW196 = KH9551 | MUL | Multiply with implicit load command (constant register KON1: dividend of the gearing factor = 13)) |
| Command 4 | DW197 = KH9652 | DIV | Divide with implicit load command (constant register KON2: divisor of the gearing factor = 8) |
| Command 5 | DW198 = KH9143 | T | Transfer to result register ERG3 |

The S5 can now read out the length from result register 3.

## 8.4   Compare Two Counter Values/Results

**Use**

The speed of a second conveyor belt is not to exceed the speed of the first.  The speeds are acquired in accordance with the principle of measurement described in section 8.2.1, and both results are compared. The outputs are disabled if conveyor belt 2 moves faster than conveyor belt 1.  This type of comparison can also be used as an emergency stop function.

**Calculation**

Conveyor belt 1:

Control Word Sequence in Parameterization Data Block for Interrupt from Counter Gate

| | | | |
|---|---|---|---|
| Command 1 | DW20 = KH1602 | SV | Save counter<br>Transfer the counting value from the counter to internal hold register 1 |
| Command 2 | DW21 = KH3702 | LD | Load counter 1<br>The counter is loaded with the contents of load register 1.  The counter remains started. |
| Command 3 | DW22 = KH8202 | BB | Process command list 1 |
| Command 4 | DW23 = KH0000 | BE | End of the control word sequence |

Control Word Sequence in Parameterization Data Block for Calculation of the Speed (Additional Command List 1)

| | | | |
|---|---|---|---|
| Command 1 | DW186 = KH9050 | L | Load from constant register KON0<br>The registers must have been supplied with the corresponding values beforehand.  In our example, this is the distance between the two light barriers in KON0. |
| Command 2 | DW187 = KH9021 | L | Load from internal hold register 1<br>(the run time) |
| Command 3 | DW188 = KH9600 | DIV | Divide accu 2 by accu 1 |
| Command 4 | DW189 = KH9141 | T | Transfer to result register ERG1 |

Conveyor belt 2:

<u>Control Word Sequence in Parameterization Data Block for Interrupt from Counter Gate</u>

| Command 1 | DW48 = KH1608 | SV | Save counter 3<br>Transfer the counting value from the counter to internal hold register 3 |
|---|---|---|---|
| Command 2 | DW49 = KH3708 | LD | Load counter 3<br>The counter is loaded with the contents of load register 3.  The counter remains started. |
| Command 3 | DW50 = KH8210 | BB | Process command list 4 |
| Command 4 | DW51 = KH0000 | BE | End of the control word sequence |

<u>Control Word Sequence in Parameterization Data Block for Calculation of the Speed (Additional Command List 4)</u>

| Command 1 | DW199 = KH9053 | L | Load from constant register KON3<br>(distance between the two light barriers on conveyor belt 2) |
|---|---|---|---|
| Command 2 | DW200 = KH9623 | DIV | Divide with implicit load command<br>(internal hold register 3: run time) |
| Command 3 | DW201 = KHA541 | ≤D | Compare for "less than/equal to" with implicit load command<br>(result register ERG1) |
| Command 4 | DW202 = KHA002 | SPR | Relative jump to address +2<br>(≙ DW204) |
| Command 5 | DW203 = KH0200 | SA | Disable the outputs<br>(emergency stop function) |
| Command 6 | DW204 = KH9171 | T | Transfer to accu 1<br>(zero operation) |

### 8.5    Start a Counter with the Adjusted Counter Value of a Second Counter

**Use**

Two dependent counters can be used to solve automation tasks such as dosing, controlled positioning, etc.

The first counter handles the acquisition of "coarse" areas (coarse dosing, high–speed procedures). When the first counter expires, a command list is activated (via interrupt from the counter output) which starts the second counter for the "fine" values (fine dosing, creep–speed procedures).

> Starting the second counter must be dependent on when the first counter reaches a certain number of pulses.

The output of the second counter is controlled via the comparator function.

The counting values can be converted for indication in volumes, weight, speed, etc.  This is done in an additional command list which is called after the interrupt value is reached for the second counter.

> The specified sequence must be adhered to to prevent the results from being falsified.

**Calculation**

The following example shows a coarse/fine dosing and the subsequent calculation for plaster weight (density: 2.3 kg/dm$^3$).

Control Word Sequence in Parameterization Data Block for Interrupt from Counter Output (Counter 4 $\triangleq$ "Coarse Area")

| Command 1 | DW67 = KH3120 | LS | Load and start counter 5<br>Counter 5 (for "fine area") is loaded with the contents of the load register, and started. |
|---|---|---|---|
| Command 2 | DW68 = KH3420 | AW | Accept interrupt value for counter 5 (with enabled comparator function) |
| Command 3 | DW69 = KH3710 | LD | Load counter 4<br>The counter is loaded with the contents of load register 4.  The counter remains started. |
| Command 4 | DW70 = KH0000 | BE | End of the control word sequence |

Control Word Sequence in Parameterization Data Block for Interrupt from Counter Output (Counter 5 ≙ "Fine Area")

| Command 1 | DW81 = KH3920 | SL | Stop and read counter 5 |
| Command 2 | DW82 = KH8220 | BB | Process command list 5 |
| Command 3 | DW83 = KH0000 | BE | End of control word sequence |

Control Word Sequence in Parameterization Data Block for the Calculation of the Weight (Additional Command List 5)

| Command 1 | DW205 = KH9065 | L | Load from counter value register 5 |
| Command 2 | DW206 = KH9654 | DIV | Divide with implicit load command (constant register KON4: divisor for recalculation of fine dosing) |
| Command 3 | DW207 = KH9304 | ADD | Add with implicit load command (load register 4) |
| Command 4 | DW208 = KH9555 | MUL | Multiply with implicit load command (constant register KON5: dividend = density value 23) |
| Command 5 | DW209 = KH9656 | DIV | Divide with implicit load command (constant register KON6: divisor = density value 10) |
| Command 6 | DW210 = KH9145 | T | Transfer to result register ERG5 |

## 8.6    Buffering Results

**Use**

The contents of a result register are buffered when the corresponding block of the measured value data block is enabled with the entry of a start address in the directory of the measured value memory ($\rightarrow$ section 3.7.1).

The speed is determined as described in section 8.2.1.  Storing the results requires that measured value series 1 (block 1) be enabled beforehand (e.g., start address 16 in the measured value data block).  The recording depth can be specified to meet your requirements (maximum of 100 x 2 data words).

The values accepted from result register ERG1 in measured value series 1 can, for example, be output on a printer as a curve (for monitoring the process), or as an error analysis from the S5.

**Sequence**

1. Enable Block 1 in Parameterization Data Block

   DW247 = KY0,16                 Enable block 1
                                  (block size: 30 x 2 DW)
                                  Start at DW16

   DW248 = KY0,0

   DW249 = KY0,0

   DW250 = KY0,0
                                  All blocks which are not required must be disabled.
   DW251 = KY0,0

   DW252 = KY0,0

   DW253 = KY0,0

   DW254 = KY0,76                 End of block: DW76 in our example

2. Transfer Block Enable to IP 242B Module:

   KH7101            PA           Parameterize counter with the global registers

3. <u>Control Word Sequence in Parameterization Data Block for Interrupt from Counter Gate</u>

| Command 1 | DW20 = KH1602 | SV | Save counter 1<br>Transfer the counting value from the counter to internal hold register 1 |
|---|---|---|---|
| Command 2 | DW21 = KH3702 | LD | Load counter 1<br>The counter is loaded with the contents of load register 1.  The counter remains started. |
| Command 3 | DW22 = KH8202 | BB | Process command list 1 |
| Command 4 | DW23 = KH0000 | BE | End of the control word sequence |

4. <u>Control Word Sequence in Parameterization Data Block for Calculation of the Speed (Additional Command List 1)</u>

| Command 1 | DW186 = KH9050 | L | Load from constant register KON0<br>The registers must have been supplied with the corresponding values beforehand.  In our example, this is the distance between the two light barriers in KON0. |
|---|---|---|---|
| Command 2 | DW187 = KH9021 | L | Load from internal hold register 1<br>(the run time) |
| Command 3 | DW188 = KH9600 | DIV | Divide accu 2 by accu 1 |
| Command 4 | DW189 = KH9141 | T | Transfer to result register ERG1 |

5. The S5 can read the measured value series, and output the buffered speed values (e.g., as curves) on a printer.

## 8.7 Prepare for Load in a Command List with Conditional Jumps

**Use**

A 10–m wooden beam is to be cut into the following lengths:

> 8 x 1.20 m
> 1 x 0.30 m

The procedure can be repeated as often as desired.

Counter gate

10 m

| 1.20 | . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 1.20 | 1.20 |

0.30
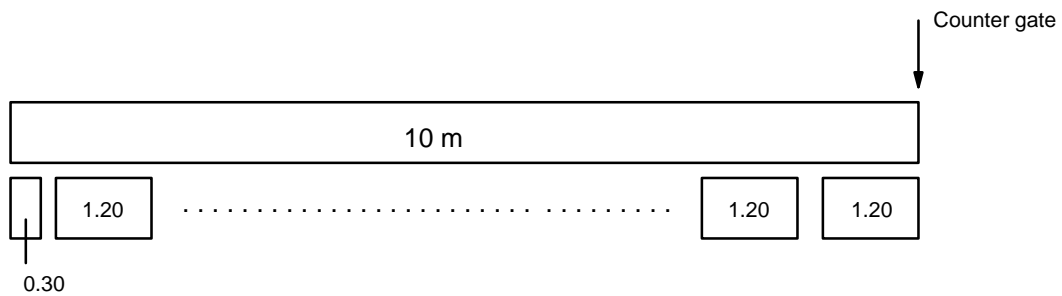
➡ This procedure cannot be performed by loading alternately from the load and hold registers. ⬅

**Procedure**

1. Following reparameterization of counter 1, the processing of additional command list 6 is triggered with the "process command list" control word. An internal "pass" counter is loaded, and counter 1 is initialized and started. Counter 1 is enabled with the counter gate.

Triggering Additional Command List 6

Control word:          KH = 8240          BB          Process command list 6

Control Word Sequence in Parameterization Data Block for Initialization of Counter 1 (Additional Command List 6)

| Command 1 | DW211 = KH905A | L | Load from constant register KON10 (value 1) |
| Command 2 | DW212 = KH9180 | T | Transfer to flag 0 ( ≙ load "pass" counter ) |
| Command 3 | DW213 = KH9057 | L | Load from constant register KON7 (value 1.20 m) |
| Command 4 | DW214 = KH9161 | T | Transfer to counter value register 1 ( ≙ load counter 1) |
| Command 5 | DW215 = KH3602 | ST | Start counter 1 |

2. When counter 1 expires, the interrupt from the counter output causes additional command list 7 to be called.

Control Word Sequence in Parameterization Data Block for Interrupt from Counter Output

| Command 1 | DW25 = KH8280 | BB | Process command list 7 |
| Command 2 | DW26 = KH0000 | BE | End of control word sequence |

Control Word Sequence in Parameterization Data Block for a Single Cutting Procedure (Additional Command List 7)

| Command 1 | DW216 = KH9080 | L | Load from flag 0 (value of the "pass" counter) |
| Command 2 | DW217 = KH935A | ADD | Add with implicit load command (constant register KON10: 1) |
| Command 3 | DW218 = KH9180 | T | Transfer to flag 0 |
| | | | DW216 to 218 increment the "pass" counter. |
| Command 4 | DW219 = KHA259 | =D | Compare for "equal" with implicit load command (constant register KON9: 8 = number of passes) |
| Command 5 | DW220 = KHA003 | SPR | Jump relative to address + 3 ( ≙ DW223) conditional jump |
| Command 6 | DW221 = KH9057 | L | Load from constant register KON7 (value 1.20 m) |

| Command 7 | DW222 = KHA004 | SPR | Jump relative to address + 4 ( ≙ DW226) unconditional jump |
|---|---|---|---|
| Command 8 | DW223 = KH905B | L | Load from constant register KON11 (value 0) |
| Command 9 | DW224 = KH9180 | T | Transfer to flag 0 |
| | | | DW223 and 224 reset the "pass" counter. |
| Command 10 | DW225 = KH9058 | L | Load from constant register KON8 (value 0.30 m) |
| Command 11 | DW226 = KH9101 | T | Transfer to load register 1 |

3. The following is performed each time counter 1 expires.
   – The load value is automatically accepted.
     and
   – Another processing pass of additional command list 7 is triggered by the interrupt from the counter output.
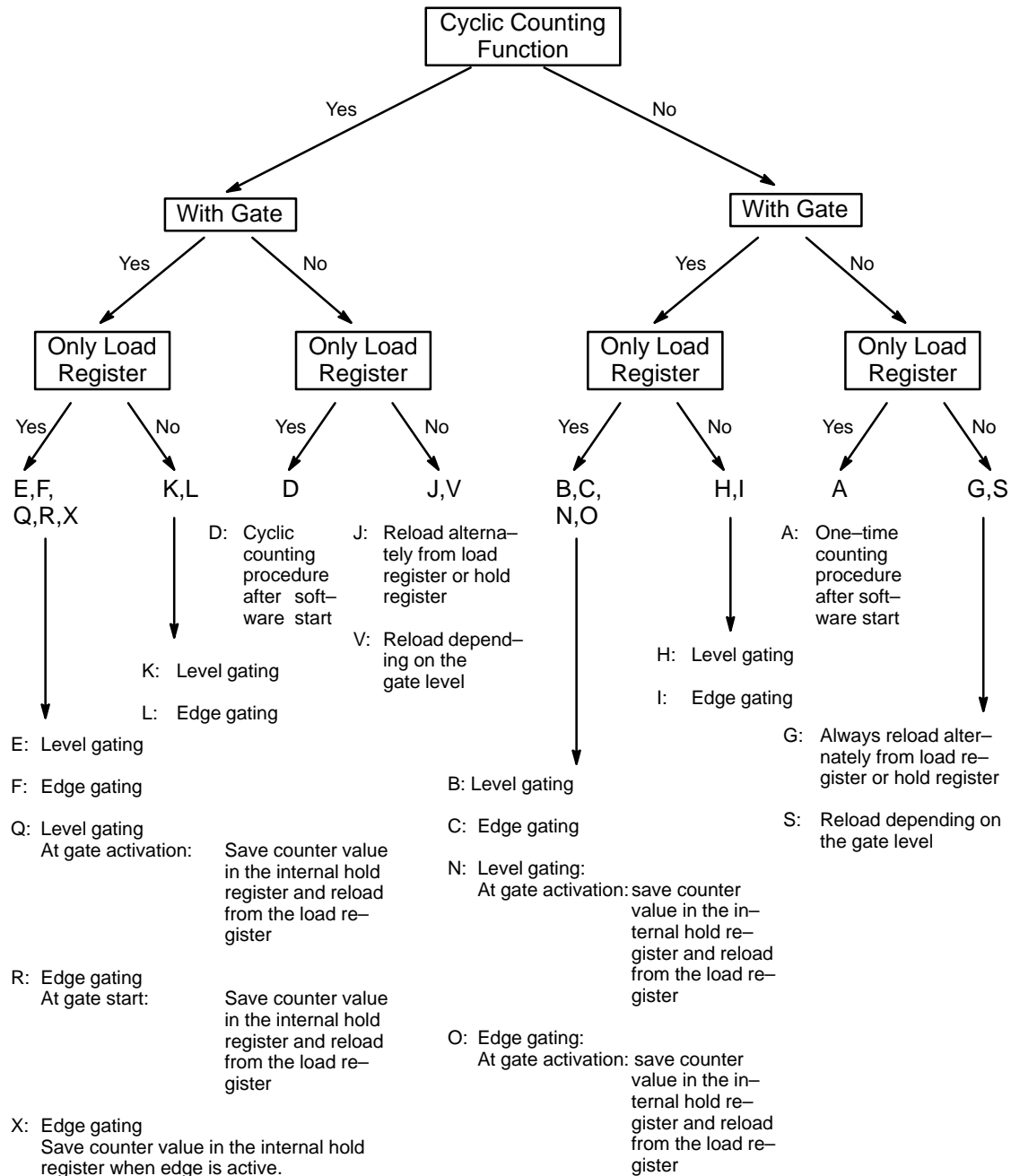
# 9 Operation Modes

## 9.1    Operating Modes for Counters 1 to 5
## 9.1.1    Overview

**IP 242A/242B**

Select your operation modes according to the following criteria:

1. Do you want cyclic counting?
2. Do you want gated counting?
3. Is the counter to be loaded only by the load register
   or alternately by the load register and the load and hold register?

```
                          Cyclic Counting
                             Function
                   Yes                        No
              With Gate                          With Gate
         Yes          No                    Yes          No
    Only Load      Only Load            Only Load      Only Load
    Register       Register             Register       Register
   Yes    No      Yes    No            Yes    No      Yes    No
   E,F,   K,L      D      J,V          B,C,   H,I      A       G,S
   Q,R,X                               N,O
```

D: Cyclic counting procedure after soft-ware start

J: Reload alternately from load register or hold register

V: Reload depending on the gate level

K:  Level gating

L:  Edge gating

E:  Level gating

F:  Edge gating

Q:  Level gating
At gate activation:    Save counter value in the internal hold register and reload from the load register

R:  Edge gating
At gate start:    Save counter value in the internal hold register and reload from the load register

X:  Edge gating
Save counter value in the internal hold register when edge is active.

B: Level gating

C: Edge gating

N:  Level gating:
At gate activation: save counter value in the internal hold register and reload from the load register

O:  Edge gating:
At gate activation: save counter value in the internal hold register and reload from the load register

A:  One-time counting procedure after soft-ware start

H:  Level gating

I:   Edge gating

G:  Always reload alternately from load register or hold register

S:  Reload depending on the gate level

All operating modes are activated with the "start counter" command.

| Mode | Description | Operating Mode |
|:---:|---|:---:|
| 1 | Counting function **without** hardware **gating**; counts only once after "start counter" command | A |
| 2 | Counting function **with** level **gating**; counts only while gate is active | B |
| 3 | Counting function **with** edge **gating**; counts only once after first active gate edge | C |
| 4 | Frequency generator **without gating** | D |
| 5 | Frequency generator **with** level **gating**; counts only while gate is active | E |
| 6 | Single pulse encoder **with** edge **gating**; counts after first active gate edge | F |
| 7 | Single pulse encoder **with** delayed pulse; counts twice (i.e., first with load register, then with hold register) | G |
| 8 | Single pulse encoder **with** delayed pulse and level **gating**; counts twice when the gate is active (i.e., once with load register, then with hold register) | H |
| 9 | Single pulse encoder **with** delayed pulse and edge **gating**; counts twice after first active gate edge (i.e., first with load register, then with hold register) | I |
| 10 | Frequency generator **with** variable duty cycle rate without **gating** | J |
| 11 | Frequency generator with variable duty cycle rate **with** level **gating**; counts only while gate is active | K |
| 12 | Single pulse encoder with delayed pulse; counts only after first active gate edge | L |
| 13 | Counting function **with** level **gating**; counts only while gate is active. At every active gate edge, the contents of the counter are saved in the hold register, and counting is continued with the contents of the load register. Counting is stopped at every terminal count. | N |
| 14 | Counting function **with** edge **gating**; counts after first active gate edge. At every active gate edge, the contents of the counter are saved in the hold register, and counting is continued with the contents of the load register. Counting is stopped at every terminal count. | O |

| Mode | Description | Operating Mode |
|------|-------------|----------------|
| 15 | Counting function **with** level **gating**;<br>counts only when gate is active.<br>At every active gate edge or terminal count, the contents of the counter are saved in the hold register, and counting is continued with the contents of the load register. | Q |
| 16 | Counting function **with** edge **gating**;<br>counts after first active gate edge.<br>At every active gate edge or terminal count, the contents of the counter are saved in the hold register, and counting is continued with the contents of the load register. | R |
| 17 | Single pulse encoder with delayed pulse;<br>counts twice; reloads from the load or hold register depending on the gate level | S |
| 18 | Frequency generator with two different frequencies;<br>the counter is reloaded from the load or hold register depending on the gate level. | V |
| 19 | Counting function **with** edge **gating**;<br>counts from first active gate edge or from terminal count with the value from the load register. The counting value is saved in the hold register at each additional gate edge and counting continues. | X |

## 9.1.2 Description of the Operating Modes

The operating mode is selected for each counter with bits CM15 to CM13 and CM7 to CM5 of the counter mode register. (See section 9.1.1.) To simplify reference to a specific operating mode, each operation mode is assigned a letter from A to X.

Representative signal sequences for the different counter operating modes are shown in figures 9.1a to 9.1x. (Figures 9.1m, 9.1p, 9.1t, 9.1u, and 9.1w are omitted because the index letter in the figure numbers corresponds to the various operating modes.)

The figures are based on the following parameters: **down counting** and **rising edge of the clock pulse as the active counting edge**.

The signal form for the terminal count pulse output and the terminal count square wave output are shown for each operating mode.

For those operating modes in which the counter is automatically disarmed (C15 = 0: single or two time counting function), the "start counter" command is required for each activation of the counting function ($\overline{WR}$ pulse).

For operating modes in which counting is repeated (CM5 = 1: cyclical counting function), the "start counter" command is omitted in the drawing. This command is required only once at the initial activation of the operating mode.

For retriggerable operating modes (i.e., N, O, Q, and R), a newly triggered operation is shown.

**The below table defines some of the abbreviations used.**

| TC | Terminal count<br>Counting cycle in which the counter is automatically reloaded |
|---|---|
| TC output | Counting output, pulse |
| TC switch–over output | Counting output, square wave |
| $\overline{WR}$, arm command | "Start counter" command |

Symbols "L" and "H" represent the counter values corresponding to the contents of the load registers or hold registers. Symbols "K" and "N" represent any counter values. The required bit pattern in the counter mode register is specified for each operating mode.

Bits whose values are disregarded are marked with an "X".

To make the following description of operating modes brief and concise, the term "pulse edge" refers only to active clock pulse edges of the source pulses and not to inactive edges. Similarly, the term "gate edge" refers only to active gate edges.

## Summary of the Counter Operating Modes

| Operating mode | A | B | C | D | E | F | G | H | I | J | K | L |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit CM7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bit CM6 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| Bit CM5 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| Gate monitoring (CM15 to CM13) | 000 | Level | Edge | 000 | Level | Edge | 000 | Level | Edge | 000 | Level | Edge |
| One–time terminal count then **counter stop** | X | X | X | | | | | | | | | |
| Two–time terminal count then **counter stop** | | | | | | | X | X | X | | | |
| Cyclical terminal count without **counter stop** | | | | X | X | X | | | | X | X | X |
| Gate input is not effective | X | | | X | | | X | | | X | | |
| Counter control by gate level | | X | | | X | | | X | | | X | |
| Counter start with gate edge Counter stop at first terminal count | | | X | | | X | | | | | | |
| Counter start with gate edge Counter stop at se–cond terminal count | | | | | | | | | X | | | X |
| No retriggering | X | X | X | X | X | X | X | X | X | X | X | X |
| At terminal count, counter is loaded from the L re–gister. | X | X | X | X | X | X | | | | | | |
| At terminal count, counter is loaded alternately from the L re–gister and the H register. | | | | | | | X | X | X | X | X | X |
| If gate is low at terminal count, the L register is transferred to the counter. If gate is high at terminal count, the H register is transferred to the counter. | | | | | | | | | | | | |
| At gate edge, transfer counter value to H register and then load the counter with the L register. | | | | | | | | | | | | |
| At gate edge, transfer counter value to H register and continue counting. | | | | | | | | | | | | |

| Operating mode | M | N | O | P | Q | R | S | T | U | V | W | X |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bit CM7 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Bit CM6 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| Bit CM5 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| Gate monitoring (CM15 to CM13) | 000 | Level | Edge | 000 | Level | Edge | 000 | Level | Edge | 000 | Level | Edge |
| One−time terminal count then **counter stop** |  | X | X |  |  |  |  |  |  |  |  |  |
| Two−time terminal count then **counter stop** |  |  |  |  |  |  | X |  |  |  |  |  |
| Cyclical terminal count without **counter stop** |  |  |  |  | X | X |  |  |  | X |  | X |
| Gate input is not effective |  |  |  |  |  |  | X |  |  | X |  |  |
| Counter control by gate level |  | X |  |  | X |  |  |  |  |  |  |  |
| Counter start with gate edge Counter stop at first terminal count |  |  | X |  |  | X |  |  |  |  |  | X |
| Counter start with gate edge Counter stop at se−cond terminal count No retriggering |  |  |  |  |  |  | X |  |  | X |  | X |
| At terminal count, counter is loaded from the L re−gister. |  | X | X |  | X | X |  |  |  |  |  | X |
| At terminal count, counter is loaded alternately from the L re−gister and the H register. |  |  |  |  |  |  |  |  |  |  |  |  |
| If gate is low at terminal count, the L register is transferred to the counter. If gate is high at terminal count, the H register is transferred to the counter. |  |  |  |  |  |  | X |  |  | X |  |  |
| At gate edge, transfer counter value to H register and then load the counter with the L register. |  | X | X |  | X | X |  |  |  |  |  |  |
| At gate edge, transfer counter value to H register and continue counting. |  |  |  |  |  |  |  |  |  |  |  | X |

Do not use operating modes M, P, T, U, and W.

## Operating Mode A

Software–Triggered Counting without Hardware Gating.

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| 0    | 0    | 0    | X    | X    | X    | X   | X   |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   | 0   | 0   | X   | X   | X   | X   | X   |

Operating mode A, shown in figure 9.1a, is one of the simplest operating modes. The counter is always ready to count clock pulse edges as soon as it is armed (software start command).

The counter is loaded from the load register at every TC and is disarmed automatically to prevent further counting. Counting starts again as soon as it is rearmed (software start).
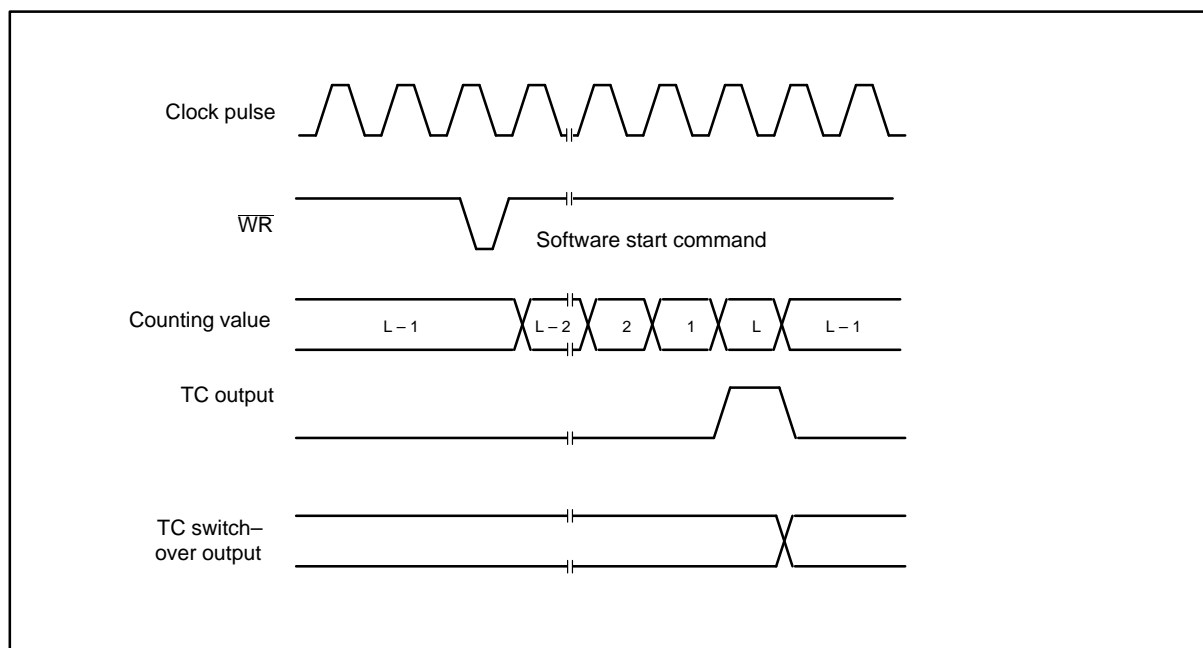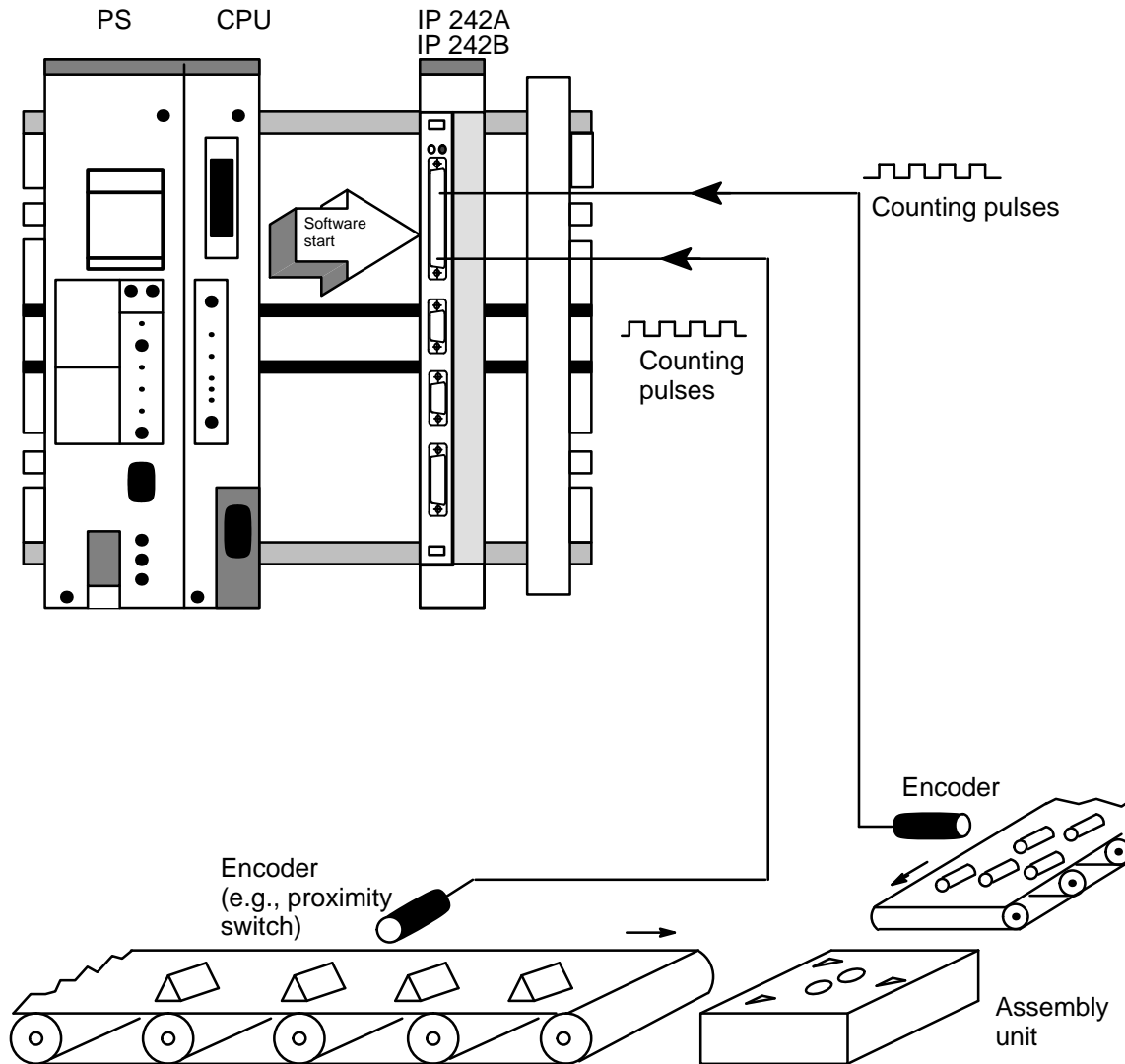


**Figure 9.1a: Signal form, operating mode A**

## Example of Operating Mode A

PS          CPU          IP 242A
                         IP 242B

Software
start

Counting pulses

Counting
pulses

Encoder

Encoder
(e.g., proximity
switch)

Assembly
unit

**Portioning prior to further processing (i.e., completion)**

1. The CPU triggers the IP 242A/242B with the "start counter" command.

2. The assembly unit is activated as soon as the counter reaches the desired number of parts. The IP 242A/242B module uses an interrupt, for example, to inform the CPU of this.

3. The CPU starts the counting function again with another software start and the counter is automatically reloaded with the desired number of parts.

## Operating Mode B

Software–Triggered Counting with Level Gating

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| Level | | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 0 | 0 | X | X | X | X | X |

Operating mode B, shown in figure 9.1b, is identical to operating mode A except that the clock pulse edges are counted only when the assigned gate is active. A software start command is necessary to arm the counter before counting can begin.

When the counter is armed, all clock pulse edges which occur while the gate is active are counted. Those edges which occur while the gate is inactive are not counted. This makes it possible to switch counting on and off with the gate.

The counter is loaded from the load register at every TC and the counter is disarmed automatically to prevent further counting until it is armed again (software start).
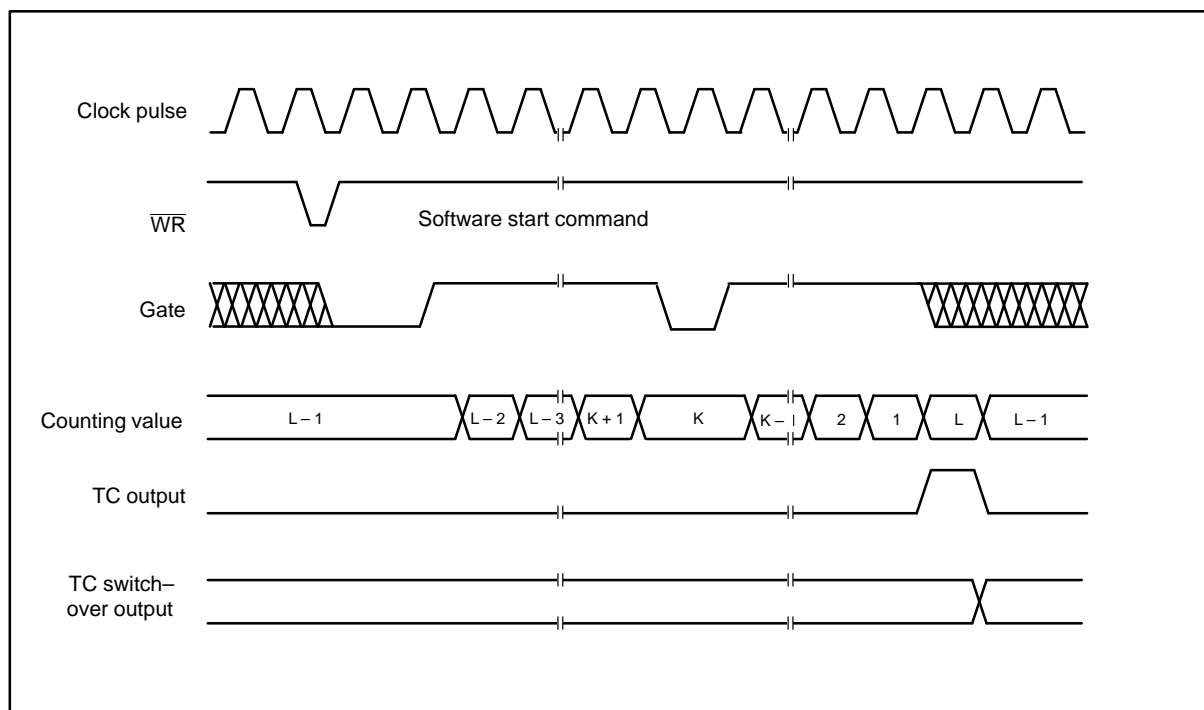


**Figure 9.1b: Signal form, operating mode B**

## Operating Mode C

Hardware–Triggered Counting

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| Edge | | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 0 | 0 | X | X | X | X | X |

Operating mode C, shown in figure 9.1c, is identical to operating mode A except that counting does not begin until the gate edge is applied to the armed counter.

The counter must be armed (software start) before the trigger gate edge is applied. Gate edges which are applied to a disarmed counter are disregarded. The counter starts counting at the first gate edge after application of the trigger gate edge and continues counting until TC.

At TC, the counter is loaded from the load register and is automatically disarmed (software start). Counting remains disabled until the counter is armed again (software start command) and a new gate edge is applied in that order.

Remember that, after the trigger gate edge is applied, the gate input has no effect on the rest of the counting cycle. This differs from operating mode B where the gate can be modulated during the counting cycle to start and stop the counter.
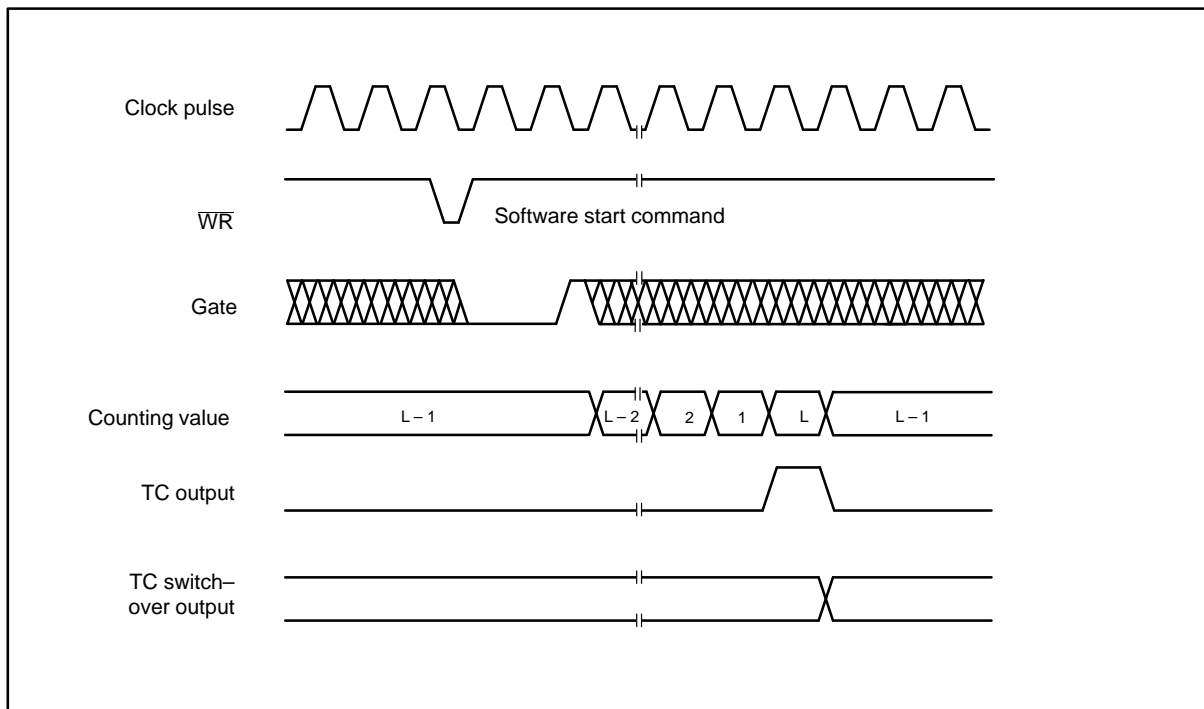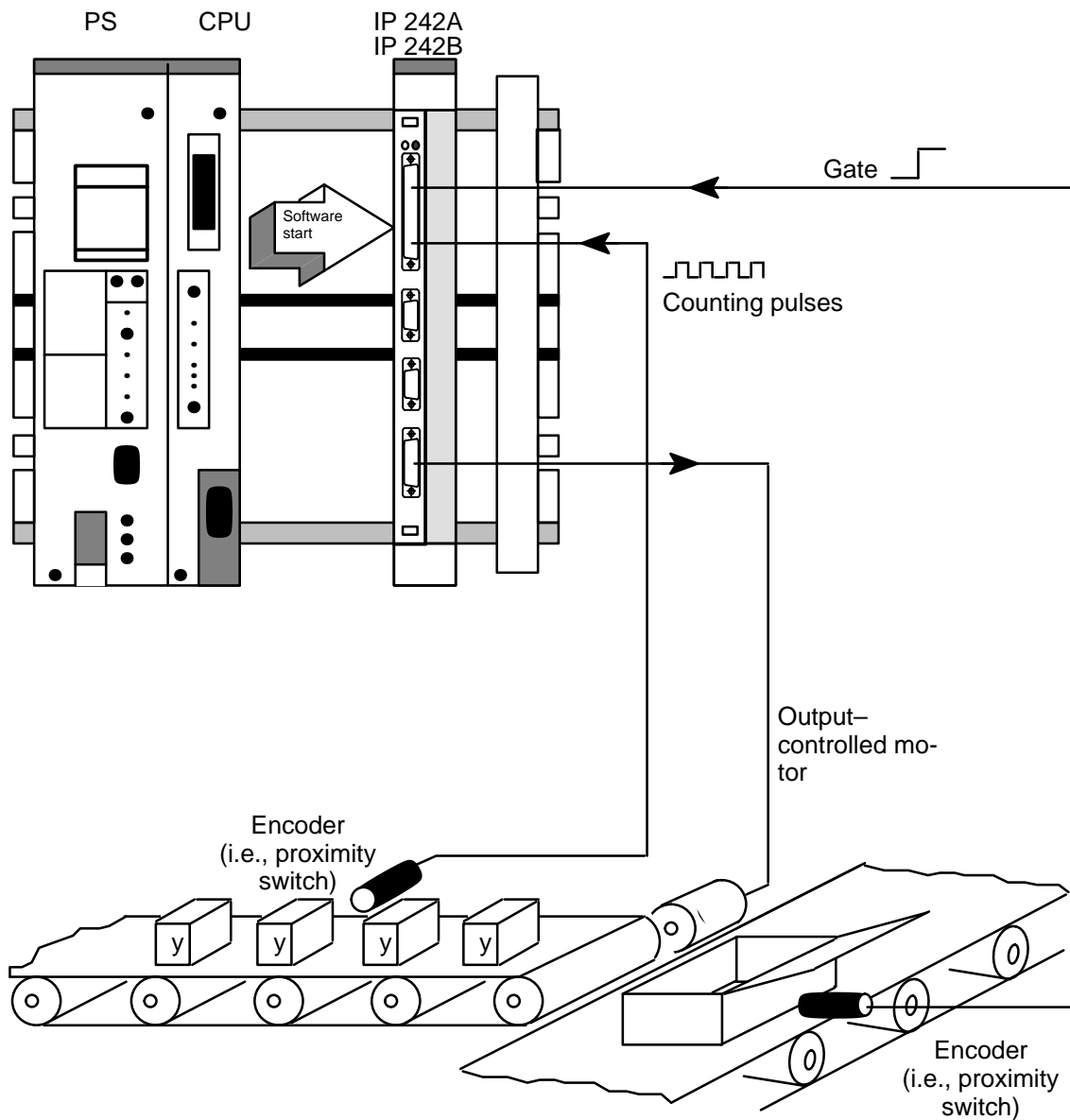


**Figure 9.1c: Signal form, operating mode C**

## Example of Operating Mode C



**Simple piece counting for packing**

1. The CPU triggers the IP 242A/242B with the "start counter" command.

2. Counting is started as soon as a carton enters the packing area.

3. The counter proceeds automatically; the output is set and this then stops the conveyor belt.

4. The counter continues processing (already loaded automatically with the same counting end va–
   lues) as soon as another software start is made.

## Operating Mode D

Frequency Generator without Hardware Gate Control

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| 0 | 0 | 0 | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 0 | 1 | X | X | X | X | X |

Operating mode D, shown in figure 9.1d, is usually used to generate frequencies. In this operating mode, the gate input has no effect on counting. The counter counts repeatedly up to TC as soon as it is armed (software start).

The counter is automatically loaded from the load register at every TC. The value of the load register, therefore, determines the time interval between TCs. A square wave determines the time interval between TCs. A square wave generator is thus created if operating mode TC switch–over output is specified in the counter mode register.
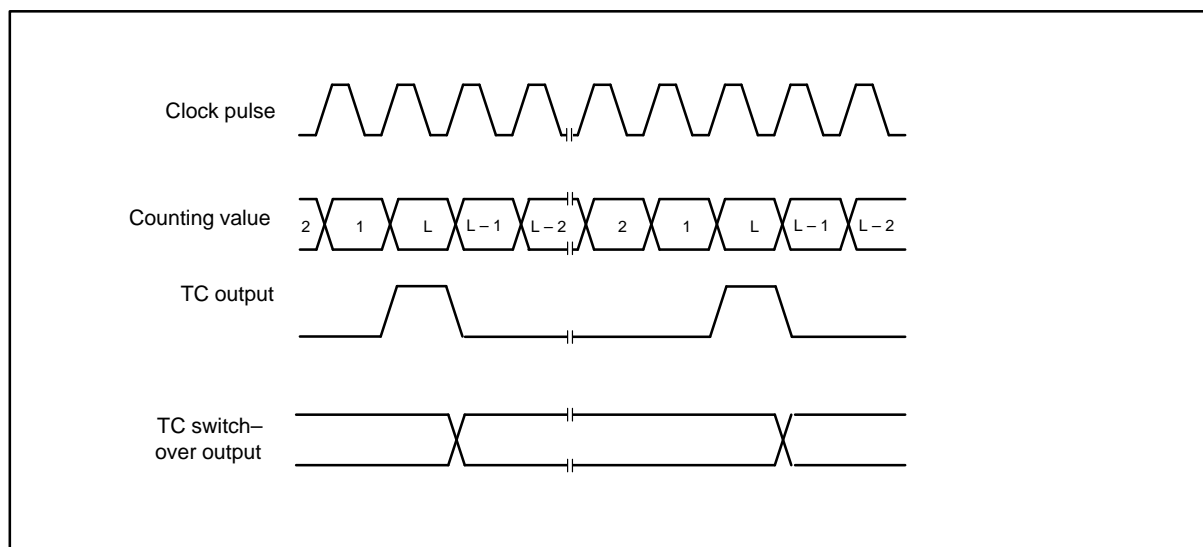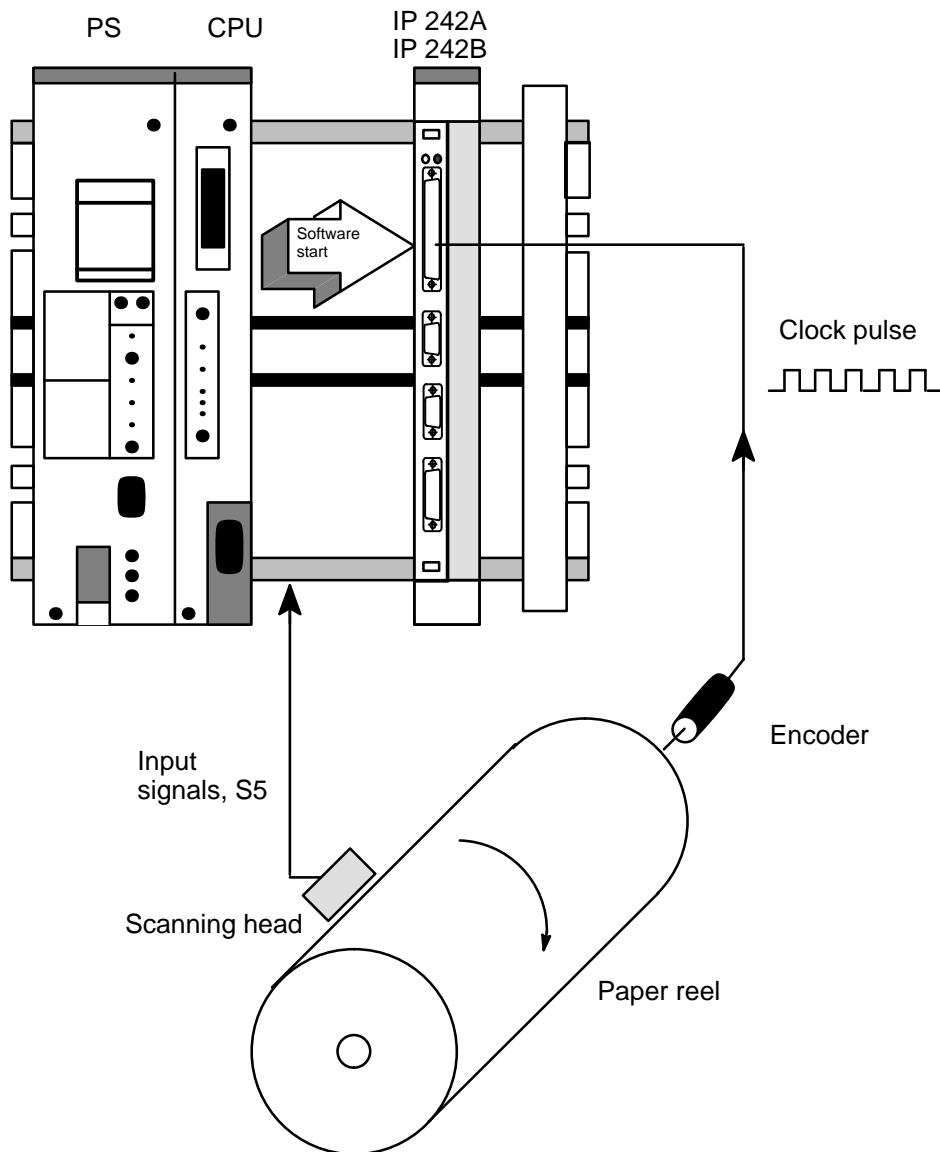


**Figure 9.1d: Signal form, operating mode D**

## Example of Operating Mode D



**Generation of an interrupt grid pattern for acquisition of measuring values (based on angle)**

1. Based on the diameter of the paper reel, a pulse grid pattern is calculated for each 10˚ that the paper reel is turned.

2. The CPU triggers the IP 242A/242B with a software start.

3. IP 242A/242B generates an interrupt every 10˚.

4. Input signals are stored in the 10˚ grid pattern in a data block (36 measured values per revolution).

## Operating Mode E

Frequency Generator with Level Gating

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| Level | | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 0 | 1 | X | X | X | X | X |

Operating mode E, shown in figure 9.1e, is identical to operating mode D except that the counter counts only those clock pulse edges which occur when the gate input is active. This makes it possible to enable and disable counting through control of the hardware.

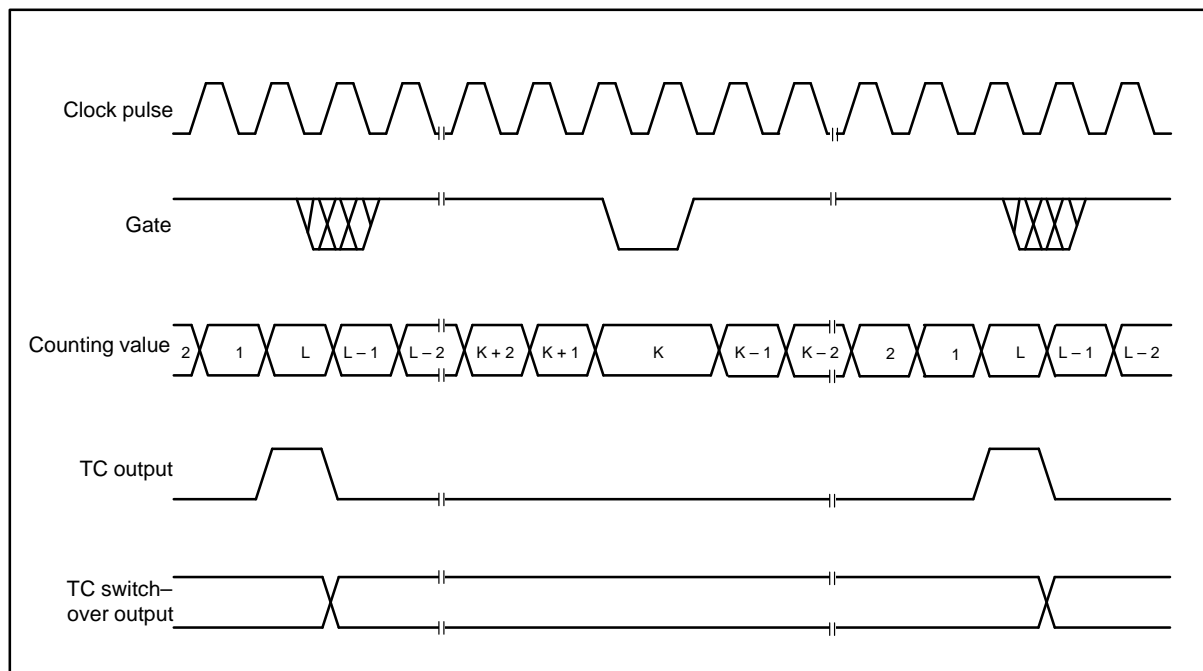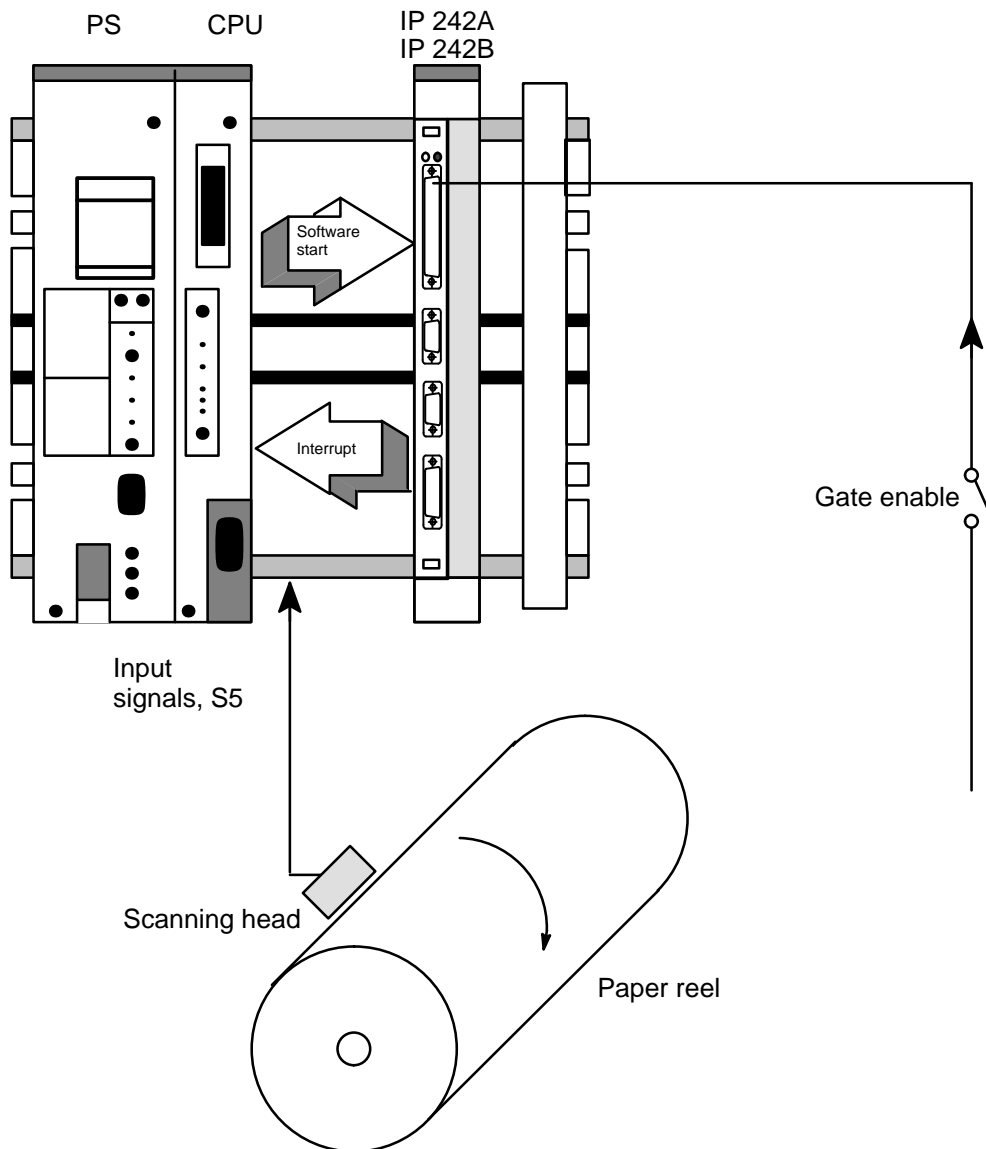A square wave generator is created if the operating mode is specified as TC switch–over output.



