Function module RMD
75720010

Technical Dokumentation

The function module is a maintenance-free, programmable controller for use in the EIB. As a supplement to the decentral system the extensive functions library can be used to solve various control and automation tasks. EIB telegrams from different EIS are received, processed and the results transmitted to the bus within a defined period. The function modules have an internal real-time clock which can be used both as a time switch and as a timer.
The function module can be supplied in two types:

1. **Function module RMD (75720010)**
2. **Mini-function module RMD (75720011)**

The types differ mainly in the extent of the functions they realise.

Configuring and start-up is carried out with the help of a software oriented towards a functions plan (FM tool with dongle / mini FM tool). The program is processed cyclically and therefore operates the module in real time mode. The module is connected to the system via the integrated bus coupling unit. It is installed on the DIN rail and supplied with power via the 29 V/DC contacts of the data rail (outer pair of conductors).

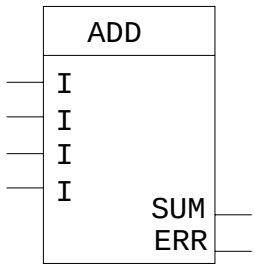
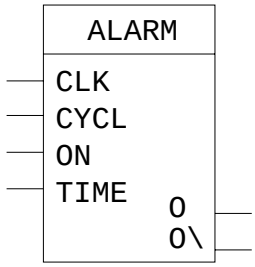
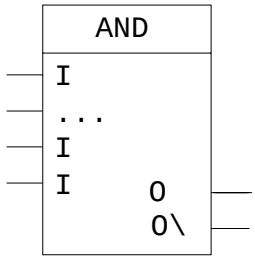
Technical data:

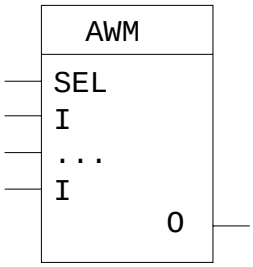
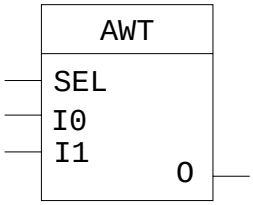
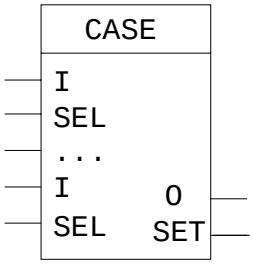
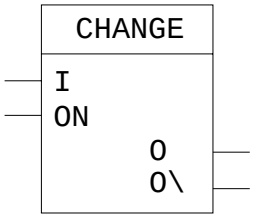
Product type	**Function module RMD**	**Mini-function module RMD**
Order no.:	75720010	75720011
Programming software:	Tool software with dongle and manual (75700011) not included in price	Mini-tool software with online documentation included in price
EIB connector:	Integrated bus coupling device	
Programming:	Interface and bus	Only bus
Distribution voltage:	External data rails with 29V D (20...30 V unrestricted)	
Power consumption:	Normal operations 40 mA (ca. 8 device units), initializing / programming: < 100 mA	
Back-up for real-time clock:	Capacitative back-up for min 24 hours	
Dimensions:	90 x 54 x 57 (H,B,T), 3TE	
No. of functions blocks:	Depends on the number of group addresses and the functions blocks used e.g.: 2000 functions blocks with 500 group addresses input group addresses output	max. 150 per appliance with max. 200 group addresses (inputs/outputs)
Program processing cycles: **(task times)**	Multitasking (min. 20 ms) Power-Fail-Task and initializing task	1 task: fixed at 100 ms
Telegram buffer:	Input: 50 telegrams Output: 100 telegrams	

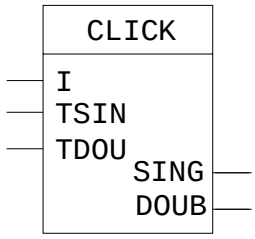
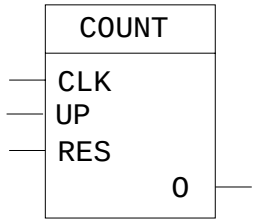
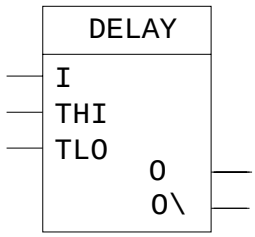
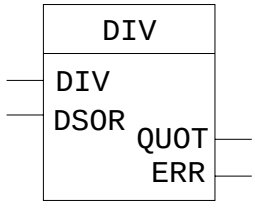
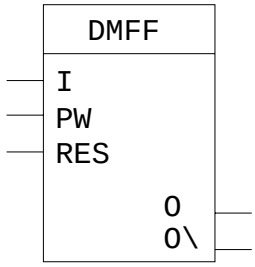
Data types:

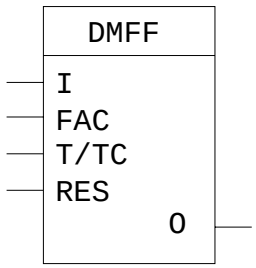
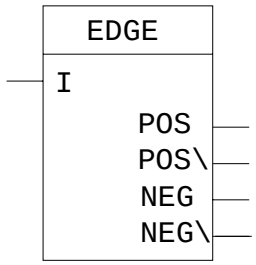
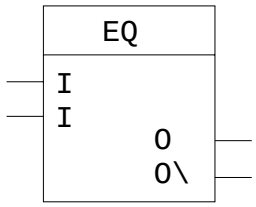
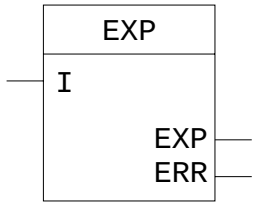
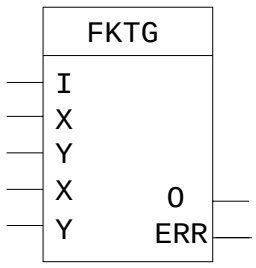
Data type	Function module RMD	Mini-function-module RMD	Format	Value range	For EIS type
Bit	**X**	**X**	1 bit , smallest unit of representation (ON / OFF; UP / DOWN; TRUE / FALSE)	0/1	EIS 1 EIS 2 EIS 7
byte	**X**	**X**	8 bit, Not interpreted or as an unsigned integer	0...255	EIS 2 (VALUE) EIS 6
Word	**X**	**X**	16 bit (2 byte) as number	0...65535	EIS 10 (CODE 10000)
Sint	**X**	**-**	16 bit (2 byte) as number with sign	- 32768 .....0.....+32767	EIS 10 (CODE 100001)
Value	**X**	**-**	16 bit (2 byte), only required in EIB systems, used for representation of floating decimal point	- 671088.64.....0..... +670760.96	EIS 5
Time	**X**	**X**	24 bit (3 byte) used for representation of time	24 hours, day of week	EIS 3
Date	**X**	**X**	24 bit (3 byte) used for representation of date	Day, month, year	EIS 4
Forced guidance	**X**	**X**	2 bit only required in EIB systems, Priority	Control/no control Enable/disable	EIS 8
Control	**X**	**X**	4 bit The name of the first bit is up/down, the other 3 bits specify the step size for dimming	Dimming direction, step	EIS 2
Max	**X**	**-**	To 14 bytes It is possible to specify how many bytes are to be received or transmitted on the bus	Different, free	EIS 9, EIS 11

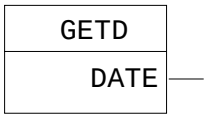
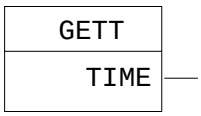
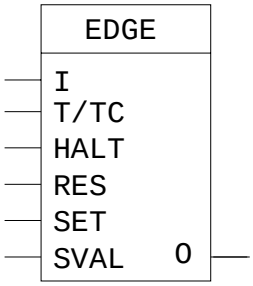
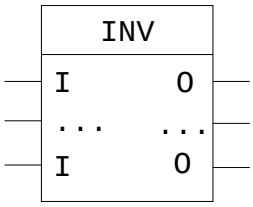
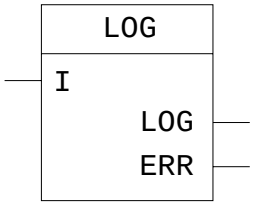
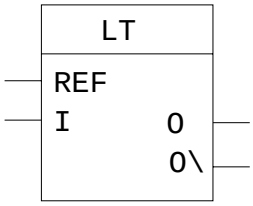
Functions block library