**Figure 9.1e: Signal form, operating mode E**

## Example of Operating Mode E



**Generation of an interrupt grid pattern for acquisition of measured values (based on external gate enabling)**

1. The CPU triggers the IP 242A/242B with a software start.

2. An interrupt is generated for the CPU (e.g., every 200 msec or 5 Hz).

3. The S5 acquires the measured values when the interrupt is active.

## Operating Mode F

Single Pulse Encoder without Retriggering

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| Edge | | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 0 | 1 | X | X | X | X | X |

Operating mode F, shown in figure 9.1f, has a single pulse–time encoder function without retriggering. The counter must be armed (software start) before this function is possible. The application of a gate edge to an armed counter enables counting.

When the counter reaches TC, it automatically reloads itself from the load register and counting is stopped until a new gate edge occurs.

Remember that, in contrast to operating mode C, the counter must not be rearmed (software command) after TC. Only a new gate edge is necessary. After application of the trigger gate edge, the gate input has no effect until TC is reached.
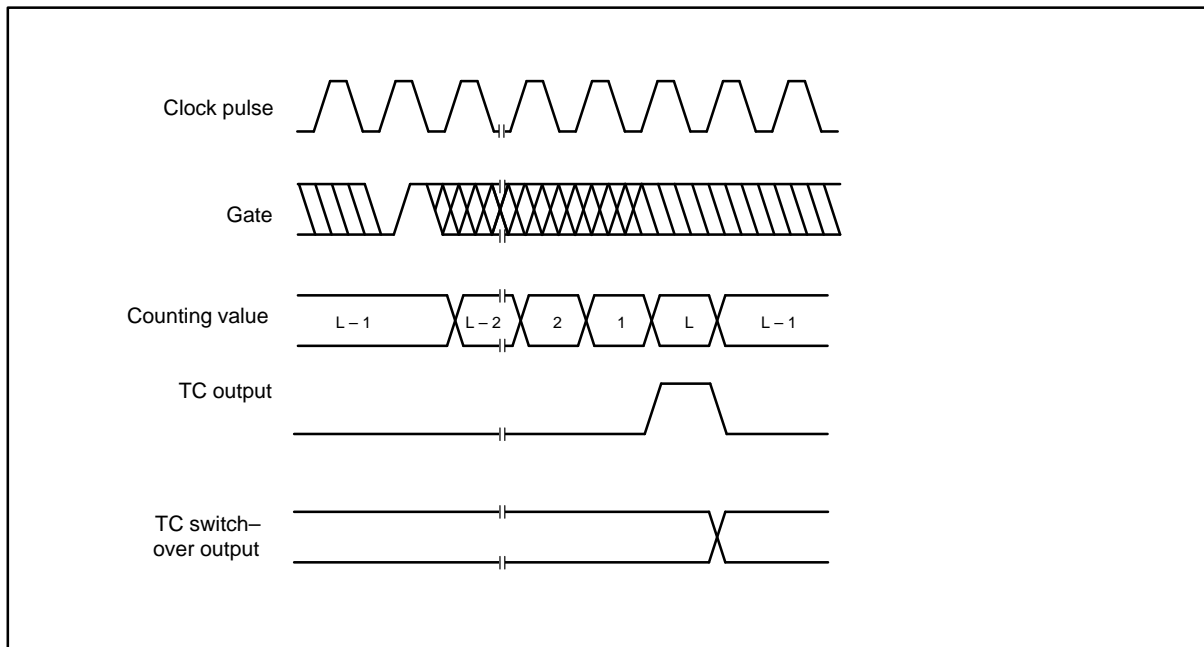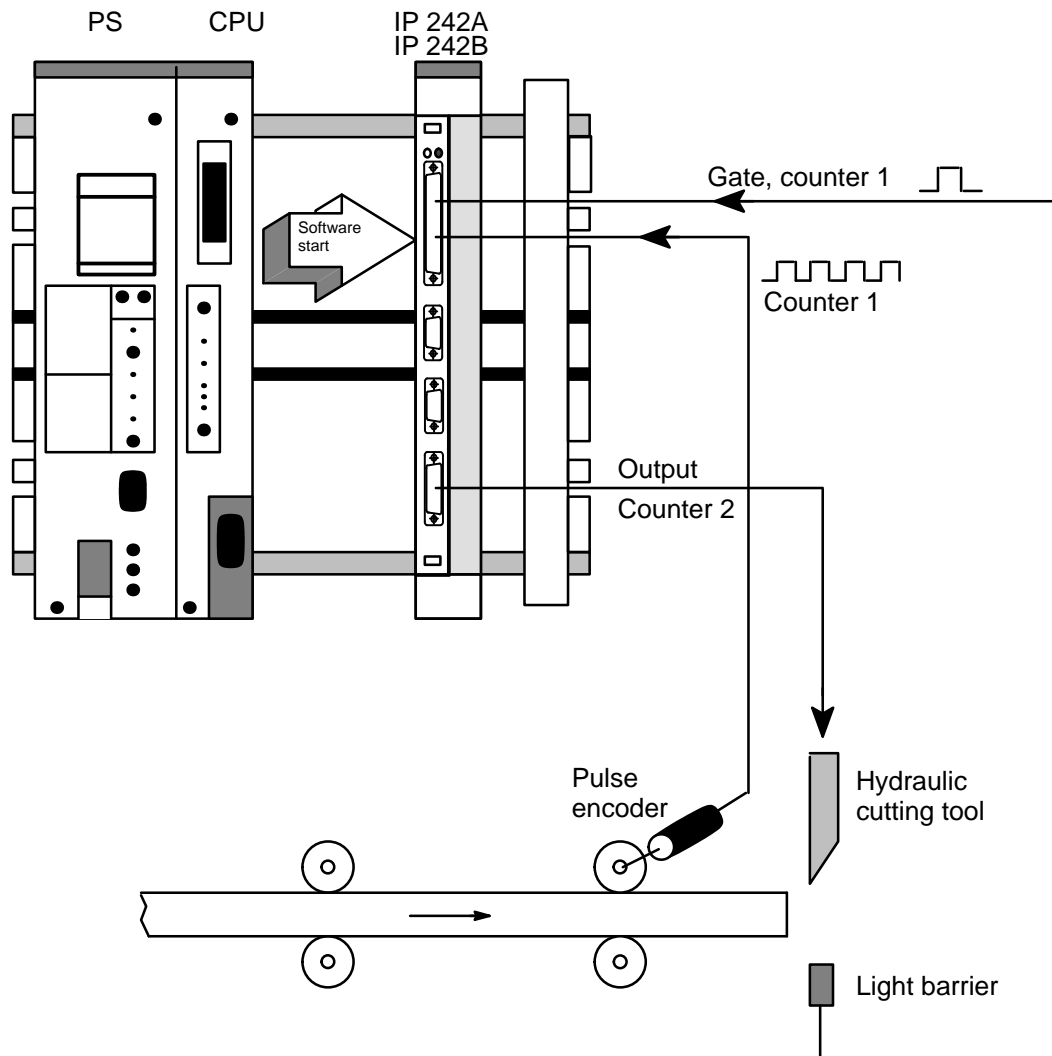


**Figure 9.1f: Signal form, operating mode F**

## Example of Operating Mode F



**Cutting of work pieces into certain lengths**

1. Counter is enabled with a software start.

2. Light barrier supplies hardware triggering (gate input for counter 1).

3. Counter 1 counts down with a certain value proportional to the length.

4. The gate of counter 2 is activated with the output pulse when counter 1 crosses zero.

5. Counter 2 runs with the internal frequency and determines the operating time of the cutting tool.

## Operating Mode G

Single Pulse Encoder with Software–Triggered Delayed Pulse

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| 0 | 0 | 0 | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 1 | 0 | X | X | X | X | X |

In operating mode G, counter operation is not affected by the gate. As soon as the counter is armed, it counts up to TC twice and then automatically disarms.

For most applications, the counter is initially loaded from the load register either by a load command or by the last TC of a previous counting cycle. After counting to the first TC, the counter is automatically re-loaded from the load register and then disarmed to prevent further counting.

Counting is started again when the counter is rearmed. A software–triggered single pulse encoder with delayed pulse is created when operating mode TC switch–over output is specified in the operation mode register. The initial value of the counter controls the delay of the arming command up to the start of the output pulse. The contents of the hold register control the duration of the pulse. Operating mode G is shown in figure 9.1g.
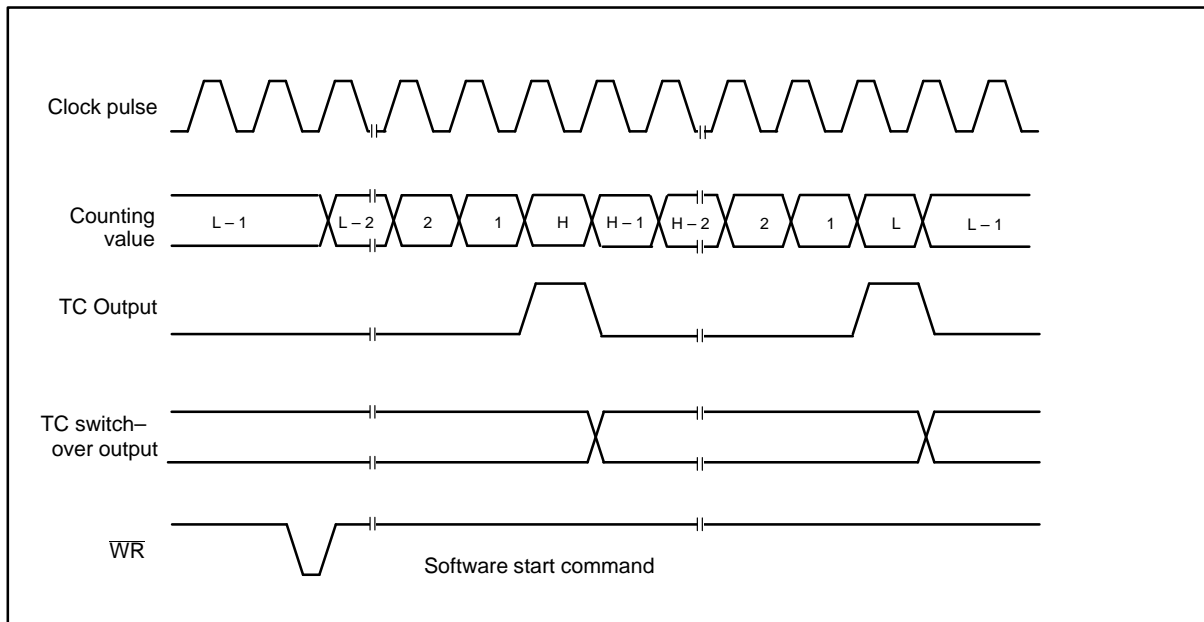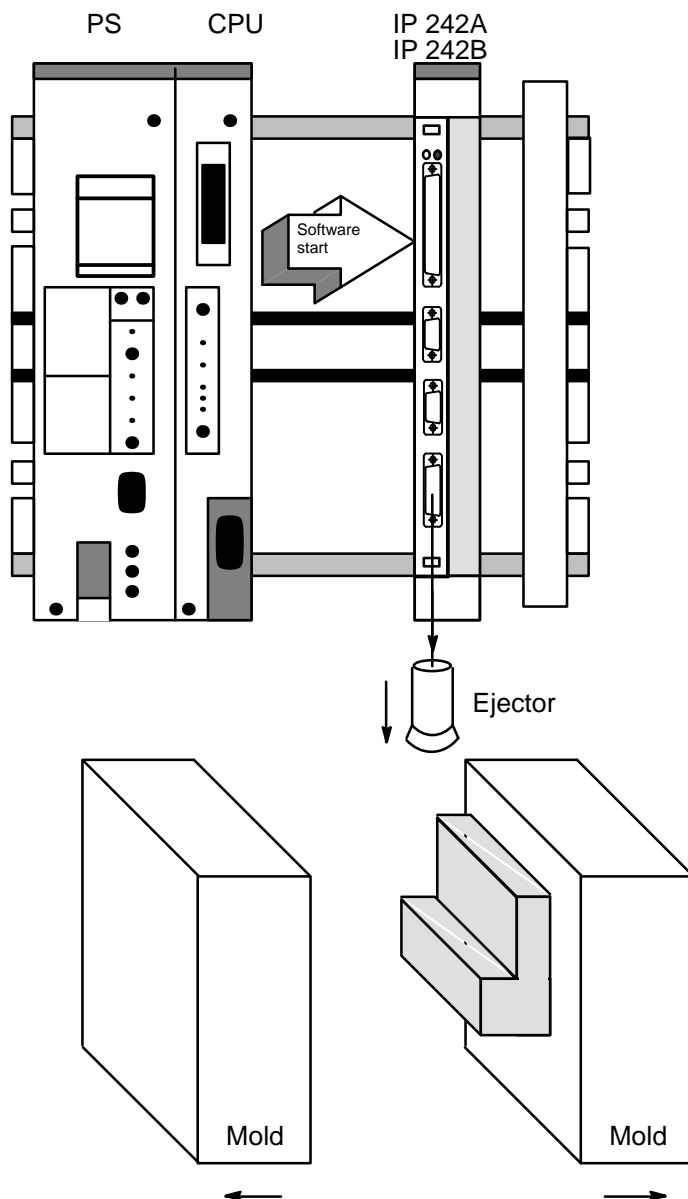


**Figure 9.1g: Signal form, operating mode G**

## Example of Operating Mode G



**Ejecting a plastic piece from a casting mold**

1. The CPU triggers the IP 242A/242B with a software start.

2. The counter output is switched on after the time contained in the load register is up (opening of the mold).

3. The ejector is activated for the time contained in the hold register.

9 – 22

## Operating Mode H

Single Pulse Encoder with Software–Triggered Delayed Pulse and Hardware Gating

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| Level | | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 1 | 0 | X | X | X | X | X |

Operating mode H, shown in figure 9.1h, is identical to operating mode G except that the gate input is used to determine which clock pulse edges are counted. The counter must be armed before counting can begin.

When armed, the counter counts all clock pulse edges which occur while the gate is active. Those clock pulse edges which occur while the gate is inactive are disregarded. In this way, counting can be stopped and started by the gate.

As in operating mode G, the counter is reloaded from the hold register at the first TC. At the second TC, the counter is reloaded from the load register and then disarmed. In this operating mode, extension of the initial delay and the pulse width can be controlled by gating.
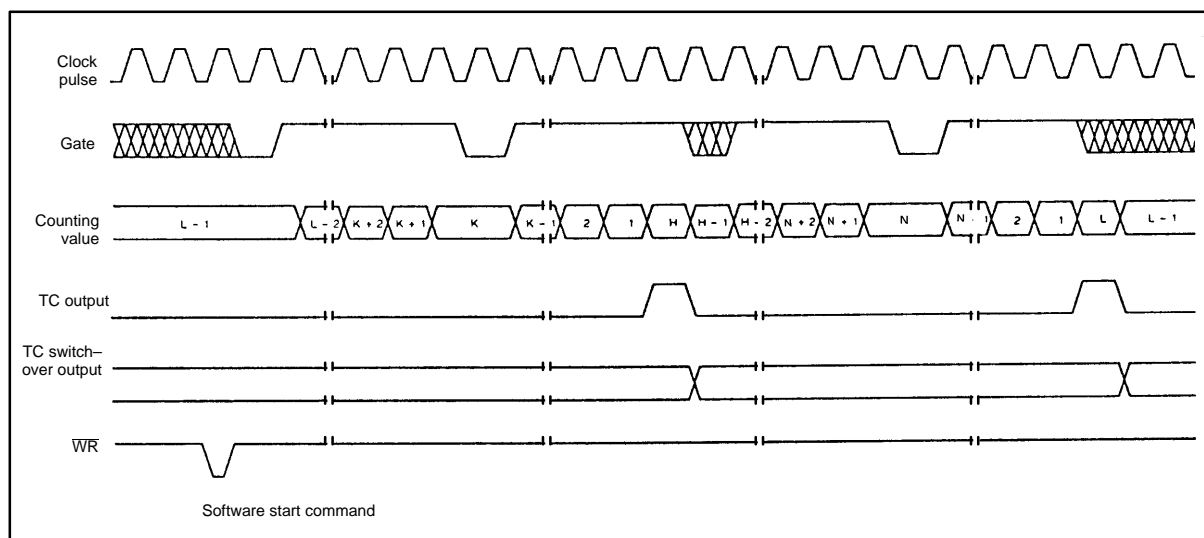


**Figure 9.1h: Signal form, operating mode H**

## Operating Mode I

Hardware–Triggered Delayed Strobe Pulse

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| Edge | | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 1 | 0 | X | X | X | X | X |

Operation mode, shown in figure 9.1i, is identical to operating mode G except that counting does not begin until a gate edge is applied to an armed counter.

The counter must be armed (software start) before application of the trigger gate edge. Gate edges which are applied to a disarmed counter (software start) are disregarded. An armed counter (software start) starts counting at the first clock pulse edge after the trigger gate edge is applied. Counting is then performed as for operating mode G.

The counter is automatically disarmed after the second TC. To start counting again, an arming command and a gate edge must be applied in that order. Remember that, after a triggered gate edge is applied, the gate input has no further effect until the second TC. This differs from operating mode H where the gate can be modulated during the counting cycle to switch the counter on and off.
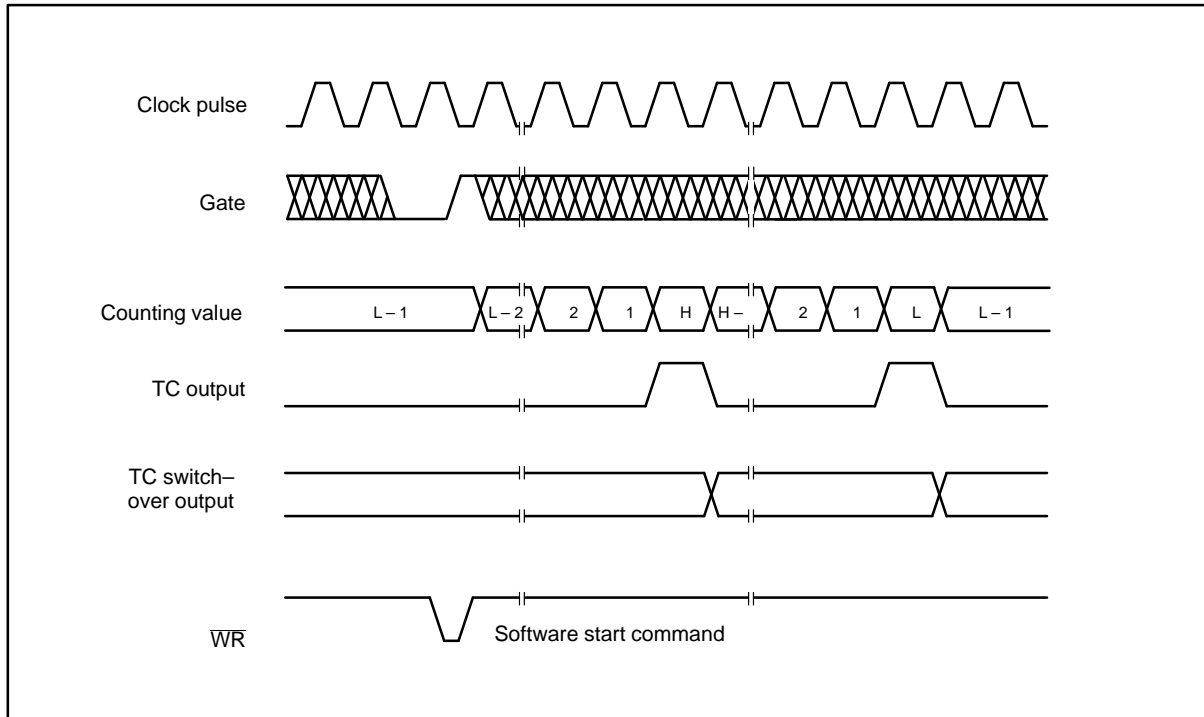


**Figure: 9.1i: Signal form, operating mode I**

## Example of Operation Mode I
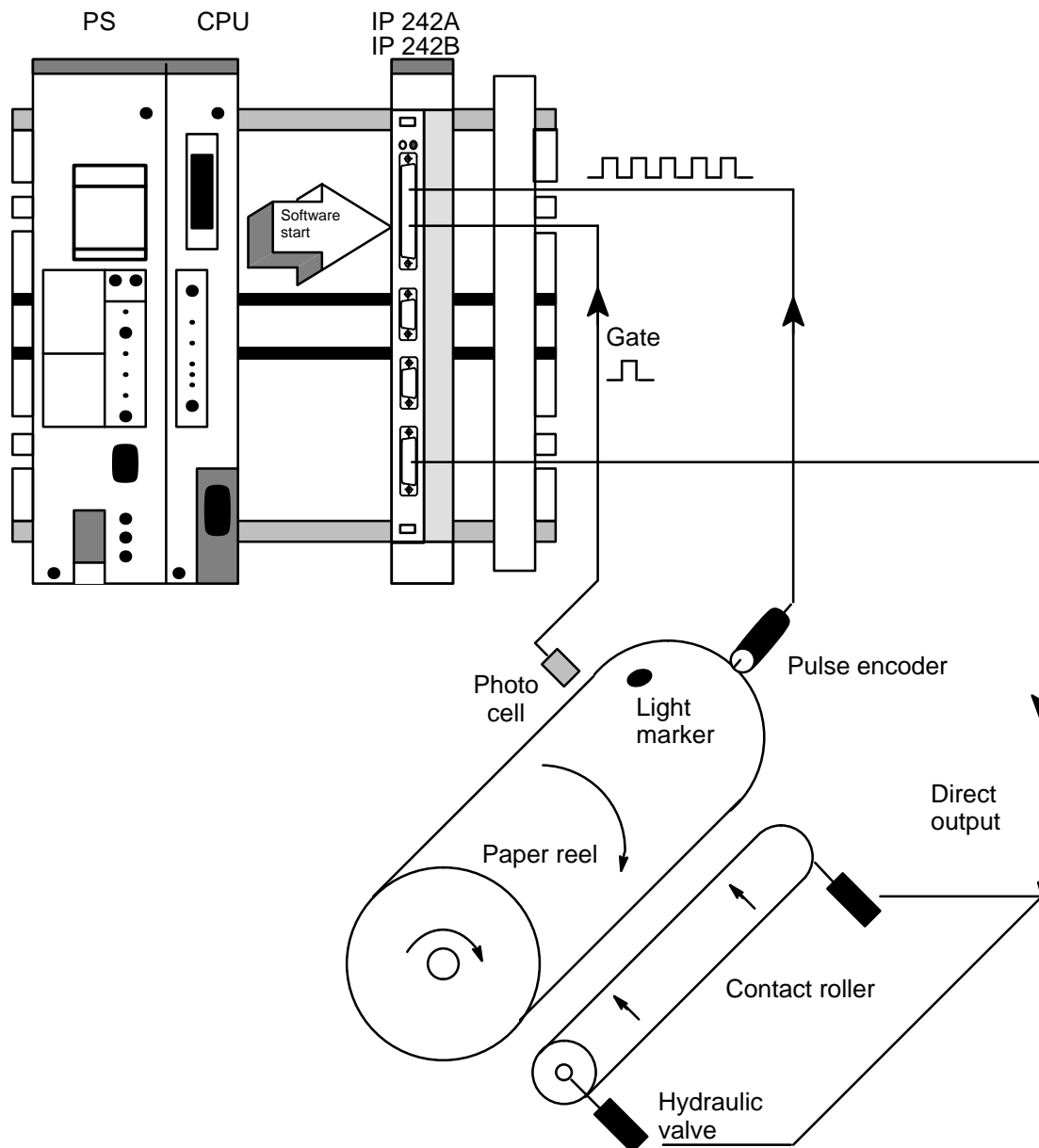


**Pulse shifting and stretching for correct placing of a contact roller on a paper reel**

1. Calculation of the point of placing based on the light marker and placing length of the contact roller.

2. Software start from the CPU to the IP 242A/242B

3. Down counting of the load register begins after a gate edge (photo cell), and the direct output is set when L = 0 (contact roller swings into position).

4. The direct output is reset again after the time specified in the hold register is up.

## Operating Mode J

Frequency Generator with Variable Duty Cycle Rate without Hardware Gating

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| 0    | 0    | 0    | X    | X    | X    | X   | X   |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0   | 1   | 1   | X   | X   | X   | X   | X   |

Operating mode J, shown in figure 9.1j, is mainly used for frequency generator applications which require a variable duty cycle rate. As soon as the count is armed, it counts continuously until it is disarmed (i.e., software stop). At the first TC, the counter is loaded again from the hold register. Counting is performed until the second TC at which time the counter is loaded again from the load register.

Counting continues until the counter is disarmed (i.e., software stop). During counting, the counter is re-loaded at each TC alternating between the two sources (i.e., the counter is reloaded from the hold register at the third TC, from the load register at the fourth TC and so on).

An output with a variable duty cycle rate is generated when the TC switch–over output is specified in the counter mode register. The load values and hold values then directly control the duty cycle rate. A high resolution is obtained when relatively high counter values are used.
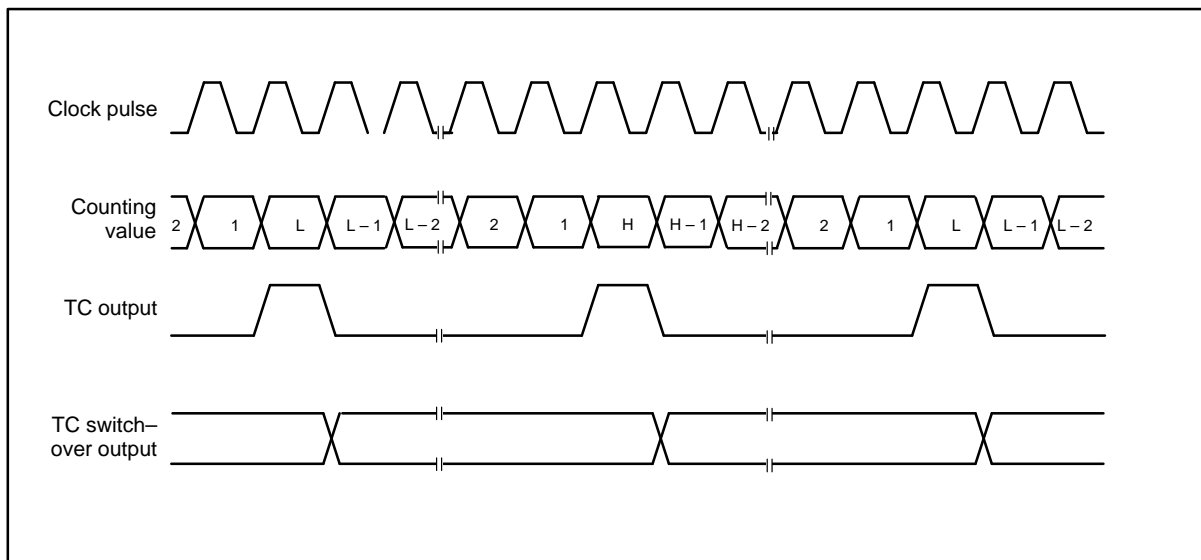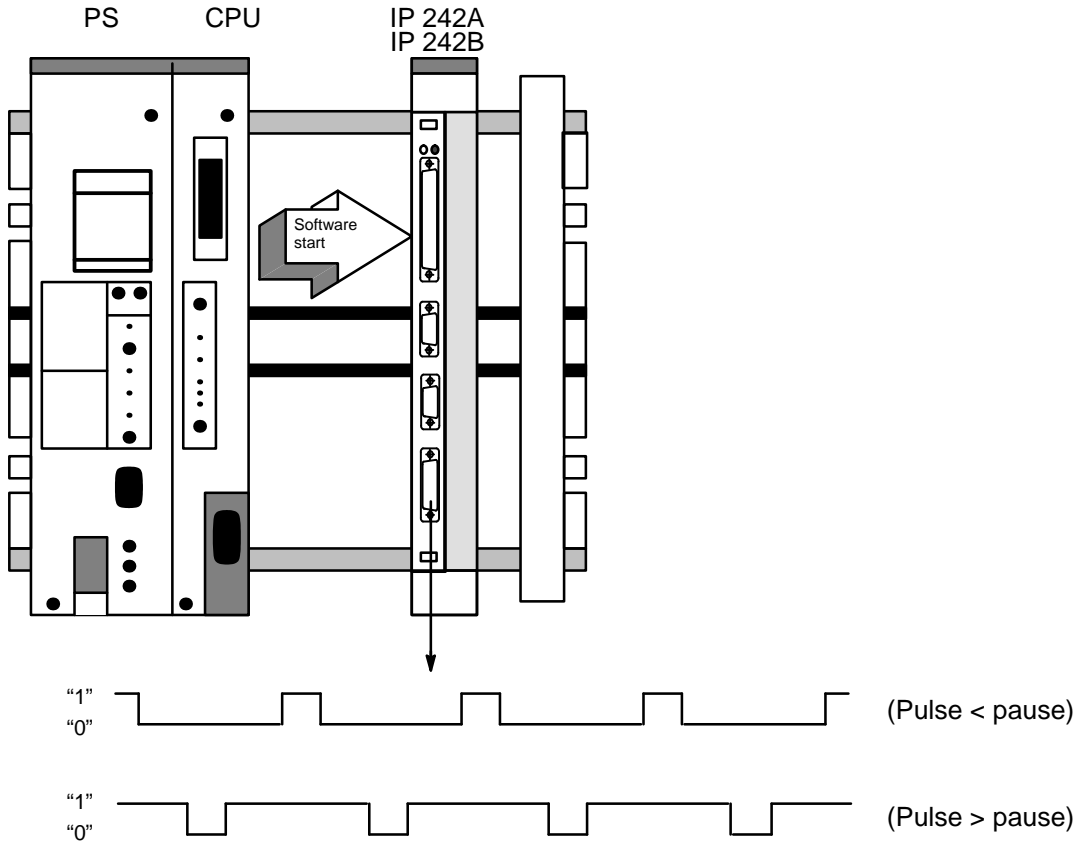


**Figure 9.1j: Signal form, operating mode J**

## Example of Operating Mode J



**Setting of the pulse width for asymmetrical pulses (e.g., to trigger and start a thyristor)**

9 − 28

## Operating Mode K

Frequency Generator with Variable Duty Cycle Rate and Level Gating

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| Level | | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 1 | 1 | X | X | X | X | X |

Operating mode K, shown in figure 9.1k, is identical to operating mode J except that the clock pulse edges are counted only when the gate is active. The counter must be armed (i.e., software start) before counting can begin.

As soon as the counter is armed, it counts all source pulse edges which occur as long as the gate is active. Clock pulse edges which occur when the gate is inactive are disregarded. This makes it possible to start and stop counting with the gate.

As in operating mode J, the sources for reloading the counter at every TC alternate starting with the hold register at the first TC after an arm command (i.e., software start command). When the TC switch–over output is used, this operating mode offers the possibility to modulate the duty cycle rate of the output frequencies. Both the positive and the negative portions of the output signal form can be modified.
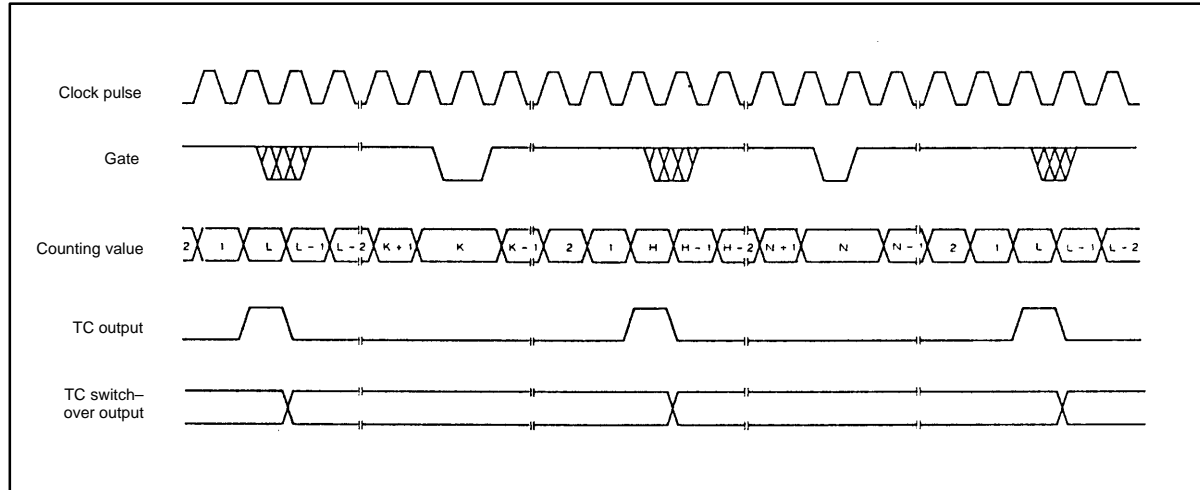


**Figure 9.1k: Signal form, operating mode K**

## Operating Mode L

Hardware–Triggered Single Pulse Encoder with Delayed Pulse

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| | Edge | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 1 | 1 | X | X | X | X | X |

Operating mode L, shown in figure 9.1l, is similar to operating mode J except that counting begins first when a gate edge is applied to an armed counter.

The counter must be armed (i.e., software start) before the trigger gate edge is applied. Gate edges which are applied to a disarmed counter are disregarded. Counting of the clock pulse edges starts after the trigger gate edge and continues until the second TC.

Remember that, after a trigger gate edge is applied, the gate input has no effect for the rest of the counting cycle. This differs from operating mode K where the gate can be modulated during the counting cycle to switch the counter on and off. The counter is loaded again from the hold register at the first TC after the trigger gate edge is applied. At the second TC, the counter is reloaded from the load register and counting is stopped until a new gate edge is applied. Remember that, in contrast to operating mode K, new gate edges are required after every second TC if counting is to continue.
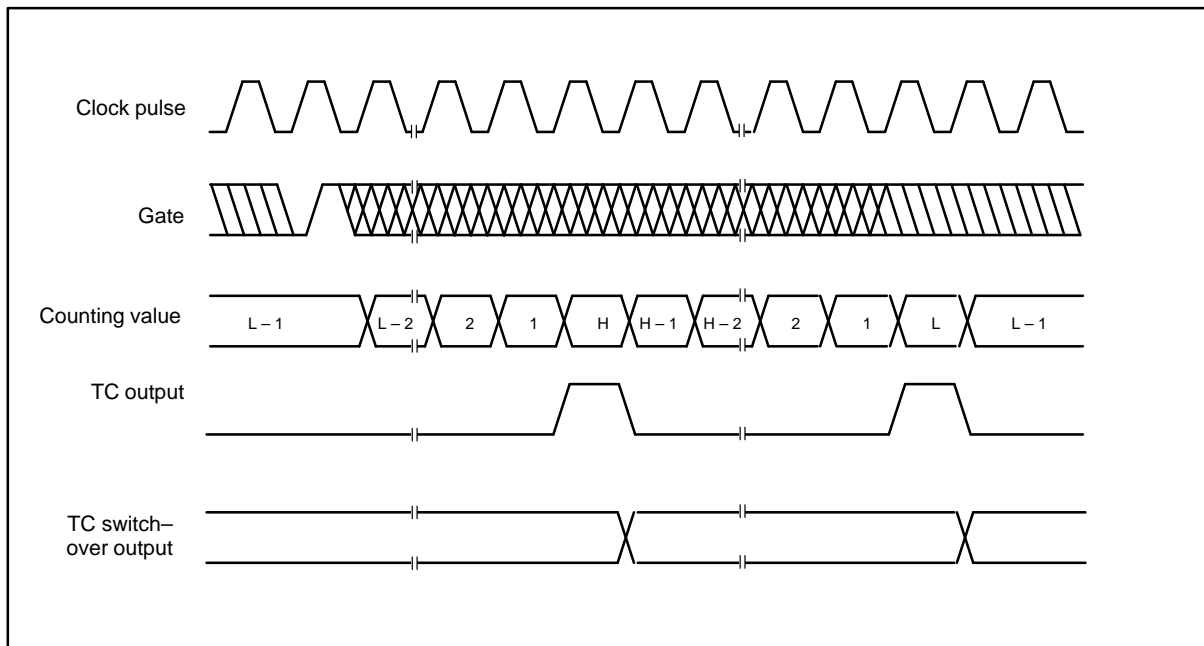


**Figure 9.1l: Signal form, operating mode L**

## Example of Operating Mode L
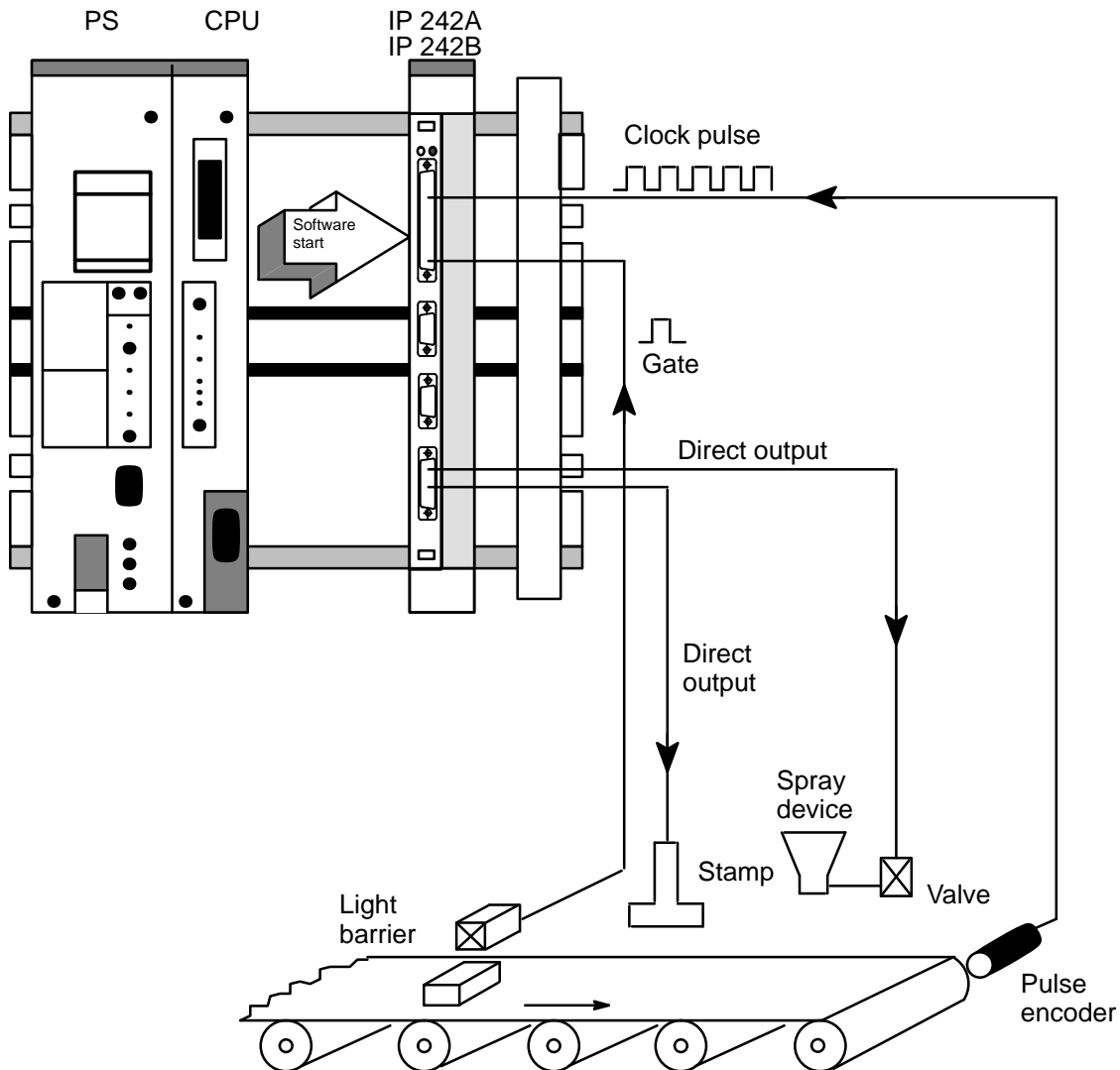


**Processing of work pieces on a conveyor belt**

1. The load register corresponds to the distance between the light barrier and the processing unit in pulses. The hold register corresponds to the running time (e.g., the spray valve).

2. Software start by the CPU to the IP 242A/242B

3. The counter is started when a work piece passes through the light barrier.

4. The part is processed via direct output regardless of the programmable controller cycle and the speed of the conveyor belt.

5. The command SA "disable the outputs" turns off the spray mist when the conveyor belt stops.

## Operating Mode N

Software–Triggered Counting with Level Gating and Hardware Retriggering

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| | Level | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 1 | 0 | 0 | X | X | X | X | X |

Operating mode N, shown in figure 9.1n, contains a software–triggered counter with level gating which can also be retriggered by the hardware. The counter must be armed (i.e., software start command) before it can begin counting.

When the counter is armed, all clock pulse edges are counted which occur when the gate is active. Those clock pulse edges which occur when the gate is inactive are disregarded. Counting can thus be started and stopped with the gate. After the arm command (i.e., software start command) is given and an active gate level is applied, the counter counts until the TC is reached. When the TC is reached, the counter is reloaded from the load register and automatically disarms to prevent further counting.

Counting starts again when a new arm command (i.e., software start command) is applied. All active gate edges applied to an armed counter (i.e., software start) cause retriggering. **The contents of the counter are stored in the internal hold register when the gate edge is applied.** The contents of the load register are transferred to the counter at the first valid clock pulse edge which occurs after the gate edge is applied for retriggering. Counting begins at the second valid clock pulse edge after the gate edge for retriggering. Valid clock pulse edges are those edges which are active while the gate is active.
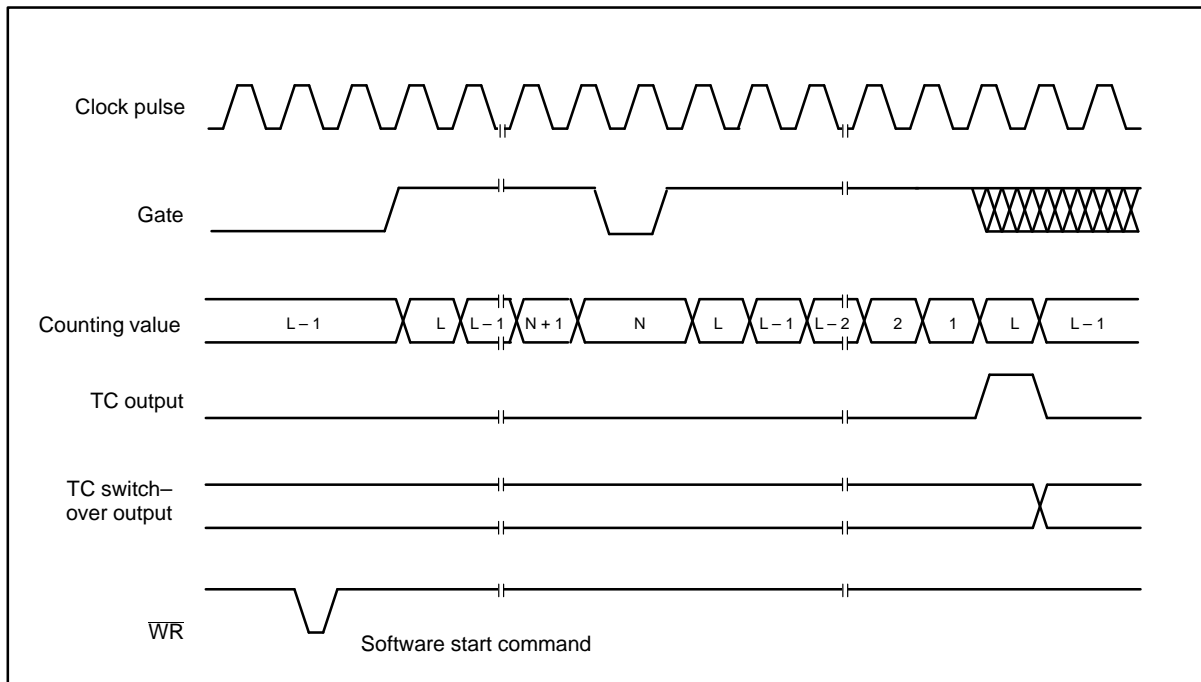


**Figure 9.1n: Signal form, operating mode N**

## Example of Operating Mode N
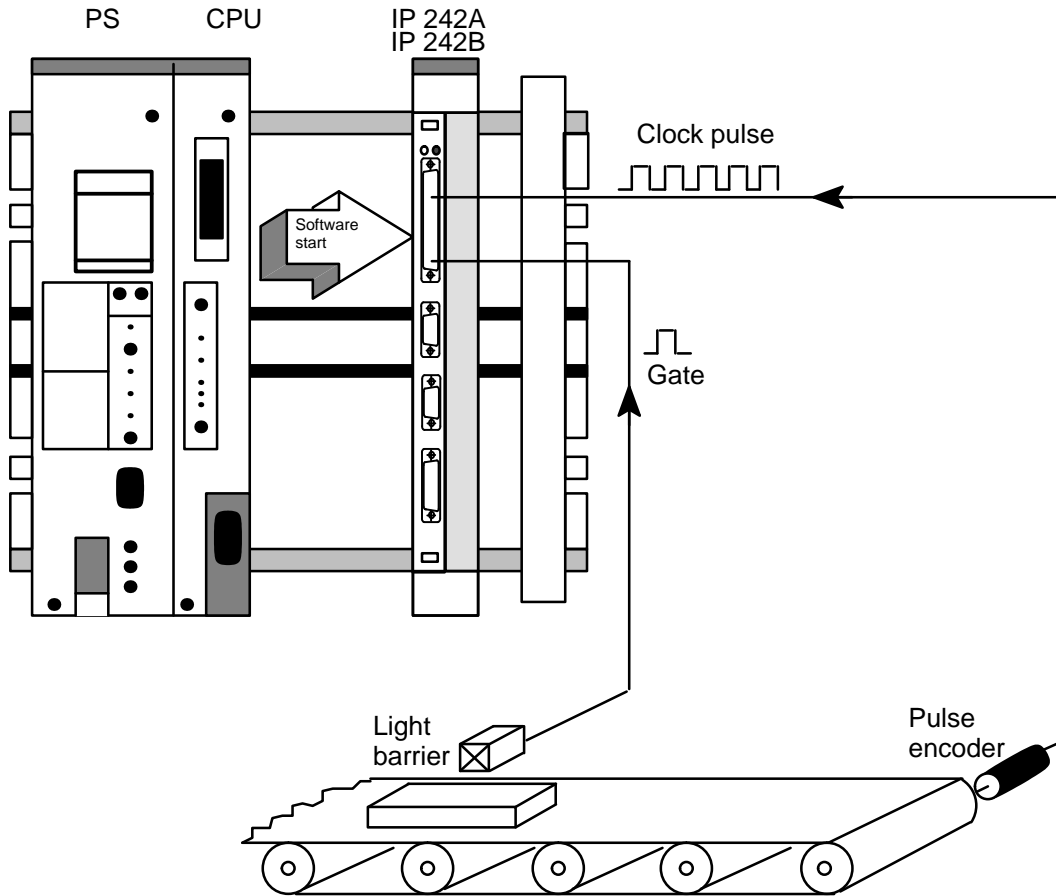


**Monitoring the length of a work piece**

1. The length of the work piece in pulses at which a reaction is required must be loaded in the load register.

2. Software start by the CPU to the IP 242A/242B

3. The counter counts the length of the work piece in pulses. The output becomes active when the maximum length is exceeded. Another software start is required after the maximum length is exceeded.

## Operating Mode O

Software–Triggered Counting with Edge Gating and Hardware Retriggering

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| Edge | | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 1 | 0 | 0 | X | X | X | X | X |

Operating mode O, shown in figure 9.1o, is similar to operating mode N except for two differences. First, counting does not start until an active gate edge is applied to an armed counter. Second, the gate level is not used to modulate counting.

The counter must be armed (i.e., software start) before the triggered gate edge is applied. Gate edges which are applied to a disarmed counter are disregarded. Regardless of the gate level, the counter counts all clock pulse edges after the triggered gate edge up to the first TC. At the first TC, the counter is reloaded from the load register and is disarmed.

A new arm command (i.e., software start command) and a new gate edge must be applied in that order before a new counting cycle can begin. In contrast to operating modes C, F, I, and L in which the gate input is disregarded after counting starts, counting in operating mode O is retriggered by all active gate edges, including the first gate edge which starts the counting process. The contents of the counter are transferred to the internal hold register at every retriggering by a gate edge. The contents of the load register are transferred to the counter at the first clock pulse edge after a gate edge which caused the retriggering. Counting starts again at the second clock pulse edge after the retriggering.
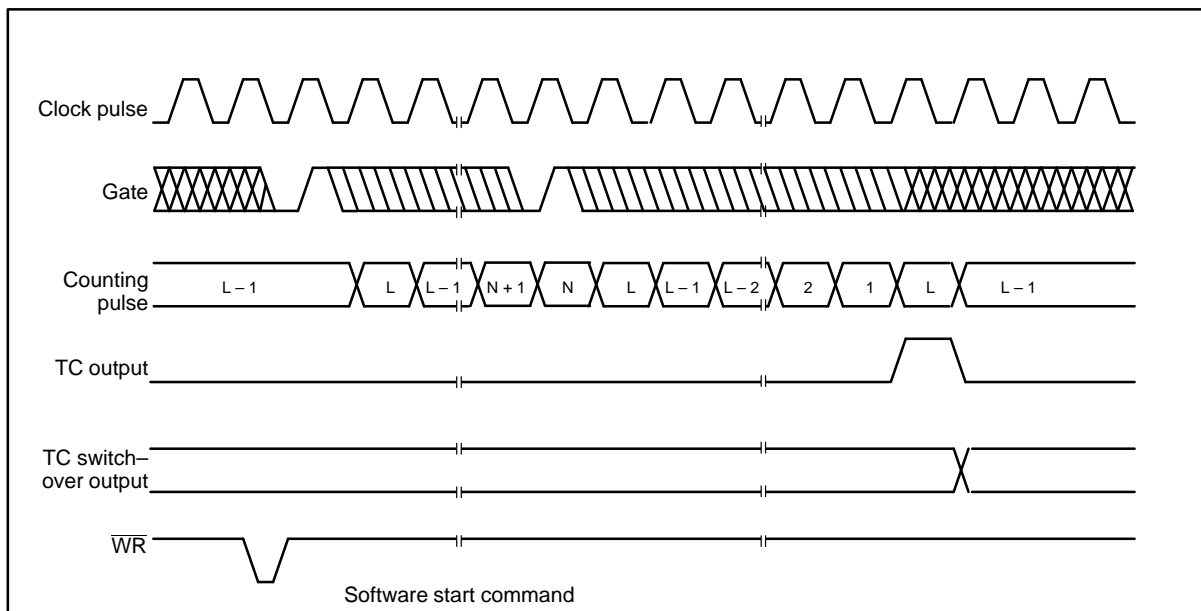


**Figure 9.1o: Signal form, operating mode O**

## Example of Operating Mode O
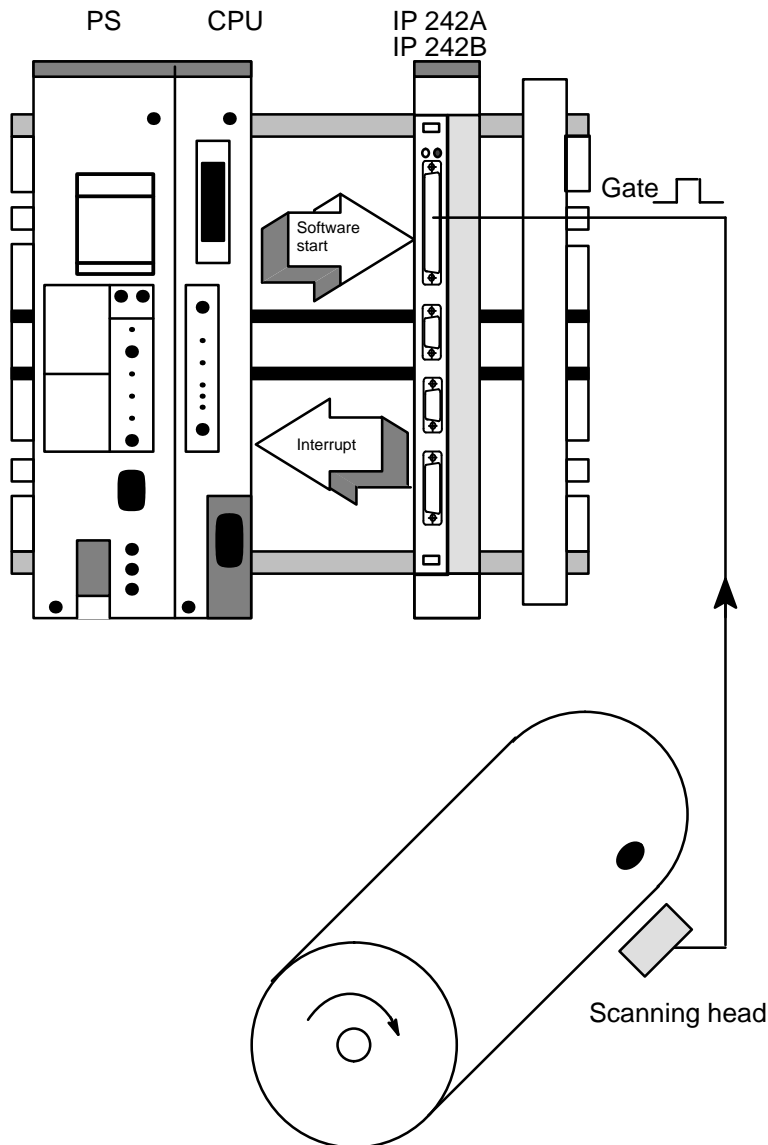


Scanning head

**Monitoring of a minimum number of revolutions**

1. After the start, the software start activates the monitoring function.

2. A pulse from the scanning head is applied to the gate once for each revolution. The active edge of the gate signal resets the counter.

3. An internal frequency is used as the input signal of the counter. The load register contains the monitoring value at which the counter output is activated.

## Operating Mode Q

Frequency Generator with Synchronization (Event Counter with Automatic Read/Reset)

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| Level | | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 1 | 0 | 1 | X | X | X | X | X |

Operating mode Q, shown in figure 9.1q, contains a frequency generator with synchronization or an event counter with automatic read/reset. The counter must be armed (i.e., start software command) before counting can begin.

When armed (i.e., software start), the counter counts all clock pulse edges which occur while the gate is active. Those clock pulse edges which occur when the gate is inactive are disregarded. Counting can thus be started and stopped by the gate. After an arm command (i.e., software start command) and the application of an active gate signal, the counter counts continuously up to the TC. The counter is automatically reloaded from the load register at every TC.

The counter can be retriggered at any time by applying an active gate edge to the gate input. The retriggered gate edge causes the contents of the counter to be transferred to the internal hold register. The contents of the load register are transferred to the counter at the first valid clock pulse edge after the retriggering gate edge. Counting starts with the second valid clock pulse edge after the retriggering gate pulse edge. Valid clock pulse edges are active edges which occur when the gate is active.
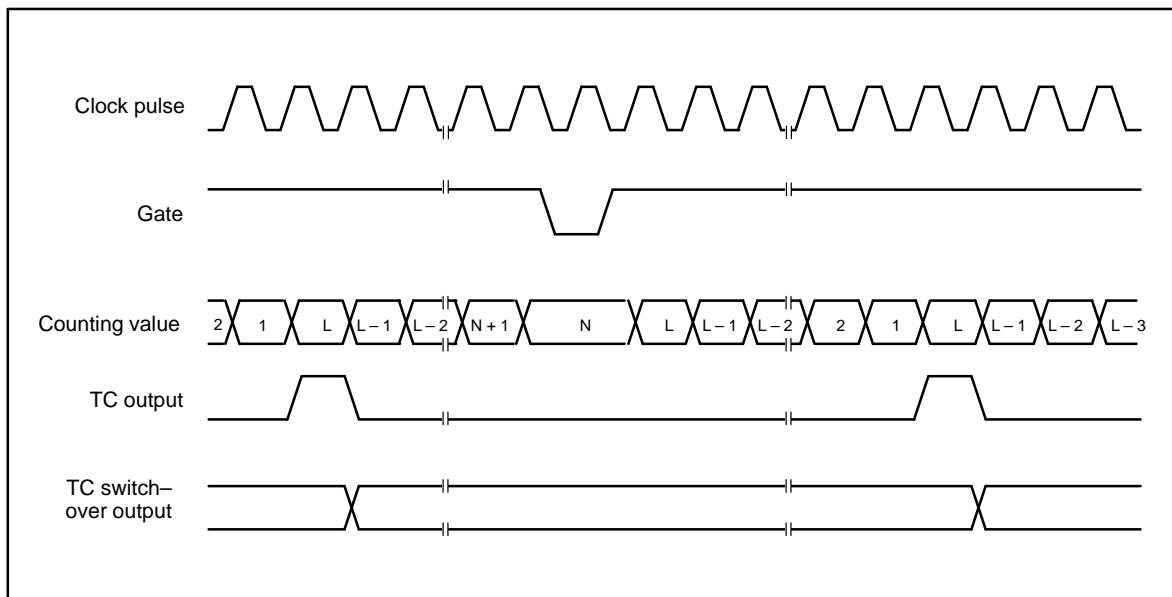


**Figure 9.1q: Signal form, operating mode Q**

## Example of Operating Mode Q
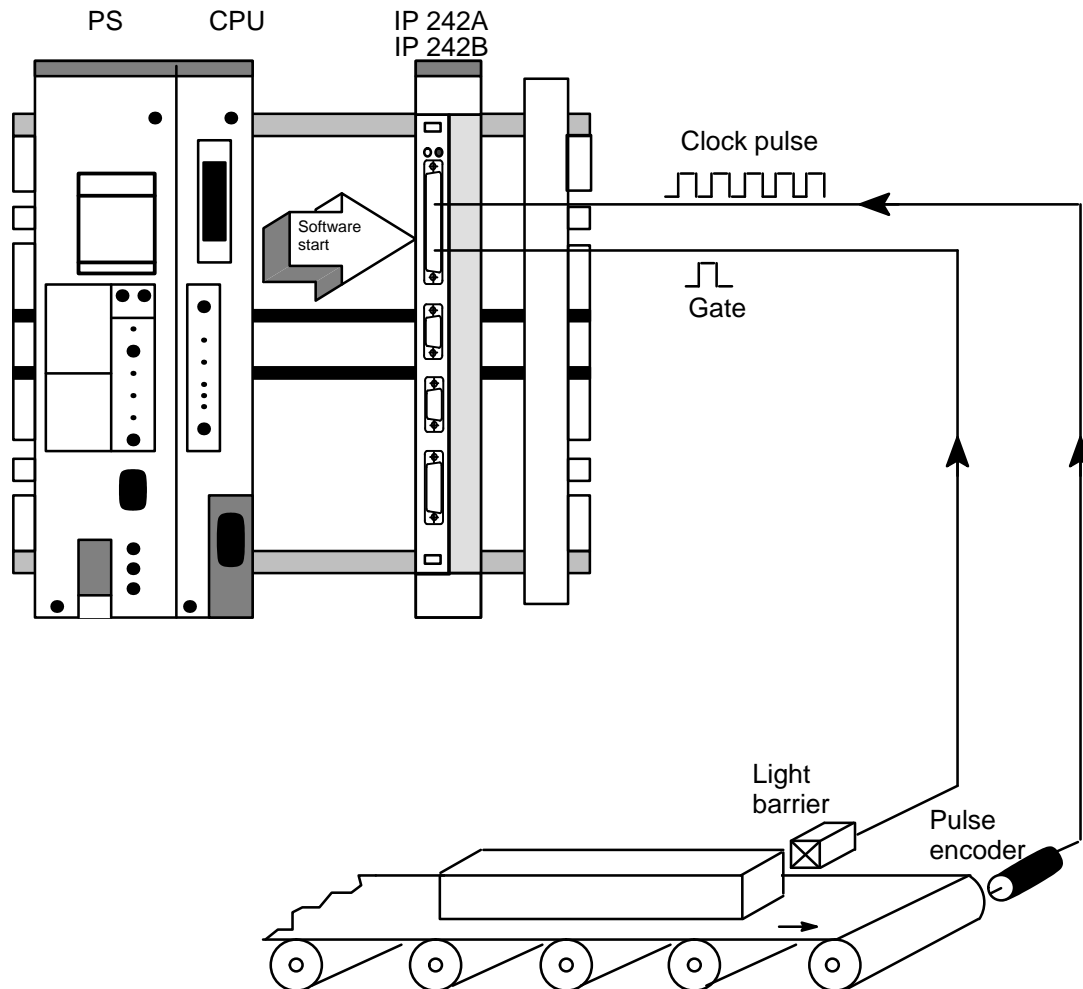


**Measuring the length of work pieces**

1. Software start from the CPU to the IP 242A/242B

2. The encoder pulses are counted as long as the gate input is activated by the light barrier.

3. When the light barrier registers the next work piece, the length value of the last work piece is stored in the internal hold register of the counter. The "copy counter" command causes the contents of the internal hold register to be transferred to the data block.

4. The contents of the hold register are proportional to the length of the work piece.

## Operating Mode R

Retriggerable Single Pulse Encoder

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| | Edge | | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 1 | 0 | 1 | X | X | X | X | X |

Operating mode R, shown in figure 9.1r, is similar to operating mode Q except that edge gating and not level gating is used (i.e., instead of the gate level determining which clock pulse edges are counted, gate edges are used to start counting).

The counter must be armed before a trigger gate edge is applied. Gate edges which are applied to a disarmed counter are disregarded. After a gate edge is applied, an armed counter (i.e., software start) counts all clock pulses up to TC regardless of the gate level.

At the first TC, the counter is reloaded from the load register and then stopped. Counting starts again only after a new gate edge is applied to the counter, including the first gate edge with which the counting begins. When a gate edge is applied, the contents of the counter are stored in the internal hold register.

The contents of the load register are transferred to the counter at the first clock pulse edge after the gate edge which caused the retriggering. Counting starts at the second clock pulse edge after the gate edge which caused the retriggering.
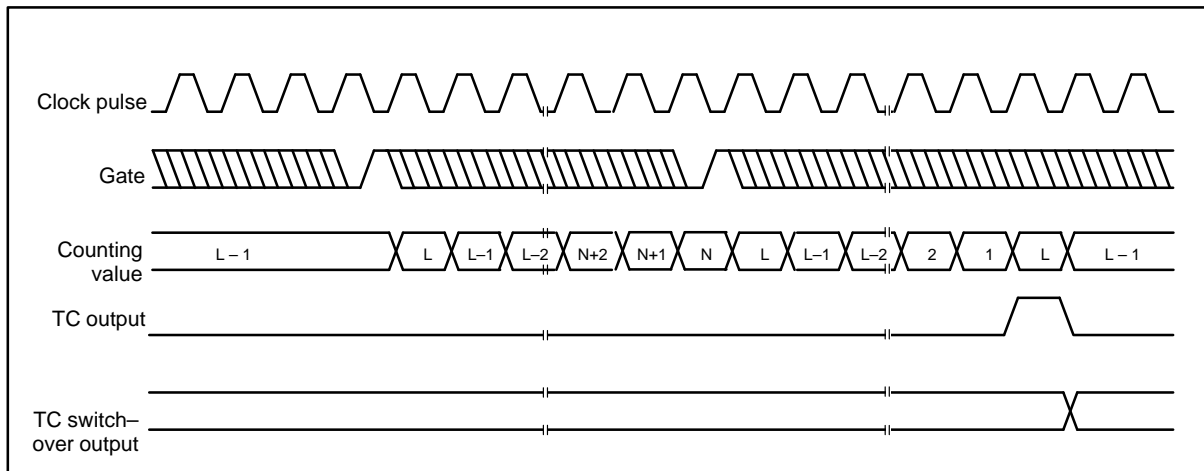


**Figure 9.1r: Signal form, operating mode R**

## Operating Mode S

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| 0 | 0 | 0 | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 1 | 1 | 0 | X | X | X | X | X |

In this operating mode, the gate input determines the source for reloading triggered by load commands (regardless of whether the counter is armed or not) and for reloading caused by the TC. In operation mode S, the gate input is used only to select the source for reloading and not to start or modulate the counting process.

The load register is used when an L level is available at the gate; the hold register is used when an H level is available at the gate. L level is used for the load register and H level is used for the hold register because these terms are easier to remember.

As soon as the counter is armed (i.e., software start), it counts to TC twice and then automatically disarms (i.e., software start command). At every TC, the counter is reloaded from the source determined by the gate.

After the second TC, an arm command (i.e., software start command) is required to start another counting cycle. Operating mode S is shown in figure 9.1s.
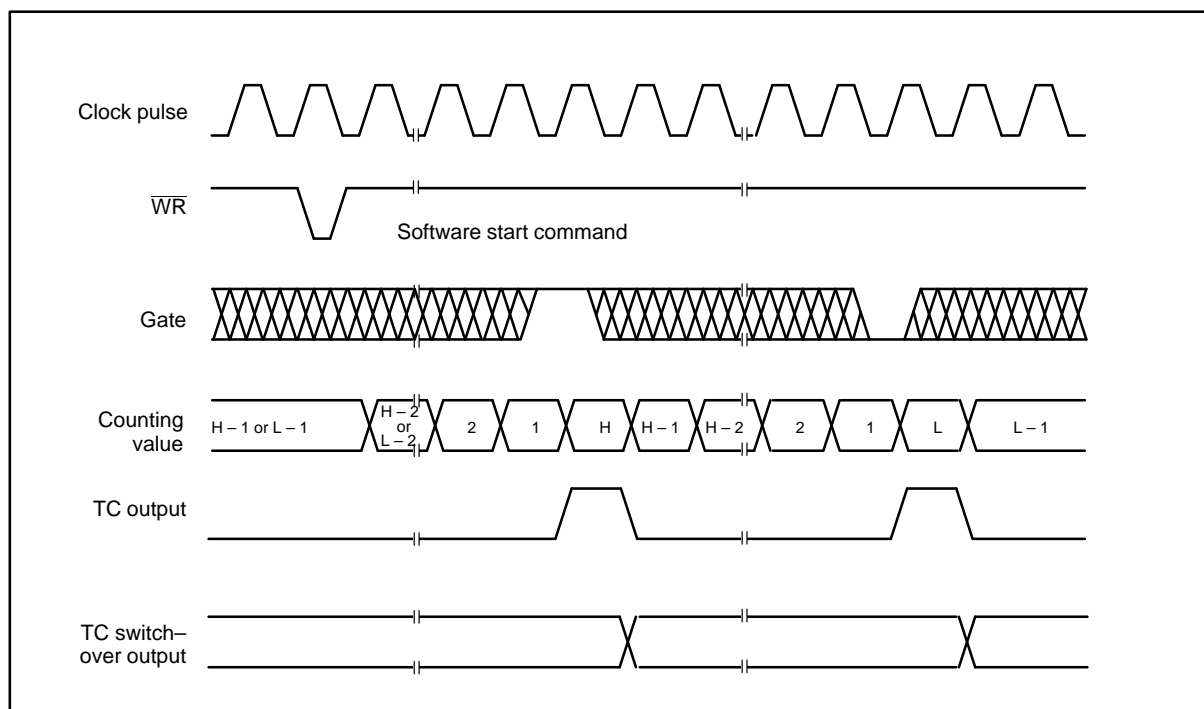


**Figure 9.1s: Signal form, operating mode S**

## Operating Mode V

Frequency Shift Keying

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
| 0 | 0 | 0 | X | X | X | X | X |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 1 | 1 | 1 | X | X | X | X | X |

Operating mode V, shown in figure 9.1v, offers modulation capability by frequency shift keying. Operation of the gate in this operating mode is identical to that in operating mode S. If an L level is available at the gate, the counter is reloaded from the load register when a load command or a reloading triggered by a TC occurs. If an H level is available at the gate, loading and reloading are performed from the hold register.

The polarity of the gate selects only the source from which to load. The polarity does not start or modulate counting. In armed status (i.e., software start), the counter counts continuously up to the TC. At every TC, the counter is automatically reloaded from the register determined by the polarity gate.

Counting continues in this manner until the counter is disarmed (i.e., software stop). Frequency shift keying is obtained by specifying operating mode TC switch–over output in the counter mode register. Frequency switching is done by modulating the gate.
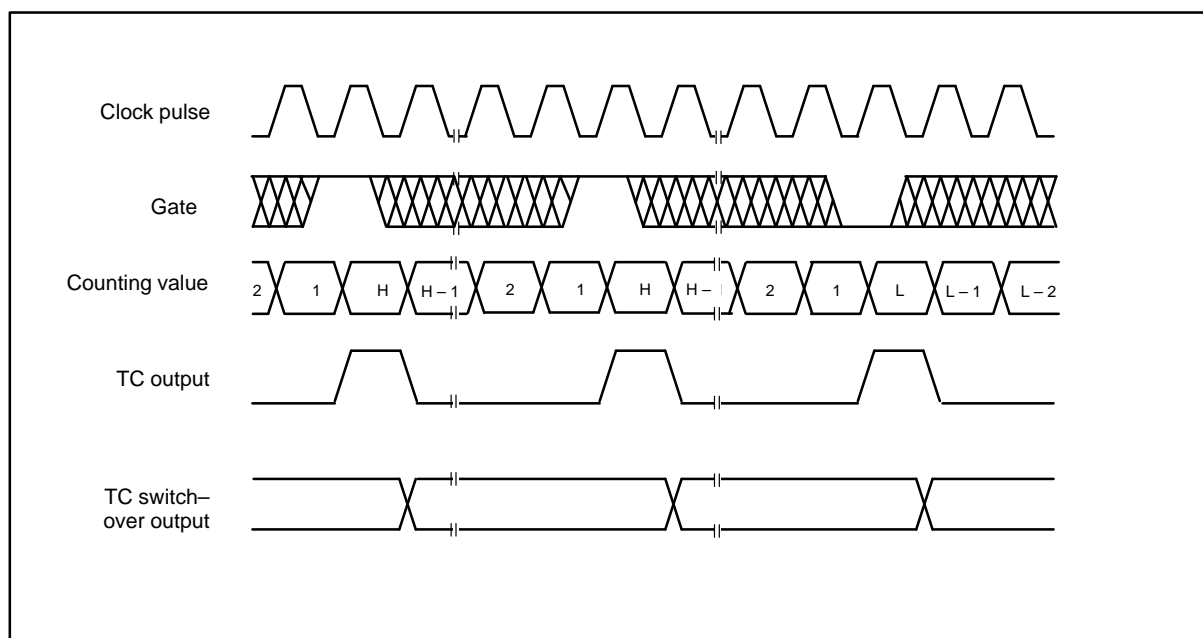


**Figure 9.1v: Signal form, operating mode V**

## Operating Mode X

Hardware Memory Storage

| CM15 | CM14 | CM13 | CM12 | CM11 | CM10 | CM9 | CM8 |
|------|------|------|------|------|------|-----|-----|
|      | Edge |      | X    | X    | X    | X   | X   |

| CM7 | CM6 | CM5 | CM4 | CM3 | CM2 | CM1 | CM0 |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 1   | 1   | 1   | X   | X   | X   | X   | X   |

In operating mode X, shown in figure 9.1x, the contents of the counter are scanned by the hardware without interrupting counting. A load and arm command or a load command followed by an arm command (software start) are required to initialize the counter. As soon as the counter is armed, counting is started by a gate edge. Gate edges applied to a disarmed counter are disregarded.

After application of the trigger gate edge, all valid gate edges are counted up to the second TC regardless of the gate level. The actual counting value is stored in the internal hold register for all gate edges which are applied during counting but counting is not interrupted. At every TC, the counter is reloaded from the load register and stopped at every second TC. Another trigger gate edge is required for further counting. Counting begins at the first clock pulse edge after the trigger gate edge.
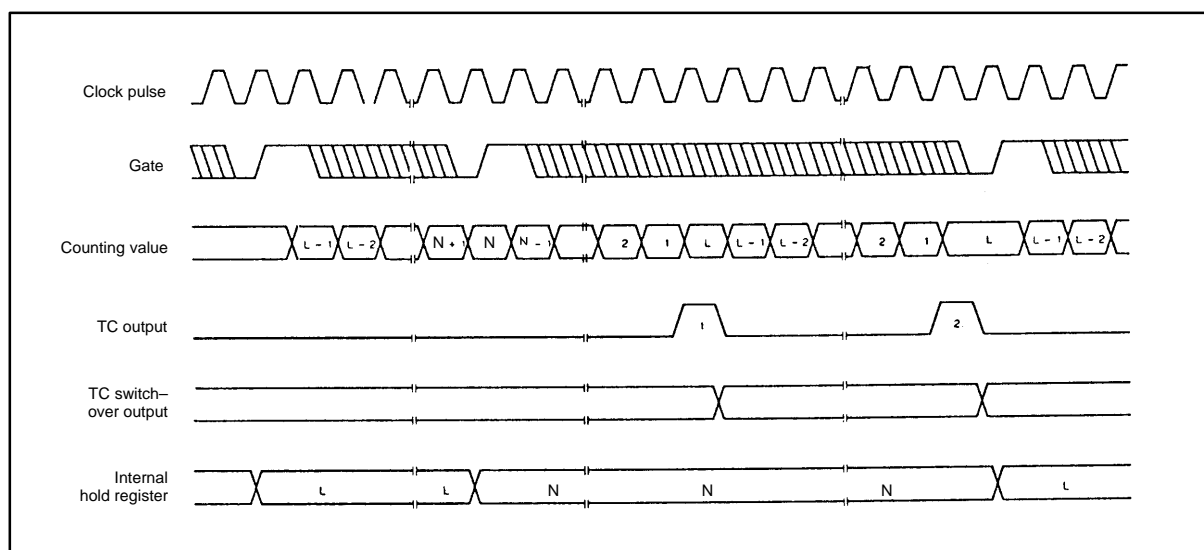


**Figure 9.1x: Signal form, operating mode X**

## 9.2    Operating Modes for Counters 6 and 7

## 9.2.1  Counting

**IP 242A/242B**

**A**  **B**   The operating modes described in section 9.1 cannot be used for counters 6 and 7.

By using the counter mode register assignments shown in the following figures, you can choose between single, double, or quadruple edge evaluation of counting pulse A or B.

This allows you to adjust the encoder resolution to your requirements.

**Counting up:**
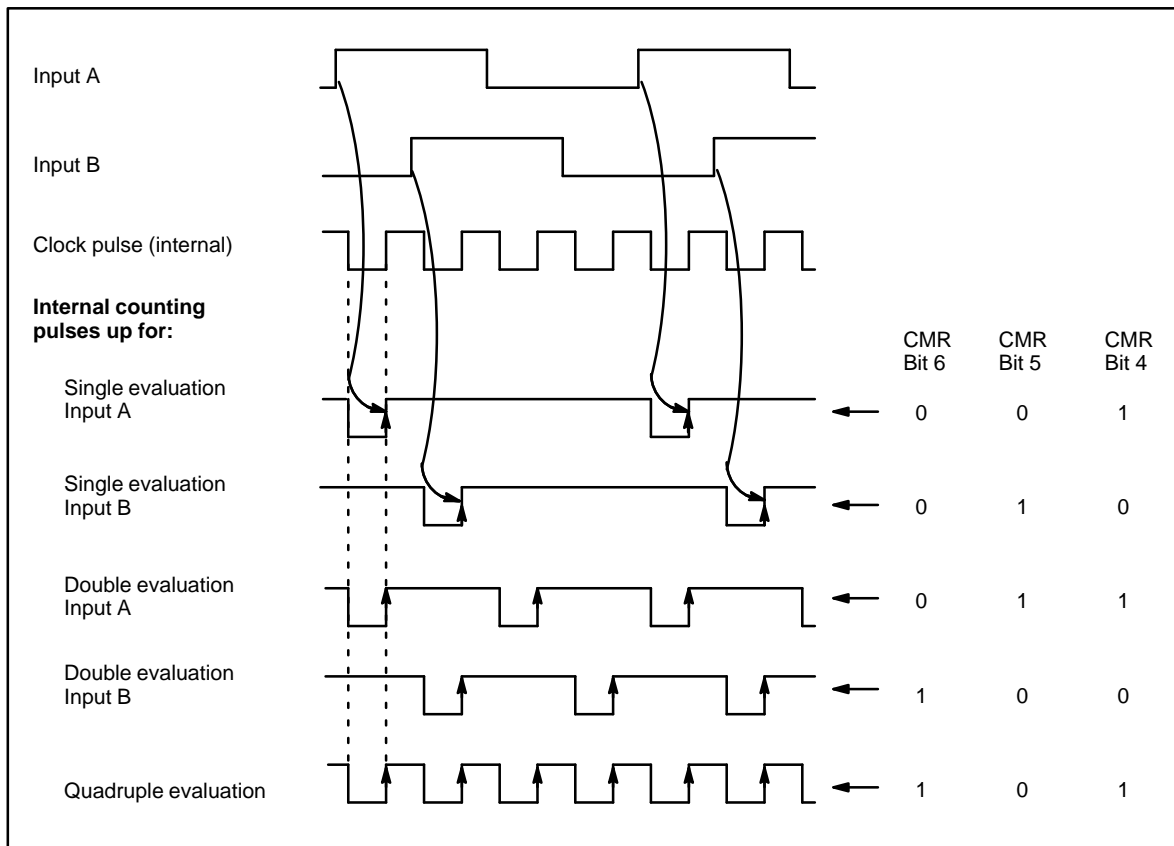


Notes:

⌐_   =  Contents of the counter decreased by 1

CMR  =  Counter mode register

**Counting Down:**

Input A

Input B

Clock pulse, internal

**Internal counting pulses down for:**

|  | CMR<br>Bit 6 | CMR<br>Bit 5 | CMR<br>Bit 4 |
|---|---|---|---|
| Single evaluation<br>Input A | 0 | 0 | 1 |
| Single evaluation<br>Input B | 0 | 1 | 0 |
| Double evaluation<br>Input A | 0 | 1 | 1 |
| Double evaluation<br>Input B | 1 | 0 | 0 |
| Quadruple evaluation | 1 | 0 | 1 |
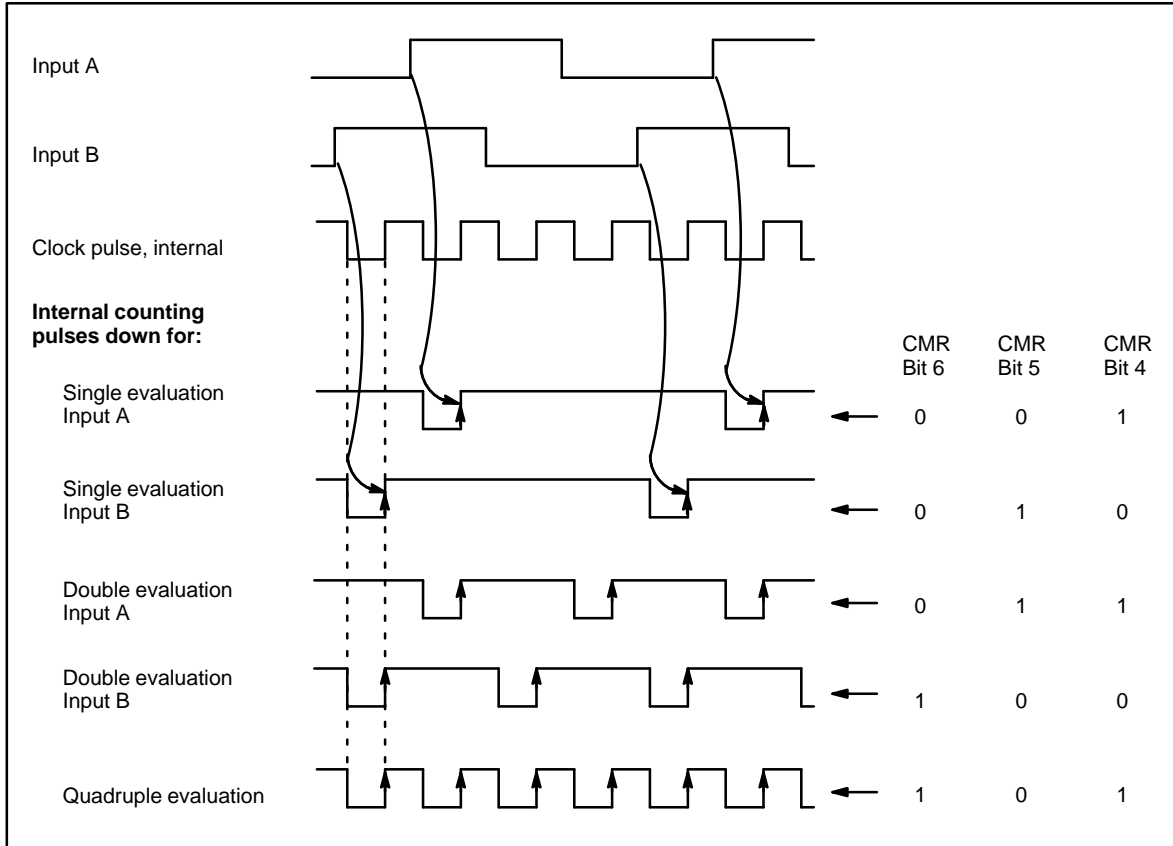
Notes:

　┌┐　 =   Contents of the counter decreased by 1

CMR   =   Counter mode register

## 9.2.2 Gate Time Measurement

**IP 242B**

**B** In this operating mode, the duration of a gate signal at input A is measured with a resolution of 100 nsec. Input B is used for a switch in direction.

| Level at Input B | Counting Direction |
|------------------|--------------------|
| H | Up |
| L | Down |

When the external reset via inputs N and SYN is used, a counting error of $\pm1$ can occur (i.e., the counter value after a reset is either $-1$, 0 or $+1$).

Allocation of the Counter Mode Register

| 15 | | | | | | | | 8 | 7 | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | 1 | 1 | 0 | x | x | x | x |

Pulse Diagram

IP 242A/242B Equipment Manual
Ⓒ Siemens AG 1993, Order No: 6ES5 998−0KM21
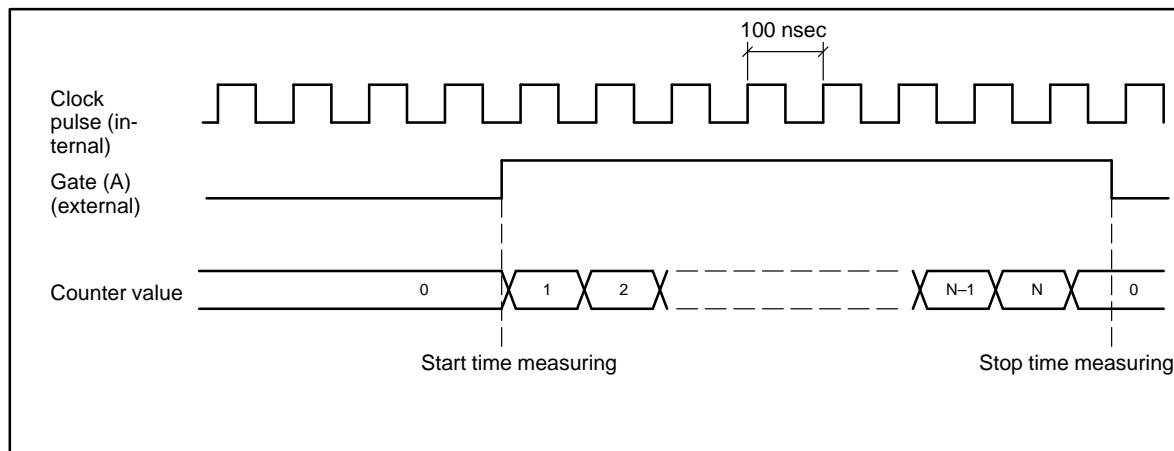
The measured gate time can be read into the counter value register after the "read counter" (LE) control word. The contents of the counter value registers (ZSZ) change each time a falling gate edge occurs. The ZSZ contains the measured gate time if the gate is "closed". The counter starts automatically at "0" when a new gate signal occurs.

Since counters 6 and 7 are already started after the parameterization, an additional "start counter (ST)" is not required.

If the counter is stopped (by SP) in this operating mode, the first gate time measurement after another start will be incorrect. For this reason, the counter should always remain started.

The load register ought to be preset with zero. Any other presetting in this operating mode is not realistic.

In contrast to the gate functions of counters 1 to 5, an interrupt **cannot** be triggered here at the gate start or end.

The counting width is 24 bits.

## 9.2.3    Frequency Measurement

**IP 242B**

**B** In this operating mode, the frequency at input A is acquired until the gate at input B is high. The direction of counting is **always up**.

When the external reset via inputs N and SYN is used, a counting error of $\pm 1$ can occur (i.e., the counter value after a reset is either –1, 0 or +1).

Allocation of the Counter Mode Register:

| 15 | | | | | | | 8 | 7 | | | | | | | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | 1 | 1 | 1 | x | x | x | x |
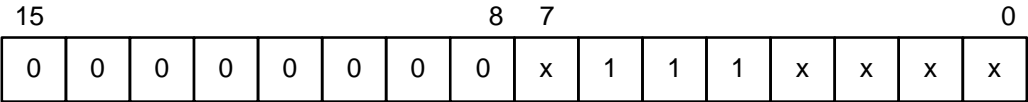
Pulse Diagram
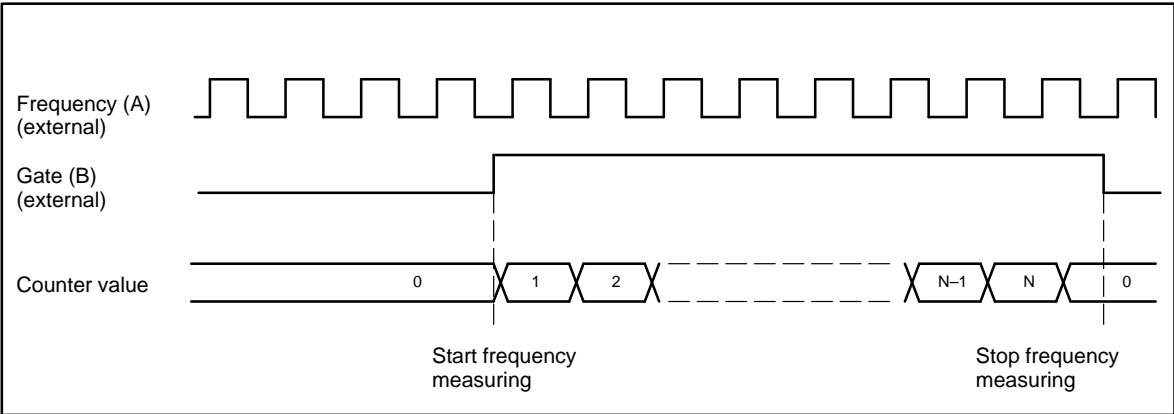


The measured gate time can be read into the counter value register after the "read counter" (LE) control word. The contents of the counter value registers (ZSZ) change each time a falling gate edge occurs. The ZSZ contains the measured frequency if the gate is "closed". The counter starts automatically at "0" when a new gate signal occurs.

Since counters 6 and 7 are already started after the parameterization, an additional "start counter (ST)" is not required.

If the counter is stopped (by SP) in this operating mode, the first frequency measurement after another start will be incorrect. For this reason, the counter should always remain started.

The load register ought to be preset with zero. Any other presetting in this operating mode is not realistic.

In contrast to the gate functions of counters 1 to 5, an interrupt **cannot** be triggered here at the gate start or end.

The counting width is 24 bits.

# 10 Technical Specifications

## 10.1　　Inputs for Counters 1 to 5

**Counter Channels 1 to 5:**

- Number of CNT Inputs : 5
- Number of STA Inputs : 5
- Number of STO Inputs : 5
- Potential Isolation
    - Between Two Inputs : yes
    - Between Input and Output : yes
    - Between Input and S5 bus : yes

**Two Different Input Voltages Are Available**

24 V Level:

- Input Rated Voltage : 24 V
- Input Voltage for Signal "0" : −3 V to 4.5 V
- Input Voltage for Signal "1" : 13 V to 30 V
- Signal Status for Nonconnected Inputs : low
- Input Resistance : typically 1.8 kΩ
- Input Current for Signal "1" (Rated Voltage 24 V) : typically 13 mA
- Input Current (13 V to 30 V) : 10 mA to 30 mA

5 V Level:

- Input Rated Voltage : 5 V
- Input Voltage for Signal "0" : −3 V to 1.5 V
- Input Voltage for Signal "1" : 4.0 V to 6.5 V
- Signal Status for Nonconnected Inputs : low
- Input Resistance : typically 420 Ω
- Input Rated Voltage at Signal "1" (5 V) : typically 12 mA
- Input Current (4 V to 6.5 V) : 10 mA to 30 mA

## 10.2   Inputs for Counters 6 and 7

**Differential Inputs A,$\overline{\text{A}}$ B,$\overline{\text{B}}$ N,$\overline{\text{N}}$**

- Maximum Voltage to Ground : +13 V
- Minimum Voltage to Ground : −13 V
- Maximum Differential Voltage : ±9 V
- Minimum Differential Voltage : ±0.6 V
- Signal Status for Nonconnected Input : Not defined
- Input Resistance : 150 Ω
- Potential Isolation
  - Between Two Differential Inputs : no
  - Between Two Differential Inputs and S5 Bus : no
  - Between Differential Input and Digital Output : yes

**Synchronization Input SYN**

- Input Rated Voltage : 5 V, 24 V
- Input Voltage for Signal "0" : −3 V to 1.5 V
- Input Voltage for Signal "1" : 4.0 V to 30 V
- Signal Status for Nonconnected Input : low
- Input Resistance : typically 3.8 kΩ
- Input Rated Current at Signal "1" (5 V) : typically 1.2 mA
- Input Rated Current at Signal "1" (24 V) : typically 6 mA
- Input Current (4 V to 30 V) : 1 mA to 8 mA
- Input Time Constant : typically 1 msec
- Minimum Pulse Width : 3 msec
- Potential Isolation
  - Between SYN Input and S5 Bus : no
  - Between SYN Input and Digital Output : yes

## 10.3    Digital Outputs (P Switch)

- Number of Outputs                                          :                                    7
- Output Voltage at Signal Status "1"                         :                $\geq U_{24Vexternal} - 2.5$ V
- Output Voltage at Signal Status "0"                         :                              < 3 V
- Maximum Output Current                                      :                            200 mA
- Maximum Total Current of All Outputs                        :                            700 mA
- Maximum Delay Time for Switch On (Low to High)              :                           15 $\mu$sec
- Maximum Delay Time for Switch Off (High to Low)             :                           30 $\mu$sec
- Maximum Output Frequency (at Scan Ratio of 1:1)             :                        Approx. 40 kHz
- Short Circuit Protection                                    :                               yes
  Short–circuited outputs (i.e., output current > 200 mA)
  are switched off
- Overload Protection                                        :                               yes
  All outputs are switched off when an overload
  (i.e., total current > 200 mA) occurs.
- Switch–Off Time at Enabling of the Outputs                 :                           $\leq$ 30 $\mu$sec
- Enabling of the Inputs                                     :                      S5 signal BASP = 0
                                                                                  and $U_{5Vinternal}$ within
                                                                                      tolerance range
- Potential Isolation
  - Between Two Digital Outputs                              :                                no
  - Between Digital Output and S5 Bus                        :                               yes
- Free–Wheel Diode Required for Inductive Loads
  See section 1.8 for circuitry.

## 10.4   Counting Frequencies

**Counters 1 to 5:**

The data applies to all CNT, STA, and STO inputs.

- Maximum Counting Frequency (without Capacitors)             :                    480 kHz
- Pulse Width (High and Low Pulse)                            :                  $\geq$ 1.04 $\mu$sec

**Counters 6 and 7:**

- Maximum Encoder Frequency                                   :                    500 kHz
- Pulse Width A,$\overline{A}$ B,$\overline{B}$ N,$\overline{N}$ (High and Low Pulse)   :                  $\geq$ 1.0 $\mu$sec
- Pulse Width SYN Input (High Pulse)                          :                    > 3 msec

**Internal Frequency Generator**

- Maximum Frequency F1                                        :                      1 mHz
- Maximum Cycle Duration of the Internal Frequency           :                  68718 sec
- Maximum Gate Time                                          :                  34359 sec
- Accuracy of the Internal Frequency (0˚ C to 70˚ C)         :                  $\pm$ 100 ppm

## 10.5    Power Supply

**Supply Voltage U$_{5\ V\ internal}$ via Basic Connector X1**

- Rated Value                                          :                              +5 V
- Lower Limit                                          :                           +4.75 V
- Upper Limit                                          :                           +5.25 V
- Power Consumption                                    :              approximately 1.1 A
- Maximum Power Consumption                            :                             1.2 A

**Supply Voltage U$_{24\ V\ external}$ via Front Connector X6**

- Static Limits (Including Ripple)
  - Lower Limit                                        :                            +20 V
  - Upper Limit                                        :                            +30 V
- Dynamic Limits
  - Lower Limit
    Value                                              :                         +14.25 V
    Duration                                           :                          5 msec
    Recovery Time                                      :                          10 sec
  - Upper Limit
    Value                                              :                            +35 V
    Duration                                           :                        500 msec
    Recovery Time                                      :                          50 sec
- Ripple                                               :                    $\leq$ 3.6 V$_{ss}$
- Power Consumption                                    :              maximum 800 mA
  (dependent on  circuitry of the outputs)
- Power Consumption without Load                       :             approximately 50 mA
- Overcurrent Protection                               :                    not available
- Voltage Monitor                                      :                    not available

The module can be operated without a fan subassembly.

## 10.6    General Data

- Differences in Potential:

  – 24V / TTL Inputs <−> Module          :                           $\leq$ 500 V

  – Outputs <−> Module                        :                           $\leq$ 500 V

  – Differential Inputs <−> Module        :                            $\leq$ 13 V

- Duration of the Red LED during Self Test    :                           < 0.5 sec

- Flash Frequency of Green LED on an Unparameterized Module          :                  approximately 1.5 Hz

- Environment Temperature                    :                       $0°$ C to $55°$ C

- Storage Temperature                          :                     $-40°$ C to $70°$ C

- Humidity Rating                                  :                       F in accordance with DIN 40040

- Degree of Protection                           :                      IP 20 when mounted in module rack

- Components                                        :                      Components listed in the UL or CSA are used.

- PCB                                                  :                       double Europe format, con–struction height 1–1/3 SEP

- Module Dimensions                           :                     233 mm x 160 mm x  20 mm

S Front Connector                               :

| Plug Connec. | Type Used |
|---|---|
| X3 | 37–way sub D pin plug connector |
| X4 | 9–way sub D pin plug connector |
| X5 | 9–way sub D pin plug connector |
| X6 | 15–way sub D socket plug connec. |

A plug connector set is available for plug connectors  X3  to  X6  under  order  number 6ES5 983–2AB11. A stub line for the connection of a Siemens incremental encoder is also available ($\rightarrow$ section 10.10).

## 10.7    Program and Data Memory

- Program Memory                        :                          $64 \times 2^{10}$–byte EPROM

- Data Memory                              :                          $16 \times 2^{10}$–byte SRAM
                                                                                  $1 \times 2^{10}$–byte dual port RAM

- Parameter Memory                      :                          $8 \times 2^{10}$–byte EEPROM

## 10.8    Processing Times for Control Words
## 10.8.1   IP 242A

| Control Word | Meaning | Running Time |
|---|---|---|
| 0100 | Reset module | 350 $\mu$sec |
| 0200 | Disable outputs | 45 $\mu$sec |
| 0300 | Enable outputs | 50 $\mu$sec [1] |
|      |                 | 75 $\mu$sec [2] |
| 0400 | Mask interrupt | 150 $\mu$sec |
| 1102 | Start counter | 50 $\mu$sec |
| 113E |               | 50 $\mu$sec |
| 1202 | Load counter | 80 $\mu$sec |
| 123E |              | 120 $\mu$sec |
| 1302 | Stop counter | 50 $\mu$sec |
| 133E |              | 50 $\mu$sec |
| 1402 | Stop and read counter | 85 $\mu$sec |
| 143E |                       | 100 $\mu$sec |
| 1502 | Step counter | 55 $\mu$sec |
| 153E |              | 55 $\mu$sec |
| 1602 | Save counter | 47 $\mu$sec |
| 163E |              | 47 $\mu$sec |
| 1702 | Copy counter | 75 $\mu$sec |
| 173E |              | 95 $\mu$sec |
| 3102 | Load and start counter | 95 $\mu$sec |
| 3140 |                        | 100 $\mu$sec |
| 31FE |                        | 160 $\mu$sec |
| 3202 | Read counter | 85 $\mu$sec |
| 3240 |              | 70 $\mu$sec |
| 32FE |              | 140 $\mu$sec |
| 3302 | Reset counter | 105 $\mu$sec |
| 3340 |               | 60 $\mu$sec |
| 33FE |               | 190 $\mu$sec |
| 3402 | Take over interrupt value | 90 $\mu$sec |
| 3440 |                           | 100 $\mu$sec |
| 34FE |                           | 130 $\mu$sec |
| 3506 | Generate difference | 90 $\mu$sec |
| 3542 |                     | 100 $\mu$sec |
| 35C0 |                     | 90 $\mu$sec |

[1] After turn–on
[2] After short circuit

| Control Word | Meaning | Running Time |
|---|---|---|
| 7101 | Parameterize counter | 350 μsec |
| 7102 | | 400 μsec [1] |
| 7102 | | 530 μsec [2] |
| 7140 | | 400 μsec [1] |
| 7140 | | 530 μsec [2] |
| 71FF | | 1.75 msec [1] |
| 71FF | | 2.55 msec [2] |
| 7201 | Store parameter | 10 msec |
| 7202 | | 10 msec |
| 7240 | | 20 msec |
| 72FF | | 100 msec |
| 7301 | Rewrite parameter | 410 μsec |
| 7302 | | 540 μsec [1] |
| 7302 | | 670 μsec [2] |
| 7340 | | 610 μsec [1] |
| 7340 | | 740 μsec [2] |
| 73FF | | 2.3 msec [1] |
| 73FF | | 3.0 msec [2] |
| 7401 | Take over basic setting | 400 μsec |
| 7402 | | 480 μsec |
| 7440 | | 510 μsec |
| 74FF | | 2.1 msec |
| 8101 | Execute test function | 88 msec |
| 8102 | | 12 msec |
| 8104 | | 135 msec |
| 8108 | | 1.6 msec |
| 8110 | | 100 msec |
| 811F | | 335 msec |

[1] With empty command lists
[2] With full command lists

## 10.8.2  IP 242B

**General Module Functions**

| Abbrevi- ation | Control Word | Run Time | |
|---|---|---|---|
| | | **On the Module [1]** | **In Command List** |
| RB | 0100 | 405 msec | – |
| SA | 0200 | 245 μsec | 35 μsec |
| FA | 0300 | 250 μsec | 40 μsec |
| IM | 0400 | 365 μsec | 155 μsec |
| KS | 0500 | 390 μsec | – |
| ZA | 0600 | 235 μsec | – |
| SZ | 1502<br>153E | 255 μsec<br>260 μsec | 45 μsec<br>50 μsec |
| SV | 1602<br>163E | 245 μsec<br>245 μsec | 35 μsec<br>35 μsec |
| CO | 1702<br>173E | 270 μsec<br>275 μsec | 65 μsec<br>65 μsec |
| LV | 1802<br>183E | 275 μsec<br>330 μsec | 70 μsec<br>120 μsec |
| LS | 3102<br>313E<br>3140<br>31FE | 340 μsec<br>390 μsec<br>350 μsec<br>435 μsec | 130 μsec<br>185 μsec<br>140 μsec<br>230 μsec |
| LE | 3202<br>323E<br>3240<br>32FE | 375 μsec<br>380 μsec<br>325 μsec<br>455 μsec | 165 μsec<br>170 μsec<br>120 μsec<br>245 μsec |
| RZ | 3302<br>333E<br>3340<br>33FE | 305 μsec<br>380 μsec<br>260 μsec<br>385 μsec | 95 μsec<br>170 μsec<br>55 μsec<br>180 μsec |
| AW | 3402<br>343E<br>3440<br>34FE | 280 μsec<br>315 μsec<br>290 μsec<br>355 μsec | 75 μsec<br>110 μsec<br>80 μsec<br>150 μsec |
| ST | 3602<br>363E<br>3640<br>36FE | 265 μsec<br>265 μsec<br>270 μsec<br>270 μsec | 60 μsec<br>60 μsec<br>60 μsec<br>65 μsec |
| LD | 3702<br>373E<br>3740<br>37FE | 300 μsec<br>350 μsec<br>305 μsec<br>390 μsec | 90 μsec<br>140 μsec<br>95 μsec<br>185 μsec |

**B**

| Abbrevi- ation | Control Word | Run Time | |
|---|---|---|---|
| | | On the Module [1] | In Command List |
| SP | 3802<br>383E<br>3840<br>38FE | 260 μsec<br>260 μsec<br>300 μsec<br>305 μsec | 55 μsec<br>55 μsec<br>95 μsec<br>95 μsec |
| SL | 3902<br>393E<br>3940<br>39FE | 415 μsec<br>415 μsec<br>385 μsec<br>495 μsec | 205 μsec<br>210 μsec<br>175 μsec<br>285 μsec |
| PA | 7101<br>7102<br>713E<br>713F<br>7140<br>71FF | 2.12 msec    [2]<br>0.87 msec    [2]<br>2.27 msec    [2]<br>3.86 msec    [2]<br>0.96 msec    [2]<br>4.70 msec    [2] | – |
| PA | 7101<br>7102<br>713E<br>713F<br>7140<br>71FF | 8.67 msec    [3]<br>1.78 msec    [3]<br>6.79 msec    [3]<br>14.95 msec   [3]<br>1.87 msec    [3]<br>17.60 msec   [3] | – |
| PS | 7201<br>7202<br>723E<br>723F<br>7240<br>72FF | 90 msec<br>40 msec<br>70 msec<br>130 msec<br>50 msec<br>170 msec | – |
| PZ | 7301<br>7302<br>733E<br>733F<br>7340<br>73FF | 0.60 msec<br>0.21 msec<br>0.42 msec<br>0.87 msec<br>0.22 msec<br>1.00 msec | – |
| GR | 7401<br>7402<br>743E<br>743F<br>7440<br>74FF | 0.81 msec<br>0.26 msec<br>0.38 msec<br>0.96 msec<br>0.27 msec<br>1.02 msec | – |
| PO | 7501<br>7502<br>753E<br>753F<br>7540<br>75FF | 0.65 msec<br>0.60 msec<br>1.20 msec<br>1.40 msec<br>0.69 msec<br>1.85 msec | 0.44 msec<br>0.39 msec<br>0.99 msec<br>1.20 msec<br>0.48 msec<br>1.64 msec |
| RL | 7601<br>7602<br>763E<br>763F<br>7640<br>76FF | 0.81 msec<br>0.36 msec<br>0.73 msec<br>1.27 msec<br>0.41 msec<br>1.58 msec | – |

| Abbrevi-ation | Control Word | Run Time | | | |
|---|---|---|---|---|---|
| | | On the Module [1] | | In Command List | |
| RS | 7701 7702 773E 773F 7740 77FF | 0.82 msec 0.37 msec 0.73 msec 1.28 msec 0.46 msec 1.66 msec | | – | |
| TF | 8101 8102 8104 8108 8110 811F | 90 msec 21 msec 182 msec 17 msec 100 msec 404 msec | | – | |
| BB | 8202 | 340 $\mu$sec | [4] | 130 $\mu$sec | [4] |
| ML | 8302 8302 83FE | 0.45 msec 1.07 msec 1.38 msec | [5] [6] [7] | – | |
| MR | 8402 8402 84FE | 0.54 msec 1.41 msec 1.90 msec | [5] [6] [7] | – | |

1)  Minimum run times are given in the table. In extreme cases, run times can be lengthened by 95 $\mu$sec.

2)  Without command lists

3)  With "full" interrupt and additional command lists, and directory of the measured value memory

4)  Basic run time of the control word; the run time of the control words contained in the additional command list must be added to this.

5)  For 1 block with 10 measured values (10 * 2 DWs)

6)  For 1 block with 100 measured values( 100 * 2 DWs)

7)  For 7 blocks with a total of 100 measured values (100 * 2 DWs)

**B**

### Calculation Functions

| Abbrevi-ation | Control Word | Run Time | |
|---|---|---|---|
| | | **On the Module [8]** | **In Command List** |
| L | 9001 | 500 μsec | 65 μsec |
| | 9006 | 500 μsec | 65 μsec |
| | 9011 | 500 μsec | 65 μsec |
| | 9016 | 500 μsec | 65 μsec |
| | 9021 | 500 μsec | 65 μsec |
| | 9026 | 500 μsec | 65 μsec |
| | 9031 | 500 μsec | 65 μsec |
| | 9036 | 500 μsec | 65 μsec |
| | 9041 | 500 μsec | 65 μsec |
| | 9046 | 500 μsec | 65 μsec |
| | 9051 | 500 μsec | 65 μsec |
| | 9056 | 500 μsec | 65 μsec |
| | 9061 | 505 μsec | 70 μsec |
| | 9066 | 600 μsec | 165 μsec |
| | 9071 | 500 μsec | 65 μsec |
| | 9081 | 500 μsec | 65 μsec |
| | 9086 | 500 μsec | 65 μsec |
| T | 9101 | 540 μsec | 105 μsec |
| | 9106 | 565 μsec | 130 μsec |
| | 9111 | 495 μsec | 60 μsec |
| | 9116 | 490 μsec | 55 μsec |
| | 9121 | 505 μsec | 70 μsec |
| | 9126 | 490 μsec | 55 μsec |
| | 9131 | 550 μsec | 115 μsec |
| | 9136 | 550 μsec | 115 μsec |
| | 9141 | 540 μsec | 105 μsec |
| | 9146 | 525 μsec | 90 μsec |
| | 9161 | 565 μsec | 130 μsec |
| | 9166 | 560 μsec | 125 μsec |
| | 9171 | 480 μsec | 45 μsec |
| | 9181 | 485 μsec | 50 μsec |
| | 9186 | 485 μsec | 50 μsec |
| TAK | 9200 | 480 μsec | 45 μsec |
| ADD | 9300 | 495 μsec [9] | 60 μsec [9] |
| SUB | 9400 | 500 μsec [9] | 65 μsec [9] |
| MUL | 9500 | 545 μsec [9] | 110 μsec [9] |
| DIV | 9600 | 575 μsec [9] | 140 μsec [9] |
| KZD | 9700 | 480 μsec [9] | 45 μsec [9] |
| ABS | 9800 | 480 μsec [9] | 45 μsec [9] |
| SPR | A000 | 480 μsec [9] | 45 μsec [9] |
| >D | A100 | 475 μsec [9] | 40 μsec [9] |
| =D | A200 | 475 μsec [9] | 40 μsec [9] |
| <D | A300 | 475 μsec [9] | 40 μsec [9] |
| ≥D | A400 | 475 μsec [9] | 40 μsec [9] |
| ≤D | A500 | 475 μsec [9] | 40 μsec [9] |
| ><D | A600 | 475 μsec [9] | 40 μsec [9] |

| Abbrevi-ation | Control Word | Run Time | |
|---|---|---|---|
| | | **On the Module** [8] | **In Command List** |
| FIN | A700 | 485 µsec            [9] | 50 µsec            [9] |
| FAUS | A800 | 485 µsec            [9] | 50 µsec            [9] |

8) Maximum run times are given in the table. They apply to the "BB" control word with the respective control word (9001 to A800) in the additional command list.

9) When the implicit load command is used, the processing time is increased by the run time of the load command for the corresponding register.

## 10.9    Basic Plug Connector Allocation

**A**  **B**  The basic plug connector (i.e., 48–way multipoint terminal strip in accordance with DIN 41612) is located on the back of the module and has the following allocations:

| Pin | d | b | z |
|-----|-----|-----|-----|
| | Signal | Signal | Signal |
| 2 | | 0 V | P + 5 V |
| 4 | | | |
| 6 | ADB 12 | ADB 0 | $\overline{RESET}$ |
| 8 | ADB 13 | ADB 1 | $\overline{MEMR}$ |
| 10 | ADB 14 | ADB 2 | $\overline{MEMW}$ |
| 12 | ADB 15 | ADB 3 | $\overline{RDY}$ |
| 14 | $\overline{IRA}$ | ADB 4 | DB 0 |
| 16 | $\overline{IRB}$ | ADB 5 | DB 1 |
| 18 | $\overline{IRC}$ | ADB 6 | DB 2 |
| 20 | $\overline{IRD}$ | ADB 7 | DB 3 |
| 22 | | ADB 8 | DB 4 |
| 24 | | ADB 9 | DB 5 |
| 26 | | ADB 10 | DB 6 |
| 28 | | ADB 11 | DB 7 |
| 30 | | BASP | |
| 32 | | 0 V | |

Basic plug connector X1 (system interface)

## 10.10   Stub Line for Siemens Incremental Encoder

Name: Stub line 705 for connection of Siemens position encoders 6FC9 320

Order no.: 6ES5 705–2xxx0          (see catalog ST 52.3/54.1)

                 xxx  =  Length code      5 m BF0
                                      10 m CB0
                                      20 m CC0

**A**   **B**



IP 242A
IP 242B

Encoder

Shielding to housing

Shielding to housing

Line 4 x 2 x 0.18 + 4 x 0.5 mm$^2$

| IP 242A/242B pin | color | Encoder pin |
|---|---|---|
| 1 | bk | 5 |
| 6 | or | 6 |
| 2 | br | 8 |
| 7 | ye | 1 |
| 3 | rd | 3 |
| 8 | gn | 4 |
| 9  wt | rd | 12 |
| 4  br | bk | 10 |

1) Twisted pairs

SYN  $\overline{\text{SYN}}$  5V  0V

Open lines 2 x 0.5 mm$^2$, 0.5 m long (marked)

Sub D, plug connector
9–way pin (crimp type)
Connection side
Metallized housing with
screw lock

Round plug connector
12–way socket
Connection side
SIEMENS

1) Twisted pairs

## 10.11   24 V Asymmetric To 5 V (RS422) Symmetric Converter

**A**   **B**   The 24 V asymmetric to 5 V (RS422) symmetric converter with integrated level conditioning is used with the IP 242A/242B module for the following purposes.

– The external circuiting of the digital outputs of counters 1 to 5 with the inputs of counters 6 and 7

– The circuiting of counters 6 and 7 with external, 24 V signals

The converter is equipped with two channels, and prepared for mounting on a top hat rail.

**Order number:**   6ES5 242–1AU11

**Technical Specifications:**

| Dimensions | |
|---|---|
| L x W x H | 180 x 112 x 90 mm |
| **Input Signal (A\*, B\*, N\*)** | |
| Input nominal voltage | 24 V |
| Input voltage for signal "0" | –3 V to 4.5 V |
| Input voltage for signal "1" | 13 V to 30 V |
| Signal state for uncircuited inputs | Low |
| Input resistance | 1.8 K$\Omega$, average |
| Input current for signal "1" (nominal voltage: 24 V) | 12 mA, average |
| Input current (13 V to 30 V) | 2.5 mA to 15 mA |
| Transmission frequency | 2 MHz (max.) |
| **Input Signals (SYN)** | |
| Data $\rightarrow$ section 10.2 | |
| **Voltage Supply (Via X1)** | |
| Nominal value | 24 V |
| Limits, static (ripple included) <br> – Lower limit <br> – Upper limit | <br> 20 V <br> 30 V |
| Limits, dynamic <br> – Lower limit <br>    Value <br>    Duration <br>    Recovery time <br> – Upper limit <br>    Value <br>    Duration <br>    Recovery time | <br><br> 14.25 V <br> 5 msec <br> 10 sec <br><br> 35 V <br> 500 msec <br> 50 sec |
| Ripple | $\leq$ 3.6 V$_{ss}$ |
| Maximum current consumption (depends on the circuiting of the outputs) | 200 mA |
| Overvoltage protection | Present |
| Voltage monitoring | Not present |

**Setup**

Channel 1          Channel 2

X3               X5



Outputs

Inputs

24 V GND

X6        X1            X2              X4
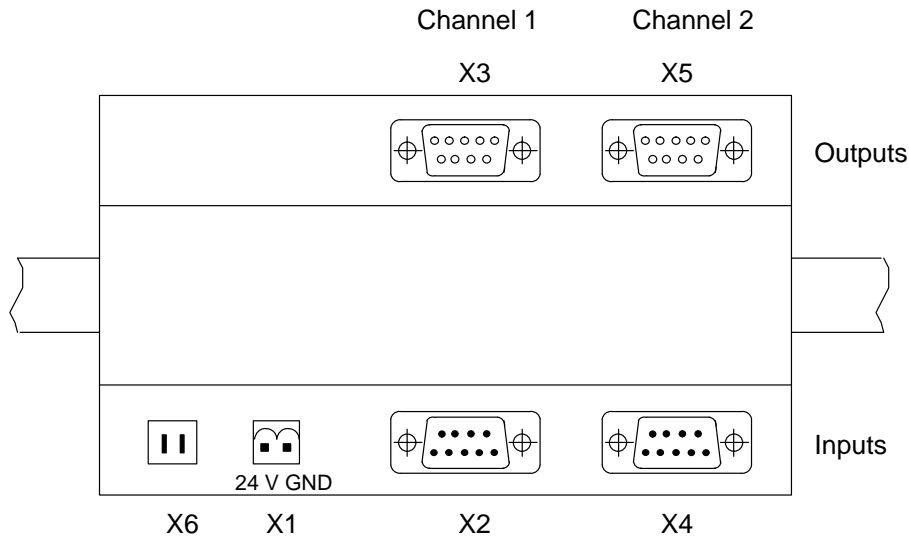
**Plug Connector Allocation**

X1

1

2–way, pin, plug connector

| Pin | Signal |
|-----|--------|
| 1 | +24 V |
| 2 | GND |

X2, X4

9    6

5    1
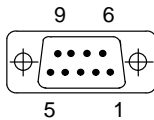
9–way, sub D, pin plug connector

| Pin | Input Signal |
|-----|--------------|
| 1 | A* |
| 2 | B* |
| 3 | N* |
| 4 | SYN |
| 5 | – |
| 6 | $A^*_{GND}$ |
| 7 | $B^*_{GND}$ |
| 8 | $N^*_{GND}$ |
| 9 | $SYN_C$ |

X6          Reference potential
            (Faston plug connector)

X3, X5

5    1

9    6
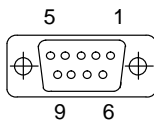
9–way, sub D, socket plug connector

| Pin | Output Signal |
|-----|---------------|
| 1 | A |
| 2 | B |
| 3 | N |
| 4 | SYN |
| 5 | – |
| 6 | $\overline{A}$ |
| 7 | $\overline{B}$ |
| 8 | $\overline{N}$ |
| 9 | $SYN_C$ |

## 10.12   In Which Slots Can the Counter Module Be Operated?

| Programmable Controller | | Slot Designation<br>▢ ≙ The IP 242A/242B can be used in this slot. |
|---|---|---|
| Central controller S5–115U with module rack: | CR 700–0LA | PS \| CPU \| **0** \| 1 \| 2 \| 3 \| IM (slot 0 usable) |
| | CR 700–0LB | PS \| CPU \| **0** \| **1** \| **2** \| **3** \| IM (slots 0–2 usable) |
| | CR 700–1 | PS \| CPU \| **0** \| 1 \| 2 \| 3 \| 4 \| 5 \| 6 \| IM (slot 0 usable) |
| | CR 700–2 | PS \| CPU \| **0** \| **1** \| **2** \| **3** \| **4** \| **5** \| 6 \| IM (slots 0–5 usable) |
| | CR 700–3 | PS \| CPU \| **0** \| **1** \| **2** \| **3** \| **4** \| **5** \| 6 \| IM (slots 0–5 usable) |
| Expansion unit | ER 701–3 [1] | PS \| **0** \| **1** \| **2** \| **3** \| **4** \| **5** \| **6** \| 7 \| IM (slots 0–6 usable) |

Central controller S5–135U

| 3 | 11 | 19 | 27 | 35 | 43 | 51 | 59 | 67 | 75 | 83 | 91 | 99 | 107 | 115 | 123 | 131 | 139 | 147 | 155 | 163 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  |  |  | 2 |  | 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Central controller S5–155U

| 3 | 11 | 19 | 27 | 35 | 43 | 51 | 59 | 67 | 75 | 83 | 91 | 99 | 107 | 115 | 123 | 131 | 139 | 147 | 155 | 163 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  |  | 3 | 3 |  |  | 2 | 2 | 2 |  |  | 2 | 2 |  |  |  |  | 2 | 2 |  |  |

Expansion unit 185U

| 3 | 11 | 19 | 27 | 35 | 43 | 51 | 59 | 67 | 75 | 83 | 91 | 99 | 107 | 115 | 123 | 131 | 139 | 147 | 155 | 163 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |  |  |  |

Expansion unit 186U [1]

| 3 | 19 | 35 | 51 | 67 | 83 | 99 | 115 | 131 | 147 | 163 |
|---|---|---|---|---|---|---|---|---|---|---|
|  | 2 |  |  |  |  |  |  |  |  |  |

➡ Counter module IP 242A/242B cannot be inserted in central controller S5–150U/S and in expansion units ER 701–1, ER 701–2, 183U, 184U and 187U. ⬅

1) Interrupts can be processed in expansion devices starting at release 6ES 701–3LA13 when optical fiber links 6ES5 307–3UA11 and 6ES5 317–3UA11 are used.

2) No interrupt lines available. Functionality very restricted

3) Only after changing jumpers on the bus PCB

4) Interrupts only connected via IM 307 and IM 317

# 11 Programming Instructions, FB 178/179
**IP 242A**

## 11.1     Overview

These programming instructions describe the following function blocks:

**FB 178 (PER:ZSTK)**     "Control Counter Module IP 242A"
                          (for page frame addressing)

**FB 179 (PER:ZSTL)**     "Control Counter Module IP 242A"
                          (for linear addressing)

These function blocks are used with the following module:

**Counter Module IP 242A**

in the following programmable controllers:

| FB 178 | FB 179 | PLC/CPU |
|--------|--------|---------|
| X | X | **S5–115U**  (CPU 941A/B to CPU 944A/B) |
| X |   | **S5–135U**  (CPU 922, CPU 928A/B) |
| X |   | **S5–150U/S** |
| X | X | **S5–155U** |

These programming instructions require you to be familiar with sections 1 to 7, 9 and 10 of this manual and the operating instructions of your programmable controller.

The example in section 12 shows a test setup with the IP 242A counter module which provides you with an easy way to check jumper allocations and the functions. You can also use this test program as the basis of a real automation task.

The files of the function blocks with example, and the English and French commentary blocks for the respective programmable controllers are located on the S5–DOS floppy disk.

| PLC S5– | File | | |
|---------|------|------|------|
|         | Function Block | Commentary Block | |
|         | German | English | French |
|         | S5LxxxST.S5D | ECLxxxST.S5D | FCLxxxST.S5D |
| 115U | S5LC50ST.S5D | ECLC50ST.S5D | FCLC50ST.S5D |
| 135U | S5LC22ST.S5D | ECLC22ST.S5D | FCLC22ST.S5D |
| 150U/S | S5LC40ST.S5D | ECLC40ST.S5D | FCLC40ST.S5D |
| 155U[1] | S5LC60ST.S5D | ECLC60ST.S5D | FCLC60ST.S5D |

When the commentary blocks are copied to file S5LxxxST.S5D, you will receive the commentaries in the applicable language when the example is printed out.

The corresponding title block files are:
S5LxxxF1.INI
ECLxxxF1.INI
FCLxxxF1.INI

1)   Use the xxLC22ST.S5D files when using a CPU 922 or 928A/B in PLC S5–155U.

## 11.2    Function Description

The following standard function blocks are available for the control of the IP 242A counter module:

FB 178 (PER:ZSTK) for page frame addressing and
FB 179 (PER:ZSTL) for linear addressing.

Both function blocks have the same function and are supplied with the same parameters (with the exception of the module address).
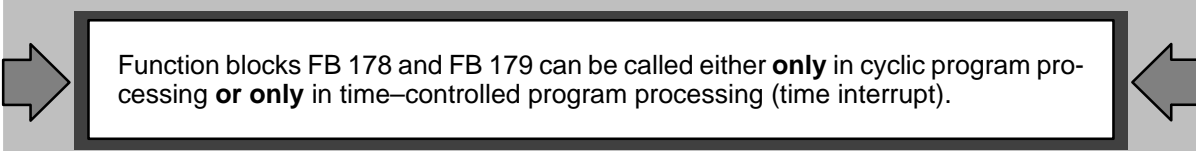
The "control counter module" function blocks can be used to execute the following functions:

– Parameterizing of counters

– Loading, starting, and reading of counters

– Processing of interrupts

Function blocks FB 178 and FB 179 are usually called in the start–up program (organization blocks OB 20, OB 21, and OB 22) to parameterize the counter module.

Control of the counter module (e.g., starting counters or reading actual values) then takes place in the cyclic program (organization block OB 1).

Function blocks FB 178 and FB 179 are called in an organization block of interrupt–controlled processing (OB 2 to OB 9, depending on the programmable controller) to evaluate the process interrupts and/or interrupts. Function blocks FB 180, FB 181 and FB 182 can be used for faster processing of the BS, BL and IN commands during interrupt processing ($\rightarrow$ section 13).

> Function blocks FB 178 and FB 179 can be called either **only** in cyclic program processing **or only** in time–controlled program processing (time interrupt).

**Function**

Controlling the IP 242A counter module

## 11.3   Calling Function Blocks FB 178 and FB 179

```
        :JU FB 178              Parameter
NAME    :PER:ZSTK              Class Type

SSNR    :                       D     KY
BEF     :                       D     KS
PAR     :                       D     KM
STEU    :                       I     W
DBNR    :                       D     KY
ABIT    :                       D     KY        (*)
IIR     :                       Q     W
ERR     :                       Q     BY
MELD    :                       Q     BY
F–NR    :                       Q     BY
```

| Parameter | | | **FB 178** | | | Parameter | |
| Class | Type | | PER:ZSTK | | | Class | Type |
|---|---|---|---|---|---|---|---|
| D | KY | | SSNR | IIR | | Q | W |
| D | KS | | BEF | ERR | | Q | BY |
| D | KM | | PAR | MELD | | Q | BY |
| I | W | | STEU | F–NR | | Q | BY |
| D | KY | | DBNR | | | | |
| D | KY | | ABIT (*) | | | | |
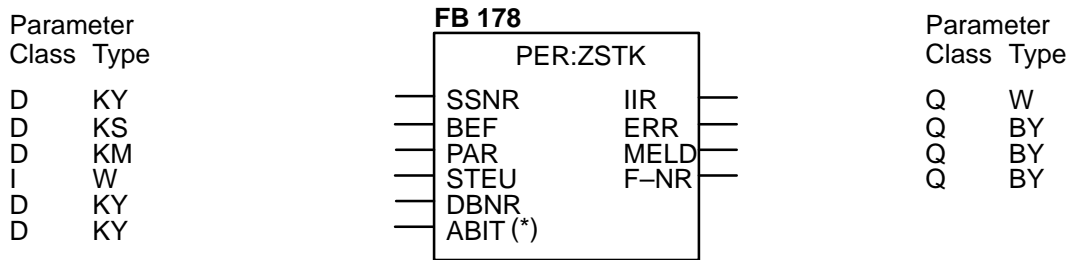
(*)   The ABIT parameter is only available for programmable controllers S5–150U/S and S5–155U.

```
        :JU FB 179              Parameter
NAME    :PER:ZSTL              Class Type

LADR    :                       D     KH
BEF     :                       D     KS
PAR     :                       D     KM
STEU    :                       I     W
DBNR    :                       D     KY
ABIT    :                       D     KY        (*)
IIR     :                       Q     W
ERR     :                       Q     BY
MELD    :                       Q     BY
F–NR    :                       Q     BY
```

| Parameter | | | **FB 179** | | | Parameter | |
| Class | Type | | PER:ZSTL | | | Class | Type |
|---|---|---|---|---|---|---|---|
| D | KH | | LADR | IIR | | Q | W |
| D | KS | | BEF | ERR | | Q | BY |
| D | KM | | PAR | MELD | | Q | BY |
| I | W | | STEU | F–NR | | Q | BY |
| D | KY | | DBNR | | | | |
| D | KY | | ABIT (*) | | | | |
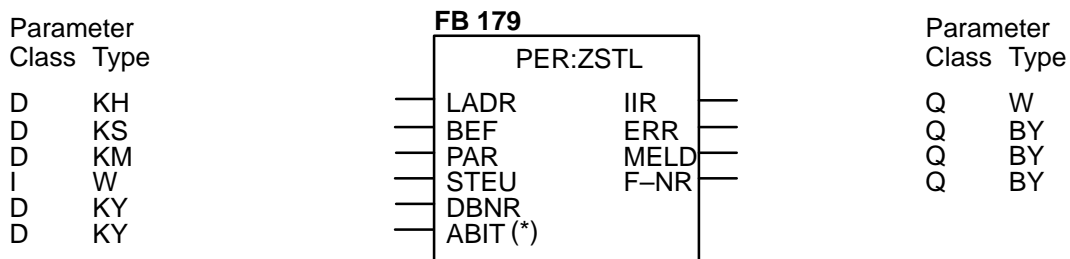
(*)   The ABIT parameter is only available for programmable controller S5–155U.

> Function block FB 179 can only be used with programmable controllers S5–115U and S5–155U.

## 11.4    Explanation of the Parameters

| NAME | CLASS | TYPE | DESIGNATION |
|------|-------|------|-------------|
| SSNR | D | KY | Interface number for page frame addressing (FB 178 only) |
| LADR | D | KH | Address for linear addressing (FB 179 only) |
| BEF | D | KS | Command; specification of which function the function block is to perform (direct parameterization) |
| PAR | D | KM | Parameter (e.g., specification of the counters which are to be controlled simultaneously) (direct parameterization) |
| STEU | I | W | Control word; specification of which function the function block is to perform (indirect parameterization) |
| DBNR | D | KY | Number of the parameterization data block |
| ABIT (*) | D | KY | Number of the interrupt bit for process interrupt processing |
| IIR | Q | W | Interrupt information register |
| ERR | Q | BY | Error information register |
| MELD | Q | BY | Message byte |
| F–NR | Q | BY | Error number |

(*)   The ABIT parameter is only available for programmable controllers S5–150U/S and S5–155U.

When a data word or a data byte is specified as the actual parameter for parameters STEU, IIR, ERR, MELD, and F–NR, this data word or data byte is located in the parameterization data block (parameter DBNR). In this case, allocation is first permitted starting at DW 120. Parameterization data for the counter module is found in data word locations up to and including data word DW 119.

| ! | Scratchpad flags FY 200 to FY 255 cannot be assigned to parameters STEU, IIR, ERR, MELD, and F–NR. | ! |
|---|---|---|

## 11.5    Assignment of the Parameters

**SSNR   :    Interface number for page frame addressing**
(FB 178 PER:ZSTK only)

KY = x, 0 to 255          Page frame address which you set at switch S4.
(x = not applicable)

**LADR   :    Module basic address for linear addressing**
(FB 179 PER:ZSTL only)

Linear addressing is only possible with programmable controllers S5–115U and S5–155U.
Only certain areas are permitted for the individual programmable controllers as follows:

PLC S5–115U        KH    =    0000
KH    =    0400
KH    =    0800
KH    =    0C00
PLC S5–155U        KH    =    0000 to EC00in pattern of 1k addresses

For S5–155U programmable controllers, the linear address is always located in I/O memory page F. Only
bits 0 to 15 then have to be specified at parameter LADR.

Switch S3 on the module must have been previously appropriately set.

**BEF     :    Command for controlling the module (direct parameterization)**

The commands are specified in KS format.

A list of possible commands is found in the following table.

To switch to indirect parameterization via parameter STEU, assign KS = xx to parameter
BEF.

See section 11.8 (Application of the Function Block) for the meaning of the individual commands.

## Table of Valid Commands

| Meaning | Com-mand BEF KS = | Direct Parameterization Parameter PAR                KM = | Indirect Parameter-ization Control word STEU KH = |
|---|---|---|---|
| Reset module | RB* | – | 0100 |
| Disable the outputs | SA | – | 0200 |
| Enable the outputs | FA | – | 0300 |
| Mask interrupt | IM | – | 0400 |
| Start counter | ST | Counters 1 to 5 | 11xx |
| Load counter | LD | Counters 1 to 5 | 12xx |
| Stop counter | SP | Counters 1 to 5 | 13xx |
| Stop and read counter | SL | Counters 1 to 5 | 14xx |
| Step counter | SZ | Counters 1 to 5 | 15xx |
| Save counter | SV | Counters 1 to 5 | 16xx |
| Copy counter | CO | Counters 1 to 5 | 17xx |
| Load and start counter | LS | Counters 1 to 7 | 31xx |
| Read counter | LE | Counters 1 to 7 | 32xx |
| Reset counter | RZ | Counters 1 to 7 | 33xx |
| Take over interrupt value | AW | Counters 1 to 7 | 34xx |
| Generate difference | DF | Counters 1 to 7 | 35xx |
| Parameterize counter | PA* | Counters 1 to 7, global register | 71xx |
| Store parameter | PS* | Counters 1 to 7, global register | 72xx |
| Rewrite parameter | PZ* | Counters 1 to 7, global register | 73xx |
| Take over basic setting | GR* | Counters 1 to 7, global register | 74xx |
| Execute test function | TF* | Test functions | 81xx |
| Read module | BL | Counters 1 to 7, global register | F1xx |
| Write module | BS | Counters 1 to 7, global register | F2xx |
| Process interrupt | IN | – | – |
| Switch to STEU | XX | – | – |

See bits 0 to 7
of parameter PAR

*)   The module may sometimes require more than 1 msec to process these commands. When this happens, bit 2 (module busy) of parameter MELD is set and processing of the block is ended.

**PAR   :   Parameter**

Parameter for the command (right byte of the control word);
(e.g., specification of the counter or test function to be controlled)

Allocation for the counter:

KM = 0000 0000 xxxx xxxx
                          └── Global register
                        └──── Counter 1
              └────────────── Counter 7
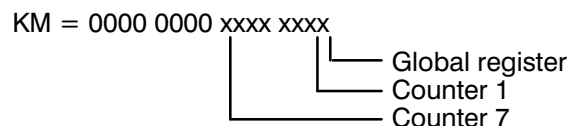
The counter to be controlled, the global registers and/or test function are selected with x = "1". Be sure that at least one counter or the global registers and at least one test function are specified or a parameterization error occurs. The number of counters to be specified varies. See above table.

**STEU : Control word**

> Parameter STEU is evaluated only if switch–over parameter KS = xx is specified at parameter BEF.
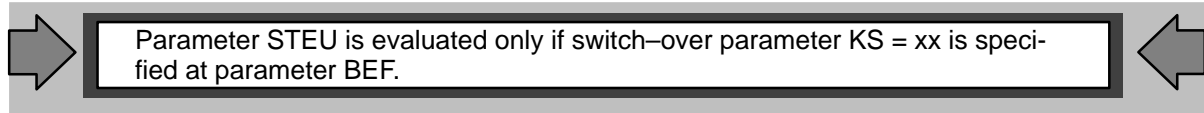
The control word must be completely specified. For example, the counter and/or the test functions to be controlled must also be included. If the specified command (left byte of the control word) is a valid command (see table of valid commands in this section), parameter allocation is checked for correctness.

If an appropriate command code is not found, the control word is transferred to the module without checking for valid parameter assignment.

> **!** Command codes KH = 00xx and KH = FFxx are illegal. **!**
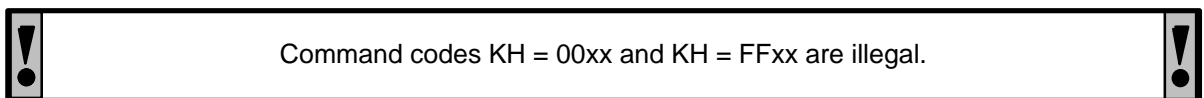
**DBNR : Number of the data block that contains the parameterization data**

KY = **x, y**   **x** = 0; **y** = 0   : The data block that is currently opened is used as the parameterization data block.

**x** = 0   : Data block DB
**x** = 1   : Data block DX (only PLC S5–135U with CPU 922/CPU 928A/B and PLC S5–155U)

**y** = 10 to 255   : Number of the data block

In all other cases, an error number is output via parameter F–NR.

### ABIT    :    Number of the interrupt bit

Applicable only for interrupt processing (BEF = IN) with PLC S5–150U/S and PLC S5–155U in S5–150U mode.

KY = **x, y**    **x**  =  Enable reset
                **x** >< 0      No reset
                **x** = 0       Reset appropriate interrupt bit in the system data

           **y**  =  Number of the interrupt bit
                $0 \leq \mathbf{y} \leq 7$

### IIR       :    Interrupt information register

During interrupt processing (BEF = IN), the contents of the interrupt information register are output at parameter IIR ($\rightarrow$ section 3.8.2).

### ERR     :    Error information register

The contents fo the counter module's error information register are assigned to parameter ERR ($\rightarrow$ section 3.8.3).

If parameter ERR is zero, no errors have occurred.

**Bit 3** of parameter **MELD** is set to signal status **"1"** when an error (parameters ERR >< zero) is detected.

The assignment of parameter ERR occurs at two different times (i.e., before writing a control word and after writing a control word).

Assignment of parameter **after** the writing of a control word is probably due to an error which occurred in connection with writing the control word. Processing of the control word in which the error occurred is terminated. The signal status of **bit 4** in parameter **MELD** is then **"0"**.

If parameter ERR is assigned **before** the control word is written, a module malfunction (e.g., short circuit at the outputs) probably caused an error number to be set. The signal status of **bit 4** in parameter **MELD** is then **"1"** and the control word is not transferred.

**MELD   :   Message byte**

The function block files the following messages in this byte:

Bit 0    No access to dual port RAM

Bit 1    Watchdog monitoring is triggered.

Bit 2    Module is busy.

Bit 3    Error information register is allocated (group bit).

Bit 4    Error information register is allocated before a control word is written.

Bit 5    Dual port RAM is busy with another processor (for multiprocessor operation only).

Bit 6    Module is busy before a write access of a control word.

Bit 7    Group error bit. Signal status is "1" when an error or a message occurs. Parameter ERR, MELD, or F–NR is not zero.


**F–NR   :   Error number**

Parameter F–NR is allocated with an error number when parameterization is illegal or when incorrect parameterization causes an error.

| KH = | Error |
|------|-------|
| 01 | Parameter BEF is incorrectly allocated. |
| 02 | Parameter PAR is incorrectly allocated. |
| 03 | Parameter STEU is incorrectly allocated. |
| 04 | Parameter DBNR is incorrectly allocated. |
| 05 | Data block does not exist. |
| 06 | Data block is too short. |
| 07 | Parameter ABIT is incorrectly allocated. |
| 08 | Module identification is wrong (bit pattern on the page frame). |
| 09 | Acknowledgement delay |
| 0A | Parameter LADR is incorrectly allocated (FB 179 only). |
| 0B | CPU identification or firmware status is wrong (for PLC S5–135U). |

KH = 00 at parameter F–NR indicates that no errors occurred during processing of the function block.

## 11.6     Assignment of the Data Area

The data block specified at parameter DBNR is assigned with counter data as described below.

Function blocks FB 178 and FB 179 transmit data between the data block and the counter module, depending on the parameters BEF and PAR or STEU.

For this reason the data block must be present in its full length. This is checked even when not all counters are used.

The data area can be located in a DB data block or in a DX data block. A DX data block can only be used with programmable controllers S5–135U and S5–155U.

The data block which is current at the moment can also be used as the parameterization data block. (See assignment of parameter DBNR.)

> **!** Under no circumstances may the data words or data bytes specified as actual parameters
> be located in the module's parameterization area from DW 0 up to and including DW 119 (is
> not checked). **!**

Overview of the assignment of the parameterization data block:

| Starting at data word | Assignment |
|---|---|
| 0 | Global register |
| 16 | Parameterization data, counter 1 |
| 30 | Parameterization data, counter 2 |
| 44 | Parameterization data, counter 3 |
| 58 | Parameterization data, counter 4 |
| 72 | Parameterization data, counter 5 |
| 86 | Parameterization data, counter 6 |
| 103 | Parameterization data, counter 7 |
| 120 | = for the user = |

**Allocation of the Global Registers:**

Bit No.                    15 ...                                                      ... 0

| | | | | |
|---|---|---|---|---|
| DW | 0 | – | | |
| DW | 1 | – | | |
| DW | 2 | – | | |
| DW | 3 | – | | |
| DW | 4 | Master mode register | MMR | KM |
| DW | 5 | Prescaler register | VTR | KM |
| DW | 6 | Gate control register | TSR | KM |
| DW | 7 | Interrupt enable register | IFR | KM |
| DW | 8 | Interrupt polarity register | IPR | KM |
| DW | 9 | | DR | KH |
| DW | 10 | Difference register | DR | KH |
| DW | 11 | Version number register | VNR | KH |
| DW | 12 | == Reserved == | | |
| DW | 13 | == Reserved == | | |
| DW | 14 | == Reserved == | | |
| DW | 15 | == Reserved == | | |

## Parameterization Data of a Counter *(1 to 5)*:

Bit No:                        15 ...                                                                  ... 0

| | | | |
|---|---|---|---|
| DW | n | Counter mode register                                    CMR | KM |
| DW | n + 1 | Load register                                            LR | KH/KF |
| DW | n + 2 | Hold register                                            HR | KH/KF |
| DW | n + 3 | Interrupt register [1)]                                  AR | KH/KF |
| DW | n + 4 | Command 1 for interrupt from counter gate | KH/KM |
| DW | n + 5 | Command 2 for interrupt from counter gate | KH/KM |
| DW | n + 6 | Command 3 for interrupt from counter gate | KH/KM |
| DW | n + 7 | Command 4 for interrupt from counter gate | KH/KM |
| DW | n + 8 | Command 5 for interrupt from counter gate | KH/KM |
| DW | n + 9 | Command 1 for interrupt from counter output | KH/KM |
| DW | n + 10 | Command 2 for interrupt from counter output | KH/KM |
| DW | n + 11 | Command 3 for interrupt from counter output | KH/KM |
| DW | n + 12 | Command 4 for interrupt from counter output | KH/KM |
| DW | n + 13 | Command 5 for interrupt from counter output | KH/KM |

1  "0" cannot be used as the interrupt value ($\rightarrow$ section 1.9).

Counter 1:  n = 16
Counter 2:  n = 30
Counter 3:  n = 44
Counter 4:  n = 58
Counter 5:  n = 72

**Parameterization Data of a Counter** *(6 and 7)*:

| Bit No. | | 15 ... | ... 0 | |
|---|---|---|---|---|
| DW | n | Counter mode register | CMR | KM |
| DW | n + 1 | (Sign) | (Bits 16 to 23) | KH |
| DW | n + 2 | Load register (bits 8 to 15) | (Bits 0 to 7) LR | KH |
| DW | n + 3 | (Sign) | (Bits 16 to 23) | KH |
| DW | n + 4 | Hold register (bits 8 to 15) | (Bits 0 to 7) HR | KH |
| DW | n + 5 | (Sign) | (Bits 16 to 23) | KH |
| DW | n + 6 | Interrupt register (bits 8 to 15) | (Bits 0 to 7) AR | KH |
| DW | n + 7 | Command 1 for interrupt from inputs SYN & zero marking pulse (simultaneous) | | KH/KM |
| DW | n + 8 | Command 2 for interrupt from inputs SYN & zero marking pulse (simultaneous) | | KH/KM |
| DW | n + 9 | Command 3 for interrupt from inputs SYN & zero marking pulse (simultaneous) | Ex: For counter reset (→ section 7.6) | KH/KM |
| DW | n + 10 | Command 4 for interrupt from inputs SYN & zero marking pulse (simultaneous) | | KH/KM |
| DW | n + 11 | Command 5 for interrupt from inputs SYN & zero marking pulse (simultaneous) | | KH/KM |
| DW | n + 12 | Command 1 for interrupt from counter output | | KH/KM |
| DW | n + 13 | Command 2 for interrupt from counter output | | KH/KM |
| DW | n + 14 | Command 3 for interrupt from counter output | | KH/KM |
| DW | n + 15 | Command 4 for interrupt from counter output | | KH/KM |
| DW | n + 16 | Command 5 for interrupt from counter output | | KH/KM |

Counter 6: n = 86
Counter 7: n = 103

### 11.7    Technical Specification

**Function Block FB 178 (PER:ZSTK) for Page Frame Addressing and Function Block FB 179 (PER:ZSTL) for Linear Addressing**

| Block number | FB 178 | | | | FB 179 | |
|---|---|---|---|---|---|---|
| Block name | PER:ZSTK | | | | PER:ZSTL | |
| PLC S5– | 115U | 135U | 150U/S | 155U | 115U | 155U |
| Library number (P71200–S ...) | –5178–A–3 | –9178–A–3 | –4178–A–3 | –6178–B–3 | –5179–A–3 | –6179–B–3 |
| Call length (words) | 11 | | 12 | | 11 | 12 |
| Block length (words) | 1356 | 1281 | 1145 | 1192 | 1402 | 1214 |
| Nesting depth | 0 | | | $1^{1)}$ | 0 | $1^{1)}$ |
| Data blocks assigned | Parameterization data block up to and including data word DW 119 (specified by parameters DBNR) | | | | | |
| Flags assigned [2] | FY 212 to FY 255 | FY 230 to FY 255 | FY 204 to FY 255 | FY 216 to FY 255 | FY 208 to FY 255 | FY 212 to FY 255 |
| System blocks assigned [3] | – | BS 60 to BS 63 | – | | – | |
| Other comments | [4] | | – | [4] | [4] | |

1    Special functions of the operating system are called which are considered "normal" block calls.

2    The flags are used as intermediate storage only. Outside of the function block they are available for use as desired.

3    The system data are used as intermediate storage only. Outside of the function block they are available for use as desired. When the function block is called during interrupt processing, these system data words must be saved and reloaded like the scratchpad flags.

> ⚠ 4    Interrupts (process interrupts, time interrupts) are disabled for approximately 1 msec in the function block. After this period of time **all** interrupts are enabled. ⚠

**Processing Times**

- PLC S5–115U (CPU 941A/B to CPU 944A/B)
- PLC S5–135U (CPU 922 from firmware release 9.0/CPU 928A from ...–3UA12/CPU 928B)
- PLC S5–150U/S
- PLC S5–155U (CPU 946/947)

**Processing Times**

The table gives the processing times in msec for FB 178 (PER:ZSTK). The processing times for FB 179 (PER:ZSTL) are approximately the same.

| PLC S5– | 115 U | | | | 135 U | | 150 U/S | 155 U |
|---|---|---|---|---|---|---|---|---|
| AG S5– CPU | 941A 941B | 942A 942B | 943A 943B | 944A 944B | 922 | 928A 928B | | |
| ST, SP, SA, FA, RB, RZ, SZ, TF, SV | 76.1 11.4 | 20.9 11.4 | 18.5 10.9 | 5.7 4.1 | 12.9 | 7.2 4.2 | 2.9 | 3.0 |
| LD, LS, SL, LE ① GR, AW, PZ, CO | 121.8 26.7 (75.5) (14.2) | 43.4 26.7 (26.3) (14.2) | 32.1 25.9 (18.6) (13.6) | 6.6 5.8 (4.5) (4.7) | 31.1 (17.8) | 15.9 11.8 (8.7) (5.2) | 3.9 (2.5) | 3.7 (2.4) |
| PA, BL, BS, PS ① | 284.0 60.2 (89.9) (19.1) | 108.4 60.2 (26.9) (19.1) | 70.0 59.6 (20.2) (18.8) | 12.0 10.7 (5.4) (7.0) | 47.6 (18.3) | 26.1 19.9 (9.4) (6.5) | 7.7 (2.7) | 6.4 (2.7) |
| IM, DF | 72.6 14.0 | 23.5 14.0 | 18.5 13.5 | 4.4 5.2 | 15.5 | 7.5 5.4 | 2.5 | 2.2 |
| IN | 18.8 4.8 | 6.1 4.8 | 4.8 4.7 | 1.1 0.8 | 3.6 | 2.1 1.6 | 1.0 | 1.2 |

The commands within the 5 groups listed have approximately the same processing times. The processing times given in the table are maximum times (e.g., when all counters and the global registers are parameterized).

The numbers in parentheses are minimum times with only one counter parameterized.

When the counter module is controlled via parameter STEU (BEF = XX), the same run times apply.

1 Cycle times can be exceeded (PLC goes into STOP status) in a user program when commands with run times of over 100 msec are used. If multiple calls of the function block cannot be avoided (e.g., by loading and starting the counters synchronously), the cycle time must be retriggered (OB 31) when CPU 941A (and possibly CPU 942A also) is used.

## 11.8     Application of the Funktion

Function block FB 178 or FB 179 is usually called in the start–up program of the programmable controller to parameterize the IP 242 A counter module. Enter the data to be transferred in the parameterization data block before parameterization (with parameter BEF = PA or appropriate assignment of the STEU parameter).

> ⚠  **After parameterization**, counters 1 to 5 are in stop status and must be started for operating. Counters 6 and 7 are started automatically after parameterization. All counter outputs are disabled. ⚠

Parameter **BEF** informs function block FB 178 or 179 what function is to be performed. The commands from FB work in both directions between parameterization data block and module or dual port RAM ($\rightarrow$ section 6.1.1).

Parameter **PAR** specifies the counters with which this function is to be performed (simultaneously).

The error information register can be read at parameter **ERR** after the selected function has been per–formed.

**RB     Reset module**

Control word "reset module" (KH = 0100) is transferred.

The counter module usually takes more than 1 msec to process this command. Bit 2 of parameter MELD ("module busy") is set and the block is exited without reading the error information register. This is done the next time the function block is called (with any command).

**SA     Disable outputs**

Control word "disable outputs" (KH = 0200) is transferred.

**FA     Enable outputs**

Control word "enable outputs" (KH = 0300) is transferred.

**IM     Mask interrupt**

The interrupt enable register and the interrupt polarity register are transferred from the parameterization data block to the module. Control word "mask interrupt" (KH = 0400) is transmitted afterwards.

**ST     Start counter**

Control word "start counter" (KH = 11xx) is transferred with the appropriate counter bits (bits 1 to 5).

**LD     Load counter**

The load register and the hold register of the counters selected are transferred from the parameterization data block to the module. Control word "load counter" (KH = 12xx) is then transferred with the appropriate counter bits (bits 1 to 5).

**SP    Stop counter**

Control word "stop counter" (KH = 13xx) is transferred with the appropriate counter bits (bits 1 to 5).

**SL    Stop and read counter**

Control word "stop and read counter" (KH = 14xx) is transferred with the appropriate counter bits (bits 1 to 5). The hold registers of the counters selected are then read by the module and entered in the parameterization data block.

**SZ    Step counter**

Control word "step counter" (KH = 15xx) is transferred with the appropriate counter bits (bits 1 to 5).

**SV    Save counter**

Control word "save counter" (KH = 16xx) is transferred with the appropriate counter bits (bits 1 to 5). This stores the current counter value in the internal hold register. The internal hold registers can be read out with the "copy counter" command.

**CO    Copy counter**

"Copy counter" can only be executed under the following conditions.
– SV was called at least once before.
– One of operating modes N, O, Q, R or X is activated. (In these operating modes, the active gate edge causes an SV on the hardware side.)

Control word "copy counter" (KH = 17xx) is transferred with the appropriate counter bits (bits 1 to 5) (entry in the hold register in the dual port RAM). The hold registers of the counters selected are then read by the module and entered in the parameterization data block (this overwrites the parameterized data).

**LS    Load and start counter**

The load register and the hold register of the counters selected are transmitted from the parameterization data block to the module. Control word "load and start counter" (KH = 31xx) is then transferred with the appropriate counter bits (bits 1 to 7).

**LE    Read counter**

Control word "read counter" (KH = 32xx) is transferred with the appropriate counter bits (bits 1 to 7). The hold registers of the counters selected are then read by the module and entered in the parameterization data block.

Since counters 6 and 7 are processed by the firmware consecutively, a difference of 15 $\mu$sec must be kept in mind when synchronously running counters are used.

**RZ    Reset counter**

Control word "reset counter" (KH = 33xx) is transferred with the appropriate counter bits (bits 1 to 7).

**AW    Take over interrupt value**

The interrupt registers of the counters selected are transferred from the parameterization data block to the module. Control word "take over interrupt value" (KH = 34xx) is then transferred with the appropriate counter bits (bits 1 to 7).

**DF    Difference generation**

Control word "difference generation" (KH = 35xx) is transferred with the appropriate counter bits (bits 1 to 7). The difference of the counter values is then read by the module and entered in the parameterization data block.

**PA    Parameterize counter**

All registers of the counters selected, including the commands for interrupt processing and the global registers (if selected), are transferred from the parameterization data block to the module. Control word "parameterization counter" (KH = 71xx) is then transferred with the appropriate counter bits (bits 1 to 7) and the bit for the global registers (bit 0).

The counter module usually takes more than 1 msec to process this command. Bit 2 of parameter MELD ("module busy") is set and the block is exited without reading the error information register. This is done the next time the function block is called (with any command).

**PS    Store parameter**

All registers of the counters selected, including the commands for interrupt processing and the global registers (if selected), are transferred from the parameterization data block to the module. Control word "store parameter" (KH = 72xx) is then transferred with the appropriate counter bits (bits 1 to 7) and the appropriate bit for the global registers (bit 0).

The counter module usually takes more than 1 msec to process this command. Bit 2 of parameter MELD ("module busy") is set and the block is exited without reading the error information register. This is done the next time the function block is called (with any command).

**PZ    Rewrite parameter**

Control word "rewrite parameter" (KH = 73xx) is transferred with the appropriate counter bits (bits 1 to 7) and the bit for the global registers (bit 0). Afterwards, all registers of the counters selected and the global registers (if selected) are read by the module and entered in the parameterization data block.

If processing in the counter module takes longer than 1 msec, bit 2 of the MELD parameter ("module busy") is set and the block is exited without transferring the registers. When this happens, the registers can be transferred later using command BL ("read module").

**PS must have been called at least once before PZ is called for the first time.**

**GR    Take over basic setting**

Control word "take over basic setting" (KH = 74xx) is transferred with the appropriate counter bits (bits 1 to 7) and the bit for the global registers (bit 0). Afterwards, all registers of the counters selected and the global registers (if selected) are read from the module and entered in the parameterization data block.

If processing in the counter module takes longer than 1 msec, bit 2 of the MELD parameter ("module busy") is set and the block is exited without transferring the registers. When this happens, the registers can be transferred later using command BL ("read module").

**TF      Execute test function**

Control word "execute test function" (KH = 81xx) is transferred with the appropriate test function bits.

If processing in the counter module takes longer than 1 msec, bit 2 of the MELD parameter ("module busy") is set and the block is exited without reading the error information register. This is then performed the next time the function block is called.

**BL      Read module**

All registers of the counters selected, including the commands for interrupt processing and the global registers (if selected), are transmitted from the module to the parameterization data block. A control word is not transferred and the error information register is not read.

**BS      Write module**

All registers of the counters selected, including the commands for interrupt processing and the global registers (if selected), are transmitted from the parameterization data block to the module. A control word is not transferred and the error information register is not read.

**IN      Interrupt processing**

The interrupt information register and the error information register are read from the module and transferred to parameters IIR and ERR. When using programmable controllers S5–150U/S and S5–155U, the system data bit for interrupt processing is also reset if necessary.

**XX      Switch–over parameter to STEU**

The control word in parameter STEU is transferred to the module instead of a command from parameter BEF supplemented by parameter PAR.

The control word must be completely specified (e.g., including the counter bits). If the command specified (left byte of the control word) corresponds to a command which could also have been specified by parameter BEF, the function block then performs a check for correct parameterization assignment. Otherwise transfer is made to the module without checking the bit pattern!

If the bit pattern corresponds to a command specified by parameters BEF and PAR, the appropriate data is either transferred to the module before writing the control word or is read by the module after writing the command, depending on the command.

## 11.9 Error Evaluation

Function block **FB 178** or **FB 179** uses parameter **ERR** as well as parameters **MELD** and **F–NR** to indicate errors which have occurred.

When one of the above parameters indicates an error, the function block sets **bit 7** of parameter **MELD** to signal status "1" (group error signal).

An error coming from counter module IP 242A (from the error information register) is passed on at parameter **ERR**. This register is scanned both before **and** after a control word is written on the module.

In both cases, bit 3 of parameter **MELD** is set to signal status "1" when the contents of the error information register are not equal to "0".

If the contents of the error information register are not equal to "0" before writing a control word (i.e., the counter module reported an error since the last processing), **bit 4** of parameter MELD is set to signal status "**1**" and the control word is not transferred.

Bit 4 is not set if the error information register contains an error number after the control word is written.

**Parameter F–NR** contains an error number if incorrect information is specified in a function block parameter or an error occurs as a result.

**Parameter MELD** is allocated bit by bit and reports operational states of the function block. The individual bits are allocated as follows:

**Bit 0:** Function block and module firmware accessing of the counter module dual port RAM is cross interlocked.

When the function block requires access to the dual port RAM, it sets a request bit and scans the appropriate bit set by the firmware. The function block can access the module dual port RAM when the RAM is not being accessed by the firmware. The function block cancels its request bit after access is completed.

When accessing by the firmware requires too much time (i.e., >1 msec), bit 0 has signal status "**1**" indicating "**no** access to dual port RAM" and the block is exited.

This bit is also set when the module does not allow access within approximately 1 msec after the control word is written. See bit 2 for exception.

**Bit 1:** The work of the module is monitored by an alternating bit (watchdog). If the bit does not change its signal status within 127 passes of the function block, bit 1 of parameter MELD is set and processing of the function block is aborted.

Processing can only be resumed when the signal status of the watchdog bit changes.

**Bit 2:** There are several commands on the module which take longer than 1 msec for the firmware to execute. See table of valid commands in section 11.5. It would require too much time to wait until one of these commands is executed before reading the error information register. Bit 2 of parameter MELD is then set and the block is exited.

> ⚠ No operations are performed by the function block as long as bit 2 is set (i.e., as long as the module is busy with the processing of the last command). ⚠

**Bit 3:** The signal status of bit 3 is always "1" when an error number is output at the ERR parameter (group bit for the ERR parameter).

**Bit 4:** The signal status of bit 4 is set to "1" when parameter ERR is assigned **before** the control word is written. (See description of parameter ERR.)

**Bit 5:** In multiprocessor operation (when several processors access the same counter module), the addressing of the dual port RAM by the function block is interlocked so that the same areas of the dual port RAM (e.g., the same counters) cannot be accessed at the same time.

If the function block is prevented from accessing the RAM for this reason, the function block waits for the enable before accessing the RAM. If the function block has to wait more than 1 msec, the signal status is set to "1" and the block is exited.

**Bit 6:** This bit can be used to determine in the user program whether the counter module is busy prior to the write access of the control word (e.g., busy with the processing of a command list), and FB 178/FB 179 has not transferred the control word to the counter module because of this. If this is the case (bit 6 is set), the function block must be called again.

**Bit 7:** The signal status of this bit is "1" if one of the three parameters ERR, MELD, or F–NR is assigned (i.e., not equal to zero).

## 11.10  Interrupt Processing
## 11.10.1 General

If, for example, a gate signal or an output signal is provided with an interrupt identification by allocating the interrupt enable register, the module jumper assignments must be set for

– either an interrupt line (S5–115U, S5–135U, and S5–155U in S5–155U mode)

– or a group interrupt bit for input byte IB 0 (S5–150U/S and S5–155U in S5–150U mode).

(See operating instructions for the respective programmable controller.)

To prevent the applicable organization block from being processed (at the rising edge of the process interrupt) more than once when using PLC S5–150U/S or PLC S5–155U in S5–150U mode, the number of this group interrupt bit must be specified in the ABIT parameter as follows:

ABIT    :    KY  =  **x, y**
$\qquad\qquad$ **x** = 255, **y** = 255 $\qquad\qquad$ No resetting
$\qquad\qquad$ **x** = 0, **y** = 0 to 7 $\qquad\qquad$ Resetting of the appropriate interrupt bit in the
$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ system data.
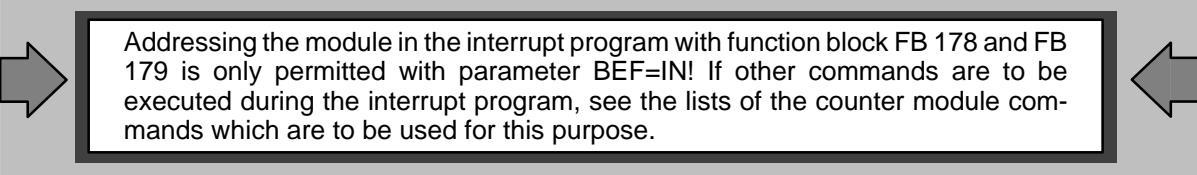
The applicable organization block is processed again when the interrupt signal disappears if this bit is not reset in the system data.

When an interrupt corresponding to jumper allocation occurs, an organization block from the interrupt controlled processing is called.

The signal states of the scratchpad flags must be saved in a data block at the beginning of the organization block and reloaded again at the end.

Function blocks FB 38 and FB 39 can be used for this purpose in connection with data block DB 255 on the floppy disk on which standard function blocks FB 178 and FB 179 are furnished.

The interrupt organization block contains the call of function block FB 178 or FB 179 with the parameter assignment of BEF = IN. After function block FB 178 or FB 179 is called, the contents of parameter IIR indicate the signal from which the interrupt came. You can process your special interrupt program now.

> Addressing the module in the interrupt program with function block FB 178 and FB 179 is only permitted with parameter BEF=IN! If other commands are to be executed during the interrupt program, see the lists of the counter module commands which are to be used for this purpose.

## 11.10.2 Special Considerations – PLC S5–115U

An interrupt of the user program always occurs at **command limits**.

Function block FB 38 and FB 39 are used in the interrupt program from data files S5LC50ST.S5D when the IP 242A counter module is used with programmable controller S5–115U. This also applies to interrupts caused by time interrupts. (See section 16.)

Function block FB 38 **saves the user system data** (BS 248 to BS 255), **the current page frame number, and the scratchpad flag area** (FY 200 to FY 255). Function block FB 39 loads the data stored by function block FB 38.

**In cyclic operation**, function block **FB 38** with the "save page frame number" parameterization must be called (e.g., via handling blocks, via direct access through the user program, or via standard function blocks (FB 178)) before the processing of a page frame access, if page frame accesses are programmed in the interrupt organization blocks (e.g., by calling FB 178 in OB 2 with BEF = IN) and if you are working with different page frame numbers.

**In interrupt organization blocks**, the scratchpad flags must always be saved (FB 38) at the beginning and loaded (FB 39) again at the end. In addition to loading the scratchpad flags, function block **FB 39** also handles the loading of user system data and the activation of the page frame number which was saved by function block **FB 38** in the cyclic program.

> ⚠ Function blocks FB 38 and FB 39 must always be used **in pairs** in the interrupt organization blocks (i.e., the interrupt organization blocks may not be prematurely exited using the BEC command for example). ⚠

## 11.10.3 Special Considerations – PLC S5–135U

An interrupt of the user program occurs at **block limits or**, when data block DX 0 is appropriately parameterized, at **command limits**.

If the user program is programmed with interrupt organization blocks in which the scratchpad flag area (flag bytes FY 200 to FY 255) is also used, be sure that this flag area is saved and then reloaded before exiting the interrupt organization block. System data BS 60 to BS 63 is handled the same way.

You can use function blocks FB 38 and FB 39, for example, from the S5LC22ST.S5D file of the program example for this purpose.

> ⚠ Function blocks FB 38 and FB 39 must always be used **in pairs** (i.e., the interrupts organization blocks may not be prematurely exited using the BEC command for example). ⚠

## 11.10.4 Special Considerations – PLC S5–150U/S

An interrupt of the user program occurs at **block limits**.

If the user program is programmed with interrupt organization blocks in which the scratchpad area (flag bytes FY 200 to FY 255) is also used, be sure that this flag area is saved and then reloaded before exiting the organization blocks.

## 11.10.5 Special Considerations – PLC S5–155U

An interrupt of the user program always occurs at **block limits** (S5–150U mode) **or**, when data block DX 0 is appropriately parameterized, at **command limits** (S5–155U mode).

If the user program is programmed with interrupt organization blocks in which the scratchpad flag area (flag bytes FY 200 to FY 255) and/or system data BS 60 to BS 63 are also used, be sure that this flag and system data area is saved and reloaded before exiting the interrupt organization blocks.

Functions blocks FB 38 and FB 39 from data file S5LC60ST.S5D must always be used to save and load the scratchpad flag area and the system data. The function blocks work together with a data block (data block DB 255 in our example in section 11). This data block must be set up up to and including data word DW 826.

> **!** Function blocks FB 38 and FB 39 must always be used **in pairs** (i.e., the interrupts organization blocks may not be prematurely exited using the BEC command for example). **!**

## 11.11   Start–Up Behavior
### 11.11.1 Start–Up Behavior – PLC S5–115U

After a manual cold restart (OB 21) and after an automatic cold restart (OB 22), cyclic program processing starts at the beginning of organization block OB 1.

Interrupts are enabled during processing of function block FB 178 or FB 179 so that interrupts and/or time interrupts can also be processed during start–up.

### 11.11.2 Start–Up Behavior – PLC S5–135U

After a cold restart (OB 20), cyclic program processing starts at the beginning of organization block OB 1.

For a manual warm restart (OB 21) or an automatic warm restart (OB 22), program processing is continued at the point of interruption after the start–up organization block has been executed.

> **A warm restart cannot be performed** when the IP 242A counter module is used with PLC S5–135U.
> The "automatic warm restart" function must be set in the "automatic cold restart after power on" function with the aid of DX 0.

> When a manual restart is performed twice in a row, this causes erroneous behavior since program processing is continued after the stop command during the second restart (i.e., a transition is made to cyclic operation).
>
> ```
> Remedy: OB21 :  JU FB nn        FB nn :
>              :  : Name STOP     Name  : STOP
>              :  : PE            F001  : STS
>                                       : JU = F001
> ```

### 11.11.3 Start–Up Behavior – PLC S5–150U/S

After a cold restart (OB 20), cyclic program processing starts at the beginning of organization block OB 1.

For a manual warm restart (OB 21) or an automatic warm restart (OB 22), program processing is continued at the point of interruption after the start–up organization block has been executed.

> An **automatic or manual warm restart cannot be performed** when the IP 242A counter module is used with PLC S5–150U/S.

```
OB 21/OB 22  :        STP      STOP at the end of cycle
                      BE
```

IP 242A/242B Equipment Manual                                           11 – 25
© Siemens AG 1993, Order No: 6ES5 998–0KM21

## 11.11.4 Start–Up Behavior – PLC S5–155U

After a cold restart (OB 20), cyclic program processing starts at the beginning of organization block OB 1.

For a manual warm restart (OB 21) or an automatic warm restart (OB 22), program processing is continued at the point of interruption after the start–up organization block has been executed.

> **!** If, during start–up organization blocks OB 21 and OB 22, blocks are called which work with the scratchpad flag area (flag bytes FY 200 to FY 255), it is mandatory that this flag area be saved and reloaded with function blocks FB 38 and FB 39 from data file S5LC60ST. S5D before the start–up organization blocks are exited. **!**

## 11.12   Multiprocessor Operation

The IP 242A counter module can be used in multiprocessor operation. Function blocks FB 178 and FB 179 for programmable controllers S5–135U and S5–155U are designed for easy operation in a multiprocessor environment.

If, during multiprocessor operation, the counter module is addressed simultaneously by several processors, the following points should be observed to ensure smooth processing:

Parameterization of the **global registers** is performed **by a single processor**.

For processes affecting the counters (e.g., parameterization loading and starting), it is a good idea to address the **respective counter from only one processor**.

**Reading** of the module can be performed by any processor without restriction.

Processing of process interrupts and interrupts can be performed by only **one** processor.

# 12 Program Example for IP 242A

## 12.1 General

A

The programming example described here is contained on the included floppy disk and can be loaded complete into the programmable controller memory for testing the module. The example shows how function block FB 178 can be used. Direct and indirect parameterization of the module is also explained. All necessary blocks are present for an executable program. At the same time, the floppy disk also provides a complete program framework which for an executable program can be utilized by the user.

The direct or indirect parameterization of the function block is selected via a digital input of the simulator. Individual commands can be transmitted to the module via additional digital inputs. Errors and counter interrupts are indicated by the digital outputs.

The example of indirect parameterization covers all possible commands whereas the example of direct parameterization is limited to the following commands:

| COMMAND<br>KS = | Meaning | Parameter<br>PAR |
|---|---|---|
| IM | Mask interrupt | – |
| SZ | Step counter | Counter 2 |
| LS | Load and start counter | Counters 2 and 6 |
| LE | Read counter | Counters 1 to 7 |
| RZ | Reset counter | Counters 2 and 6 |
| AW | Take over interrupt value | Counters 2 and 6 |
| DF | Generate difference | Counters 2 and 6 |
| PA | Parameterize counter | Counters 1 to 7, global register |
| PS | Store parameter | Counters 1 to 7, global register |
| PZ | Rewrite parameter | Counters 1 to 7, global register |
| BL | Read module | Counters 1 to 7, global register |
| IN | Process interrupt | – |

## 12.2   Device Configuration

The following devices, for example, can be used to test the IP 242A counter module:

–  One of the programmable controllers listed

–  Programmer (e.g., PG 685, PG 750)

–  Counter module IP 242A (6ES5 242–1AA31)

–  Shaft encoder (5 V) with 2 pulse trains, phase–displaced by $90°$

–  Digital input module (e.g., 6ES5 420–4UA11)

–  Digital output module (e.g., 6ES5 421–4UA11)

–  Simulator for digital inputs and outputs (e.g., 6ES5 788–0LA12)

PLC

Programmer

IP 242A

Pulse encoder

Digital out-
put module

Digital input
module

Simulator
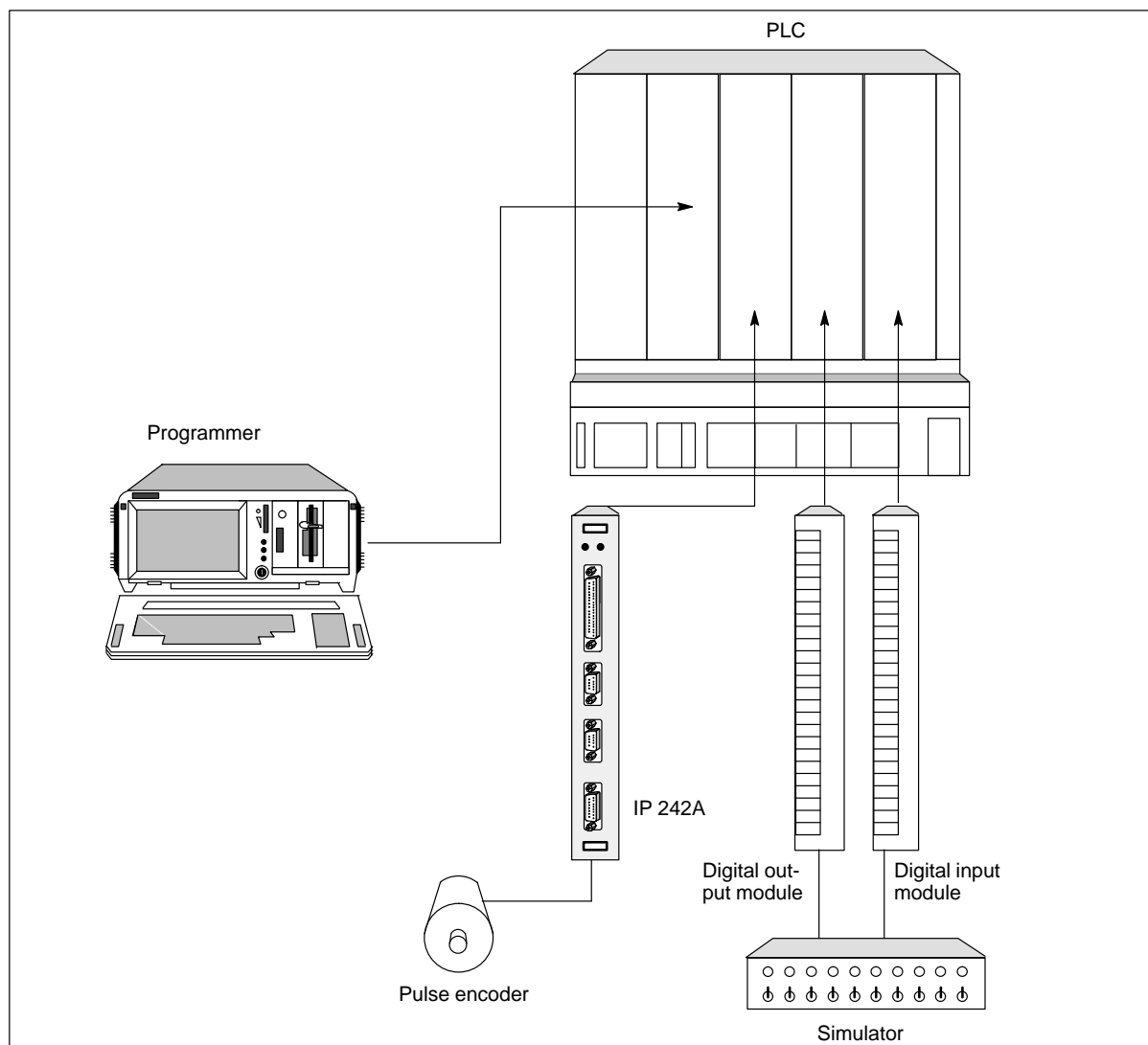
## 12.3    Jumper Allocation for Counter Module IP 242A

The following switches and jumpers must be set before the module is inserted. All other jumpers and switches remain in their original positions.

Switch S1 :
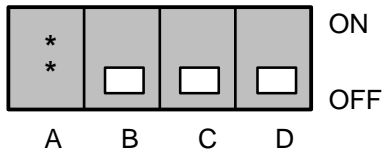
| S1.1 | Programmable controller |
|------|-------------------------|
| ON | PLC S5–115U |
| ON | PLC S5–135U |
| OFF | PLC S5–150U/S |
| ON | PLC S5–155U (155U mode) |

A = ON –> interrupt triggering
            via interrupt line A

Switch S2:

S2.10 = ON –> interrupt triggering via bit 0.0
              (coding as master module)

| S2.10 | Programmable Controller |
|-------|-------------------------|
| OFF | PLC S5–115U |
| OFF | PLC S5–135U |
| ON | PLC S5–150U/S |
| OFF | PLC S5–155U (155U mode) |

Switch S3:
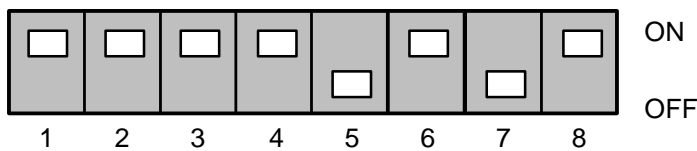
Page frame addressing
Page frame address = F400 H

Switch S4:
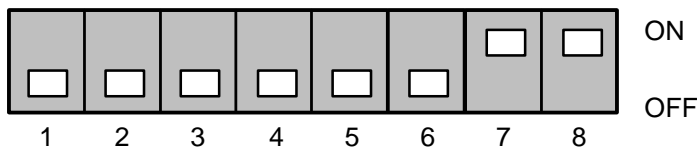
Page frame number = 3

Jumpers 1 and 2:      open

Jumpers 3 to 17:      insert in position 2 – 3

## 12.4     Allocation of the Inputs and Outputs

The program is designed so that it can easily be adapted to different input and output allocations. The program example works only with flags. The inputs and outputs used are assigned in organization block OB 1 to these flags. In our example, these are input word IW 4, input byte IB 6, output word QW 4 (or QW 8 for programmable controller S5–115U), and output word QW 6 (or QW 10 for programmable controller S5–115U).

| Input | Input Image | Corres-ponding Pulse Flag | Direct Parameterization | | Indirect Parameterization |
|-------|-------------|---------------------------|-------------------------|---|---------------------------|
| I 4.0 | F 4.0 | F 12.0 | PA | Parameterize | C2 + C6 + global register | Control word bit 8 |
| I 4.1 | F 4.1 | F 12.1 | LE | Read | C1 to C7 | Control word bit 9 |
| I 4.2 | F 4.2 | F 12.2 | – | | | Control word bit 10 |
| I 4.3 | F 4.3 | F 12.3 | LS | Load and start | C2 | Control word bit 11 |
| I 4.4 | F 4.4 | F 12.4 | SZ | Step | C2 | Control word bit 12 |
| I 4.5 | F 4.5 | F 12.5 | RZ | Reset | C2 | Control word bit 13 |
| I 4.6 | F 4.6 | F 12.6 | AW | Interrupt value | C2 | Control word bit 14 |
| I 4.7 | F 4.7 | F 12.7 | – | | | Control word bit 15 |
| | | | | | | |
| I 5.0 | F 5.0 | F 13.0 | LS | Load and start | C6 | Control word bit 0 |
| I 5.1 | F 5.1 | F 13.1 | RZ | Reset | C6 | Control word bit 1 |
| I 5.2 | F 5.2 | F 13.2 | AW | Interrupt value | C6 | Control word bit 2 |
| I 5.3 | F 5.3 | F 13.3 | – | | | Control word bit 3 |
| I 5.4 | F 5.4 | F 13.4 | – | | | Control word bit 4 |
| I 5.5 | F 5.5 | F 13.5 | DF | Difference | C2 to C6 | Control word bit 5 |
| I 5.6 | F 5.6 | F 13.6 | PS | Store parameter | | Control word bit 6 |
| I 5.7 | F 5.7 | F 13.7 | IM | Mask Interrupt | | Control word bit 7 |
| | | | | | | |
| I 6.0 | F 6.0 | F 14.0 | Acknowledge error | | | Acknowledge error |
| I 6.1 | F 6.1 | F 14.1 | Acknowledge interrupt message | | | Acknowledge interrupt message |
| I 6.2 | F 6.2 | F 14.2 | – | | | Control with FW 4 |
| I 6.3 | F 6.3 | F 14.3 | = 0 direct | | | = 1 indirect |

| Output | Output for PLC S5–115U Only | Output Image | In Programmable Controller |
|--------|------|------|------|
| Q 4.0 | (Q 8.0) | F 16.0 | Error in start–up program |
| Q 4.1 | (Q 8.1) | F 16.1 | Error in interrupt program (OB 2) |
| Q 4.2 | (Q 8.2) | F 16.2 | Error in cyclic program (OB 1) |
| Q 4.3 | (Q 8.3) | F 16.3 | Error in FB 38 or FB 39 (only S5–115U) |
| Q 4.4 | (Q 8.4) | F 16.4 | – |
| Q 4.5 | (Q 8.5) | F 16.5 | Interrupt at counter 2 |
| Q 4.6 | (Q 8.6) | F 16.6 | Interrupt at counter 6 |
| Q 4.7 | (Q 8.7) | F 16.7 | Interrupt is triggered. |

Output byte QB 5                          Error information register(for an error
(QB 9 for S5–115U)                        in the cyclic program)


| Output | Output for PLC S5–115U Only | Output Image | Meaning |
|--------|------|------|------|
| Q 6.0 | (Q 10.0) | F 18.0 | No access to dual port RAM |
| Q 6.1 | (Q 10.1) | F 18.1 | Watchdog monitoring is triggered. |
| Q 6.2 | (Q 10.2) | F 18.2 | Module busy |
| Q 6.3 | (Q 10.3) | F 18.3 | Error information register (ERR) is allocated. |
| Q 6.4 | (Q 10.4) | F 18.4 | ERR allocated before writing control word |
| Q 6.5 | (Q 10.5) | F 18.5 | Dual port RAM allocated (for multiprocessor operation only) |
| Q 6.6 | (Q 10.6) | F 18.6 | – |
| Q 6.7 | (Q 10.7) | F 18.7 | Group error bit |

Output byte QB 7                          Error number parameter F–NR
(QB 11 for S5–115U)

## 12.5     Allocation of the Flag Area

| | | |
|---|---|---|
| FW | 4 | Flag of input word IW 4 |
| FY | 6 | Flag of input byte IB 6 |
| FW | 8 | Edge flag of input word IW 4 |
| FY | 10 | Edge flag of input byte IB 6 |
| FW | 12 | Pulse flag of input word IW 4 |
| FY | 14 | Pulse flag of input byte IB 6 |
| FW | 16 | Flag of output word QW 4 (QW 8) |
| FW | 18 | Flag of output word QW 6 (QW 10) |
| | | |
| FY | 44 | Parameter   ERR    Start–up + interrupt program |
| FY | 45 | Parameter   MELD   Start–up + interrupt program |
| FY | 46 | Parameter   F–NR   Start–up + interrupt program |
| | | |
| FY | 48 | Parameter   ERR    Cyclic program |
| FY | 49 | Parameter   MELD   Cyclic program |
| FY | 50 | Parameter   F–NR   Cyclic program |
| FW | 52 | Parameterization error in FB 38 (only PLC S5–115U) |
| FW | 54 | Parameterization error in FB 39 (only PLC S5–115U) |
| | | |
| FW | 100 | Parameter   STEU   Indirect parameterization |
| | | |
| FW | 102 | Parameter   IIR      Interrupt information register |

## 12.6     Allocation of the Data Area

| | | |
|---|---|---|
| DB | 178 | Parameterization data block |
| DB | 255 | Intermediate storage for scratchpad flags |
| DX | 0 | Operating system presetting (for PLCs S5–135U and S5–155U only) |

**Block assignments**

| | | |
|---|---|---|
| FB | 38 | Save scratchpad flags |
| FB | 39 | Load scratchpad flags |
| FB | 77 | Example of indirect parameterization |
| FB | 78 | Example of direct parameterization |
| FB | 178 | Process counter module |
| | | |
| OB | 1 | Cyclic program |
| OB | 2 | Interrupt program |
| OB | 20 | Cold restart (not for PLC S5–115U) |
| OB | 21 | Automatic warm restart (cold restart for PLC S5–115U) |
| OB | 22 | Automatic warm restart or automatic cold restart |

## 12.7    Turn–On, Start-Up Behavior

After an overall reset of the programmable controller, load the entire data file into the central processor's user memory. Then perform a cold restart.

The counter module is already parameterized in the start–up organization blocks. Parameterization data block DB 178 on the floppy disk already contains valid parameterization data and does not need to be modified. If parameterization is correct, the green RUN LED on the counter module lights up continuously.

If all inputs on the simulator are in switch position "0" when the programmable controller is turned on, no outputs should be set after the programmable controller has started up. If output Q 4.0 (Q 8.0 for PLC S5–115U) is set, a parameterization error occurred during start–up. You can determine the exact cause of the error with the aid of flag bytes FY 44 (parameter ERR), FY 45 (parameter MELD), and FY 46 (parameter F–NR).

## 12.8    Cyclic Operation

Use input I 6.3 to alternate between direct and indirect parameterization of function block FB 178.

If the signal status of input I 6.3 is "0", function block FB 78 is called for direct parameterization. If the signal status of input I 6.3 is "1", function block FB 77 is executed for indirect parameterization.

**Direct parameterization**

Function block FB 78 shows the following functions: parameterizing counters, loading and starting, resetting, stepping and reading. The example also shows how interrupt values can be taken over. In addition, parameters can be permanently stored in the EEPROM of the counter module, the interrupt mask can be taken over, and the difference between two counter values can be generated.

**Indirect parameterization**

For indirect parameterization, the control word is preset at input word IW 4. The command whose control word you supplied at input word IW 4 is executed at the rising edge of input I 6.2.

**Error messages**

Output Q 4.2 (Q 8.2 for PLC S5–115U) is set when an error occurs during the cyclic program. You can determine the cause of the error with the aid of flag bytes FY 48 (parameter ERR), FY 49 (parameter MELD), and FY 50 (parameter F–NR).

These flag bytes are also shown at output bytes QB 5 to QB 7 (QB 9 to QB 11 for PLC S5–115U).

You can erase the error message again with input I 6.0.

## 12.8.1  Direct Parameterization

**Parameterize the counters**

Counter 2, counter 6, and the global registers are parameterized with input I 4.0.

– DW  0   to   DW  15    Global register
– DW 30   to   DW  43    Parameterization data, counter 2
– DW 86   to   DW 102    Parameterization data, counter 6

**Reading the counter values**

Using the "control variable" programmer function, the counter values on the programmer screen can be read from data block DB 178 and indicated in the following data words:

DW   18     Hold register for counter   1
DW   32     Hold register for counter   2
DW   46     Hold register for counter   3
DW   60     Hold register for counter   4
DW   74     Hold register for counter   5
DW   89     Hold register for counter   6  (sign, bits 16 to 23)
DW   90     Hold register for counter   6  (bits 0 to 15)
DW  106     Hold register for counter   7  (sign, bits 16 to 23)
DW  107     Hold register for counter   7  (bits 0 to 15)

Function block FB 178 is called with command BEF = LE (read counters 1 to 7), as long as the signal status of input I 4.1 is "1" (static function). If you trigger command BEF = SZ (step counter), for example, during reading, reading of the counters is interrupted for this one S5 cycle (dynamic function).

DW   18    KH  =  0100
DW   32    KH  =  0200
DW   46    KH  =  0300
DW   60    KH  =  0400
DW   74    KH  =  0500
DW   89    KH  =  0000
DW   90    KH  =  0600
DW  106    KH  =  0000
DW  107    KH  =  0700

Leave input I 4.1 on so that you can observe changes in the counter values.

**Step counters**

In the example, the counter value of counter 2 is stepped by one pulse with the rising edge at input I 4.4. Counting is performed unconditionally.

The parameterization of the counter mode register determines the direction of counting (counting down, in the example).

**Change counter value 6**

There is no STEP command for counters 6 and 7. Connect the pulse encoder to counting input 6 to change the counter value of counter 6. Turn the encoder to change the counter value. The counter value is increased/decreased depending on the direction in which the encoder is turned.

**Load and start counter 2 or counter 6**

By activating input I 4.3 for counter 2 or input I 5.0 for counter 6, the corresponding counter is loaded with the value in the respective load register and is started.

The counter values now indicate actual values of KH = 0200 for counter 2 and KH = 0600 for counter 6.

**Reset counters**

By activating input I 4.5 for counter 2 or input I 5.1 for counter 6, the corresponding counter is reset to KH = 0000.

**Generate difference**

At input I 5.5 you can activate the generation of the difference between the counter values of counter 2 and counter 6.

The result (counter 2 minus counter 6) is stored in the result register of parameterization data block DB 178 as follows:

DW    9        Result register (bits 31 to 16)
DW    10       Result register (bits 15 to 0)

Since the result itself is a 24–bit binary number in dual complement format and bits 25 to 31 contain the status of bit 24, again it is only practical to use the binary counting mode (preselection in the CMR) for the counters used here.

**Store parameter**

The transmission of all registers of counter 1 to counter 7, including the interrupt processing commands and the global registers, to the counter module is triggered by the rising edge at input I 5.6.

Control word "store parameter" is then transferred. This control word causes the counter module to store the transferred data in the module's EEPROM.

> After power returns, the counter module is parameterized with the data in the EEPROM. For this reason, the "store parameter" command should be executed at least once to ensure that the EEPROM contains valid data.

## 12.8.2  Indirect Parameterization

Select the indirect parameterization of function block FB 178 with the "input I 6.3 = 1" signal. In the example, the function block itself is called with the rising edge at input I 6.2.

Command "XX" causes a switch–over to the STEU parameter (i.e., the control word contained in the STEU parameter is now transferred to the module instead of a command from the BEF parameter supplemented by the PAR parameter). In the example, you can specify the contents of the STEU parameter via the input word IW 4.

> **!** Specify the control word completely (i.e., command code with global register and counter bits). **!**

If the specified command (left byte of the control word; in the example, the left byte of input byte IB 4) is a valid command, function block FB 178 performs a check for correct parameter assignment. (In the example, the right byte of the control word is set at input byte IB 5.)

If a valid command code is not found, the block transfers the parameterized control word immediately to the module without outputting an error number.

Exception:   STEU parameter   =   KH 00xx  or
                                            KH FFxx
                 causes error message KH 03 at parameter F–NR.

### 12.8.3 Parameterization of the IP 242A Counter Module

Parameterization data block DB 178 in the program example contains the following valid parameterization data.

**Default settings for the global registers**

Master mode register (DW 4)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |

If bits 2, 3, 12, 13, 14 = 1, then comparators 1 to 5 are active.

Interrupt enable register (DW 7)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

| Bit | 0 | = | 1 | Enable group interrupt for error message |
| Bit | 8 | = | 1 | Enable group interrupt S5 for counter events (terminal count reached, comparator, gate edge) |
| Bits | 9 to15 | = | 1 | Enable outputs 1 to 7 |

**Default settings of counter 2**

Counter mode register, counter 2 (DW 30)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0  | 0  | 0  | 0  | 0  | 0  | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |

Bits  0  to   2          Square wave, start high
Bit   3            = 0   Counting direction down
Bit   4            = 0   Binary counting mode
Bits  5  to   7          Cyclic counting,
                         refill from load register
Bits  8  to  11          Counting pulse source at counting input 2
Bit  12            = 0   Rising counting pulse edge
Bits 13  to  15  = 0   Without gate control

Load register, counter 2 (DW 31)

DW  31  =  0200 H

Interrupt register, counter 2 (DW 33)

DW  33  =  0100 H

**Default settings of counter 6**

Counter mode register, counter 6 (DW 86)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0  | 0  | 0  | 0  | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |

Bits  0  to   2  = 0
Bits  8  to  15  = 0
Bit   3            = 1   Output signal active
Bits  4  to   6          Simple edge evaluation
                         with rising edge at input A for counting up and
                         falling edge at input A for counting down
Bit   7            = 0   Disable the reset input

Load register, counter 6 (DW 87, DW 88)

DW  87  =  0000 H
DW  88  =  0600 H

Interrupt register, counter 6 (DW 91, DW 92)

DW  91  =  0000 H
DW  92  =  0500 H

## 12.9    Processing of Interrupts

In this example, interrupt processing is programmed in organization block OB 2.

Depending on the programmable controller, interrupts are acquired via input byte IB 0 (S5–150U/S) or via interrupt line IR–A (S5–115U, S5–135U, S5–155U in S5–155U mode).

The scratchpad flags must be saved at the beginning of the interrupt block and reloaded before the block is exited. If programmable controllers S5–115U and S5–155U are used, this saving and reloading is performed by standard function blocks FB 38 and FB 39. These function blocks are located on the floppy disk with the standard function blocks supplied with the counter modules.

Saving and reloading of the scratchpad flags must be performed for all interrupt–controlled types of program processing (and also in the warm restart organization blocks and the error evaluation organization blocks). This has already been considered in our example.

If an error occurs during interrupt processing, output Q 4.1 (Q 8.1 for S5–115U) is set. Evaluate flag bytes FY 44 (parameter ERR), FY 45 (parameter MELD), and FY 46 (parameter F–NR) to determine the exact cause of the error.

After the error is corrected, acknowledge the error message at input I 6.0. The error indication Q 4.1 (Q 8.1) is reset with this input.

Remember that "0" cannot be used as the interrupt value for counters 1 to 5 ($\rightarrow$ section 1.9).

**Interrupt Activation with the Interrupt Value**

Each counter of the counter module has an interrupt register in data block DB 178. These registers are located in the following data words for the individual counters:

DW    19        Interrupt register for counter 1
DW    33        Interrupt register for counter 2
DW    47        Interrupt register for counter 3
DW    61        Interrupt register for counter 4
DW    75        Interrupt register for counter 5
DW    91        Interrupt register for counter 6  (sign, bits 23 to 16)
DW    92        Interrupt register for counter 6  (bits 15 to 0)
DW   108        Interrupt register for counter 7  (sign, bits 23 to 16)
DW   109        Interrupt register for counter 7  (bits 15 to 0)

The interrupt values can be changed with the aid of the programmer function "control variable".

Interrupt value 0 is not permitted for counters 1 to 5 ($\rightarrow$ section 3.4.4).

Transfer of counter 2 to the counter module is activated with input I 4.6 of the interrupt value.

The interrupt value of counter 6 is transferred with input I 5.2.

If the counter interrupts are enabled in the interrupt enable register (DW 7) and if all comparators are switched on in the master mode register (DW 4), an interrupt is triggered when the counter status of one counter is the same as the contents of its interrupt register.

The comparator function of counter 6 and counter 7 is selected in bit 3 of the applicable counter mode register (DW 86, DW 103).

Default setting of the master mode register (DW 4)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |

If bits 2, 3, 12, 13, 14 = 1, then comparators 1 to 5 are active.

Default setting of the interrupt enable register (DW 7)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Bit   0         = 1   Enable group interrupt for error message
Bit   8         = 1   Enable group interrupt S5 for counter events
                       (terminal count reached, comparator, gate edge)
Bits  9  to  15 = 1   Enable outputs 1 to 7

Default setting of the counter mode register, counter 6 (DW 86)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |

Bits  0  to   2 = 0
Bits  8  to  15 = 0
Bit   3         = 1   Output signal active
Bits  4  to   6        Simple edge evaluation
                       with rising edge at input A for counting up and falling edge
                       at input A for counting down
Bit   7         = 0   Disable the reset input

**Mask interrupt**

The transmission of the interrupt enable register (DW 7) and the interrupt polarity register (DW 8) to the module is activated at input I 5.7.

Default setting of the interrupt polarity register (DW 8)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |

Bits  1  to   5 = 1   Interrupt at the rising edge of gate signals T1 to T5
Bits  9  to  13 = 1   Interrupt at the rising edge of counter outputs A1 to A5

# 13   Programming Instructions, FBs 180/181/182
**IP 242A**

## 13.1   Overview

These programming instructions describe the following function blocks:

**FB 180 (PER:BS)**        "Control counter module IP 242A"
                           (Write module)

**FB 181 (PER:BL)**        "Control counter module IP 242A"
                           (Read module)

**FB 182 (PER:IN)**        "Control counter module IP 242A"
                           (Acknowledge interrupt)

The function blocks are used with

**Counter module IP 242A**   (6ES5 242–1AA31)

in the following programmable controllers:

| PLC/CPU | | File |
|---------|---|------|
| **S5–115U**  CPU 941A/B to 944A/B | | S5LC50ST.S5D |
| **S5–135U**  CPU 922, CPU 928, CPU 928B | | S5LC22ST.S5D |
| **S5–155U**  CPU 946/947 [1] | | S5LC60ST.S5D |

[1]   Use file S5LC22ST.S5D when a CPU 922 or 928A/B is used in PLC S5–155U.

## 13.2    Function Description

The two standard function blocks below are available for the control of the IP 242A module:

FB 178 (PER:ZSTK)     For page frame addressing
FB 179 (PER:ZSTL)     For linear addressing

This program package has been expanded to include the following high–speed standard function blocks:

FB 180 (PER:BS)       Write module
FB 181 (PER:BL)       Read module
FB 182 (PER:IN)       Acknowledge interrupt

In comparison to function block FB 178, these function blocks have **significantly shorter run times** using the applicable command. This was achieved by removing the remaining block parameters and functions, and omitting various tests (e.g., parameter test, test for acknowledgement delay, watchdog test and others).

Function block **FB 180** corresponds to function block FB 178 (PER:ZSTK) with the BEF = BS command. This allows the **data** in the parameterization data block to be **written on the page frames**, and **no** control word is transferred to the IP 242A counter module.

Function block **FB 181** corresponds to function block FB 178 (PER:ZSTK) with the BEF = BL command. The **data** are **read from the page frames** and written in the parameterization data block. **No** control word is transferred to the IP 242A counter module.

Function block **FB 182** corresponds to function block FB 178 (PER:ZSTK) with the BEF = IN command. The interrupt information register and the error information register are read with function block FB 182: the **interrupt is acknowledged**.

Function blocks FB 180, FB 181 and FB 182 can only be used for interrupt processing (e.g., in OB 2).

This means however, that you will implement your task with FB 178, and not replace it with the "fast" FBs in the interrupt branch until after commissioning.

## 13.3    Calling Function Blocks FB 180, FB 181 and FB 182

Function Block FB 180

```
                                    Parameter
            :JU FB180               Class  Type
NAME  :PER:BS
SSNR  :                               D     KY
PAR   :                               D     KM
MELD  :                               Q     BY
```

```
 Parameter              FB 180
Class   Type
                         PER:BS               Parameter
  D     KY                                   Class  Type
  D     KM          ─── SSNR
                    ─── PAR    MELD ───         Q     BY
```

Function Block FB 181

```
                                    Parameter
            :JU FB181               Class  Type
NAME  :PER:BL
SSNR  :                               D     KY
PAR   :                               D     KM
MELD  :                               Q     BY
```

```
 Parameter              FB 181
Class   Type
                         PER:BL               Parameter
  D     KY                                   Class  Type
  D     KM          ─── SSNR
                    ─── PAR    MELD ───         Q     BY
```

Function Block FB 182

```
                                    Parameter
            :JU FB182               Class  Type
NAME  :PER:IN
SSNR  :                               D     KY
ABIT  :                               D     KY  (*)
IIR   :                               Q     W
ERR   :                               Q     BY
MELD  :                               Q     BY
```
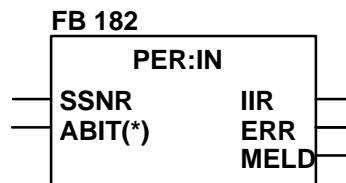
```
 Parameter              FB 182
Class   Type
                         PER:IN               Parameter
  D     KY                                   Class  Type
  D     KM          ─── SSNR    IIR  ───
                    ─── ABIT(*)  ERR  ───       Q     W
                               MELD ───         Q     BY
                                                Q     BY
```

(*)   The ABIT parameter is only available on programmable controller S5–155U.

## 13.4 Explanation of the Parameters

| NAME | CLASS | TYPE | DESIGNATION |
|------|-------|------|-------------|
| SSNR | D | KY | Interface number |
| PAR | D | KM | Parameter (e.g., specification of the counters to be controlled simultaneously) |
| ABIT (*) | D | KY | Number of the interrupt bit during interrupt processing |
| IIR | Q | W | Interrupt information register |
| ERR | Q | BY | Error information register |
| MELD | Q | BY | Message byte |

(*) The ABIT parameter is only available with programmable controller S5–155U.

| **!** | Do not assign the scratchpad flags to the stated parameters. | **!** |

Function blocks FB 180 and FB 181 use the current data block as the parameterization data block. This means that the parameterization data block must have been opened with A DB x before function block FB 180 or FB 181 is called.

When a data word or a data byte is specified as the current parameter during parameterization of the function block in the IIR, ERR and MELD parameters, this data word or data byte is located in the current data block (i.e., in the parameterization data block). No check is made to determine whether data for the counter module may be overwritten!

Example:

```
        :A    DB 178                Parameterization data block DB 178
        :JU   FB 180
NAME  :PER:BS                       Write module
SSNR  :KY  0,8                      Interface number 8
PAR    :KM  00000000  11111111      All counters and global registers
MELD  :FY   10                      Messages
```

---

### 13.5    Assignment of the Parameters

---

**SSNR   :    Interface number**

   KY = **x**, 0 to 255          Page frame address
                                (x = disregard)

**PAR     :    Parameter**

   Parameter (e.g., specification of the counters to be controlled)

   Assignment for the Counter Channels

   KM = 0000 0000 xxxx xxxx
                            └──── Global register
                            └──── Counter 1
                            └──── Counter 7

   The counters or global registers to be controlled are selected with x = "1". Remember that at least one counter and/or the global registers are always specified. The selection of several counters and, in addition, the global registers is also permitted.

> **!** To minimize run times, the function block does not check for erroneous assignments. **!**

**ABIT    :    Number of the Interrupt Bit**

(Function block FB 182: Applicable when interrupt processing is used with programmable controller S5–155U in S5–150U mode)

   KY = **x, y**    x  =  Enable for reset
                    x >< 0          No reset
                    x = 0            Reset the applicable interrupt bit in the system data

                   **y**  =  Number of the interrupt bit
                    $0 \leq y \leq 7$

> **!** To minimize run times, the function block does not check for erroneous assignments (e.g., y > 7). **!**

**IIR        :        Interrupt Information Register**

The contents of the interrupt information register of the counter module are output in the IIR parameter during interrupt processing (FB 182).

**ERR      :        Error Information Register**

The ERR parameter is output with the contents of the error information register of the counter module during interrupt processing (FB 182).

**MELD   :        Message Byte**

The function block stores the following messages in this byte.

Bit 0    No access to the dual port RAM

Bit 1    Not assigned

Bit 2    Not assigned

Bit 3    The error information register is not zero.

Bit 4    Not used

Bit 5    Dual port RAM is occupied with another processor (only for multi–processor operation).

Bit 6    Not used

Bit 7    Group error bit. Has signal status "1" when a message occurred (bit 0, 3 or 5 is "1")

## 13.6    Technical Specifications

**Function Block FB 180** (BS: Write Module)

| | Programmable Controller | | | | | | |
|---|---|---|---|---|---|---|---|
| | **S5–115U** | | | | **S5–135U** | | **S5–155U** |
| | CPU 941<br>A   &#124;B | CPU 942<br>A   &#124; B | CPU 943<br>A   &#124;B | CPU 944<br>A   &#124;B | CPU 922 | CPU 928<br>A   &#124;B | |
| Block number | FB 180 | FB 180 | FB 180 | FB 180 | FB 180 | FB 180 | FB 180 |
| Block name | PER:BS | PER:BS | PER:BS | PER:BS | PER:BS | PER:BS | PER:BS |
| Library number (P71200–S ...) | –5180–A–1 | –5180–A–1 | –5180–A–1 | –5180–A–1 | –9180–A–1 | –9180–A–1 | –6180–B–1 |
| Call length (in words) | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| Block length (in words) | 466 | 466 | 466 | 466 | 364 | 364 | 327 |
| Nesting depth | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Run times (in msec) | 52.9   14.3<br>(18.9) (3.9) | 37.1   14.3<br>(9.5) (3.9) | 18.8   13.8<br>(4.9) (3.7) | 4.1   1.7<br>(3.0) (0.7) | 10.1<br>(3.6) | 5.4   4.8<br>(2.1) (1.5) | 1.4<br>(0.7) |
| Assignment in the data area | Parameterization data block up to and including DW 119 (current data block open prior to the call of the function block) | | | | | | |
| Assignment in the flag area [1] | from FY200 to FY255 | from FY200 to FY255 | from FY200 to FY255 | from FY200 to FY255 | from FY232 to FY255 | from FY232 to FY255 | from FY232 to FY255 |

[1]  The flags are used as intermediate storage.

The run time given in the table corresponds to the maximum time (when, for example, all counters and the global registers are parameterized at the same time). The minimum time is given in parentheses (only one counter parameterized).

**Function Block FB 181** (BL: Read Module)

| | Programmable Controller | | | | | | |
|---|---|---|---|---|---|---|---|
| | **S5–115U** | | | | **S5–135U** | | **S5–155U** |
| | CPU 941<br>A ⎮ B | CPU 942<br>A ⎮ B | CPU 943<br>A ⎮ B | CPU 944<br>A ⎮ B | CPU 922 | CPU 928<br>A ⎮ B | |
| Block number | FB 181 | FB 181 | FB 181 | FB 181 | FB 181 | FB 181 | FB 181 |
| Block name | PER:BL | PER:BL | PER:BL | PER:BL | PER:BL | PER:BL | PER:BL |
| Library number (P71200–S ...) | –5181–A–1 | –5181–A–1 | –5181–A–1 | –5181–A–1 | –9181–A–1 | –9181–A–1 | –6181–B–1 |
| Call length (in words) | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| Block length (in words) | 247 | 247 | 247 | 247 | 224 | 224 | 183 |
| Nesting depth | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Run times (in msec) | 17.2 ⎮ 4.0<br>(12.7) ⎮ (2.5) | 9.6 ⎮ 4.0<br>(5.4) ⎮ (2.5) | 6.4 ⎮ 3.9<br>(3.2) ⎮ (2.3) | 3.1 ⎮ 0.7<br>(2.7) ⎮ (0.4) | 4.2<br>(2.8) | 2.3 ⎮ 1.4<br>(1.4) ⎮ (1.0) | 0.7<br>(0.6) |
| Assignment in the data area | Parameterization data block up to and including DW 119<br>(current data block open prior to the call of the function block) | | | | | | |
| Assignment in the flag area    1) | from FY236 to FY255 | from FY236 to FY255 | from FY236 to FY255 | from FY236 to FY255 | from FY232 to FY255 | from FY232 to FY255 | from FY232 to FY255 |

1 The flags are used as intermediate storage.

The run time given in the table corresponds to the maximum time (when, for example, all counters and the global registers are parameterized at the same time). The minimum time is given in parentheses (only one counter parameterized).

**Function Block FB 182** (IN: Acknowledge Interrupt)

| | Programmable Controller | | | | | | |
|---|---|---|---|---|---|---|---|
| | **S5–115U** | | | | **S5–135U** | | **S5–155U** |
| | CPU 941<br>A \| B | CPU 942<br>A \| B | CPU 943<br>A \| B | CPU 944<br>A \| B | CPU 922 | CPU 928<br>A \| B | |
| Block number | FB 182 | FB 182 | FB 182 | FB 182 | FB 182 | FB 182 | FB 182 |
| Block name | PER:IN | PER:IN | PER:IN | PER:IN | PER:IN | PER:IN | PER:IN |
| Library number<br>(P71200–S ...) | –5182–A–1 | –5182–A–1 | –5182–A–1 | –5182–A–1 | –9182–A–1 | –9182–A–1 | –6182–B–1 |
| Call length<br>(in words) | 6 | 6 | 6 | 6 | 6 | 6 | 7 |
| Block length<br>(in words) | 203 | 203 | 203 | 203 | 154 | 154 | 149 |
| Nesting depth | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Run times<br>(in msec) | 14.0 \| 3.3 | 5.7 \| 3.3 | 4.0 \| 3.1 | 2.7 \| 0.6 | 2.7 | 1.3 \| 1.1 | 0.7 |
| Assignment in the<br>data area | (No data block is used.) | | | | | | |
| Assignment in the<br>flag area    1) | from FY236<br>to FY255 | from FY236<br>to FY255 | from FY236<br>to FY255 | from FY236<br>to FY255 | from FY232<br>to FY255 | from FY232<br>to FY255 | from FY232<br>to FY255 |

1 The flags are used as intermediate storage.

# 14 Programming Instructions, FB 183/184

**IP 242B**

| **14.1** | **Overview** |
|---|---|

**B**

These programming instructions describe the following two function blocks.

**FB 183 (ZYK:242B)**      "Control IP 242B counter module"
(with page frame addressing)

**FB 184 (INT:242B)**      "Process interrupts"
(with page frame addressing)

The function blocks are used with the module

**IP 242B counter module**

in the following programmable controllers:

| **FB 183** | **FB 184** | **PLC/CPU** |
|:---:|:---:|---|
| X | X | **S5–115U**   (CPU 941A/B to CPU 944A/B) |
| X | **X** | **S5–135U**   (CPU 922, CPU 928A/B) |
| X | X | **S5–155U** |

These programming instructions assume knowledge of sections 1 to 10, and the operating instructions of your programmable controller.

The example in section 15 gives a test setup with the IP 242B counter module which provides an easy method of testing the jumper assignments and functions. This test program can also be used as the basis of an automation task to be implemented.

The files of the function blocks with example and commentary blocks in English and French for the respective programmable controller are included on the S5 DOS floppy disk.

| | File | | |
|---|---|---|---|
| | Function Block | Commentary Bock | |
| PLC S5– | German | English | French |
| | S5LxxxST.S5D | ECLxxxST.S5D | FCLxxxST.S5D |
| 115U | S5LD50ST.S5D | ECLD50ST.S5D | FCLD50ST.S5D |
| 135U | S5LD24ST.S5D | ECLD24ST.S5D | FCLD24ST.S5D |
| 155U[1] | S5LD69ST.S5D | ECLD69ST.S5D | FCLD69ST.S5D |

Copy the commentary blocks in the S5LxxxST.S5D file to obtain the commentary in your language when the example is printed out.

The corresponding title block files are listed below:

S5LxxxF1.INI
ECLxxxF1.INI
FCLxxxF1.INI

[1]   The xxLD24ST.S5D files must be used when a CPU 922 or a 928A/B is used in PLC S5–155U.

## 14.2    Function Description

Two standard function blocks are available for processing the IP 242B counter module.

FB 183 (ZYK:242B)    For controlling the IP 242B counter module with page frame addressing
FB 184 (INT:242B)     For processing interrupts with page frame addressing

The following functions can be executed with the "control counter module" function block.

–   Parameterize the counter

–   Load, start and read the counter

The "process interrupts" function block can be used to process interrupts and process–interrupts.

Function block FB 183 is usually called in the startup program (organization blocks OBs 20,21 and 22) to parameterize the counter module.

Counter module control (e.g., start counter or read actual values) then takes place during the cyclic program (organization block OB 1).

Function block FB 184 is called in an organization block of the interrupt–controlled processing (OB 2 to OB 9 depending on the programmable controller) to evaluate process interrupts or interrupts.

> Standard function block FB 183 may only be called in the time–controlled program processing if it is not called in cyclic program processing and if the interrupt priority of the timed interrupts is lower than that of the process interrupts or interrupts.
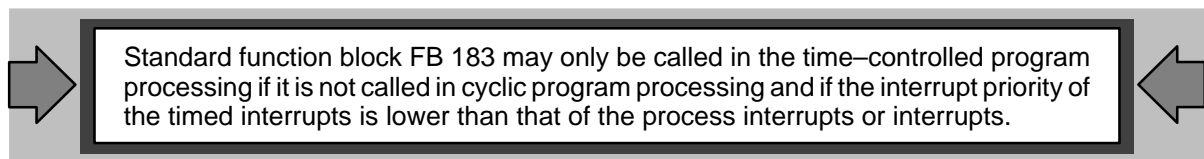
**Functions**

–   Controlling the IP 242B counter module

–   Processing interrupts

## 14.3    Calls of Function Block FB 183 and FB 184

```
        JU FB183                Parameter
NAME  :ZYK:242B                Class Type

SSNR  :                         D    KY
BEF   :                         D    KS
PAR   :                         D    KM
STEU  :                         I    W
DBNR  :                         D    KY
ERR   :                         Q    W
MELD  :                         Q    BY
F–NR  :                         Q    BY
```

| Parameter Class | Type | | **FB 183** | | | Parameter Class | Type |
|---|---|---|---|---|---|---|---|
| | | | ZYK:242B | | | | |
| D | KY | | SSNR | | | | |
| D | KS | | BEF | ERR | | Q | W |
| D | KM | | PAR | MELD | | Q | BY |
| I | W | | STEU | F–NR | | Q | BY |
| D | KY | | DBNR | | | | |
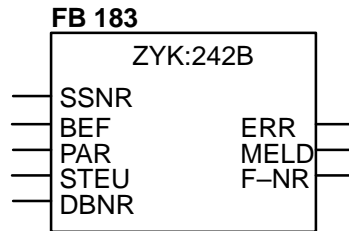
```
        JU FB184                Parameter
NAME  :INT:242B                Class Type

SSNR  :                         D    KY
FKT   :                         D    KM
PAR   :                         D    KM
DBNR  :                         D    KY
ABIT  :                         D    KY      (*)
IIR   :                         Q    W
ERR   :                         Q    W
MELD  :                         Q    BY
F–NR  :                         Q    BY
```
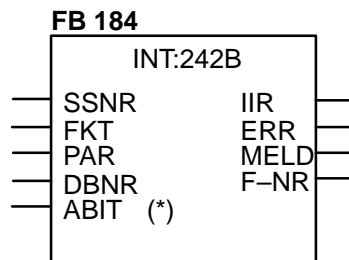
| Parameter Class | Type | | **FB 184** | | | Parameter Class | Type |
|---|---|---|---|---|---|---|---|
| | | | INT:242B | | | | |
| D | KY | | SSNR | IIR | | Q | W |
| D | KM | | FKT | ERR | | Q | W |
| D | KM | | PAR | MELD | | Q | BY |
| D | KY | | DBNR | F–NR | | Q | BY |
| D | KY | | ABIT  (*) | | | | |

(*)   The ABIT parameter is only available with programmable controller S5–155U.

## 14.4    Explanation of the Parameters

| NAME | Class | TYPE | FB 183 | FB 184 | DESIGNATION |
|------|-------|------|--------|--------|-------------|
| SSNR | D | KY | x | x | Interface number for page frame addressing |
| FKT | D | KM | – | x | Function to be executed by FB 184 (interrupt processing) |
| BEF | D | KS | x | – | Command; specification of the control word to be executed by FB 183 (direct parameterization) |
| PAR | D | KM | x | x | Parameter; for example, specification of the counters to be controlled simultaneously (direct parameterization) |
| STEU | I | W | x | – | Control word; specification of the function to be executed by function block FB 183, and the corresponding parameter (indirect parameterization) |
| DBNR | D | KY | x | x | Number of the parameterization data block |
| ABIT (*) | D | KY | – | x | Number of the interrupt bit for process interrupt processing |
| IIR | Q | W | – | x | Interrupt information register |
| ERR | Q | W | x | x | Error information register |
| MELD | Q | BY | x | x | Message byte |
| F–NR | Q | BY | x | x | Error number |

x:   Parameter available,   –: Parameter not available

(*)  The ABIT parameter is only available with programmable controller S5–155U.

> **!** Parameters STEU, IIR, ERR, MELD and F–NR may not be assigned with scratchpad flags FY 200 to FY 255 or data words/data bytes. **!**

## 14.5    Assignment of the Parameters
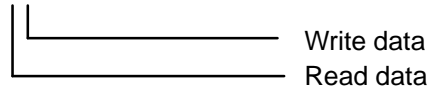
**SSNR   :    Interface Number for Page Frame Addressing**

KY = x, 0 to 255          Page frame address which you set on switch S4
                          (x = disregard)

**FKT     :    Function of Function Block FB 184**

This parameter is used to specify processing of function block FB 184.

KM = 0000 0000 0000 00XX

—————— Write data
—————— Read data

Bit 0                 Transfer new parameterization data to the module ( ≙ write data) (i.e.
                      setting of bit 0 causes the RS command to be triggered at interrupt).
                      The transferred data must be activated with a parameterization com-
                      mand (e.g., PO) so they they take effect. Together with the counter bits
                      or the bit for the global register, the PO command must be entered in
                      the appropriate interrupt command lists on the module.

                      Interrupt Procedure:

                      1. Command list is executed on the counter module.

                      2. Interrupt is sent to the PLC.

                      3. In the interrupt branch, "bit 0 = 1" causes new parameterization data
                         to be transferred to the counter module.

                      4. New parameterization data are availlable when the next interrupt
                         occurs.

Bit 1                 Read data from the counter module
                      If this bit is set, all counter value registers, the counter status registers
                      and all result registers are read from the module. The PAR parameter is
                      not evaluated.

Bit 2 to 15           These bits are not evaluated.

When several functions of the function block are selected by setting several bits in the PAR parameter,
these are processed in the following order:

First, the parameterization data are transferred to the module. The counter value registers, the counter
status registers and the result registers are then read. Last, the interrupt is acknowledged.

The FKT parameter is not checked for plausibility.

FB 184 may only be called once in the process interrupt–interrupt program.

### BEF : Command for Controlling the Module (Direct Parameterization)

The command is specified in KS format.

See the table below for a list of possible commands.

The BEF parameter is assigned with KC = XX to switch to indirect parameterization with the STEU parameter.

See section 14.8 (use of function block FB 183) for the meaning of the individual commands.

**Table of Possible Commands of FB 183:**

| Meaning | BEF KC = | Direct Parameterization Parameter PAR KM = | Indirect Parameterization Control word STEU KH = | Permitted for S5 | Permitted for Command List |
|---|---|---|---|---|---|
| Reset module | RB* | – | 0100 | • | – |
| Disable the outputs | SA | – | 0200 | • | • |
| Enable the outputs | FA | – | 0300 | • | • |
| Mask interrupt | IM | – | 0400 | • | • |
| Write constant register | KS | – | 0500 | • | – |
| Update counter values | ZA | – | 0600 | • | – |
| Step counter | SZ | Counters 1 to 5 | 15xx | • | • |
| Save counter | SV | Counters 1 to 5 | 16xx | • | • |
| Copy counter | CO | Counters 1 to 5 | 17xx | • | • |
| Prepare for load | LV | Counters 1 to 5 | 18xx | • | • |
| Load and start counter | LS | Counters 1 to 7 | 31xx | • | • |
| Read counter | LE | Counters 1 to 7 | 32xx | • | • |
| Reset counter | RZ | Counters 1 to 7 | 33xx | • | • |
| Accept interrupt value | AW | Counters 1 to 7 | 34xx | • | • |
| Start counter | ST | Counters 1 to 7 | 36xx | • | • |
| Load counter | LD | Counters 1 to 7 | 37xx | • | • |
| Stop counter | SP | Counters 1 to 7 | 38xx | • | • |
| Stop and read counter | SL | Counters 1 to 7 | 39xx | • | • |
| Parameterize counter | PA* | Counters 1 to 7, global register | 71xx | • | – |
| Store parameter | PS* | Counters 1 to 7, global register | 72xx | • | – |
| Rewrite parameter | PZ | Counters 1 to 7, global register | 73xx | • | – |
| Accept basic setting | GR | Counters 1 to 7, global register | 74xx | • | – |
| Parameterize counter (without command list) | PO | Counters 1 to 7, global register | 75xx | • | • |
| Read register | RL | Counters 1 to 7, global register | 76xx | • | – |
| Write register | RS | Counters 1 to 7, global register | 77xx | • | – |
| Execute test function | TF* | Test functions | 81xx | • | – |
| Process command list | BB | Additional command list 1 to 7 | 82xx | • | • |
| Read meas. value series | ML | Meas. value series 1 to 7, meas. value directory | 83xx | • | – |
| Read and reset meas. value series | MR | Meas. value series 1 to 7, meas. value directory | 84xx | • | – |
| Switch over to STEU | XX | – | – | – | – |

See parameter PAR, bits 0 to 7

*) These commands may take longer than 2 msec to process on the module. If this happens, bits 1, 6, and 7 of the MELD parameter are set and processing of the module is concluded.

### PAR : Parameter

Parameter for the command (righthand byte of the control word)
Example: specification of the counter to be controlled or the test function

Assignment tor the Counters:

KM = 0000 0000 xxxx xxxx
└─ Global register
└─ Counter 1
└─ Counter 7

"x = 1" causes the counters to be controlled, the global registers or the test function to be selected. Remember that at least one counter or the global registers and at least one test function must always be specified (otherwise parameterization error). The number of counters to be specified varies (see table).

### STEU : Control Word

> The STEU parameter is not evaluated unless the KC = XX switchover parameter has been specified at the BEF parameter.

The control word be specified completely. For example, it must also contain the counters to be controlled or the test functions. A check for correct parameter assignment is performed if the specified command (lefthand byte of the control word) corresponds to a valid command (see table).

When no corresponding command code is found, the control word is transferred to the module without checking for valid parameter assignment.

### DBNR : Number of the Data Block with the Parameterization Data

KY = **x, y**    **x** = 0; **y** = 0 [*)]    : The currently opened data block is used as the parameterization data block.

**x** = 0    : Data block DB
**x** = 1    : Data block DX (only for PLC S5–135U with CPU 922/CPU 928A/B and PLC S5–155U)

**y** = 10 to 255    : Number of the data block

In all other cases, an error number is output via the F–NR parameter.

> *) The combination KY = 0.0 cannot be used with PLC S5–115U. It will cause an error message in the F–NR parameter.

> The indicator to the measured value data block is stored in DW 255 of the para–meterization data block.

### ABIT    :    Number of the Interrupt Bit

Only relevant for process interrupt processing with PLC S5–155U in 150U mode.

| | | |
|---|---|---|
| KY = **x, y** | **x** = | Enable for reset |
| | **x** >< 0 | No reset |
| | **x** = 0 | Reset applicable interrupt bit in the system data |
| | **y**= | Number of the interrupt bit |
| | $0 \leq \mathbf{y} \leq 7$ | |

### IIR    :    Interrupt Information Register

The contents of the interrupt information register are output at the IIR parametr when interrupt or process interrupt processing occurs ($\rightarrow$ section 3.8.2).

### ERR    :    Error Information Register

The group error bit (bit 7) is set in the MELD parameter each time an error occurs.

The ERR parameter contains the contents of the error information register of the counter module ($\rightarrow$ section 3.8.3).

The value zero at the ERR parameter means that no error has occurred.

When an error is determined (ERR parameter >< zero), **bit 5** of the **MELD** parameter is set to signal status **"1"**.

The assignment of the ERR parameter occurs at two different times: before a control word is written and after the control word is written.

The assignment of the ERR parameter **after writing** a control word has in all probability been caused by an error connected with writing the control word. Processing of the control word in which the error occurred is terminated. **Bit 3** of the **MELD** parameter also has signal status **"1"** in addition to **bits 5** and **7**.

The assignment of the ERR parameter **before writing** the control word has probably been triggered by a module malfunction (e.g., short circuit at the outputs or computer error during processing of a command list) which caused an error number to be set. **Bit 2** in the **MELD** parameter is then also assigned with **"1"** in addition to **bits 5** and **7**, and the control word is not transferred.

If applicable, the lefthand byte of the ERR parameter contains the data word number of the command of a command list which caused the error ($\rightarrow$ section 3.8.4).

**MELD  :   Message Byte**

The function stores the following messages in this byte:

Bit 0 ⎫
Bit 1 ⎬  Assignment depends on bits 5 and 6 (see table).
Bit 2 ⎪
Bit 3 ⎭
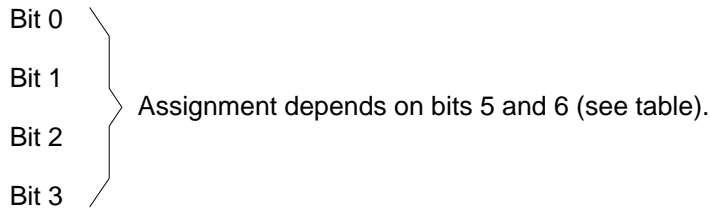
Bit 4   Not used

Bit 5   **Module error**. Has signal status "1" when a module error has occurred

Bit 6   **Repeat command**. Has signal status "1" when the specified command could not be transferred to the module

Bit 7   **Group error bit**. Has signal status "1" when an error or a message occurred (the ERR, MELD or F–NR parameters have a value not equal to zero)

**Evaluation Principle:**

Bit 7 = 1          An error has occurred.

Bit 6 to 0 must be evaluated to locate the error.

– Bits 6 to 0 = 0   See F–NR parameter for more information.

– Bit 6 = 1          See bits 3 to 0 for more information.

| Bit 3 2 1 0 | Message |
|---|---|
| 0 0 0 1 | No access to dual port RAM in multi–processor operation |
| 0 0 1 0 | Module is busy. |

– Bit 5 = 1          See bits 3 to 0 for more information.

| Bit 3 2 1 0 | Message | |
|---|---|---|
| 0 0 0 1 | Watchdog monitoring | |
| 0 0 1 0 | Loop monitoring | |
| 0 1 0 0 | Error before a control word is written | ⎫ 1) |
| 1 0 0 0 | Error after a control word is written | ⎭ |

1) The type of error is stored in the ERR parameter

### F–NR : Error Number

The F–NR parameter is assigned with an error number when an error is caused by illegal parameterization or incorrect parameterization.

| KH = | Error |
|------|-------|
| 01 | BEF parameter assignment is inccorect. |
| 02 | PAR parameter assignment is incorrect. |
| 03 | STEU parameter assignment is incorrect. |
| 04 | DBNR parameter assignment is incorrect. |
| 05 | Parameterization data block does not exist. |
| 06 | Parameterization data block is too short |
| 07 | ABIT parameter assignment is incorrect. |
| 08 | Module identifier is incorrect (bit pattern on the page frame). |
| 09 | Acknowledgement delay |
| 0B | CPU identifier or firmware status is incorrect (for PLC S5–135U) |
| 0E | Release status or the FB and the IP 242B firmware are not compatible. |
| 0F | Measured value data block does not exist. |
| 10 | Measured value data block is too short. |
| 11 | Error in the directory of the measurement value memory |

Assignment of the F–NR parameter with KH = 00 means that no error has occurred during processing of the function block.

## 14.6    Assignment of the Data Area

The parameterization data block specified at the DBNR parameter is assigned with counter data as shown below.

Depending on the assignment of the BEF and PAR or STEU parameters, function block FB 183 transfers data between the parameterization data block or measured value data block, and the counter module. FB 184 transfers data in accordance with the assignment of the FKT parameter.

This requires that the parameterization data block and the measured value data block be present in their full length (is checked even when not all counters are used).

The data area can be located in a DB data block or in a DX data block (DX only for programmable controllers S5–135U and S5–155U).

It is also possible to use the currently active data block as the parameterization data block (see assignment of the DBNR parameter).

## 14.6.1  Assignment of the Parameterization Data Block

**Overview**

| From DW | Assignment |
|---------|------------|
| 0 | Global register |
| 16 | Parameterization data, counter 1 |
| 30 | Parameterization data, counter 2 |
| 44 | Parameterization data, counter 3 |
| 58 | Parameterization data, counter 4 |
| 72 | Parameterization data, counter 5 |
| 86 | Parameterization data, counter 6 |
| 103 | Parameterization data, counter 7 |
| 120 | Counter status register |
| 123 | Counter value register |
| 132 | Calculation register |
| 178 | Additional command lists |
| 247 | Directory of the measurd value memory |
| 255 | Indicator to the measured value data block |

## Assignment the Global registers

| Bit No. | | 15 ... | ... 0 | Data Format |
|---|---|---|---|---|
| DW | 0 | – | | |
| DW | 1 | – | | |
| DW | 2 | – | | |
| DW | 3 | – | | |
| DW | 4 | Master mode register | MMR | KM |
| DW | 5 | Prescaler register | VTR | KM |
| DW | 6 | Gate control register | TSR | KM |
| DW | 7 | Interrupt enable register | IFR | KM |
| DW | 8 | Interrupt polarity register | IPR | KM |
| DW | 9 | | | KH |
| DW | 10 | == Reserved == | | KH |
| DW | 11 | Version number register | VNR | KH |
| DW | 12 | FB version identifier | FBV | KH |
| DW | 13 | Data record number, EEPROM (0 to 7) | DSN | KH |
| DW | 14 | Interrupt filter register | AFR | KM |
| DW | 15 | Pulse duration register | PDR | KT |

**Parameterization Data of a Counter** *(1 to 5)***:**

| Bit No. | | 15 ... ... 0 | Data Format |
|---|---|---|---|
| DW | n | Counter mode register                               CMR | KM |
| DW | n+1 | Load register                                      LR | KH/KF |
| DW | n+2 | Hold register                                      HR | KH/KF |
| DW | n+3 | Interrupt register [1)]                             AR | KH/KF |
| DW | n+4 | Command 1 for interrupt from counter gate | KH/KM |
| DW | n+5 | Command 2 for interrupt from counter gate | KH/KM |
| DW | n+6 | Command 3 for interrupt from counter gate | KH/KM |
| DW | n+7 | Command 4 for interrupt from counter gate | KH/KM |
| DW | n+8 | Command 5 for interrupt from counter gate | KH/KM |
| DW | n+9 | Command 1 for interrupt from counter output | KH/KM |
| DW | n+10 | Command 2 for interrupt from counter output | KH/KM |
| DW | n+11 | Command 3 for interrupt from counter output | KH/KM |
| DW | n+12 | Command 4 for interrupt from counter output | KH/KM |
| DW | n+13 | Command 5 for interrupt from counter output | KH/KM |

1) The interrupt value "0" is not permitted ($\rightarrow$ section 1.9).

Counter 1: n = 16
Counter 2: n = 30
Counter 3: n = 44
Counter 4: n = 58
Counter 5: n = 72

## Parameterization Data of a Counter *(6 and 7)*:

| Bit No. | | 15 ... ... 0 | Data Format |
|---------|------|--------------------|-------------|
| DW | n | Counter mode register                                    CMR | KM |
| DW | n+1 | (Bits 31 to 24) \| (Bits 23 to 16) | KH |
| DW | n+2 | Load register (bits 15 to 8) \| (Bits 7 to 0)      LR | KH |
| DW | n+3 | (Bits 31 to 24) \| (Bits 23 to 16) | KH |
| DW | n+4 | Hold register (bits 8 to 15) \| (Bits 7 to 0)      HR | KH |
| DW | n+5 | (Bits 31 to 24) \| (Bits 23 to 16) | KH |
| DW | n+6 | Interrupt register (bits 15 to 8) \| (Bits 7 to 0)      AR | KH |
| DW | n+7 | Command 1 for interrupt from inputs SYN & zero marking pulse (simultaneous) | KH/KM |
| DW | n+8 | Command 2 for interrupt from inputs SYN & zero marking pulse (simultaneous) | KH/KM |
| DW | n+9 | Command 3 for interrupt from inputs SYN & zero marking pulse (simultaneous) — Ex: For counter reset (→ section 7.6) | KH/KM |
| DW | n+10 | Command 4 for interrupt from inputs SYN & zero marking pulse (simultaneous) | KH/KM |
| DW | n+11 | Command 5 for interrupt from inputs SYN & zero marking pulse (simultaneous) | KH/KM |
| DW | n+12 | Command 1 for interrupt from counter output | KH/KM |
| DW | n+13 | Command 2 for interrupt from counter output | KH/KM |
| DW | n+14 | Command 3 for interrupt from counter output | KH/KM |
| DW | n+15 | Command 4 for interrupt from counter output | KH/KM |
| DW | n+16 | Command 5 for interrupt from counter output | KH/KM |

Counter 6: n = 86
Counter 7: n = 103

## Assignment of the Counter Status Registers

| Bit No: | | 15 ... | ... 0 | Data Format |
|---|---|---|---|---|
| DW | 120 | Counter status register 1 | ZSR1 | KM |
| DW | 121 | Counter status register 2 | ZSR2 | KM |
| DW | 122 | Counter status register 3 | ZSR3 | KM |

## Assignment of the Counter Value Registers

| Bit No: | | 15 ... | | ... 0 | Data Format |
|---|---|---|---|---|---|
| DW | 123 | Counter value, cyclic C1 | | ZSZ1 | KH/KF |
| DW | 124 | Counter value, cyclic C2 | | ZSZ2 | KH/KF |
| DW | 125 | Counter value, cyclic C3 | | ZSZ3 | KH/KF |
| DW | 126 | Counter value, cyclic C4 | | ZSZ4 | KH/KF |
| DW | 127 | Counter value, cyclic C5 | | ZSZ5 | KH/KF |
| DW | 128 | | (Bits 31 to 16) | | KH |
| DW | 129 | Counter value, cyclic C6 | (Bits 15 to 0) | ZSZ6 | KH |
| DW | 130 | | (Bits 31 to 16) | | KH |
| DW | 131 | Counter value, cyclic C7 | (Bits 15 to 0) | ZSZ7 | KH |

## Assignment of the Calculation Registers

| Bit No: | | 15 ... | ... 0 | Data Format |
|---------|--------|--------|-------|-------------|
| DW | 132 | | (Bits 31 to 16) | KH |
| DW | 133 | Result register 1 | (Bits 15 to 0) ERG1 | KH |
| DW | 134 | | (Bits 31 to 16) | KH |
| DW | 135 | Result register 2 | (Bits 15 to 0) ERG2 | KH |
| DW | 136 | | (Bits 31 to 16) | KH |
| DW | 137 | Result register 3 | (Bits 15 to 0) ERG3 | KH |
| DW | 138 | | (Bits 31 to 16) | KH |
| DW | 139 | Result register 4 | (Bits 15 to 0) ERG4 | KH |
| DW | 140 | | (Bits 31 to 16) | KH |
| DW | 141 | Result register 5 | (Bits 15 to 0) ERG5 | KH |
| DW | 142 | | (Bits 31 to 16) | KH |
| DW | 143 | Result register 6 | (Bits 15 to 0) ERG6 | KH |
| DW | 144 | | (Bits 31 to 16) | KH |
| DW | 145 | Result register 7 | (Bits 15 to 0) ERG7 | KH |
| DW | 146 | | (Bits 31 to 16) | KH |
| DW | 147 | Constant register 0 | (Bits 15 to 0) KON0 | KH |
| DW | 148 | | (Bits 31 to 16) | KH |
| DW | 149 | Constant register 1 | (Bits 15 to 0) KON1 | KH |
| DW | 150 | | (Bits 31 to 16) | KH |
| DW | 151 | Constant register 2 | (Bits 15 to 0) KON2 | KH |
| DW | 152 | | (Bits 31 to 16) | KH |
| DW | 153 | Constant register 3 | (Bits 15 to 0) KON3 | KH |
| DW | 154 | | (Bits 31 to 16) | KH |
| DW | 155 | Constant register 4 | (Bits 15 to 0) KON4 | KH |
| DW | 156 | | (Bits 31 to 16) | KH |
| DW | 157 | Constant register 5 | (Bits 15 to 0) KON5 | KH |
| DW | 158 | | (Bits 31 to 16) | KH |
| DW | 159 | Constant register 6 | (Bits 15 to 0) KON6 | KH |
| DW | 160 | | (Bits 31 to 16) | KH |
| DW | 161 | Constant register 7 | (Bits 15 to 0) KON7 | KH |
| DW | 162 | | (Bits 31 to 16) | KH |
| DW | 163 | Constant register 8 | (Bits 15 to 0) KON8 | KH |
| DW | 164 | | (Bits 31 to 16) | KH |
| DW | 165 | Constant register 9 | (Bits 15 to 0) KON9 | KH |
| DW | 166 | | (Bits 31 to 16) | KH |
| DW | 167 | Constant register 10 | (Bits 15 to 0) KON10 | KH |
| DW | 168 | | (Bits 31 to 16) | KH |
| DW | 169 | Constant register 11 | (Bits 15 to 0) KON11 | KH |
| DW | 170 | | (Bits 31 to 16) | KH |
| DW | 171 | Constant register 12 | (Bits 15 to 0) KON12 | KH |
| DW | 172 | | (Bits 31 to 16) | KH |
| DW | 173 | Constant register 13 | (Bits 15 to 0) KON13 | KH |
| DW | 174 | | (Bits 31 to 16) | KH |
| DW | 175 | Constant register 14 | (Bits 15 to 0) KON14 | KH |
| DW | 176 | | (Bits 31 to 16) | KH |
| DW | 177 | Constant register 15 | (Bits 15 to 0) KON15 | KH |

**Assignment of the Additional Command Lists**

DW 178 to DW 185        Directory for DZB additional command lists

DW 186 to DW 246        ZB additional command lists

Use the DZB to specify the data word (from the range from DW 186 to DW 246) at which a command list begins. See section 3.6.1 for the setup of the DZB.

**Assignment of the Directory for Measured Values**

DW 247 to DW 254        Directory of DM measured value memory

Use the DM to specify the data word in the measured value data block at which a measured value series (block) begins. See section 3.7.1 for the setup of the DM.

**Assignment of the Indicator to the Measured Value Data Block**

DW 255                    Indicator to the measured value DB

| DB type | DB–no./DX no. |
|---------|---------------|

DB type:    0 = DB
            1 = DX

DX no:    Only available for PLCs S5–135U and S5–155U

Store the indicator in KY format in data word 255.

## 14.6.2   Assignment of the Measured Value Data Block

Specify the structure and size of the measured value blocks in the directory of the measured value memory in accordance with your requirements.  Block 1 always starts at DW 16. Each individual measured value occupies two data words.

The measured value data block must be set up to the length specified by you in the directory of the measured value memory ($\rightarrow$ section 3.7.1).

---

## 14.7    Technical Specifications

**Function Block FB 183 (ZYK:242B) for the Control of the Module and
Function Block 184 (INT:242B) for the Interrupt Processing**

| Block number | FB 183 | | | FB 184 | | |
|---|---|---|---|---|---|---|
| Block name | ZYK: 242B | | | INT: 242B | | |
| PLC S5– | 115U | 135U | 155U | 115U | 135U | 155U |
| Library number (P71200–S ...) | –5183–A–1 | –9183–A–1 | –6183–B–1 | –5184–A–1 | –9184–A–1 | –6184–B–1 |
| Call length (in words) | 10 | | | 10 | | 11 |
| Block length (in words) | 1111 | 1293 | 1234 | 375 | 394 | 360 |
| Nesting depth | 0 | | 1[1] | 0 | | 1[1] |
| Assignment in the data area | Parameterization data block up to and including data word DW 255 (specified in the DBNR parameter); measurd value data block up to DW...) | | | | | |
| Assignment in the flag area [2] | FY 212 to FY 255 | | | FY 228 to FY 255 | | |
| Assignment in the system area [3] | – | BS 60 | – | – | | |
| Other | 4) | | | – | | |

1) Special functions of the operating system are called.

2) The flags are only used for intermediate storage. They can be used as desired outside the function block.

3) The system data are only used for intermediate storage. They can be used as desired outside the function block. When the function block is called during interrupt processing, these system data words must be saved and then reloaded (just as the scratchpad flags).

⚠ 4) Interruptions (process interrupts, interrupts and timed interrupts) are disabled in the function block for approximately 1 msec. **All** interruptions are enabled afterwards. ⚠

**Areas of Use**

- PLC S5–115U (CPU 941A/B to CPU 944A/B)
- PLC S5–135U (CPU 922 starting at firmware release 9.0/CPU 928A from ...–3UA12/CPU 928B)
- PLC S5–155U (CPU 946/947)

## Processing Times for FB 183 (ZYK:242B) (in msec)

| PLC–S5 | 115 U | | | | 135 U | | 155 U |
|---|---|---|---|---|---|---|---|
| CPU | 941A<br>941B | 942A<br>942B | 943A<br>943B | 944A<br>944B | 922 | 928A<br>928B | 946/947 |
| **RB** [1] | 108.9<br>17.4 | 19.4<br>17.4 | 16.0<br>13.7 | 8.4<br>6.1 | 17.1 | 9.1<br>4.4 | 4.0 |
| **SA** | 71.6<br>12.4 | 15.3<br>12.4 | 12.9<br>10.7 | 5.0<br>3.4 | 13.6 | 7.0<br>4.3 | 2.0 |
| **FA** | 71.0<br>12.5 | 16.8<br>12.5 | 12.7<br>10.8 | 5.1<br>3.4 | 13.8 | 6.9<br>4.4 | 2.0 |
| **IM** | 74.9<br>14.4 | 18.4<br>14.4 | 14.5<br>12.5 | 5.7<br>3.7 | 16.0 | 8.2<br>5.2 | 2.3 |
| **KS** | 79.7<br>16.5 | 21.2<br>16.5 | 16.9<br>14.3 | 5.8<br>4.3 | 18.7 | 10.2<br>6.2 | 2.4 |
| **ZA** | 37.5<br>9.9 | 12.5<br>9.9 | 9.9<br>8.5 | 2.2<br>1.6 | 10.0 | 4.6<br>3.4 | 1.7 |
| **SZ** [2] | 62.1<br>11.9 | 16.0<br>11.9 | 12.0<br>10.4 | 4.7<br>3.3 | 12.9 | 6.6<br>4.2 | 2.0 |
| **SV** [2] | 41.0<br>10.9 | 12.9<br>10.9 | 10.3<br>8.7 | 2.6<br>1.9 | 10.5 | 4.8<br>3.6 | 1.7 |
| **CO** [2] | 60.3<br>12.0 | 14.6<br>12.0 | 12.0<br>10.0 | 4.9<br>3.1 | 12.5 | 6.2<br>4.4 | 1.7 |
| **LV** | | | | | | | |
| All counters<br>(1 to 5) | 60.9<br>13.1 | 17.5<br>13.1 | 13.7<br>11.4 | 3.2<br>2.3 | 15.1 | 7.4<br>4.7 | 2.2 |
| One counter<br>(from 1 to 5) | 54.5<br>10.9 | 13.8<br>10.9 | 11.2<br>9.5 | 3.1<br>2.3 | 12.8 | 6.2<br>4.0 | 2.0 |
| **LS** | | | | | | | |
| All counters<br>(1 to 7) | 68.7<br>15.7 | 19.7<br>15.7 | 15.9<br>13.4 | 3.7<br>2.8 | 16.5 | 8.3<br>5.5 | 2.5 |
| One counter<br>(from 1 to 7) | 58.1<br>12.1 | 14.1<br>12.1 | 11.5<br>9.7 | 3.4<br>2.5 | 13.3 | 6.8<br>4.2 | 2.1 |
| **LE** [2] | 39.2<br>10.3 | 12.7<br>10.3 | 10.5<br>8.6 | 2.5<br>2.0 | 10.0 | 4.8<br>3.6 | 1.9 |
| **RZ** [2] | 63.9<br>12.6 | 16.2<br>12.6 | 12.4<br>10.3 | 4.5<br>3.1 | 13.1 | 6.8<br>4.4 | 2.0 |
| **AW** | | | | | | | |
| All counters<br>(1 to 7) | 87.9<br>16.4 | 21.4<br>16.4 | 17.3<br>14.4 | 5.4<br>3.8 | 18.7 | 9.9<br>6.1 | 2.5 |
| One counter<br>(from 1 to 7) | 77.4<br>13.4 | 17.1<br>13.4 | 13.1<br>11.1 | 4.9<br>3.6 | 15.5 | 7.8<br>4.8 | 2.2 |
| **ST** [2] | 52.8<br>11.1 | 13.2<br>11.1 | 10.7<br>9.0 | 2.9<br>2.2 | 10.9 | 5.1<br>4.0 | 1.8 |
| **LD** | | | | | | | |
| All counters<br>(1 to 7) | 75.9<br>15.7 | 19.0<br>15.7 | 16.5<br>13.8 | 4.3<br>3.1 | 17.1 | 8.7<br>5.6 | 2.5 |
| One counter<br>(from 1 to 7) | 65.0<br>12.0 | 14.7<br>12.0 | 12.0<br>10.1 | 3.9<br>3.0 | 13.9 | 6.9<br>4.4 | 2.1 |
| **SP** [2] | 48.7<br>11.3 | 13.5<br>11.3 | 11.0<br>9.3 | 3.3<br>2.5 | 11.4 | 5.4<br>4.0 | 1.7 |
| **SL** [2] | 51.6<br>11.3 | 13.8<br>11.3 | 11.4<br>9.5 | 3.7<br>2.6 | 11.6 | 5.7<br>4.1 | 1.8 |

1)
2)  } See end of table for explanations.

| PLC–S5 | 115 U | | | | 135 U | | 155 U |
|---|---|---|---|---|---|---|---|
| CPU | 941A 941B | 942A 942B | 943A 943B | 944A 944B | 922 | 928A 928B | 946/947 |
| **PA** [1) | | | | | | | |
| All counters (1 to 7 + global reg.) | 140.3 40.9 | 51.7 40.9 | 43.1 36.2 | 9.2 6.9 | 46.1 | 24.9 13.6 | 6.6 |
| Only global registers | 107.1 28.7 | 36.6 28.7 | 29.4 24.9 | 8.2 6.2 | 33.0 | 17.8 9.5 | 5.2 |
| **PS** [1) | | | | | | | |
| All counters (1 to 7 + global reg.) | 149.0 41.0 | 51.9 41.0 | 43.3 36.4 | 9.1 7.1 | 46.4 | 25.0 20.1 | 6.1 |
| Only global registers | 117.3 27.0 | 36.9 27.0 | 29.5 25.1 | 8.2 6.4 | 33.2 | 17.9 10.4 | 4.9 |
| **PZ** [1) | | | | | | | |
| All counters (1 to 7 + global reg.) | 137.0 40.2 | 53.6 40.2 | 44.9 37.7 | 9.2 7.0 | 45.0 | 24.7 19.7 | 5.5 |
| Only global registers | 114.6 27.6 | 38.5 27.6 | 31.1 26.2 | 8.6 6.7 | 31.4 | 17.1 11.8 | 3.9 |
| **GR** [1) | | | | | | | |
| All counters (1 to 7 + global reg.) | 139.9 40.4 | 53.7 40.4 | 45.1 37.8 | 9.5 7.1 | 45.0 | 24.8 20.1 | 5.9 |
| Only global registers | 118.5 28.0 | 38.6 28.0 | 31.4 26.4 | 8.7 6.7 | 31.7 | 17.2 12.0 | 4.2 |
| **PO** | | | | | | | |
| All counters (1 to 7 + global reg.) | 81.2 18.8 | 22.9 18.8 | 19.0 15.9 | 5.4 3.9 | 20.0 | 10.5 7.4 | 4.2 |
| Only global registers | 60.7 13.4 | 17.7 13.4 | 13.5 11.5 | 4.5 3.6 | 14.2 | 7.4 4.9 | 2.6 |
| **RL** [1) | | | | | | | |
| All counters (1 to 7 + global reg.) | 121.2 40.3 | 52.5 40.3 | 43.9 36.8 | 8.1 6.2 | 43.5 | 23.8 16.0 | 6.7 |
| Only global registers | 100.9 24.6 | 36.5 24.6 | 29.8 25.1 | 7.3 5.6 | 30.3 | 16.5 10.7 | 4.5 |
| **RS** [1) | | | | | | | |
| All counters (1 to 7 + global reg.) | 131.9 39.4 | 51.2 39.4 | 42.6 35.8 | 8.9 6.5 | 45.5 | 24.5 16.0 | 6.3 |
| Only global registers | 99.7 26.2 | 33.9 26.2 | 28.7 24.3 | 7.7 5.8 | 32.7 | 17.6 10.7 | 4.3 |
| **TF** EPROM | 86.5 16.9 | 19.3 16.9 | 15.8 13.6 | 9.0 6.0 | 16.8 | 9.0 5.7 | 3.7 |
| **BB** [2) | 47.2 11.7 | 14.2 11.7 | 11.7 10.8 | 4.0 2.0 | 12.0 | 6.0 4.2 | 3.3 |

1)
2)  } See end of table for explanations.

| PLC–S5 | 115 U | | | | 135 U | | 155 U |
|---|---|---|---|---|---|---|---|
| CPU | 941A 941B | 942A 942B | 943A 943B | 944A 944B | 922 | 928A 928B | 946/947 |
| **ML/MR** [1] | | | | | | | |
| Only directory | 90.0 15.5 | 20.7 15.5 | 16.0 13.5 | 7.1 5.0 | 29.0 | 10.1 5.8 | 2.8 |
| 1 * 2 DW | 103.2 17.3 | 22.4 17.3 | 17.0 14.5 | 7.4 5.3 | 29.8 | 10.8 6.0 | 2.9 |
| 50 * 2 DW | 117.2 29.3 | 37.2 29.3 | 29.5 25.0 | 8.3 6.0 | 41.6 | 17.0 10.4 | 4.7 |
| 100 * 2 DW | 134.1 40.1 | 50.6 40.1 | 42.2 35.6 | 8.9 6.6 | 52.9 | 23.6 14.4 | 6.7 |

The same run times apply when the counter module is controlled with the STEU parameter (BEF= XX).

1) Commands with run times of more than 100 msec can cause cycle times to be exceeded in the user program (PLC in STOP status). If multiple calls of the FB cannot be avoided (e.g. by loading and starting the counters synchronously), the cycle time must be retriggered (OB 31) when the CPU 941A is used.

2) The command has the same run time regardless of the number of permissible parameters selected.

**Processing Times for FB 184 (INT:242B) (in ms)**

| | Righthand Byte of the Parameter | | PLC S5– | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | 115 U | | | | 135 U | | 155 U |
| Task | FKT | PAR | CPU 941A 941B | CPU 942A 942B | CPU 943A 943B | CPU 944A 944B | CPU 922 | CPU 928A 928B | CPU 946/947 |
| Acknowledge interrupt | 0000 0000 | 0000 0000 | 9.0 3.5 | 3.4 3.5 | 2.7 2.4 | <0.5 <0.5 | 1.8 | 0.9 <0.5 | <0.5 |
| Write data (global register) | 0000 0001 | 0000 0001 | 20.0 6.4 | 7.5 6.4 | 5.2 4.5 | 1.5 1.1 | 4.9 | 2.4 1.4 | 1.4 |
| Write data (Counter 1) | 0000 0001 | 0000 0010 | 20.0 4.5 | 7.0 4.5 | 4.9 4.2 | 1.2 1.0 | 4.6 | 2.3 1.3 | 1.0 |
| Write data (C1 to C7 + global reg.) | 0000 0001 | 1111 1111 | 28.3 9.1 | 12.3 9.1 | 10.3 8.4 | 2.5 2.4 | 7.0 | 3.6 2.7 | 2.2 |
| Read data | 0000 0010 | XXXX XXXX | 17.8 6.1 | 8.5 6.1 | 6.4 5.5 | 0.7 0.5 | 12.3 | 1.9 1.6 | 0.4 |
| Read and write data | 0000 0011 | 1111 1111 | 33.2 12.2 | 17.0 12.2 | 13.5 11.3 | 3.0 2.5 | 17.0 | 4.7 3.4 | 2.3 |

X: Is not evaluated

## 14.8     Use of Function Block FB 183

Function block FB 183 is usually called in the startup program of the programmable controller to parame-
terize the IP 242B counter module.  Before the parameterization (with the BEF = PA parameter or ap-
propriate assignment of the STEU parameter), the data to be transferred must be entered in the parame-
terization data block.  If the measured value series function is used, a second data block (measured value
DB) must be set up to accept the measured values.

> **!** Counters 1 to 5 are in STOP status **after parameterization**, and must be started for opera-
> tion.  Counters 6 and 7 are automatically started after parameterization.  All counter outputs
> are disabled. **!**

Function block FB 183 uses the **BEF** parameter to receive the function to be executed.

The commands from the FB work in both directions between parameterization data block and module or
dual port RAM ($\rightarrow$ section 6.1.1).

The PAR parameter specifies the counters for which this function is to be executed (simultaneously).

The assignment of the error information register can be read at the ERR parameter after the selected
function has been executed.

The following data are read from the module each time the function block is called regardless of the speci-
fied command.

- – Counter status register
- – Counter value of counters 1 to 7
- – Result registers 1 to 7.

**RB     Reset module**

The "reset module" ( KH = 0100) control word is transferred.

Processing on the counter module for this command usually takes longer than approximately
2 msec.  Bits 1, 6 and 7 of the MELD parameter are set and the block is exited without reading the
error information register.  This is performed the next time the function block is called (with any
command).

The counter values, the counter status registers and the result register have the value 0 after the
"reset module" command.  These values are set to zero in the parameterization data block.  All
other values are not changed.

**SA     Disable the outputs**

The "disable the outputs" (KH = 0200) control word is transferred.

**FA     Enable the outputs**

The "enable the outputs"  (KH = 0300) control word is transferred.

**IM      Mask interrupt**

All global registers are transferred from the parameterization data block to the module.  The "mask interrupt" (KH = 0400) control word is then transferred.

**KS      Write constant register**

All constant registers are transferred from the parameterization data block to the module.  The "write constant register" (KH = 0500) is then transferred to the module.  When this command is used in a command list, the constant registers must already be available on the module (implementation with the RS command).

**ZA      Update counter values**

The "update counter values" (KH = 0600) control word is transferred to the module.  The counter bits are not evaluated.  In contrast to the LE command, all counters with an operating mode which reloads from the hold register supply the value of the hold register specified during the programming of the counter as the counter value.  This prevents the internal counter hold register from being overwritten.

**SZ      Step counter**

The "step counter" (KH = 15xx) control word is transferred with the appropriate counter bits (bits 1 to 5).

**SV      Save counter**

The "save counter" (KH = 16xx) control word is transferred with the appropriate counter bits (bits 1 to 5).  This stores the current counter value in the internal hold register. The internal hold registers can be read out with the "copy counter" command.

**CO      Copy counter**

"Copy counter" can only be executed under the following conditions.
– SV was called at least once before.
– One of operating modes N, O, Q, R or X is activated. (In these operating modes, the active gate edge causes an SV on the hardware side.)

The "copy counter" (KH = 17xx control word is transferred with the appropriate counter bits (bits 1 to 5) (entry in the ZSZ).  The counter value registers of the selected counters are then read from the module and entered in the parameterization data block.

**LV      Prepare to load**

The load and hold registers of the selected counters are transferred from the parameterization data block to the module, but the counters themselves are not yet loaded.  The "prepare to load" (KH = 18xx) control word is then transferred with the appropriate counter bits (bits 1 to 5).
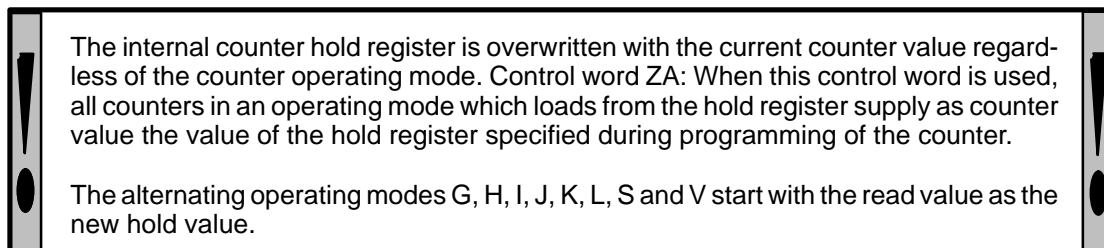
Advantage:  The counter processes the "last" value in this register and accepts (internally in the module) the values transferred by LV as soon as the next start occurs (e.g., gate or TC).
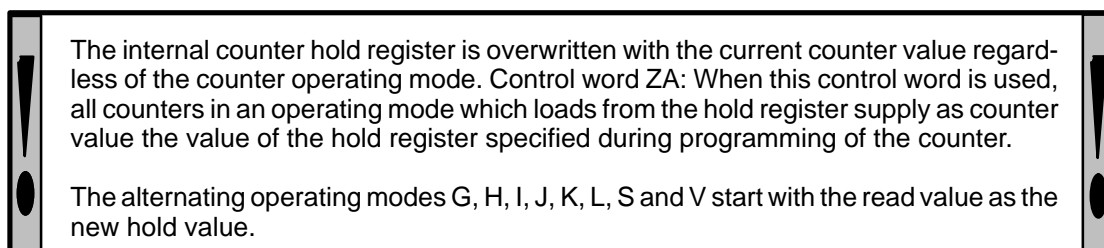
**LS      Load and start counter**

All counter registers of the selected counters are transferred from the parameterization data block to the module.  The "load and start counter" (KH = 31xx) control word is then transferred with the appropriate counter bits (bits 1 to 7).

**LE    Read counter**

The "read counter" (KH = 32xx) control word is transferred with the appropriate counter bits (bits 1 to 7).

> **!**  The internal counter hold register is overwritten with the current counter value regardless of the counter operating mode. Control word ZA: When this control word is used, all counters in an operating mode which loads from the hold register supply as counter value the value of the hold register specified during programming of the counter.
>
> The alternating operating modes G, H, I, J, K, L, S and V start with the read value as the new hold value.

Since counters 6 and 7 are processed by the firmware in succession when all counters are read, remember that there is a difference of 15 μsec for synchronously running counters.

**RZ    Reset counter**

The "reset counter" (KH = 33xx) control word is transferred with the appropriate counter bits (bits 1 to 7).

**AW    Accept interrupt value**

The interrupt registers of the selected counters are transferred from the parameterization data block to the module. The "accept interrupt value" (KH = 34xx) control word is then transferred with the appropriate counter bits (bits 1 to 7).

Advantage:  Fast reparameterization of the interrupt value even when the counters are running.

**ST    Start counter**

The "start counter" (KH = 36xx) control word is transferred with the appropriate counter bits (bits 1 to 7).

**LD    Load counter**

The load and hold registers of the selected counters are transferred from the parameterization data block to the module. The "load counter" (KH = 37xx) control word is then transferred with the appropriate counter bits (bits 1 to 7).

**SP    Stop counter**

The "stop counter" (KH = 38xx) control word is transferred with the appropriate counter bits (bits 1 to 7).

**SL    Stop and read counter**

The "stop and read counter" (KH = 39xx) control word is transferred with the appropriate counter bits (bits 1 to 7).

> **!**  The internal counter hold register is overwritten with the current counter value regardless of the counter operating mode. Control word ZA: When this control word is used, all counters in an operating mode which loads from the hold register supply as counter value the value of the hold register specified during programming of the counter.
>
> The alternating operating modes G, H, I, J, K, L, S and V start with the read value as the new hold value.

## PA    Parameterize counter

All registers of the selected counters, including the commands for interrupt processing and (if selected) the global registers, constant registers, the additional command lists and the measured value directory, are transferred from the parameterization data block to the module. The "parameterize counter" (KH = 71xx) control word is then transferred with the appropriate counter bits (bits 1 to 7) and the bit for the global registers (bit 0).

Processing on the counter module with this command usually takes longer than approximately 2 msec. Bits 1, 6 and 7 of the MELD parameter are set and the block is exited without reading the error information register. This is performed the next time the function block is called (with any command).

## PS    Store parameter

All registers of the selected counters, including the commands for interrupt processing and (if selected) the global registers, constant registers, the additional command lists and the measured value directory, are transferred from the parameterization data block to the module. The "store parameter" (KH = 72xx) control word is then transferred with the appropriate counter bits (bits 1 to 7) and the bit for the global registers (bit 0).

The number under which the parameter record (data record) is to be stored is specified in data word DW 13 of the parameterization data block.

Processing on the counter module which this command usually takes longer than approximately 2 msec. Bits 1, 6 and 7 of the MELD parameter are set and the block is exited without reading the error information register. This is performed the next time the function block is called (with any command).

> Data word DW 255 with the indicator to the measured value data block is **not** stored.
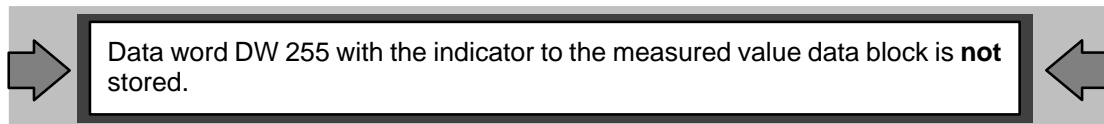
## PZ    Rewrite parameter

The "rewrite parameter" (KH = 73xx) control word is transferred with the appropriate counter bits (bits 1 to 7) and the bit for the global registers (bit 0). All registers of the selected counters, including the commands for interrupt processing and (if selected) the global registers, constant registers, the additional command lists and the measured value directory are then read from the module and entered in the data block.

The number of the data record is specified in data word DW 13 of the parameterization data block.

> The indicator to the measured value data block cannot be intermediately stored (since it is not stored).
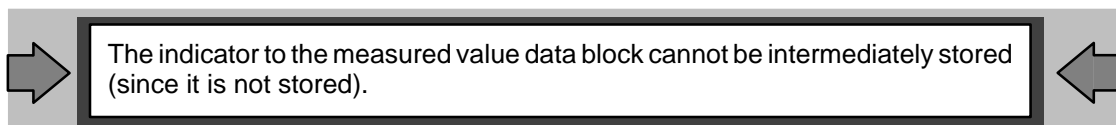
**PS must have been called at least once before PZ is called for the first time.**

**GR**    **Accept basic setting**

The "accept basic setting" (KH = 74xx) control word is transferred with the appropriate counter bits (bits 1 to 7) and the bit for the global registers (bit 0).  All registers of the selected counters and (if selected) the global registers, constant registers, the additional command lists and the measured value directory are then read from the module and entered in the parameterization data block.

**PO**    **Parameterize counter (without command list)**

This command allows you to "reparameterize" the selected counters and the global registers.  First, the counter registers without the command lists and (if selected) the global registers are transferred from the parameterization data block to the module.  The "parameterize counter" (KH = 75xx) control word is then transferred with the appropriate counter bits (bits 1 to 7) and the bit for the global registers (bit 0).

The applicable counter registers:        – Counter mode register
                                         – Hold register
                                         – Load register
                                         – Interrupt register

The applicable global registers:         – Master mode register
                                         – Prescaler register
                                         – Gate control register
                                         – Interrupt enable register
                                         – Interrupt polarity register
                                         – Pulse duration register
                                         – Interrupt filter register

**RL**    **Read register**

The "read register" (KH = 76xx) control word is transferred with the appropriate counter bits (bits 1 to 7) and the bit for the global registers (bit 0).  All registers of the selected counters, including the counter command list and (if selected) the global registers together with the measured value directory, the constant registers and the additional command lists, are then transferred from the module to the parameterization data block.

**RS**    **Write register**

All registers of the selected counters including the counter command list and (if selected) the global registers together with the measured value directory, the constant registers and the additional command lists, are transferred from the parameterization data block to the module.

The "write register" (KH = 77xx)  control word is then transferred with the appropriate counter bits (bits 1 to 7) and the bit for the global registers (bit 0).

This command is used in preparation to transfer data to the page frame (dual port RAM) of the module without reparameterizing the module. This can be performed later with the PO command in a command list, for example.

**TF    Execute test function**

The "execute test function" (KH = 81xx) control word is transferred with the appropriate test function bits.

Processing on the counter module with this command usually takes longer than approximately 2 msec. Bits 1, 6 and 7 of the MELD parameter are set and the block is exited without reading the error information register. This is performed the next time the function block is called (with any command).

**BB    Process command list**

Up to seven additional command lists are available to you in the parameterization data block. Processing of these additional command lists is triggered with the BB command. Which of the additional command lists (bits 1 to 7) is to be processed is specified in the PAR parameter. When several bits are set, the corresponding command lists are processed in succession in ascending numerical sequence.

> A recurrent call of one command list is not permitted.

Since only the "process command list" (KH = 82xx) control word is transferred to the module, the command lists to be processed and any required data must have been transferred to the module before (e.g., with the "write register" control word).

**ML    Read measured value series**

An area of 200 x 2 data words in the measured value data block is available to you for the storage of measured values. This area can be allocated to the seven measured value series as desired. The allocation of the individual sections of the measured value memory to a measured value series is performed in the measured value directory of the parameterization data block.

When data is being supplied by only one result register for the measured value memory, this measured value series can use the entire measured value memory. When data is being supplied by several result registers for the measured value memory, the measured value memory must be divided.

The PAR parameter (bits 1 to 7) determines which of the seven measured value series is to be read. Several measured value series can also be read simultaneously.

The "read measured value series" (KH = 83xx) control word is transferred with the appropriate measured value series bits (bits 1 to 7).

When only bit 0 is set in the PAR parameter, only the measured value directory is read and written in the parameterization data block.

### MR    Read and reset measured value series

The "read and reset measured value series" (KH = 84xx) control word is transferred with the appropriate measured value series bits (bits 1 to 7).  The selected measured value series are then read.

In contrast to the ML command, the MR command resets (deletes) the measured value series which has been read.  The fill status indicator of the measured value series read is set to its initial value.

When only bit 0 is set in the PAR parameter, only the measured value directory is read and written in the parameterization data block, but not reset.

### XX    Switchover parameter to STEU

The control word in the STEU parameter is transferred to the module instead of a command from the BEF parameter with command supplements from the PAR parameter.

The control word must be completely specified (e.g., including  the counter bits). When the specified command (lefthand byte of the control word) corresponds to a command which could also be specified with the BEF parameter, a check for correct parameter assignment is performed by the function block.  Otherwise, transfer to the module is performed without checking the bit pattern.

When the bit pattern corresponds to a command specified via the BEF and PAR parameters, the appropriate data are transferred to the module before the control word is written, or the appropriate data are read from the module after the command is written (depending on the command).

## 14.9    Application of Function Block FB 184

This function block may only be used together with function block FB 183. Function block FB 184 is called in the interrupt or process interrupt program (e. g., OB 2).

The interrupt information register and the error information register are read from the module and transferred to the IIR and ERR parameters. When programmable controller S5–155U is used, the system data bit for interrupt processing is reset if necessary.
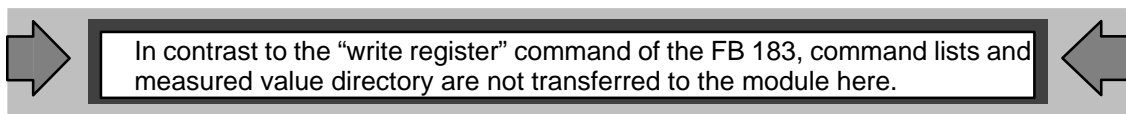
The source of the interrupt can be determined with the IIR parameter.

Depending on the assignment of the **FKT parameter**, the function block also assumes additional functions.

– **Transfer new parameterization data to the module (bit 0 = "1"  ≙  write data)**

This can be used to transfer the counter registers and (if selected) the global registers to the mo–dule.  The "write register" is then transferred to the module with the appropriate counter bits. The data to be transferred are specified in the PAR parameter. The data are fetched from the data block specified in the DBNR parameter.

The transferred data must be activated with a parameterization command (e.g.,  PO) so that they become effective. The PO command for reparameterization of the counters can be located in a command list on the counter module.

> In contrast to the "write register" command of the FB 183, command lists and measured value directory are not transferred to the module here.

– **Read data from the counter module (bit 1 = "1")**

When this bit is set, all counter value registers, the counter status registers and all result registers are read from the module.  The PAR parameter is not evaluated.

The data which have been read are copied to the data block specified in the DBNR parameter.  The data which have been read are copied to the same data words as for the call of function block FB 183.

> When the number of the parameterization data block is specified in the DBNR parameter, the counter status registers, the counter value registers and the re-sult registers of the cyclic program are overwritten when function block FB 184 is called.
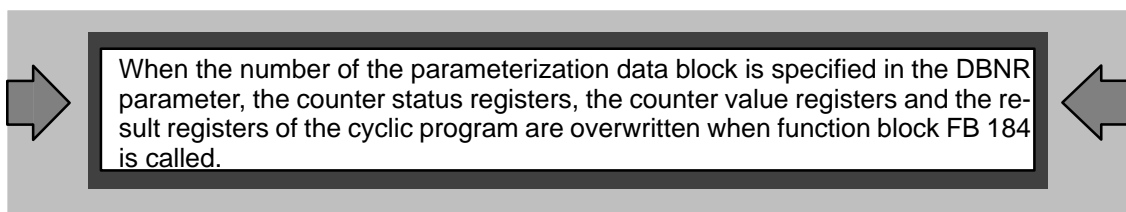
When evaluating data in this situation, remember that they can change within one cycle.  A piece of data scanned at two different points in the program can produce different results.  When this is not desired, another data block must be opened when function block FB 184 is called.

If, during the interrupt or process interrupt program when function block FB 184 is called, the same parameterization data block is used as during the cyclic program, and if the counter values, the counter status register and the result registers are read (bit 1 = "1") during the interrupt or process interrupt program, accesses to double–word data in the cycle must be protected with "disable/enable interrupts".

Example for PLC S5–115U:

|       |                                  |
|-------|----------------------------------|
| AS    | Disable interrupts               |
| A DB x | Open parameterization data block |
| L DW y | Load HIGH WORD                  |
| T .... |                                  |
| L DW z | Load LOW WORD                   |
| T .... |                                  |
| AF    | Enable interrupts                |

When several functions of function block FB 184 are selected by setting bits 0 and 1 in the FKT parameter, these are processed in the following order.

1. Transfer parameterization data to the module

2. Read counter value registers, counter status registers and result registers

3. Acknowledge interrupt (is always executed)

## 14.10 Error Evaluation

Function blocks FB 183 and FB 184 use both the **ERR** parameter and the **MELD** and **F–NR** parameters to report errors which have occurred.

When an error is reported with one of the above stated parameters, the function block sets **bit 7** of the **MELD** parameter to signal status **"1"** (group error bit).

An error coming from the IP 242B counter module (from the error information register) is forwarded with the **ERR parameter**. This register is scanned before a control word is written to the module **and** after the control word is written.

When the error information register is not equal to zero (the counter module reported an error since the last processing) before writing a control word, **bit 5** and **bit 2** of the **MELD** parameter are set to signal status **"1"** (module error before writing a control word) and the control word is not transferred.

When the error information register is assigned with an error number after a control word is written, **bits 5 and 3** of the **MELD** parameter are set to signal status **"1"** (module error after writing a control word).

The **F–NR parameter** contains an error number when an incorrect specification was made for a parameter of function block FB 183 or when an error results from this.

**Evaluation of the MELD Parameter**

The **MELD parameter** is assigned by bit and reports operating states of the function block. The assignment of the individual bits is described below:

**Bits 0 to 3**  These bits have different assignments. They supply additional information for bits 5 and 6..

**Bit 4**  Not assigned

**Bit 5**  Module error

A module error has occurred. The cause can be determined with bits 0 to 3 and the ERR parameter if necessary. The counter module may not be processed again until the cause of the error has been corrected.

**Bit 0 = "1":** Watchdog has been triggered.
The work of the module is monitored with an alternating bit (watchdog). If the bit does not change its signal status within 127 calls of the function block, bit 0 of the MELD parameter is also set in addition to bit 5 and processing of the function block is terminated. Processing is not started again until the signal status of the watchdog bit has changed.

**Bit 1 = "1":** Loop monitor has been triggered.
After the function block has transferred a control word to the module, an acknowledgement from the counter module is waited for for up to 2 msec. Bit 1 is set in addition to bit 5 and the wait loop is exited if the module supplies neither acknowledgement nor an identifier indicating that it is busy with an extra long command. The error information register of the module is not read.

**Bit 2 ="1":** Error information register is assigned before the control word is written.
The error information register of the counter module is read before the control word is written. The module assigns the error information register if it has detected an error since the last call. The exact error can be read in the ERR parameter when bit 5 and bit 2 of the MELD parameter are set. Error sources can include a short circuit on the outputs of the counter module, bad parameterization data of the previous "parameterize module" command, or a calculation error while processing a command list.

**Bit 3 = "1":** Error information register is assigned after the control word is written.
The error information register is read after a control word is written. If bit 5 and bit 3 of the MELD parameter are set, this means in all probability that an error resulted from the control word which was just written. The cause of the error can be determined with the ERR parameter.

**Bit 6**      Repeat command

This bit is set when the function block was not able to process the specified control word within a wait time of 2 msec. The cause can be determined with bits 0 to 3. The function block must be called again when the bit is set.

**Bit 0 = "1":**
When this bit is set in addition to bit 6, the page frame (dual port RAM on the counter module) is busy with another processor (only in multi–processor operation).

**Bit 1 = "1":**
The function block is exited without transferring the specified control word if the module is busy at the moment (e. g., processing a command list or an extra long command). The function block must be called again.

**Bit 7**      This is a group error bit This bit has signal status "1" when one of the parameters ERR, MELD or F–NR is assigned (not equal to zero).

## 14.11    Interrupt Processing
## 14.11.1 General

If, for example, a gate signal or an output signal is to be provided with an interrupt identification (by assignment of the interrupt enable register), the jumpers of the module must be set in one of the following ways.

–    To an interrupt line (S5–115U, S5–135U and S5–155U in 155U mode)

–    To a group interrupt bit of input bytes IB 0(S5–155U in 150U mode)

(See operating instructions of your programmable controller.)

To ensure that the applicable organization block is processed only once (at the rising edge of the process interrupt) when PLC S5–155U in 150U mode is used, the number of this group interrupt bit must be specified at the ABIT parameter:

ABIT   :   KY  =  **x, y**
                 **x** = 255, **y** = 255              No reset (module without process interrupt)
                 **x** = 0, **y** = 0 to 7              Reset the applicable interrupt bit in the system data.

When this bit is not reset in the system data, the applicable organization block will be processed again when the interrupt signal disappears (a process interrupt is evaluated via input byte IB 0 based on the edge when PLC S5–155U in 150U mode is used).

An organization block (depending on the jumper settings) of the interrupt–controlled processing is called when an interrupt occurs.

The signal states of the scratchpad flags must be saved in a data block at the beginning of this organization block, and reloaded again at the end of this block.

Function blocks FB 38 and FB 39 can be used for this purpose together with data block DB 255 from the floppy disk containing standard function blocks FB 183 and FB 184.

The interrupt organization block contains the call of function block FB 184.  After the call, the IRR parameter contains the signal from which the interrupt came.  Your own interrupt program can now be processed.

The module may not be addressed by function block FB 183 during the interrupt program.

A command list can be processed (assignment in the interrupt enable register and the interrupt filter register) when an interrupt occurs on the counter module. The interrupt or process interrupt is not triggered in the programmable controller **until** processing of the command list is complete (i.e., the results of a command list are already available in the interrupt or process interrupt program (e. g., OB 2)).

## 14.11.2 Special Features of PLC S5–115U

An interruption of the user program always occurs at the **instruction boundaries**.

When the IP 242B counter module is used in PLC S5–115U, function blocks FB 38 and FB 39 are used in the interrupt program from the S5LD50ST.S5D file. This also applies to interruptions caused by timed interrupts (see section 16).

Function block FB 38 handles the **saving of user system data** (BS 248 to BS 255), **the current page frame number and the scratchpad flag area** (FY 200 to FY 255). Function block FB 39 handles the loading of the data stored by FB 38.

**In cyclic operation,** function block FB 38 with the "save page frame number" parameterization must be called (e. g., via handling blocks, direct access by the user program or via standard function blocks (FB 184)) before a page frame access is performed if page frame accesses are programmed in the interrupt organization blocks (e.g., by calling FB 184 in OB 2), and if processing with different page frame numbers is used.

**In interrupt organization blocks,** the scratchpad flags must always be saved at the beginning (FB 38), and loaded again at the end (FB 39). **FB 39** also handles the loading of user system data in addition to the loading of the scratchpad flag area, and the provision of the page frame numbers saved by **FB 38** in the cyclic program.

> **!** Function blocks FB 38 and FB 39 must be used **in pairs** in the interrupt organization blocks (i.e., the interrupt organization blocks may not be exited prematurely with the BEB instruction, for example).

## 14.11.3 Special Features of PLC S5–135U

An interruption of the user program occurs at **block boundaries or at instruction boundaries** when data block DX 0 is parameterized appropriately.

When interrupt OBs are programmed in the user program in which the scratchpad flag area (flag bytes FY 200 to FY 255) are also used, be sure that this scratchpad area is saved and reloaded before the interrupt OB is exited. This also applies to the system data (BS 60 to BS 63).

You can use function blocks FB 38 and FB 39, for example, from the S5LD24ST.S5D file of the program example for this purpose.

> **!** Function blocks FB 38 and FB 39 must always be used **in pairs** (i.e., the interrupts organization blocks may not be prematurely exited using the BEB command for example).

## 14.11.4 Special Features of PLC S5–155U

An interruption of the user program occurs at **block boundaries (150U mode) or at instruction boundaries (155U mode)** when data block DX 0 is parameterized appropriately.

When interrupt OBs are programmed in the user program in which the scratchpad flag area (flag bytes FY 200 to FY 255) or the system data (BS 60 to BS 63) are also used, be sure that this flag and system data area is saved and reloaded again before the interrupt OBs are exited.

It is mandatory that function blocks FB 38 and FB 39 from the S5LD69ST.S5D be used to save and load the scratchpad flag area and the system data. The function blocks work together with a data block (DB 255 in our example). This block must be set up up to and including DW 826.

> **!** Function blocks FB 38 and FB 39 must be used **in pairs** (i.e., the interrupt OBs may not be exited prematurely with the BEB instruction, for example). **!**

## 14.12   Startup Behavior
## 14.12.1 Startup Behavior for PLC S5–115U

Cyclic program processing starts at the beginning of OB 1 after the "manual new start" (OB 21) and the "automatic new start" (OB 22).

Enabling of the interrupts during processing of FB 183 permits interrupts or timed interrupts which occur during startup to also be processed with FB 184.

## 14.12.2 Startup Behavior for PLC S5–135U

Cyclic program processing starts at the beginning of OB 1 after a "new start" (OB 20).

Program processing is continued at the point of interruption after the startup OB has been processed during a manual restart (OB 21) or during an automatic restart (OB 22).

**Do not use** a **restart** when using the IP 242B counter module in PLC S5–135U.
The "automatic restart" function must be set in the "automatic new start after power on" function with the aid of DX 0.

When a manual restart is performed twice in a row, this causes erroneous behavior since program processing is continued after the stop command during the second restart (i.e., a transition is made to cyclic operation).

```
Remedy: OB21 : JU FB nn        FB nn :
        :    : Name STOP       Name : STOP
        :    : PE              F001 : STS
                                    : JU = F001
```

## 14.12.3 Startup Behavior for PLC S5–155U

Cyclic program processing starts at the beginning of OB 1 after a "new start" (OB 20).

Program processing is continued at the point of interruption after the startup OB has been processed during a manual restart (OB 21) or during an automatic restart (OB 22).

> **!** If blocks are called in startup organization blocks OB 21 and OB 22 which use the "scratchpad flag area" (flag bytes FY 200 to FY 255), it is mandatory that this flag area be saved and reloaded again (with function blocks FB 38 and FB 39 from the S5LD69ST.S5D file) before the startup OBs are exited. **!**

## 14.13    Multi–Processor Operation

The IP 242B counter module can be used in multi–processor operation.  Function blocks FB 183 and FB 184 for programmable controllers S5–135U and S5–155U are set up to permit this.

If the counter module is to be addressed by several processors simultaneously during multi–processor operation, remember the following points to ensure smooth operation.

Parameterization of the **global registers** should be performed **by only one processor**.

When controlling the counters (e.g., parameterizing, loading and starting), **only one processor** should be used to address **one counter**.

**Reading** from the module can be performed by every processor without restriction.

Process interrupt or interrupt processing can only be performed by **one** processor.

---

# 15    Program Example for IP 242B

<div style="border:1px solid">

## 15.1    General

</div>

**B**

This program example is located on the included floppy disk, and can be loaded completely in the PLC memory to test the module.  The example shows how function blocks FB 183 (ZYK:242B) and FB 184 (INT:242B) can be used.  It also covers direct and indirect parameterization of the module with function block FB 183 (ZYK:242B).  All blocks required for an executable program are included.  The floppy disk also provides a complete "program framework" which you can use.

Direct or indirect parameterization of function block FB 183 (ZYK:242B) is selected via a digital input of the simulator. Individual commands can be transferred to the module with additional digital inputs. Any errors which occur and the counter interrupts are indicated on the digital outputs.

When indirect parameterization of function block FB 183 (ZYK:242B) is used, you can use every command in the program example.  When direct parameterization is used, the following commands are preset.

| Command KC = | Meaning | Parameter PAR |
|---|---|---|
| PA | Parameterize counter | Counters 2 and 6, global registers |
| PO | Parameterize counter (without command lists) | Counters 2 and 6, global registers |
| PS | Store parameter | Counters 1 to 7, global registers |
| PZ | Rewrite parameter | Counters 1 to 7, global registers |
| KS | Write constant registers | – |
| BB | Process command list | Additional command list 1 |
| MR | Read and reset measured value series | Measured value series 1, 2 and 3 |
| ML | Read measured value series | Measure value series 1,2 and 3 |
| LS | Load and start counter | Counters 2, 3, 4 and 6 |
| RZ | Reset Counter | Counters 2 and 6 |
| AW | Accept interrupt value | Counters 2 and 6 |
| SZ | Step counter | Counter 2 |
| ZA | Update counter values | – |

Function block FB 184 (INT:242B) is used with the "acknowledge interrupt" function for interrupt processing.

## 15.2    Device Configuration

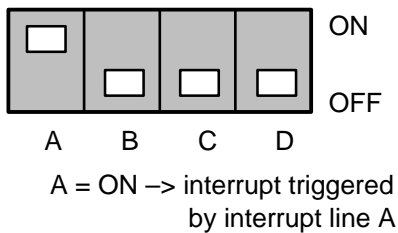The following devices can be used, for example, to run the program example for the IP 242B counter module.

– One S5–115U, S5–135U or S5–155U programmable controller

– PG 730/750/770 programmer

– IP 242B counter module (6ES5 242–1AA41)

– Encoder (5 V) with two pulse trains displaced by 90°

– Digital input module (e.g., 6ES5 430–4UA12)

– Digital output module (e.g., 6ES5 451–4UA12)

– Simulator for digital inputs and outputs (e.g., 6ES5 788–0LA12)

IP 242A/242B Equipment Manual
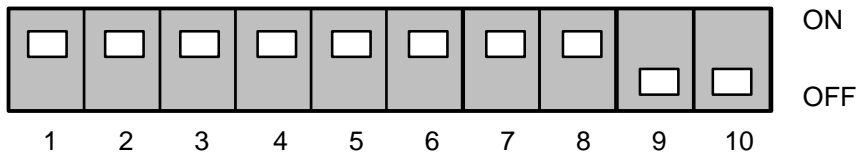© Siemens AG 1993, Order No: 6ES5 998–0KM21
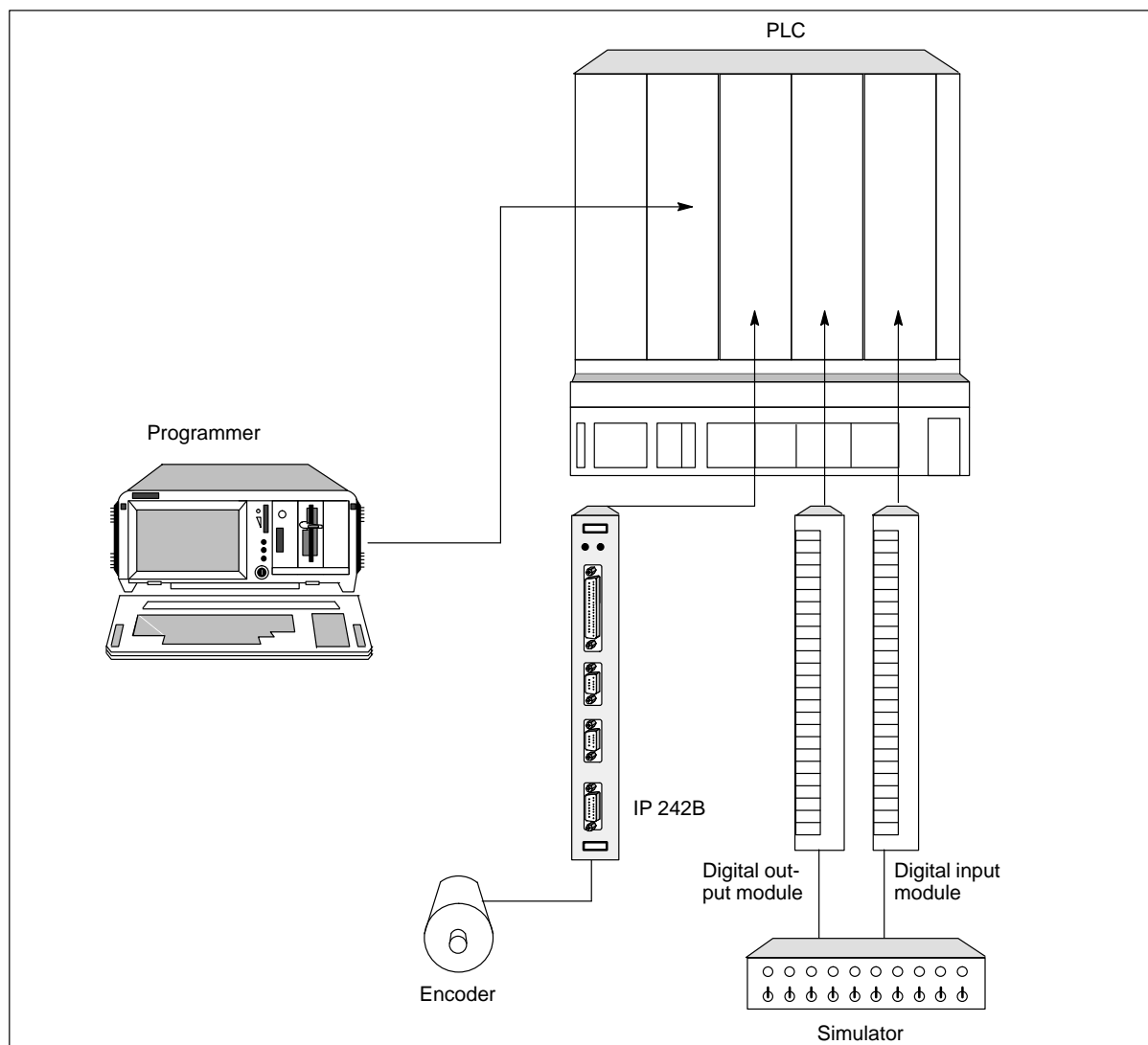
## 15.3    Jumper Assignment of the IP 242B Counter Module

The following switch and jumper settings must be made before the module is installed.  All jumpers and switches which are not mentioned are to remain set as delivered.
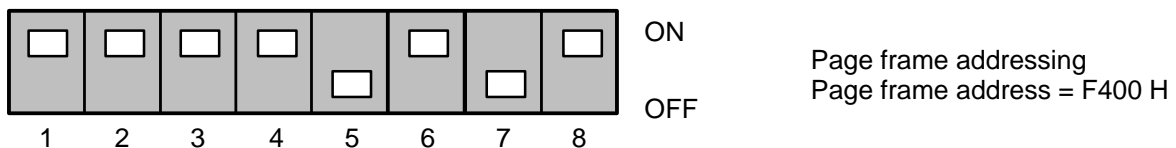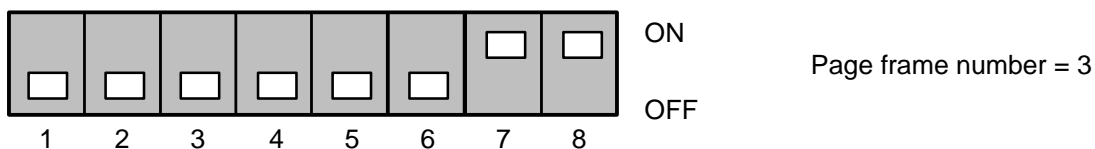
Switch S1 :

A = ON –> interrupt triggered
             by interrupt line A

Switch S2:

S2.10 = OFF –> Process interrupts disabled via input byte IB0

Switch S3:

Page frame addressing
Page frame address = F400 H

Switch S4:

Page frame number = 3

Jumpers 1 and 2:        Open

Jumpers 3 to 17:        Insert in position 2 – 3

The central controller must be operated in 155U mode when the program example is tested in PLC S5–155U.  If not, the settings of switches S1 and S2 specified here must be changed.

## 15.4    Assignment of the Inputs and Outputs

The program is set up for easy adaptation to different input and output assignments.  The program example works with flags only.  The inputs and outputs used are assigned in organization block OB 1 to these flags.  In our example, these are input word IW 4, input byte IB 6, output word QW 8 and output word QW 10.

| Input | Input Image | Corresponding Pulse Flag | Direct Parameterization | Indirect Parameterization |
|---|---|---|---|---|
| I 4.0 | F 4.0 | F 12.0 | PA Parameterize C2+C6+global reg. | Control word, bit 8 |
| I 4.1 | F 4.1 | F 12.1 | PO Parameterize w/o command lists C2+C6+global reg. | Control word, bit 9 |
| I 4.2 | F 4.2 | F 12.2 | KS Write constant register | Control word, bit 10 |
| I 4.3 | F 4.3 | F 12.3 | PS Store parameter C1 to C7+global reg. | Control word, bit 11 |
| I 4.4 | F 4.4 | F 12.4 | BB Process command list, list 1 | Control word, bit 12 |
| I 4.5 | F 4.5 | F 12.5 | – | Control word, bit 13 |
| I 4.6 | F 4.6 | F 12.6 | MR Read and reset measured pulse series Series 1, 2 and 3 | Control word, bit 14 |
| I 4.7 | F 4.7 | F 12.7 | ML Read measured value series 1, 2 and 3 | Control word, bit 15 |
| I 5.0 | F 5.0 | F 13.0 | LS Load and start C2 to C4 | Control word, bit 0 |
| I 5.1 | F 5.1 | F 13.1 | SZ Step C2 | Control word, bit 1 |
| I 5.2 | F 5.2 | F 13.2 | RZ Reset C2 | Control word, bit 2 |
| I 5.3 | F 5.3 | F 13.3 | AW Accept interrupt value, C2 | Control word, bit 3 |
| I 5.4 | F 5.4 | F 13.4 | LS Load and start C6 | Control word, bit 4 |
| I 5.5 | F 5.5 | F 13.5 | RZ Reset C6 | Control word, bit 5 |
| I 5.6 | F 5.6 | F 13.6 | AW Accept interrupt value, C6 | Control word, bit 6 |
| I 5.7 | F 5.7 | F 13.7 | ZA Update counter values | Control word, bit 7 |
| I 6.0 | F 6.0 | F 14.0 | Acknowledge error | Acknowledge error |
| I 6.1 | F 6.1 | F 14.1 | Acknowledge interrupt | Acknowledge interrupt |
| I 6.2 | F 6.2 | F 14.2 | – | Control with IW 4 |
| I 6.3 | F 6.3 | F 14.3 | = 0 direct | = 1 indirect |

| Output | Output Image | |
|---|---|---|
| Q 8.0 | F 16.0 | Error in startup program |
| Q 8.1 | F 16.1 | Error in interrupt program (OB 2) |
| Q 8.2 | F 16.2 | Error in cyclic program (OB 1) |
| Q 8.3 | F 16.3 | Error in FB 38 or FB 39 (for PLC S5–115U only) |
| Q 8.4 | F 16.4 | – |
| Q 8.5 | F 16.5 | Interrupt was triggered. |
| Q 8.6 | F 16.6 | Interrupt at counter 2 |
| Q 8.7 | F 16.7 | Interrupt at counter 6 |

Output byte QB  9      MELD parameter (for error in cyclic program)

Output byte QB  10     F–NR parameter (for error in cyclic program)

## 15.5    Assignment of the Flag Area

| | | |
|---|---|---|
| FW | 4 | Flag of input word IW 4 |
| FY | 6 | Flag of input byte IB 6 |
| FW | 8 | Edge flag of input word IW 4 |
| FY | 10 | Edge flag of input byte IB 6 |
| FW | 12 | Pulse flag of input word IW 4 |
| FY | 14 | Pulse flag of input byte IB 6 |
| FW | 16 | Flag of output word QW 8 |
| FW | 18 | Flag of output word QW 6 |
| | | |
| FW | 20 | Loop counter of startup program |
| | | |
| FW | 30 | ERR parameter,      startup + interrupt program |
| FY | 32 | MELD parameter,   startup + interrupt program |
| FY | 33 | F–NR parameter,    startup + interrupt program |
| | | |
| FW | 34 | ERR parameter      cyclic program |
| FY | 36 | MELD parameter    cyclic program |
| FY | 37 | F–NR parameter,    cyclic program |
| | | |
| FW | 40 | IRR parameter,       interrupt information register |
| | | |
| FW | 52 | Parameterization error in FB 38 (for PLC S5–115U only) |
| FW | 54 | Parameterization error in FB 39 (for PLC S5–115U only) |

## 15.6    Assignment of the Data Area

| | | |
|---|---|---|
| DB | 182 | Measured value data block |
| DB | 183 | Parameterization data block |
| DB | 255 | Intermediate storage for scratchpad flags |
| DX | 0 | Operating system presetting (for PLCs S5–135U and S5–155U only) |

**Block Allocation**

| | | |
|---|---|---|
| FB | 38 | Save scratchpad flag |
| FB | 39 | Load scratchpad flag |
| FB | 82 | Example of indirect parameterization of FB 183 (ZYK:242B) |
| FB | 83 | Example of direct parameterization of FB 183 (ZYK:242B) |
| FB | 183 | Process counter module (cyclic program processing) |
| FB | 184 | Process counter module (interrupt processing) |
| FB | 220 | Is called by OB 20 |
| FB | 221 | Is called by OB 21 |
| FB | 222 | Is called by OB 22 |
| | | |
| OB | 1 | Cyclic program |
| OB | 2 | Interrupt program |
| OB | 20 | New start (not for PLC S5–115U) |
| OB | 21 | Manual restart (new start for PLC S5–115U) |
| OB | 22 | Automatic restart or automatic new start |

## 15.7    Switchon, Startup Behavior

After an overall reset of the programmable controller, the entire file can be loaded in the user memory of the programmable controller. A new start must then be performed.

The counter module is already parameterized in the startup organization blocks. Parameterization data block DB 183 on the floppy disk contains valid parameterization data and requires no modification. The green RUN LED on the counter module goes on continuously to indicate that parameterization was successful.

If all inputs of the simulator were in switch position "0" when the programmable controller was switched on, outputs may not be set after startup of the programmable controller. When output Q 8.0 is set, a parameterization error occurred during startup. The exact cause of the error can then be determined with flag word FW 30 (ERR parameter), flag byte FY 32 (MELD parameter) and flag byte FY 33 (F–NR parameter).

## 15.8    Cyclic Operation

Input I 6.3 can be used to switch between direct and indirect parameterization of function block FB 183 (ZYK:242B).

Function block FB 83 will be called for direct parameterization when input I 6.3 has status "0". Function block FB 82 will be processed for indirect parameterization (controlled operation) when input I 6.3 has signal status "1".

Direct parameterization (I 6.3 = "0") should be selected for the first test since the commands can be triggered with the switches of the simulator.

**Direct Parameterization**

When direct parameterization of function block FB 183 is used, the command to be executed is specified at the BEF parameter in KS format. Selection of the counters or the global registers is performed in the PAR parameter.

The commands programmed in function block FB 83 can be triggered with the switches of the simulator.

**Indirect Parameterization**

When indirect parameterization is used, the command to be executed together with the counter bits or the bit for the global registers is specified at the STEU parameter of function block FB 183.

In our program example, the control word is preset via input word IW 4. When a rising edge occurs at input I 6.2, the command whose control word was specified at input word IW 4 is executed.

**Error Messages**

When an error occurs in the cyclic program, output Q 8.2 is set. The cause of the error can be determined with flag word FW 30 (ERR parameter), flag byte FY 32 (MELD parameter) and flag byte FY 33 (F–NR parameter).

In addition, these flag bytes (FY 32 and FY 33) are imaged on output bytes QB 9 and QB 10.

The error indication is deleted again with input I 6.0.

## 15.8.1  Direct Parameterization

**PS Store parameter (I 4.3)**

This command is used to store the parameterization data of all counters, the global registers, all constant registers, the additional command lists and the measured value directory permanently in the EEPROM of the counter module. In addition, the counter module is parameterized with these data. The parameterization data are located in parameterization data block DB 183. The program example contains a parameterization data block with valid parameterization data.

Since, in our program example, the counter module is parameterized with the data of the EEPROM after return of power, the "store parameter" command should be executed at least once to ensure that valid data are available in the EEPROM.

The counter module offers the capability of storing several data records in the EEPROM. The example works with EEPROM record zero. The value "0" is already entered in data word DW 13 of parameterization data block DB 183 during the startup program.

**PA Parameterize counters 2 and 6 and the global registers (I 4.0)**

This command can be used to parameterize counters 2 and 6 and the global registers. The parameterization data are located in parameterization data block DB 183.

**PO Parameterize counters 2 and 6 and the global registers (I 4.1)**

This command is used to parameterize counter 2, counter 6 and the global registers. The parameterization data are located in parameterization data block DB 183.

In contrast to the PA command, the command lists are not transferred to the module with the PO command. This makes the processing time of the PO command shorter than that of the PA command.

**KS Write constant register (I 4.2)**

This command can be used to transfer the 16 constant registers to the counter module. The constant registers are located in the parameterization data block in data words DW 146 to DW 177.

Each constant occupies two data words in the parameterization data block. The constant registers can be used for calculations in the command lists.

The KS command offers the capability of making fast changes in the constant registers specified during parameterization (PA command).

**BB Process command list (I 4.4)**

This command can be used to process additional command list 1 on the module. The commands of command list 1 are located in parameterization data block DB 183 starting with data word DW 186.

In order to be able to execute the command list, it must have been transferred to the module at least once with the PA command (parameterize module). The bit for the global registers must be set at the PAR parameter.

The following command list was programmed in our example:

```
DW 186    9063    L      ZSZ3     Load counter value of counter 3
DW 187    9464    SUB    ZSZ4     Subtract counter value of counter 4
DW 188    9141    T      ERG1     Transfer result to result register 1
DW 189    9051    L      KON1     Load constant 1
DW 190    9052    L      KON2     Load constant 2
DW 191    A741    FIN    ERG1     ERG1 within KON1, KON2 window?
DW 192    A003    SPR    +3
DW 193    9142    T      ERG2     No: Transfer ERG1 TO ERG2
DW 194    A002    SPR    +2
DW 195    9143    T      ERG3     Yes:Transfer ERG1 TO ERG3
```

**MR Read and reset measured value series (I 4.6)**
**ML Read measured value series (I 4.7)**

These commands can be used to read measured value series 1, 2 and 3 . In contrast to the ML command, the MR command also causes the measured value series to be reset during reading. "Reset" means that the fill status indicator is set to the first value of the measured value series.

**LS Load and start counters 2, 3 and 4 (I 5.0)**
**LS Load and start counter 6 (I 5.4)**

These commands are used to load and start the respective counter with the value of the corresponding load register.

**RZ Reset counter 2 (I 5.2)**
**RZ Reset counter 6 (I 5.5)**

These commands are used to reset the respective counter to the value KH = 0000.

**AW Accept interrupt value for counter 2 (I 5.3)**
**AW Accept interrupt value for counter 6 (I 5.6)**

These commands are used to accept the interrupt value of the respective counter.

**SZ Step counter 2 (I 5.1)**

This command is used to step the counter value of counter 2 by one pulse.  In our example, the direction of counting is down (decrementation).  Counting is performed unconditionally.

**ZA Update counter values (I 5.7)**

This command is used to read the counter values of all counters, the result registers and the counter status registers from the counter module, and enter them in the parameterization data block.

Read Data Areas in Parameterization Data Block DB 183

|  |  |  |  |  |
|---|---|---|---|---|
| DW | 120 | to | DW | 122 | Counter status register |
| DW | 123 | to | DW | 127 | Counter values of counters 1 to 5 (one word each) |
| DW | 128 | to | DW | 131 | Counter values of counters 6 and 7 (two words each) |
| DW | 132 | to | DW | 145 | Result registers 1 to 7 (two words each) |

This command should always be selected (I 5.7 = "1") for the test so that you can directly observe the affects of the individual commands on the counters (the STEU VAR function of the programmer can be used to make the contents of the data words visible).

---

## 15.8.2  Indirect Parameterization

Indirect parameterization of function block FB 183 is selected with input I 6.3 = "1".  In our example, the function block itself is called with the rising edge at input I 6.2.

The KC = XX command is used to switch to the STEU parameter (i.e., the control word in the STEU parameter in now transferred to the module instead of a command from the BEF parameter with the command supplement from the PAR parameter).  In our example, the contents of the STEU parameter can be specified in bits via input word IW 4.

The control word must be completely specified (command code with bit for the global registers or counters 1 to 7).

If the specified command (lefthand byte of the control word – input byte IB 4 in our example) corresponds to a permissible command, a check for correct parameter assignment is performed by function block FB 183 (in our example, the righthand byte of the control word is set at input byte IB 5).

If no permissible command code is found, the function block transfers the parameterized control word to the module without changes and without a check.  In this case, the counter module checks the transferred control word and, if necessary, rejects it with an error message at the ERR parameter.

### 15.8.3  Parameterization of the IP 242B Counter Module

In our example, parameterization data block DB 183 contains the following valid parameterization data:

**Preassignments of the Global Registers**

Master mode register, MMR (DW 4)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1  | 0  | 0  | 0  | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

Bit   3  =                          1          Comparator 2 active
Bit  15  =                          1          BCD scaling (scaling factor = 10)

Prescaler register, VT (DW 5)

DW  5  =                    KF+100    Prescaler = 1:100

The setting in the prescaler register and in bit 15 of the master mode register results in a clock pulse frequency of 1 MHz:100:10:10:10 = 10 Hz for the internal module clock pulse F4. A clock pulse frequency of 100 Hz results for the clock pulse F3.  Counting clock pulses F3 and F4 are required for counters 3 and 4.

Interrupt enable register, IFR (DW 7)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1  | 1  | 1  | 1  | 1  | 1  | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Bit        0    =              1       Enable group interrupt for error message
Bit        8    =              1       Enable S5 group interrupt for counter events
                                       (overflow, comparator, gate edge)
Bits   9 to 15  =              1       Enable outputs 1 to 7

**Preassignments of Counter 2**

Counter mode register, CMR (DW 30)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0  | 0  | 0  | 0  | 0  | 0  | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |

| Bits | 0 to 2 | = | 001 | Output signal: pulse, active high |
|------|--------|---|-----|-----------------------------------|
| Bit | 3 | = | 0 | Counting direction: down |
| Bit | 4 | = | 1 | Counting mode: BCD |
| Bits | 5 to 7 | = | 001 | Counter operating mode D (cyclic counting procedure, reload from the load register) |
| Bits | 8 to 11 | = | 0010 | Counting pulse source = counter input C2 |
| Bit | 12 | = | 0 | Counting pulse edge = rising |
| Bits | 13 to 15 | = | 000 | Without gate control |

Load register LR (DW 31)

DW 31             =         KH 0010    Load register = 10

Interrupt register AR (DW 33)

DW 33             =         KH 0005    Interrupt register = 5

**Preassignments of Counter 3**

Counter mode register CMR (DW 44)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0  | 0  | 0  | 0  | 1  | 1  | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |

| Bits | 0 to 2 | = | 000 | Output signal switched off (low level) |
|------|--------|---|-----|----------------------------------------|
| Bit | 3 | = | 0 | Counting direction: down |
| Bit | 4 | = | 1 | Counting mode: BCD |
| Bits | 5 to 7 | = | 001 | Counter operating mode (cyclic counting procedure, reload from the load register) |
| Bits | 8 to 11 | = | 1110 | Counting pulse source = frequency F4 (10 Hz) |
| Bit | 12 | = | 0 | Counting pulse edge = rising |
| Bits | 13 to 15 | = | 000 | Without gate control |

Load register LR (DW 45)

DW 45             =         KH 1000    Load register = 1000

**Preassignments of Counter 4**

Counter mode register CMR (DW 58)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0  | 0  | 0  | 0  | 1  | 1  | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |

| Bits | 0 to 2 | = | 000 | Output signal switched off (low level) |
|------|--------|---|-----|----------------------------------------|
| Bit  | 3      | = | 0   | Counting direction: down |
| Bit  | 4      | = | 1   | Counting mode: BCD |
| Bits | 5 to 7 | = | 001 | Counter operating mode (cyclic counting procedure, reload from the load register) |
| Bits | 8 to 11| = | 1101| Counting pulse source = frequency F3 (100 Hz) |
| Bit  | 12     | = | 0   | Counting pulse edge = rising |
| Bits | 13 to 15| = | 000 | Without gate control |

Load register LR (DW 59)

| DW 59 | = | KH 0100 | Load register = 100 |
|-------|---|---------|---------------------|

**Preassignments of Counter  6**

Counter mode register CMR (DW 86)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0  | 0  | 0  | 0  | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |

| Bit  | 3      | = | 1   | Output signal active |
|------|--------|---|-----|----------------------|
| Bits | 4 to 6 | = | 001 | Single edge evaluation at input A |
|      |        |   |     | Input B determines the counting direction. |
| Bit  | 7      | = | 0   | Counter reset disabled |

Load register LR (DD 87)

| DD 87 | = | KH 0000 0600 | Load register = 600 |
|-------|---|--------------|---------------------|

Interrupt register AR (DD 91)

| DD 91 | = | KH 0000 0500 | Interrupt register = 500 |
|-------|---|--------------|--------------------------|

### 15.9    Interrupt Processing

In our example, interrupt processing is programmed in organization block OB 2.

The interrupts are acquired with interrupt line IR–A.

The scratchpad flags must be saved at the beginning of the interrupt block and reloaded again before the block is exited. This is performed with standard function blocks FB 38 and FB 39. These function blocks are included on the floppy disk with the standard function blocks for the counter module.

Saving and loading the scratchpad flags must be performed for all interrupt–controlled types of program processing (and also in the restart and error evaluation organization blocks if scratchpad flags are used here). This has already been taken care of in our example.

Output Q 8.1 is set when an error occurs during interrupt processing. Flag word FW 30 (ERR parameter) and flag bytes FY 32 (MELD parameter) and FY 33 (F–NR parameter) must be evaluated to determine the exact cause of the error.

The error indication can be acknowledged with input I 6.0 when the error is corrected. Error indication Q 8.1 is reset with this input.

**Triggering Interrupts with the Interrupt Value**

Each counter in the counter module has an interrupt register in parameterization data block DB 183. The data words for the individual counters are listed below.

DW    19        Interrupt register for counter    1
DW    33        Interrupt register for counter    2
DW    47        Interrupt register for counter    3
DW    61        Interrupt register for counter    4
DW    75        Interrupt register for counter    5
DW    91        Interrupt register for counter    6
DW   108        Interrupt register for counter    7

PG function STEU VAR can be used to change the interrupt values.

Interrupt value 0 is not permitted for counters 1 to 5 ($\rightarrow$ section 3.4.4).

Input I 5.3 is used to transfer the interrupt value of counter 2 to the counter module. Input I 5.6 is used to transfer the interrupt value of counter 6 to the counter module.

When the counter interrupts are enabled in the interrupt enable register (DW 7) and all comparators are switched on in the master mode register (DW 4), an interrupt is triggered when the counter value of a counter is equal to the contents of its interrupt register.

The comparator function of counters 6 and counter 7 is selected in bit 3 of the appropriate counter mode register.

---

# 16    Function Blocks FB 38 and FB 39 (Only for PLC S5–115U)

**IP 242A/242B**

---

## 16.1    Overview

These programming instructions describe the following function blocks:

**FB 38  (SAVE)**          Save scratchpad flags/system data and page frame number
**FB 39  (LOAD)**          Load scratchpad flags/system data and page frame number

These function blocks are used with programmable controller S5–115U (CPU 941A/B to CPU 944A/B).

These function blocks are found on the S5–DOS floppy disk in the following file:

**S5LD50ST.S5D**

## 16.2    Function Block FB 38

### 16.2.1  Function Description

The standard function block "save scratchpad flags/system data/page frame number" performs the following tasks depending on the parameterization:

1.  Save scratchpad flags (FY 200 to FY 255) and system data (BS 248 to BS 255) in a data block

2.  Save page frame number in a data block

The function block is called at the beginning of blocks from interrupt–controlled processing to "save scratchpad flags/system data".

When parameterized with "save page frame number", the function block is called in all blocks which work with page frame – addressed modules and which can be interrupted with process interrupts and time interrupts (call in organization blocks OB 21, OB 22, OB 1, and in blocks of the time–controlled processing). The function block does not have to be called again until the page frame changes.

Standard function block FB 38 works together with function block FB 39 which reloads the saved data. Function block FB 39 must be called at the end of interrupt–controlled processing.

## 16.2.2  Calling Function Block FB 38

In STL:

```
         :
         : L      KY     c,d
         : L      KY     a,b
         : JU     FB     38
Name : SAVE
PAFE :
         :
```

Do not call in CSF/LAD since both load commands must be programmed in STL and
would then be located in another network.
The prescribed sequence must be followed.

## 16.2.3  Explanation of the Parameters

| NAME | CLASS | TYPE | DESIGNATION |
|------|-------|------|-------------|
| PAFE | Q | BI | Parameterization error |
| Accumulator 1 (left byte) | | | Page frame number to be saved |
| Accumulator 1 (right byte) | | | Save scratchpad flags/system data or page frame number |
| Accumulator 2 (left byte) | | | Data block number |
| Accumulator 2 (right byte) | | | Identification for the organization block |

## 16.2.4  Assignment of the Parameters

**PAFE    :**          Signal status "1" indicates invalid parameterization.
                       The error can then be read from the assignment of accumulator 1 as follows:

                       KF = 1     Incorrect DB number
                       KF = 2     DB missing or too short
                       KF = 3     Assignments of parameters *"b"* and *"d"* do not match.


**Accumulator 1:**   **a,b**  **a:**   Page frame number to be saved

                             **b:**   Save scratchpad flags/system data or page frame number

                                      b=1:   Save only page frame number
                                      b=2:   Save only scratchpad flags/system data
                                      b=3:   Save everything


**Accumulator 2:**   **c,d**  **c:**   Number of the data block

                                      $10 \leq$ DB number $\leq$ 255

                             **d:**   Identification for the organization block

                                      d=1:   OB 1 / OB 21 / OB 22
                                      d=2:   OB 2 to 6 (interrupts)
                                      d=4:   OB 10 to 13 (time interrupts)

## 16.2.5  Overview of Valid Combinations

The following table assumes that you are working with data block DB 38.

The below assignments are also made:

K   :        Page frame number to be saved
x   :        Not applicable (i.e., the constant to be given here is not considered)

| Call In | Valid Function | Assignment of Accumulators |
|---|---|---|
| OB 1<br>OB 21 / OB 22 | Save page frame | Accumulator 1: KY = K,1<br>Accumulator 2: KY = 38,1 |
| OB 2 to OB 6 | Save scratchpad flags/<br>system data | Accumulator 1: KY = x,2<br>Accumulator 2: KY = 38,2 |
| OB 10 to OB 13 | Save scratchpad flags/<br>system data | Accumulator 1: KY = x,2<br>Accumulator 2: KY = 38,4 |
|  | Save page frame | Accumulator 1: KY = K,1<br>Accumulator 2: KY = 38,4 |
|  | Save both | Accumulator 1: KY = K,3<br>Accumulator 2: KY = 38,4 |

## 16.2.6  Assignment of the Data Area

Standard function block FB 38 works together with a parameterizable data block. Any number between 10 and 255 can be assigned to the data block. Set up the data block up to and including data word DW 80.

> **!**  Always specify the same data block every time FB 38 and FB 39 are called.  **!**

## 16.2.7　Technical Specifications – Function Block FB 38

Block number　　　　　　　　　: 038

Block name　　　　　　　　　　: SAVE

Library number　　　　　　　　: P71200–S 5038–A–1

Call length　　　　　　　　　　: 3 words

Block length　　　　　　　　　: 137 words

Processing times in msec

| Function | CPU 941A/B | CPU 942A/B | CPU 943A/B | CPU 944A/B |
|---|---|---|---|---|
| Save page frame number | 6.0/2.2 | 3.2/2.2 | 1.8/1.8 | 1.6/1.2 |
| Save scratchpad flags/ system data | 10.3/5.7 | 7.5/5.7 | 6.1/5.5 | 2.3/1.8 |
| Save both | 11.3/5.8 | 8.2/5.8 | 6.3/5.7 | 2.5/1.9 |

Nesting depth　　　　　　　　: 0

Blocks called　　　　　　　　 : none

Data words used　　　　　　　: parameterized data block up to and including DW 80

Flags used　　　　　　　　　　: FY 248 to FY 255

System statements　　　　　 : yes

Other　　　　　　　　　　　　 : During processing of the block, interrupts are disabled by
　　　　　　　　　　　　　　　　commands "AS" and "AF".

> **!**　　　　Use of command "AF" in FB 38 cancels out command "AS"!　　　　**!**

## 16.2.8  Application of Function Block FB 38

Depending on the parameterization, standard function block FB 38 performs the saving of the scratchpad flag area (FY 200 to FY 255), the user system data (BS 248 to BS 255), and/or the page frame number specified. It uses the accumulators for this purpose.

During cyclic operation and start–up, function block **FB 38**, when parameterized with **"save page frame number"**, must be called **before a page frame number is written** if page frame accesses are also programmed in the interrupt organization blocks and different page frame numbers are being processed.

Blocks of the time–controlled processing are handled the same way since these blocks can be stopped by an interrupt.

A page frame can be accessed, for example, by handling blocks, direct access by a user program, or by standard function blocks.

Scratchpad flags must always be saved at the beginning and reloaded at the end in interrupt organization blocks. Function block FB 39 handles the loading of the saved data.

Function blocks **FB 38 and FB 39 must always be used in pairs in the interrupt organization blocks** (i.e., these blocks cannot be exited prematurely with the "BEC" command).

## 16.3    Function Block FB 39

## 16.3.1  Function Description

Depending on the parameterization, standard function block "load scratchpad flags/system data/page frame" performs the following tasks:

1. Load scratchpad flags (FY 200 to FY 255) and system data (BS 248 to BS 255) from a data block

2. Load page frame number from a data block

The function block is called at the end of blocks of interrupt–controlled processing to load scratchpad flags/system data and/or page frame number.

The data must be saved in the data block beforehand by standard function block FB 38.

## 16.3.2  Calling Function Block FB 39

In STL:

```
      :
      : L     KY    c,d
      : L     KY    a,b
      : JU    FB    39
Name : LOAD
PAFE :
      :
```

> Do not call in CSF/LAD since both load commands must be programmed in STL and would then be located in another network.
> The prescribed sequence must be followed.

## 16.3.3  Explanation of the Parameters

| NAME | CLASS | TYPE | DESIGNATION |
|------|-------|------|-------------|
| PAFE | Q | BI | Parameterization error |
| Accumulator 1 (left byte) | | | Not applicable |
| Accumulator 1 (right byte) | | | Load scratchpad flags/system data or page frame number |
| Accumulator 2 (left byte) | | | Data block number |
| Accumulator 2 (right byte) | | | Identification of the organization block |

## 16.3.4  Assignment of the Parameters

**PAFE:**  Signal status "1" indicates invalid parameterization.
The error can then be read from the assignment of accumulator 1 as follows:

KF = 1    Incorrect DB number
KF = 2    DB missing or too short
KF = 3    Assignments of parameters *"b"* and *"d"* do not match.

**Accumulator 1:**   **a,b**   **a:**   Not applicable

 **b:**   Select what is to be loaded

b=1:   Load only page frame number
b=2:   Load only scratchpad flags/system data
b=3:   Load everything

**Accumulator 2:**   **c,d**   **c:**   Number of the data block

$10 \leq$ DB number $\leq 255$

 **d:**   Identification for the organization block

d=2:   OB 2 to 6 (interrupts)
d=4:   OB 10 to 13 (time interrupts)

## 16.3.5  Overview of Valid Combinations

The following table assumes that you are working with data block DB 38.

The below assignment is also made:

x   :        Not applicable (i.e., the constant to be given here is not considered)

| Call In | Valid Function | Assignment of Accumulators |
|---|---|---|
| OB 2 to OB 6 | Load scratchpad flags/ system data | Accumulator 1: KY = x,2<br>Accumulator 2: KY = 38,2 |
| | Load page frame | Accumulator 1: KY = x,1<br>Accumulator 2: KY = 38,2 |
| | Load both | Accumulator 1: KY = x,3<br>Accumulator 2: KY = 38,2 |
| OB 10 to OB 13 | Load scratchpad flags/ system data | Accumulator 1: KY = x,2<br>Accumulator 2: KY = 38,4 |
| | Load page frame | Accumulator 1: KY = x,1<br>Accumulator 2: KY = 38,4 |
| | Load both | Accumulator 1: KY = x,3<br>Accumulator 2: KY = 38,4 |

## 16.3.6  Assignment of the Data Area

Standard function block FB 39 works together with a parameterizable data block. Any number between 10 and 255 can be assigned to the data block. Set up the data block up to and including data word DW 80.

| ! | Always specify the same data block every time FB 38 and FB 39 are called. | ! |

## 16.3.7  Technical Specifications – Function Block FB 39

Block number                  : 039

Block name                    : LOAD

Library number                : P71200–S 5039–A–1

Call length                   : 3 words

Block length                  : 141 words

Processing times in msec

| Function | CPU 941A/B | CPU 942A/B | CPU 943A/B | CPU 944A/B |
|---|---|---|---|---|
| Load page frame number | 5.8/1.5 | 2.3/1.5 | 1.2/1.2 | 0.5/0.3 |
| Load scratchpad flags/ system data | 9.5/4.3 | 6.3/4.3 | 4.6/4.2 | 0.6/0.4 |
| Load both | 11.0/4.4 | 6.9/4.4 | 4.8/4.3 | 0.7/0.5 |

Nesting depth                 : 0

Blocks called                 : none

Data words used               : parameterized data block up to and including DW 80

Flags used                    : FY 248 to FY 255

System statements             : yes

Other                         :  During processing of the block, interrupts are disabled by
                                  commands "AS" and "AF".

| ! | Use of command "AF" in FB 39 cancels out command "AS"! | ! |
|---|---|---|

## 16.3.8  Application of Function Block FB 39

Depending on the parameterization, standard function block FB 39 performs the loading of the scratchpad flag area (FY 200 to FY 255), the user system data (BS 248 to BS 255), and/or the page frame number specified. It uses the accumulators for this purpose.

The function block **is called at the end of every interrupt–controlled program processing** (interrupts, time interrupts). It ensures that the original status of the interrupted program is restored when returning to the program. The data must be saved beforehand in the parameterized data block with FB 38.

Scratchpad flags must always be saved at the beginning and reloaded at the end in interrupt organization blocks. Function block FB 38 handles the saving of the data.

Function blocks **FB 38 and FB 39 must always be used in pairs** in the interrupt organization blocks (i.e., these blocks cannot be exited prematurely with the "BEC" command).

# 17   Glossary

**IP 242A/242B Equipment Manual – Glossary**

| | |
|---|---|
| Accu | Four accumulators are used to implement the calculation functions on the IP 242B. |
| Additional command lists | Contain command sequences for the calculation functions (IP 242B only). |
| AFR | See interrupt filter register |
| Basic address | Address set on the module |
| BASP | "Disable command output" S5 signal<br>The signal occurs when the S5 CPU goes into stop status. |
| BCD | Binary coded decimal number |
| CM register | See counter mode register |
| Command list | The user enters in the command list the control words which are to be executed when a counter interrupt occurs. See sample number G 89 09 653. |
| Control word | All necessary module and counter functions can be triggered with the control words. |
| Counter mode register | This register can be parameterized separately for each of the 7 counters. It determines counting mode, pulse source, gate control, counting edge, operating mode, reset input and output signal for each individual counter. |
| Counter values | The counter values are entered cyclically in the IP 242B by the CPU. |
| Counter status register | Contains the current module status.<br>It can be read by the S5 (IP 242B only) |
| Counting cycle zero | For down counting with counters 1 to 5, the value of the load or hold register is reloaded immediately after the value 1 is reached. For this reason, **zero** is not in the counter. |
| CPU | Central processing unit in the programmable controller |
| DB | Data block |
| DD | Data double word |
| DPR | See dual port RAM |
| Dual port RAM | Exchange of data and commands between the IP 242A/242B module and the programmable controller takes place in this memory area. |

| | |
|---|---|
| DW | Data word |
| EEPROM | Non–volatile memory in the IP 242A in which the user can store the parameters, and if necessary, read out and also overwrite |
| ERR | See error information register |
| Error information register | A register in which the IP 242A/242B enters an error number |
| FB | Function block |
| Gate | Independent, supplementary control input for counters 1 to 5 |
| Gate control register | This is where it is determined whether internal or external signals are applied to the counter gates. |
| Global register | There is one global register for each module. The following are global registers: master mode register, prescaler register, gate control register, interrupt enable register, interrupt polarity register, and version number register. For the IP 242B: also interrupt filter register, FB version identifier and pulse duration register. |
| Hold register | There is one separate register for each counter. These registers are used as intermediate storage for each counter or are used alternately with the load register. |
| H register | See hold register |
| IB 0 | Input byte IB 0 |
| IFR | See interrupt enable register |
| IIR | See interrupt information register |
| Incremental encoder | Encoder with 90° displaced pulse trains; can be connected to counters 6 and 7. |
| Input byte IB 0 | Some programmable controllers use input byte IB 0 for interrupt evaluation. |
| Interrupt | Group interrupt supplied by an interrupt line or input byte IB 0 to the CPU of the programmable controller |
| Interrupt | Interrupt via an interrupt line (PLCs S5–115U, S5–135U , and S5–155U in S5–155U mode) |
| Interrupt enable register | Internal interrupts which cause interrupts are set here. |
| Interrupt filter register | Use these for specific enabling of interrupt from the module to the S5 (IP 242B only). |
| Interrupt information register | The module enters its interrupt request in this register. |
| Interrupt polarity register | The edge at which an interrupt is triggered is set here. |
| Interrupt register | Comparison register for comparator function. The actual counter status is compared with the interrupt value specified. If equal, the digital output is set and an interrupt is sent to the S5 (if enabled). |

| | |
|---|---|
| Interrupt value | Contents of the interrupt register |
| IPR | See interrupt polarity register |
| IRA, IRB, IRC, IRD | S5 interrupt lines |
| Linear address | Address area of an I/O module which is not addressed in the page frame area |
| Load register | There is one separate register for each counter. The initial value of the respective counter is stored here. |
| L register | See load register |
| Master | The highest ranking, interrupt–generating IP module |
| Master mode register | A common register through which scaling factors and comparator functions can be specified for all counters together. |
| Measured value data block | Data block in the S5 in which the measured value series of the IP 242B is/are stored |
| MMR | See master mode register |
| Operation modes | Available for counters 1 to 5. See section 8. |
| Page frame address, page frame area | Address area starting at F400H where several modules can be addressed in parallel. Differentiation through the page frame number. |
| Page frame number | Address of a module in the page frame area |
| Parameterization data block | Data block in the S5 in which the data for the counters is stored |
| PDR | See pulse duration register |
| PLC | Programmable controller S5 |
| Prescaler | Programmable frequency reducer for the internal 1–mHz frequency generator |
| Prescaler register | The scaling factor of the prescaler is set here. |
| Process interrupt | Interrupt via input byte IB 0 (PLCs S5–150U/S and S5–155U in S5–150U mode) |
| Pulse duration register | Contains the pulse duration for software comparator (IP 242B only) |
| PY 0 | Peripheral byte zero. See input byte IB 0. |
| Result register | The results of the calculation functions are stored here (IP 242B only). |
| S5 bus | Backplane bus system of the programmable controller through which all data and commands are exchanged |
| Scaler | Programmable frequency reducer on the module |
| Slave | When several interrupt–generating IP modules are used, interrupt source priority is established according to the master slave principle. |

| | |
|---|---|
| Terminal count | Maximum counting limit (up counting) or minimum counting limit (down counting) |
| Trace function | The results are entered in the measured value memory in measured value series. This permits "past values" to be read out of the measured value series. This makes it easy to determine the cause of a malfunction (IP 242B only). |
| TSR | See gate control register |
| Version number register | The firmware status is specified here. |
| VNR | See version number register |
| VTR | See prescaler register |
| ZSR | See counter status register |
| ZSZ | See counter values |

## 18   Index

# Adapter Module (S5 Adapter) A

**In this Chapter**     In this chapter, you will learn the following:

- How you install the modules in the adapter module

- What you must observe when using the various S5 modules

## A.1 Prerequisites

**General**

The following prerequisites must be observed when using S5 modules in an S7-400:

- Check with your local Siemens office that the modules you want to use have been approved for this type of use.

- Programmable S5 modules can be linked into a STEP 7 user program only with special standard function blocks. Should you have only standard S5 function blocks for your S5 modules which are not expressly authorized for use with STEP 7 in the associated documentation (Manual or Product Information), you must order new standard function blocks for those modules.

- SIMATIC S5 and SIMATIC S7 differ from one another in their general technical specifications, particularly those relating to ambient conditions. When installing an S5 module in an S7-400, the more stringent ambient conditions for either the S5 or the S7 apply for the system as a whole.

**Permissible Racks**

You can use the adapter module only in the S7-400 central rack.

---

**Note**

Before installing an S5 module which has already been used in an S5 configuration in an S7 system, always call your local Siemens office for advice. The information provided in this chapter relates exclusively to the current versions and releases of the S5 modules covered.

---

## A.2   Installing an Adapter Module in an S7-400

**Introduction**

To install an S5 module in an S7-400, you must first install the adapter module in the S7 rack, then set the address on the S5 module, and, finally, insert the module in the adapter module.

**Installing the Adapter Module in a Rack**

Proceed as follows to install an adapter module in a rack:

1. Check to make sure that the jumpers on the back of the adapter module are closed (factory setting). These jumpers are for test purposes only, and must always remain closed.
   Figure A-1 shows the location of the jumpers.
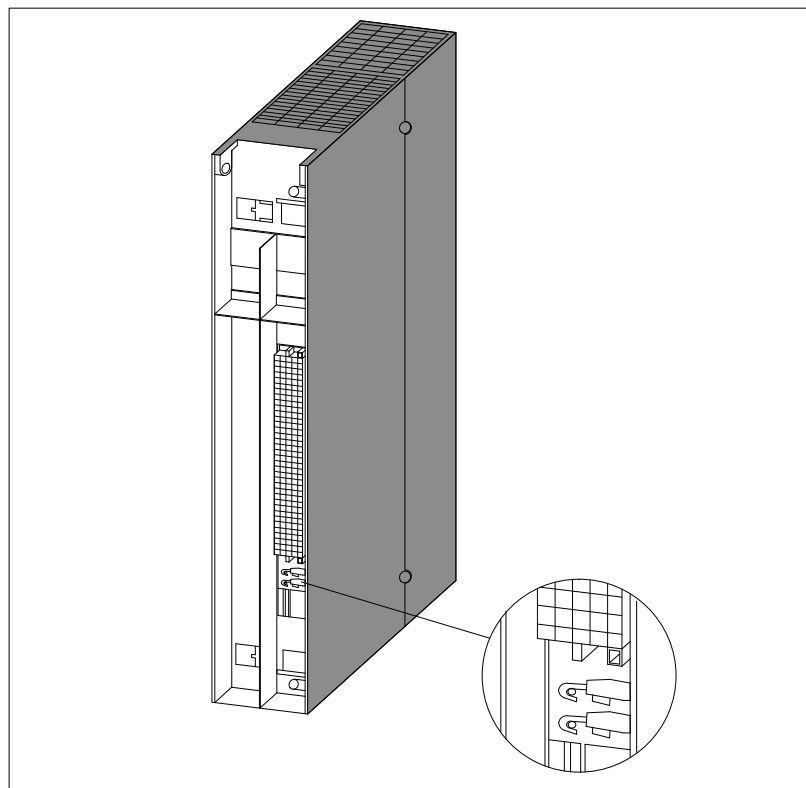


Figure A-1     Location of the Jumpers on the Adapter Module

2. Set the CPU mode switch to the STOP position.

3. Set the standby switch on the power supply module to the ⏻ position (0 V output voltages).

4. Follow the instructions in the *S7-400 and M7-400 Hardware and Installation Manual* for inserting modules in a rack.

**Setting the Address**

Set the address on the S5 module.

## A.3 Inserting S5 Modules in the Adapter Module

**Procedure**
Proceed as follows to insert an S5 module in the adapter module:

1. Set an interrupt circuit on the module, which sets the destination CPU for interrupts (in the case of interrupt-generating modules only).

| Interrupt Circuit... | ...Corresponds to Destination CPU |
|---|---|
| /INT A | CPU 1 |
| /INT B | CPU 2 |
| /INT C | CPU 3 |
| /INT D | CPU 4 |

2. Unscrew and remove the interlocking plate on the adapter module.

3. Insert the module in the adapter module's guide rails and push. The rear plug connectors snap into the adapter module's socket connectors.

4. Screw the interlocking plate back into place.

5. On S5 modules with locking screw, push the knob in and turn it so that the screw slit is vertical.

Figure A-2 shows how to insert an S5 module into the adapter module.
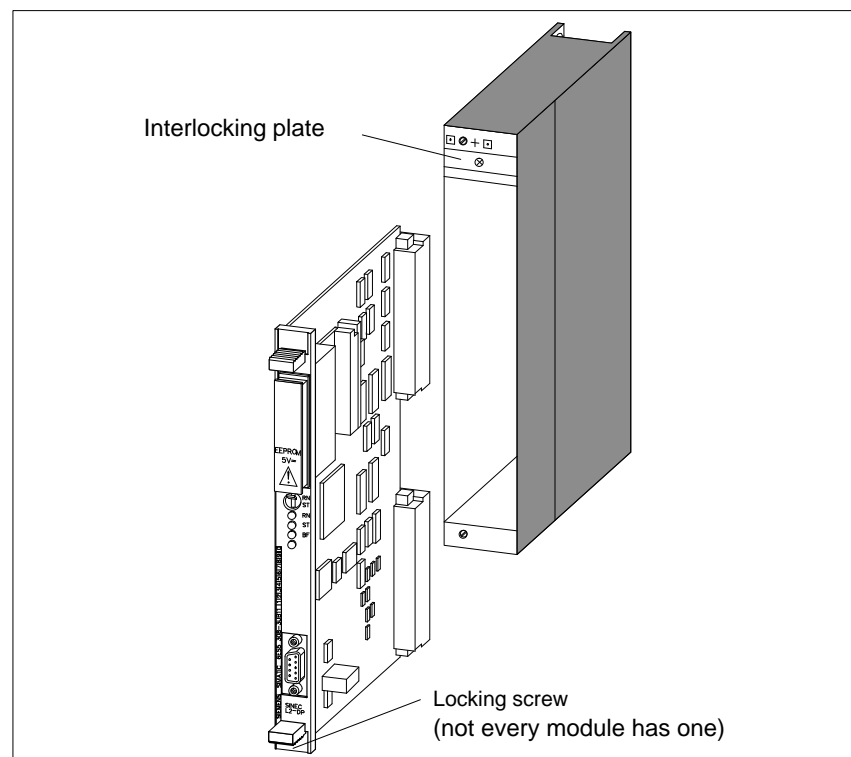


Figure A-2    Inserting an S5 Module into the Adapter Module

## A.4 Interrupt Processing

**Introduction**

The adapter module converts S5 interrupts into S7 interrupt functions and interrupt signals.

**Interrupt Routing**

All of the S5 module's interrupts are forwarded as (S7) hardware interrupts. The interrupts are routed as follows:

| S5 Interrupt Circuit | S7 Interrupt Circuit |
|:---:|:---:|
| /INT A | /I1 |
| /INT B | /I2 |
| /INT C | /I3 |
| /INT D | /I4 |

**Interrupt During Active OD**

No new interrupts are triggered while OD (OUTPUT DISABLE) is active (for example, when the CPU is in STOP). Interrupts which were already pending are processed. The falling edge of the OD signal resets the S7-specific interrupt functions.

Whether or not the S5-specific interrupt functions are reset with the falling edge of the OD signal depends on the S5 module (please refer to the relevant manuals). In the case of S5 modules in which the falling edge of the OD signal does not reset an interrupt, a new interrupt is subsequently triggered.

**Ascertaining the Interrupt-Generating Module**

When an S5 module in the adapter module triggers an interrupt, the logical address of that module is entered in the interrupt OB's local data area.

**Acknowledging an Interrupt**

You acknowledge an interrupt in the usual manner, that is, the same as in S5 systems (refer to the Manual or the Product Information for details). The CPU automatically performs the additional S7-specific interrupt functions.

## A.5 Technical Specifications

| Dimensions and Weight | |
|---|---|
| Dimensions W × H × D | 50mm × 290mm × 210 mm (1.96 in. x 11.4 in. x 8.26 in.) |
| Weight | Approx. 300 g |
| **Voltages and Currents** | |
| System voltage [1] | 5 V DC |
| • Rated voltage | 5.1 V DC |
| • Range | 4.75 V to 5.25 V DC |
| Auxiliary voltage [1] | |
| • Rated voltage | 24 V DC |
| • Range | 18 V to 32 V DC |
| Battery voltage [1] | |
| • Rated voltage | 3.4 V DC |
| • Range | 2.75 V to 4.4 V DC |

| Maximum Current Load | |
|---|---|
| The maximum power which may be drawn from the adapter module is as follows: | |
| • From the system voltage | 3 A |
| • From the auxiliary voltage | 0.5 A |
| • From the battery voltage | 0.5 mA |

[1] Looped through from the S7-400 power supply

# Addressing S5 Modules
# (Adapter Module and IM 463-2)

# B

**In this Chapter**    This chapter describes the following topics:

- How you address S5 modules inserted in the adapter module

- How you address S5 modules connected via the IM 463-2

## B.1　Addressing S5 Modules

**Introduction**　　　　There are two ways of using an IP xxx S5 module in an S7-400:

- By installing it in the adapter module in the S7 central rack

- By using an S5 expansion rack and connecting the S5 module via the IM 463-2 interface module in the S7 central rack and the IM 314 interface module in the S5 expansion rack

**Addressing**　　　　In order to be able to address an S5 module in the S7-400, you must set addresses in two different places:

- The address under which the module is to be referenced in the user program and the address set on the module must be entered in STEP 7.

- The address of the S5 module in a permissible S5 address range (address switch on the module).

**S7 Address**　　　　You set the address under which the module is to be referenced in the S7-400 under STEP 7. It is not possible to use default addresses in the S7-400.

Assign the following values for an S5 module address in the S7-400:

- S7 address:　　　　Logical address. The value range depends on the CPU used.

- S5 address:　　　　Address set on the module. Value range from 0 to 255.

- Length:　　　　Size of the address field. Value range from 0 to128 (in bytes).

- Process image subarea:　　　　Process image subarea assignment.
　　　　　　　　　　Value range:　0　　　(entire process image)
　　　　　　　　　　　　　　　　　　1 to 8　(process image subarea)

- Area:　　　　Value range P, Q, IM3, IM4.

**S5 Address Areas**  S5 modules in the S7-400 may be addressed in the following address areas:

- I/O area (P area)

- Extended I/O area (Q, IM3, IM4)

- Page area

**I/O Area**  A PESP signal (memory I/O select signal) is generated in the P area only when S5 modules are interfaced to the system via the adapter module. The signal is forwarded to the S5 module. No PESP signal is generated for the Q, IM3 or IM4 areas.

When the S5 modules are interfaced via the IM 463-2, the PESP signal is generated by the IM 314 in the S5 expansion unit (for the selected P, Q, IM3 or IM4 area).

This corresponds to the 256-byte I/O area as defined for SIMATIC S5. The S5 address of the module in these areas is set on the module using jumpers or switches. Please refer to the respective module manual for the correct setting.

For modules which reserve input and output areas, an entry must be made under STEP 7 for each area.

**Page Area**  In order to operate an S5 module with page addressing, you require the revised standard function blocks (S7 functions). These standard function blocks call special system functions which emulate the S5 page commands. These standard function blocks can be linked into your user program.

Even in the case of page addressing, you must assign a logical address. This logical address is entered in the interrupt OB's local data area as start information.

Under STEP 7, you must assign an S7 address and an S5 address in the input area with length 0. You may not assign an address for this module in the output area.

---

**Note**

When using S5 modules in your S7-400, you must observe the following carefully when setting the module addresses:

- No two S7 addresses may be the same.

- No two S5 addresses may be the same in any given area (P, Q, IM3, IM4).

- Even when an S5 module has an address area with a length of 0, its address may not lie within the address area of another S5 module.

---

**Example of Addressing in the Page Area**

The CPU and an IP (intelligent I/O module) interchange data via the S5 bus interface and a 2 Kbyte dual-port RAM which is divided into two "pages".

The addressing area in which the pages are located is set at the factory. You need only set the page number for the first page on the module.

A module's two pages always occupy two consecutive numbers. The IP thus knows the address for the second page automatically.

The same addressing area is set for page addressing on each module at the factory.

When you configure your hardware with STEP 7, you must enter the following parameters in the input area:

- – S7 address: Logical address
- – S5 address: 0 (value range from 0 to 255, may not appear more than once in the specified area)
- – Length: 0
- – Process image subarea: 0
- – Area: P (value range P, Q, IM3, IM4)

The parameter list must always be passed in its complete state. Individual parameters have no technical relevance.

**Example of Addressing in the P Area**

An IP xxx requires 32 addresses in order to pass the required parameters. Only the start address of the module need be set. The next 31 addresses are reserved by an internal decoding procedure, and are then no longer available for other modules. The addresses can be set in increments of 32.

A module's input and output addresses (S5 and S7) must be identical. This is a prerequisite which must be observed to ensure correct use of the standard function blocks.

When you configure your hardware with STEP 7, you must enter the following parameters in the input and output areas:

- – S7 address: Must be a logical address equal to or greater than 512 (which you can use in your user program to reference the module)
- – S5 address: Same as on the module
- – Length: 32 bytes
- – Process image subarea: 0
- – Area: Depends on the area set on the module or IM 314 (P, Q, IM3 or IM4)

The address of the IP 244 may not lie within the process image. There are two ways to ensure this:

- Set an S7 address equal to or greater than 512
- Select a process image subarea value equal to or greater than 0

# IP 242B Counter Module

**C**

**In this Chapter**

This chapter describes the functions of the IP 242B counter module, lists its technical specifications and provides a programming example to show you how to use the functions.

**Chapter Overview**

## C.1   Overview

**Introduction**

This addendum supplements Chapters 14 and 15 of the manual. It describes the standard blocks for the IP 242B counter module for the SIMATIC S7-400.

The counter module can be connected via the adapter module in a SIMATIC S7-400 programmable controller.

For this purpose, there are new standard blocks which can execute in the S7-400 programmable controller's CPUs.

**Standard Software**

The standard functions are provided in the form of a SETUP on a floppy disk. The SETUP can run only under Windows 95.

When the SETUP is run, it creates a library containing only the standard functions for the IP 242B, and a programming example.

An on-line help facility is provided for the standard functions.

**Prerequisite for Assigning Parameters to the Module**

Before you start assigning parameters to the IP 242B, you should make sure that the following prerequisite has been fulfilled:

• Version 2.0 or a newer version of STEP 7 must be correctly installed on your programming device or PC.

**Installation**

All the software (function blocks and example) is on a 3.5 inch floppy disk.

You install the software as follows:

1. Insert the disk in your programming device or PC disk drive.

2. Under Windows 95, start the dialog for installing software by double-clicking on the *Software* symbol in the *Control panel*.

3. Select the disk drive and the *SETUP.EXE* file and start the installation procedure.

4. Follow the step-by-step instructions displayed by the installation program.

**Result:**

The software is installed in the following directories on the target drive:

| Software | Directory |
|----------|-----------|
| • Function blocks: | STEP7_V2\S7LIBS\IP242BLI |
| • Example: | STEP7_V2\EXAMPLES\IP242BEX |

---

**Note**

If you selected a directory other than STEP 7_V2 when you installed
STEP 7, that directory will be entered.

---

**Configuring**

Before you can configure your system, you must have created a project in
which you can store the parameters. You can find additional information on
configuring modules in your *Standard Software for S7 and M7, STEP 7* User
Manual. Only the most important steps are described below.

1. Start the SIMATIC Manager and open the configuration table in your
   project.

2. Select a rack and place it at the desired position.

3. Open the rack.

4. Select the following components in the hardware catalog:

   SIMATIC 400

   |────── IM-400
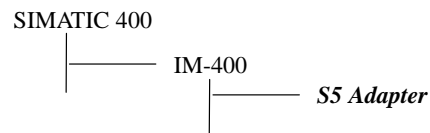
   |────── *S5 Adapter*

Please refer to the chapter in this addendum entitled "Addressing S5
Modules" (Adapter Module and IM 463-2) for additional information needed
to configure the hardware.

## C.2 Counter Processing Blocks

## Functions FC183 (ZYK_242B), FC184 (INT_242B) and FC185 (ZA_242B)

**Introduction**
The call, meaning and values of the parameters for the FC183, FC184 and FC185 functions are described below.

**Calling the Function FC183**

| Ladder Logic | Statement List |
|---|---|
| FC183<br><br>── EN        ENO ──<br>── SSNR    ERR ──<br>── BEF     MELD ──<br>── PAR     F_NR ──<br>── STEU<br>── DBNR | `CALL  FC183 (`<br>`  SSNR :=    ,`<br>`  BEF  :=    ,`<br>`  PAR  :=    ,`<br>`  STEU:=    ,`<br>`  DBNR :=    ,`<br>`  ERR  :=    ,`<br>`  MELD :=    ,`<br>`  F_NR :=    );` |

**Calling the Function FC184**

| Ladder Logic | Statement List |
|---|---|
| FC184<br><br>── EN        ENO ──<br>── SSNR    IIR ──<br>── FKT     ERR ──<br>── PAR     MELD ──<br>── DBNR    F_NR ── | `CALL  FC184 (`<br>`  SSNR :=    ,`<br>`  FKT  :=    ,`<br>`  PAR  :=    ,`<br>`  DBNR :=    ,`<br>`  IIR  :=    ,`<br>`  ERR  :=    ,`<br>`  MELD :=    ,`<br>`  F_NR :=    );` |

**Calling the
Function FC185**

| Ladder Logic | Statement List |
|---|---|
| FC185<br><br>— EN        ENO —<br>— SSNR     IIR —<br>— KMD      ERR —<br>— DBNR    MELD — | `CALL  FC185 (`<br>`  SSNR :=    ,`<br>`  KMD  :=    ,`<br>`  DBNR :=    ,`<br>`  IIR  :=    ,`<br>`  ERR  :=    ,`<br>`  MELD :=    );` |

**Parameters**

The table below provides an overview of the parameters required by the functions FC183, FC184 and FC185. The columns with the names of the functions contain an 'X' to show whether the particular parameter is available in the function.

The parameter DBNR is new for the block FC185. This means it is no longer necessary to open the parameter assignment data block before calling FC185.

| Name | Parameter Type | Data Type | Description | FC183 | FC184 | FC185 |
|---|---|---|---|---|---|---|
| SSNR | INPUT | INT | Interface number | X | X | X |
| FKT | INPUT | WORD | Function which FC184 (interrupt processing) should execute | – | X | – |
| BEF | INPUT | WORD | Command; control word which FC183 should execute (direct parameter assignment) | X | – | – |
| PAR | INPUT | WORD | Parameter; e.g. counters which should be controlled simultaneously (direct parameter assignment) | X | X | – |
| STEU | INPUT | WORD | Control word; function which FC183 should execute, and relevant parameter (indirect parameter assignment) | X | – | – |
| KMD | INPUT | WORD | Command which determines the function of FC185. (bit 8 = 1 in the interrupt OB) | – | – | X |
| DBNR | INPUT | INT | Number of the parameter ass. data block | X | X | X |
| IIR | OUTPUT | WORD | Interrupt information register | – | X | X |
| ERR | OUTPUT | WORD | Error information register | X | X | X |
| MELD | OUTPUT | BYTE | Message byte | X | X | X |
| F_NR | OUTPUT | BYTE | Error number | X | X | – |

**Parameter Values**

- DBNR: INT = x
  x = depends on the CPU used (0 is not permitted)

- BEF: WORD = B#(i,j)
  The value for the parameter *BEF* can be found in the table below:

| Description | BEF |
|---|---|
| RB: Reset module | B#(0,1) |
| SA: Disable outputs | B#(0,2) |
| FA: Enable outputs | B#(0,3) |
| IM: Mask interrupt | B#(0,4) |
| KS: Write constant registers | B#(0,5) |
| ZA: Update counter states | B#(0,6) |
| SZ: Step counter | B#(1,5) |
| SV: Save counter | B#(1,6) |
| CO: Copy counter | B#(1,7) |
| LV: Prepare to load | B#(1,8) |
| LS: Load and start counter | B#(3,1) |
| LE: Read counter | B#(3,2) |
| RZ: Reset counter | B#(3,3) |
| AW: Transfer interrupt value | B#(3,4) |
| ST: Start counter | B#(3,6) |
| LD: Load counter | B#(3,7) |
| SP: Stop counter | B#(3,8) |
| SL: Stop and read counter | B#(3,9) |
| PA: Assign counter parameters | B#(7,1) |
| PS: Save parameters | B#(7,2) |
| PZ: Rewrite parameters | B#(7,3) |
| GR: Transfer basic settings | B#(7,4) |
| PO: Assign counter parameters (without command list) | B#(7,5) |
| RL: Read register | B#(7,6) |
| RS: Write register | B#(7,7) |
| TF: Execute test function | B#(8,1) |
| BB: Process command list | B#(8,2) |
| MR: Read and reset measured value set | B#(8,3) |
| ML: Read measured value set | B#(8,4) |
| XX: Switch to STEU | B#(255,255) |

For all other parameter values, please refer to the Manual (Section 14.5
"Assignment of the Parameters").

**Technical Specifications for FC183, FC184 and FC185**

The technical specifications for FC183, FC184 and FC185 are listed below:

|                                     | **FC183**        | **FC184**      | **FC185**      |
|-------------------------------------|------------------|----------------|----------------|
| Block number                        | 183              | 184            | 185            |
| Block name                          | ZYK_242B         | INT_242B       | ZA_242B        |
| Version                             | 1.0              | 1.0            | 1.0            |
| Space reserved in work memory       | 3,228 bytes      | 978 bytes      | 1,204 bytes    |
| Space reserved in load memory       | 3,750 bytes      | 1,170 bytes    | 1,410 bytes    |
| Space reserved in local data area   | 28 bytes         | 16 bytes       | 16 bytes       |
| Space reserved in data area         | – Parameter assignment data block up to and including data byte DBB 511<br> – Measured value data block up to and including data byte DBB [1] | | |
| System functions called             | SFC41   DIS_AIRT<br>SFC42   EN_AIRT<br>SFC47   WAIT | | |
| Other                               | Interrupts (hardware interrupts, cyclic interrupts) are disabled for a maximum of 3 ms in the function block | | |

[1]   The length depends on the length of the measured values

**Execution Times**     The execution times for FC183, FC184 and FC185 shown below apply for
                        the CPU 416-1:

| Block | Function | Execution Time |
|-------|----------|----------------|
| FC183 | RB | 1.0 ms |
|       | SA, FA, IM, KS, ZA, SZ, SV, CO | 0.5 ms |
|       | LV:<br> 1 counter:<br> all counters: | <br>0.5 ms<br>0.6 ms |
|       | LS<br> 1 counter:<br> all counters: | <br>0.5 ms<br>0.7 ms |
|       | LE, RZ | 0.5 ms |
|       | AW<br> 1 counter:<br> all counters: | <br>0.5 ms<br>0.7 ms |
|       | ST, SP, SL | 0.5 ms |
|       | LD<br> 1 counter:<br> all counters: | <br>0.5 ms<br>0.6 ms |
|       | PA<br> global reg. only:<br> complete: | <br>1.0 ms<br>1.5 ms |
|       | PS<br> global reg. only:<br> complete: | <br>1.0 ms<br>1.5 ms |
|       | PZ<br> global reg. only:<br> complete: | <br>1.0 ms<br>1.0 ms |
|       | GR<br> global reg. only:<br> complete: | <br>0.7 ms<br>1.0 ms |
|       | PO<br> global reg. only:<br> complete: | <br>0.5 ms<br>0.8 ms |
|       | RL<br> global reg. only:<br> complete: | <br>0.7 ms<br>1.1 ms |
|       | RS<br> global reg. only:<br> complete: | <br>0.7 ms<br>1.2 ms |

| Block | Function | Execution Time |
|-------|----------|----------------|
| FC183<br>(cont.) | BB | 0.6 ms |
|       | TF<br> EPROM: | <br>0.8 ms |
|       | ML, MR<br> directory only:<br> 1*2 words:<br> 50*2 words:<br> 100*2 words: | <br>0.5 ms<br>0.6 ms<br>0.7 ms<br>0.8 ms |
| FC184 | Acknowl. interrupt | 0.1 ms |
|       | Write data<br> global registers:<br> 1 counter:<br> complete: | <br>0.3 ms<br>0.2 ms<br>0.3 ms |
|       | Read data | 0.2 ms |
|       | Read and write data | 0.3 ms |
| FC185 | Acknowl. interrupt | 0.3 ms |
|       | Read counter status register | 0.3 ms |
|       | Read counter states 1 to 7 | 0.3 ms |
|       | Read result register | 0.3 ms |
|       | All | 0.3 ms |

# Data Block Format

**Introduction**

In SIMATIC S7, data addresses are always counted by **byte**. The address of an S5 data word (DW n) corresponds to the address DBW (2*n) of the S7 data word.

The data block formats differ from those in S5.

**General Notes Regarding the Data Block Format**

In STEP 7, addressing of the data addresses in data blocks is done by **byte** (in contrast to STEP 5, where addressing is by **word**). For this reason, the addresses of the data must be converted accordingly (see Figure C-1).
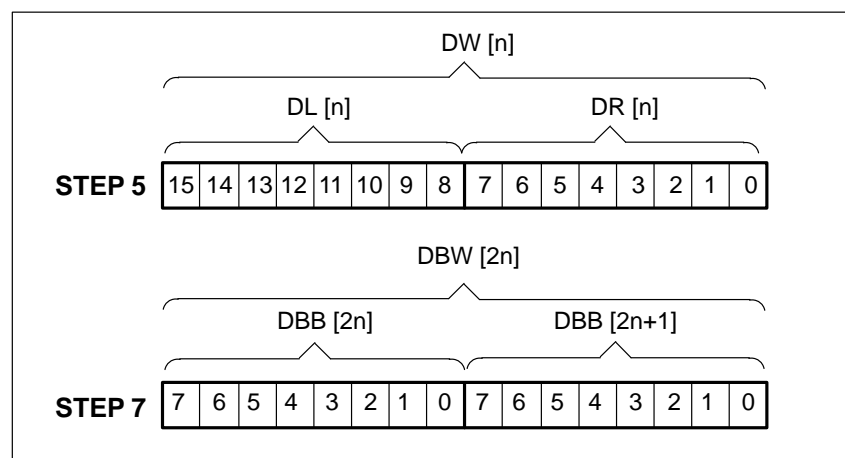


Figure C-1    Comparison of Data Addressing in STEP 5 and STEP 7

As opposed to STEP 5, the address of a data word in STEP 7 is doubled. There is no longer a division into a right (low-order) and left (high-order) byte; the bits are always numbered from 0 to 7.

**Examples:**

The STEP 5 data addresses (at the left in the table below) are converted into the STEP 7 data addresses on the right in the same table.

| STEP 5 | STEP 7 |
|--------|--------|
| DW 10 | DBW 20 |
| DL 10 | DBB 20 |
| DR 10 | DBB 21 |
| D 10.0 | DBX 20.0 |
| D 10.8 | DBX 21.0 |
| D 255.7 | DBX 511.7 |

**Differences between SIMATIC S7 and SIMATIC S5**

The following points distinguish S7 from S5:

- Directory for the additional command lists DZB:

  The DW number of the first word in the field is entered in the start address.

  The following formula is used to calculate the start address:

  "S7 DBW number / 2"

  **Example**:

  If the field starts at DBW 372; the start address has the value:
  $$372 / 2 = \mathbf{186}$$

- Directory for the measured value memory DM:

  The DW number of the first word in the field is entered in the start address.

  The following formula is used to calculate the start address:

  "S7 DBW number / 2"

  **Example**:

  If the field starts at DBW 372; the start address has the value:
  $$32 / 2 = \mathbf{16}$$

- Block parameter *F-NR*:

| Error Number, KH.. | Description |
|---|---|
| 00 | No errors have occurred while processing the function |
| 01 | Parameter *BEF* incorrectly assigned |
| 02 | Parameter *PAR* incorrectly assigned |
| 03 | Parameter *STEU* incorrectly assigned |
| 04 | Parameter *DBNR* incorrectly assigned [1] |
| 05 | Parameter assignment data block not available [1] |
| 06 | Parameter assignment data block too short [1] |
| 07 | Parameter *ABIT* incorrectly assigned [2] |
| 08 | Incorrect module ID |
| 09 | Timeout |
| 0B | Incorrect CPU ID or firmware release [2] |
| 0E | Function (FC) release and IP 242B firmware are not compatible |

| Error Number, KH.. | Description |
|---|---|
| 0F | Measured value data block not available [1] |
| 10 | Measured value data block too short |
| 11 | Error in directory for measured value memory |

[1]   The error numbers 04, 05, 06 and 0F cannot be reported via the parameter *F-NR*. With these error numbers, the CPU goes into STOP mode if no error OB was configured.

[2]   These error numbers no longer exist in SIMATIC S7.


**Note on FC185:**

In the interrupt OB, the parameter KMD must be assigned so that the interrupt for IP 242B is acknowledged.

## C.3    Programming Example

**Overview**

The programming example described below shows how the functions FC183 (ZYK_242B) and FC184 (INT_242B) can be used. The example describes how the module can have parameters assigned with the function FC183 (ZYK_242B) directly and indirectly. All blocks required for the program to execute are available.

The programming example should achieve the following:

- It is intended to show examples of the most important functions

- It allows you to check that the hardware (for example, sensors) connected is in full working order

- It is simple and easy to follow

- It can be extended for your own use with little effort on your part

The example can be run with only a minimum of hardware (3 bytes for inputs, 3 bytes for outputs). S7 Status (Monitoring and Modifying) is generally used.

**Direct/Indirect Parameter Assignment**

Direct or indirect parameter assignment of the function FC183 (ZYK_242B) is selected via a digital input. Individual commands can also be passed to the module via additional digital inputs. Any errors which may occur and the counter interrupts are displayed at the digital outputs.

With indirect parameter assignment of the function FC183 (ZYK_242B) you can use every command in the sample program. With direct parameter assignment the following commands are preset:

| Command BEF | Description | Parameter PAR |
|---|---|---|
| B#(7,1) | PA: Assign counter parameters | Counters 2, 6 + global registers |
| B#(7,5) | PO: Assign counter parameters (without command lists) | Counters 2, 6 + global registers |
| B#(7,2) | PS: Save parameters | Counters 1 to 7, global registers |
| B#(7,3) | PZ: Rewrite parameters | Counters 1 to 7, global registers |
| B#(0,5) | KS: Write constant registers | – |
| B#(8,2) | BB: Process command list | Additional command list 1 |
| B#(8,3) | MR: Read and reset measured value set | Measured value sets 1, 2 and 3 |
| B#(8,4) | ML: Read measured value set | Measured value sets 1, 2 and 3 |
| B#(3,1) | LS: Load and start counter | Counters 2, 3, 4 and 6 |
| B#(3,3) | RZ: Reset counter | Counters 2 and 6 |

| Command BEF | Description | Parameter PAR |
|---|---|---|
| B#(3,4) | AW: Transfer interrupt value | Counters 2 and 6 |
| B#(1,5) | SZ: Step counter | Counter 2 |
| B#(0,6) | ZA: Update counter states | – |

**Interrupt Processing**

For interrupt processing, the function FC184 (INT_242B) is used with "Acknowledge interrupt".

**Device Configuration**

The devices listed below are examples of those which can be used to try out the sample program:

- An S7-400 programmable controller system (rack, power supply unit, CPU)

- An adapter module

- An IP 242B counter module

- A rotary transducer (5 V) with two pulse trains in phase quadrature

- One digital input module and one digital output module

- A programming device (such as a PG 740)

It is possible to do without both the digital inputs and the digital outputs if all functions are executed with the S7 Status function. This would require changes in organization block OB1.

**Settings in the CPU**

When you configure the hardware, you must set the addresses for the adapter module via STEP 7. In the example, the following settings in the input area have been assumed:

- S7 address:        512,

- S5 address:        0,

- Length:             0,

- Process image
  subarea:            0,

- Area:                P

The following interrupt settings are required in the CPU:

- Hardware interrupt:  OB40,

- Interrupt:            I1 (S5 assignment: IA).

## Jumper Assignments on the Counter Module

**Introduction**    You must make the following switch and jumper settings before you insert the module. Any jumpers and switches not listed remain set to their factory settings.

**Switch S1**    A = on –> interrupt triggered via interrupt circuit A

| on | | | |
|----|-----|-----|-----|
| | off | off | off |
| A | B | C | D |

**Switch S2**    S2.10 = off –> hardware interrupts via input byte IB 0 disabled

| on | on | on | on | on | on | on | on | | |
|----|----|----|----|----|----|----|----|-----|-----|
| | | | | | | | | off | off |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |

**Switch S3**    Page addressing, page address = F400H

| on | on | on | on | | on | | on |
|----|----|----|----|-----|----|-----|----|
| | | | | off | | off | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

**Switch S4**    Page number = 3

| | | | | | | on | on |
|-----|-----|-----|-----|-----|-----|----|----|
| off | off | off | off | off | off | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

**Jumpers 1 and 2**    Open

**Jumpers 3 to 17**    Plug in position "2–3"

# Input, Output and Bit Memory Address Area Assignments, Block Assignments

**Introduction**  The program is designed so that it can easily be adapted to different inputs and outputs. The programming example only works with memory bits. These memory bits are assigned to the inputs and outputs used in the organization block OB1. In the example these are the input word IW 4, the input byte IB 6, the output word QW 8 and the output word QW 10.

**Inputs and Outputs**  The tables below show the signals for the digital inputs and outputs, together with the associated memory bits.

| Signal | Memory Bit | Direct Parameter Assignment | Indirect Parameter Assignment |
|--------|-----------|-----------------------------|-------------------------------|
| I 4.0 | M 4.0 | PA: Assign parameters C2 + C6 + global registers | Control word bit 8 |
| I 4.1 | M 4.1 | PO: Assign parameters without command lists C2 + C6 + global registers | Control word bit 9 |
| I 4.2 | M 4.2 | KS: Write constant registers | Control word bit 10 |
| I 4.3 | M 4.3 | PS: Save parameter C1 to C7 + global registers | Control word bit 11 |
| I 4.4 | M 4.4 | BB: Process command list List 1 | Control word bit 12 |
| I 4.5 | M 4.5 | – | Control word bit 13 |
| I 4.6 | M 4.6 | MR: Read and reset measured value set Sets 1, 2 and 3 | Control word bit 14 |
| I 4.7 | M 4.7 | ML: Read measured value set Sets 1, 2 and 3 | Control word bit 15 |

| Signal | Memory Bit | Direct Parameter Assignment | Indirect Parameter Assignment |
|--------|-----------|-----------------------------|-------------------------------|
| I 5.0 | M 5.0 | LS: Load and start C2 to C4 | Control word bit 0 |
| I 5.1 | M 5.1 | SZ: Step C2 | Control word bit 1 |
| I 5.2 | M 5.2 | RZ: Reset C2 | Control word bit 2 |
| I 5.3 | M 5.3 | AW: Transfer interrupt value C2 | Control word bit 3 |
| I 5.4 | M 5.4 | LS: Load and start C6 | Control word bit 4 |
| I 5.5 | M 5.5 | RZ: Reset C6 | Control word bit 5 |
| I 5.6 | M 5.6 | AW: Transfer interrupt value C6 | Control word bit 6 |
| I 5.7 | M 5.7 | ZA: Update counter states | Control word bit 7 |

| Signal | Memory Bit | Direct Parameter Assignment | Indirect Parameter Assignment |
|--------|------------|-----------------------------|-------------------------------|
| I 6.0 | M 6.0 | Acknowledge error | Acknowledge error |
| I 6.1 | M 6.1 | Acknowledge interrupt | Acknowledge interrupt |
| I 6.2 | M 6.2 | – | Control with IW 4 |
| I 6.3 | M 6.3 | = '0' direct parameter assignment | = '1' indirect parameter assignment |
| I 6.4 | – | Not assigned | |
| I 6.5 | – | Not assigned | |
| I 6.6 | – | Not assigned | |
| I 6.7 | – | Not assigned | |

| Signal | Memory Bit | Description |
|--------|------------|-------------|
| Q 8.0 | M 16.0 | Error in start-up program |
| Q 8.1 | M 16.1 | Error in interrupt program (OB40) |
| Q 8.2 | M 16.2 | Error in cyclic program (OB1) |
| Q 8.3 | M 16.3 | – |
| Q 8.4 | M 16.4 | – |
| Q 8.5 | M 16.5 | Interrupt at counter 2 |
| Q 8.6 | M 16.6 | Interrupt at counter 6 |
| Q 8.7 | M 16.7 | Interrupt triggered (group interrupt display) |

| Output Byte | Description |
|-------------|-------------|
| QB 9 | Parameter MELD (in case of error in the cyclic program) |
| QB 10 | Parameter F-NR (in case of error in the cyclic program) |

**Bit Memory Area Assignments**

The table below shows the assignment of the bit memory address area.

| Bit Memory Area | Description |
|---|---|
| MW 4 | Memory word for input word IW 4 |
| MB 6 | Memory byte for input byte IB 6 |
| MW 8 | Edge memory word for input word IW 4 |
| MB 10 | Edge memory byte for input byte IB 6 |
| MW 12 | Pulse memory word for input word IW 4 |
| MW 14 | Pulse memory word for input byte IB 6 |
| MW 16 | Memory word for output word QW 8 |
| MW 18 | Memory word for output word QW 6 |
| MW 20 | Loop counter for start-up program |
| MW 30 | Parameter ERR, start-up and interrupt program |
| MB 32 | Parameter MELD, start-up and interrupt program |
| MB 33 | Parameter F-NR, start-up and interrupt program |
| MW 34 | Parameter ERR, cyclic program |
| MB 36 | Parameter MELD, cyclic program |
| MB 37 | Parameter F-NR, cyclic program |
| MW 40 | Parameter IIR interrupt information register |

**Block Assignments**

The table below shows the assignment of the data blocks and logic blocks used.

| Block | Description |
|---|---|
| DB182 | Measured value data block |
| DB183 | Parameter assignment data block |
| FC82 | Example for indirect parameter assignment of the function FC183 (ZYK_242B) |
| FC83 | Example for direct parameter assignment of the function FC183 (ZYK_242B) |
| FC183 | Process counter module (cyclic program execution) |
| FC184 | Process counter module (interrupt processing) |
| FC220 | Called by OB100 |
| OB1 | Cyclic program |
| OB40 | Hardware interrupt program |
| OB100 | Complete restart |

## Start-Up Program and Error Messages

**Start-Up Program**
After executing a CPU memory reset (STOP mode), download the entire example to the CPU. Then move the mode selector from STOP to RUN_P (**start-up characteristics CRST**).

The start-up program is in OB100.

The counter module is assigned parameters in the start-up organization block. The parameter assignment data block DB183 contains valid parameter data and does not need to be changed. You can recognize whether the parameter assignments were successful if the green RUN LED on the counter module is lit continuously.

If all inputs have the signal state '0' when you switch on the S7-400, no output may be set after the CPU starts up.

**Response to Errors**
If the output Q 8.0 is set, a parameter assignment error occurred during startup. You can determine the exact cause of the error with the aid of the memory word MW 30 (parameter *ERR*), the memory byte MB 32 (parameter *MELD*) and the memory byte MB 33 (parameter *F-NR*).

# Cyclic Operation

**General Remarks**

The cyclic program is in OB1.

Using the input I 6.3 you can switch between assigning parameters directly and indirectly to the function FC183 (ZYK_242B).

If the input I 6.3 has the signal state '0', the function FC83 is called for direct parameter assignment. If the input I 6.3 has the signal state '1', the function FC82 is processed for indirect parameter assignment (controlled operation).

For the first test you should select direct parameter assignment (I 6.3 = '0') because the commands can be triggered via the respective input signals.

**Direct Parameter Assignment**

With direct parameter assignment of the function FC183, the command to be executed is specified directly with the parameter *BEF*. The counters and global registers are selected via the parameter *PAR*.

The commands programmed in the function FC83 can be triggered via the respective input signals.

**Indirect Parameter Assignment**

With indirect parameter assignment the command to be executed is specified together with the counter bits or the bit for the global registers in the parameter *STEU* in the function FC183.

In the example, the control word is preset via the input word IW 4. The command whose control word you set at input word IW 4 is executed with the rising edge of the input I 6.2.

**Error Messages**

If an error occurs in the cyclic program, the output Q 8.2 is set. You can determine the cause of the error with the aid of the memory word MW 30 (parameter *ERR*), the memory byte MB 32 (parameter *MELD*) and the memory byte MB 33 (parameter *F-NR*).

These memory bytes MB 32 and MB 33 are also mapped to the output bytes QB 9 and QB 10.

You delete the error display again using the input I 6.0.

## Direct Parameter Assignment

**PA**      **Assign parameters to counters 2, 6 and global registers (I 4.0):**

You use this command to assign parameters to the counter 2, counter 6 and the global registers. The parameter data are in the parameter assignment data block DB183.

**PO**      **Assign parameters to counters 2, 6 and global registers (I 4.1):**

You use this command to assign parameters to the counter 2, counter 6 and the global registers. The parameter data are in the parameter assignment data block DB183.

In contrast to the PA command, the command lists are not transferred to the module with the PO command. This makes the execution time for the PO command shorter than that for the PA command.

**KS**      **Write constant registers (I 4.2):**

You use this command to transfer the 16 constant registers to the counter module. The constant registers are in the parameter assignment data block in the data words DBD 292 to DBD 352 (symbols: KON0 to KON15).

Every constant occupies two data words in the parameter assignment data block. You can use the constant registers for calculations in the command lists.

The command KS enables you to quickly change the constant register specified in the parameter assignment (PA command).

**PS**      **Save parameters (I 4.3):**

You use this command to store the parameter data for all counters and the global registers, all constant registers, the additional command lists and the measured value directory permanently in the EEPROM of the counter module. The counter module then has these data assigned as parameters. The parameter data are in the parameter assignment data block DB183. The sample program contains a parameter assignment data block with valid parameter data.

The counter module allows you to store a number of data records at a time in the EEPROM. The sample program uses the EEPROM data record zero.

**BB**                            **Process command list (I 4.4):**

You use this command to process the additional command list 1 in the
counter module. The commands in command list 1 are in parameter
assignment data block DB183 from data word DBW 372 (symbols: B1 to
B11).

In order to be able to execute the command list, it must have been
downloaded to the module once with the command PA (assign module
parameters). The bit for the global registers must be set in the parameter
PAR.

The following command list was programmed in the example:

| Symbol | (Data Word) | Code (Hex.) | Command | | Description |
|--------|-------------|-------------|---------|---|-------------|
| B1 | (DBW 372) | 9063 | L | C3 | Load counter state of counter 3 |
| B2 | (DBW 374) | 9464 | SUB | C4 | Subtract counter state of counter 4 |
| B3 | (DBW 376) | 9141 | T | E1 | Transfer result to result register 1 |
| B4 | (DBW 378) | 9051 | L | K1 | Load constant 1 |
| B5 | (DBW 380) | 9052 | L | K2 | Load constant 2 |
| B6 | (DBW 382) | A741 | FIN | E1 | Is E1 within the window K1, K2? |
| B7 | (DBW 384) | A003 | JUR | +3 | |
| B8 | (DBW 386) | 9142 | T | E2 | No:    Transfer E1 to E2 |
| B9 | (DBW 388) | A002 | JUR | +2 | |
| B10 | (DBW 390) | 9143 | T | E3 | Yes:   Transfer E1 to E3 |
| B11 | (DBW 392) | 9171 | T | A1 | Transfer accumulator 1 (equals NOP) |

**MR**                            **Read and reset measured value set (I 4.6):**
**ML**                            **Read measured value set (I 4.7):**

You use these commands to read the measured values in the result registers 1,
2 and 3.

In contrast to the command ML, the command MR also resets the measured
values when it reads them. "Reset" means that the liquid level indicator is set
to the first value in the measured value set.

**LS**                            **Load and start counters 2, 3 and 4 (I 5.0):**
                                   **Load and start counter 6 (I 5.4):**

You use these commands to load and start the relevant counter with the value
in the corresponding load register.

**RZ**                            **Reset counter 2 (I 5.2):**
                                   **Reset counter 6 (I 5.5):**

You use these commands to reset counters 2 and 6 to the value 0.

**AW**                    **Transfer interrupt value for counter 2 (I 5.3):**
                          **Transfer interrupt value for counter 6 (I 5.6):**

You use these commands to transfer the interrupt value of the respective counter.

**SZ**                    **Step counter 2 (I 5.1):**

You use this command to move the counter state of the counter 2 on by one pulse. In the example, the count direction is decrementing (counting down). Counting is performed without conditions.

**ZA**                    **Update counter states (I 5.7):**

You use this command to read the counter states of all counters, the result registers and the counter status register of the counter module and enter them in the parameter assignment data block.

Data areas read in the parameter assignment data block DB183:

| ZSR1 to ZSR3 | Counter status register | DBW 240 to DBW 244 |
|---|---|---|
| C1 to C5 | Counter states of counters 1 to 5 | DBW 246 to DBW 254 |
| C6 to C7 | Counter states of counters 6 and 7 | DBD 256 and DBD 260 |
| ERG 1 to ERG7 | Result registers 1 to 7 | DBD 264 and DBD 288 |

When testing, this command should always be selected (I 5.7 = '1') so that you can directly monitor the effects of each individual command on the counters.

## Indirect Parameter Assignment

**Notes**

You select indirect parameter assignment of the function FC183 using the input I 6.3 = '1'. The function itself is called in the example with the rising edge at input I 6.2. The command BEF = 255,255 switches to the parameter *STEU*. This means that instead of a command from the parameter *BEF* with the additional command from the parameter *PAR*, the control word passed with the parameter *STEU* is now transferred to the module. The content of the parameter *STEU* can be specified bit-by-bit in the example via the input word IW 4.

The control word must be specified completely (command code with selection bit for the global registers and counters 1 to 7).

If the specified command (left byte of the control word, in the example the input byte IB 4) corresponds to a valid command, a check is run by the function FC183 to ensure that the parameters are correctly assigned. (In the example, the right byte of the control word is set at input byte IB 5.)

If no valid command code is found, the function passes the assigned control word to the module without change and without a check. In this case, the counter module checks the control word and may reject it with an error message in the parameter *ERR*.

## Assigning Counter Module Parameters

**Presetting the Global Registers**

The global registers have the following presets:

- **Master mode register MMR (DBW 8):**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1  | 0  | 0  | 0  | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

Bit 3 = 1:                   Comparator active

Bit 15 = 1:              BCD pitch (spacing factor = 10)

- **Prescaler register VTR (DBW 10):**

DBW 5 = 100

Prescaler = 1:100

The settings in the prescaler register and in the master mode register bit 15 result in a clock frequency of 1 MHz:100:10:10:10 = 10 Hz for the module-internal clock pulse F4. For the clock pulse F3, a clock frequency of 100 Hz results. The count pulses F3 and F4 are required for the counters 3 and 4.

- **Interrupt enable register IFR (DBW 14):**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 1  | 1  | 1  | 1  | 1  | 1  | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Bit 0 = 1:                   Enables group interrupt in case of error message

Bit 8 = 1:                   Enables group interrupt S7 in case of counter events (overflow, comparator, gate edge)

Bits 9 to 15:            Enables outputs 1 to 7

**Presets for Counter 2**

The counter 2 has the following presets:

- **Counter mode register CMR2 (DBW 60):**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0  | 0  | 0  | 0  | 0  | 0  | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |

| | |
|---|---|
| Bits 0 to 2 = 001: | Output signal: pulse, active high |
| Bit 3 = 0: | Count direction backwards |
| Bit 4 = 1: | Count mode BCD |
| Bits 5 to 7 = 001: | Count mode D (periodic counting process, reloading from load register) |
| Bits 8 to 11 = 0010: | Count pulse source = counter input C2 |
| Bit 12 = 0: | Count pulse edge = rising |
| Bits 13 to 15 = 000: | Without gate control |

- **Load register LR2 (DBW 62):**

  DBW 62 = W#16#10:    Load register = 10

- **Interrupt register AR2 (DBW 66):**

  DBW 66 = W#16#5:    Interrupt register = 5

**Presets for Counter 3**

The counter 3 has the following presets:

- **Counter mode register CMR3 (DBW 88):**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0  | 0  | 0  | 0  | 1  | 1  | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |

| | |
|---|---|
| Bits 0 to 2 = 000: | Output signal: switched off (low level) |
| Bit 3 = 0: | Count direction backwards |
| Bit 4 = 1: | Count mode BCD |
| Bits 5 to 7 = 001: | Count mode D (periodic counting process, reloading from load register) |
| Bits 8 to 11 =1110: | Count pulse source = frequency F4 (10 Hz) |
| Bit 12 = 0: | Count pulse edge = rising |
| Bits 13 to 15 = 000: | Without gate control |

- **Load register LR3 (DBW 90):**

  DBW 90 = W#16#1000:  Load register = 1000

**Presets for
Counter 4**

The counter 4 has the following presets:

- **Counter mode register CMR4 (DBW 106):**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |

| | |
|---|---|
| Bits 0 to 2 = 000: | Output signal: switched off (low level) |
| Bit 3 = 0: | Count direction backwards |
| Bit 4 = 1: | Count mode BCD |
| Bits 5 to 7 = 001: | Count mode D (periodic counting process, reloading from load register) |
| Bits 8 to 11 =1101: | Count pulse source = frequency F3 (100 Hz) |
| Bit 12 = 0: | Count pulse edge = rising |
| Bits 13 to 15 = 000: | Without gate control |

- **Load register LR4 (DBW 118):**

  DBW 118 = W#16#0100:  Load register = 100

**Presets for
Counter 6**

The counter 6 has the following presets:

- **Counter mode register CMR6 (DBW 172):**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |

| | |
|---|---|
| Bit 3 = 1: | Output signal active |
| Bits 4 to 6 = 001: | Simple edge evaluation at input A, input B determines count direction |
| Bit 7 = 0: | Counter reset disabled |

- **Load register LR6 (DBD 174):**

  DBD 174 = W#16#600:  Load register = 600

- **Interrupt register AR6 (DBD 182):**

  DBD 182 = W#16#500:  Interrupt register = 500

## Interrupt Processing

**Notes**

Interrupt processing is programmed in the organization block OB40 in this example.

The interrupts are detected via the interrupt circuit I1 (IA).

If an error occurs during interrupt processing, the output Q 8.1 is set. To determine the exact cause of the error, you must evaluate the memory word MW 30 (parameter *ERR*) and the memory bytes MB 32 (parameter *MELD*) and MB 33 (parameter *F-NR*).

Once the error has been corrected you can acknowledge the error display with the input I 6.0. The error display Q 8.1 is reset with this input.

**Generating Interrupts with the Interrupt Value**

Each counter in the counter module has an interrupt register in the parameter assignment data block DB183. The data for the individual counters are as follows:

| AR1 | Interrupt register for counter 1 | DBW 38 |
|-----|----------------------------------|---------|
| AR2 | Interrupt register for counter 2 | DBW 66 |
| AR3 | Interrupt register for counter 3 | DBW 94 |
| AR4 | Interrupt register for counter 4 | DBW 122 |
| AR5 | Interrupt register for counter 5 | DBW 150 |
| AR6 | Interrupt register for counter 6 | DBD 182 |
| AR7 | Interrupt register for counter 7 | DBD 216 |

You can change the interrupt values with S7 Status.

The interrupt value 0 is not permitted for the counters 1 to 5.

The interrupt value of counter 2 is transferred to the counter module with the input I 5.3 and the interrupt value of counter 6 is transferred with the input I 5.6.

If the counter interrupts are enabled in the interrupt enable register IFR (DBW 14) and all comparators are switched on in the master mode register MMR (DBW 8), an interrupt is triggered if the counter state of a counter is equal to its interrupt register content.

The comparator function of counter 6 and counter 7 is selected with bit 3 of the corresponding counter mode register.

**To:**
**Siemens AG**
**Bereich Automatisierungstechnik**
**Kombinationstechnik (B1.2−T1)**
**Postfach 2355**

**D−90713 Fürth/Bay.**

Suggestions

Corrections

**For equipment manual**

_____

_____

**Order no.** _____

From:

Name _____

Company/department _____

Address _____

Telephone _____ / _____

If you find typographical errors while reading this document, please use this form to let us know.

We would also be grateful for your suggestions, remarks or ideas.

**Please fill in the order no. of the affected document.**

**Suggestions or/and corrections:**