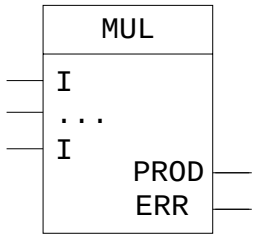
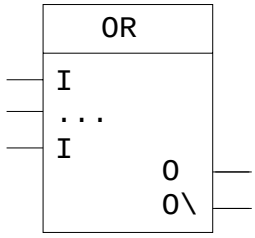
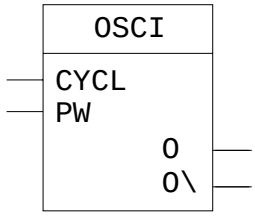
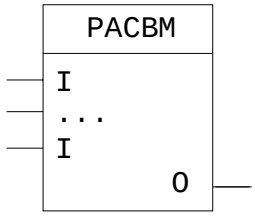
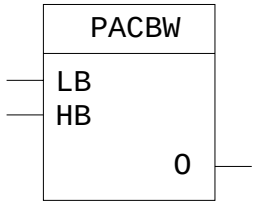
Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
ADD	**X**	**-**	ADD adds 2 to 16 summands to a sum. Overflow generates an error.	I: Summand SUM: Sum ERR: Overflow (BIT)	
ALARM	**X**	**X**	The alarm block functions like an alarm clock. A rising edge on the CLK input sets a timer according to the value which exists on the TIME input. At the same time the timer starts the countdown. Once the timer reaches zero the output is set to 1 for duration of a task cycle. If after the pulse the CYCL input is equal to 1, the timer is started again. The timer status is frozen if ON input is 0. If a second rising edge appears on CLK input before timer reaches zero, the timer is reset to the value wich is specified on the TIME input. If the CLK input is initialised to value 1, then this is interpreted as a rising edge after a Reset and the ALARM block is started.	CLK: Edge-tr. input (bit) CYCL: Cycl. output sig. (bit) ON: Switch On/Off (bit) TIMER: Time in task cycl. (Word) O: Non-inv. output (bit) O/: Inv. output (bit)	
AND	**X**	**X**	Logical AND operation with 2 to 16 inputs and non-inverted and inverted output.	I: Input signal (bit) O: Output signal (bit) O\: Inverted output (bit)	

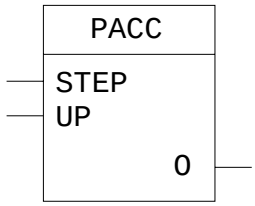
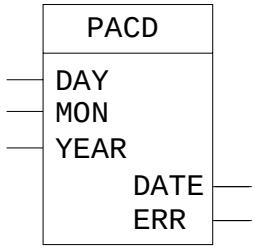
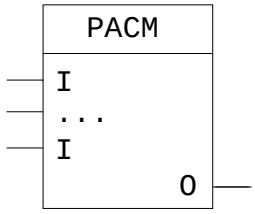
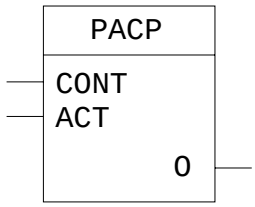
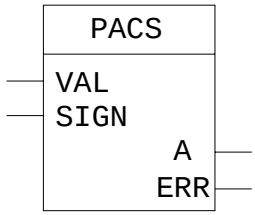
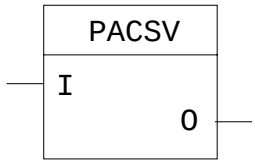
Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
AWM	**X**	**X**	The input selection switch connects one of the 1-16 inputs with the output. If SEL does not have a valid value, output remains unchanged.	SEL: Selection of input (Word) I: Input signals O: Output signals	
AWT	**X**	**X**	The double output selection switch connects one of the two inputs with the output. If SEL is 0 input I0 is selected, otherwise I1 is.	SEL: Selection selects input (bit) I0: Input signal 0 I1: Input signal 1 O: Output signals	
CASE	**X**	**X**	CASE copies the input with active selection bit to the output. The output signal SET is set when output O has changed. If more than one selection bit is active the value of the last selected input is transferred to output. The number of inputs can vary between 1 to 8. Data types: WORD, DATE, TIME, VALUE	I: Input signals SEL: Selection selects input (bit) O: Output signals SET: Output (bit) = OR links of all SEL	
Change	**X**	**X**	Shows that a signal has changed since the last task cycle. The function can be switched ON and OFF with the ON input.	I: Input signal (BIT) ON: Switch On/Off (BIT) O: Output signal (bit) O\: Inverted output (bit)	

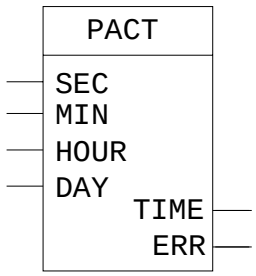
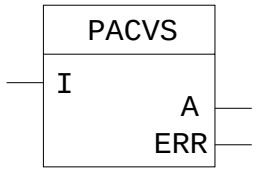
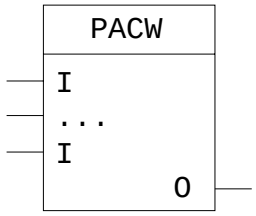
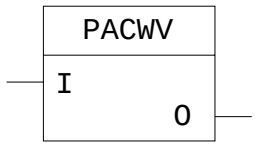
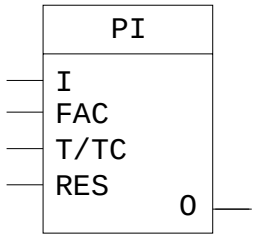
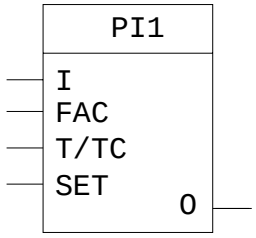
Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
CLICK	**X**	**X**	CLICK detects whether the input signal is a single or a double click. If only one pulse occurs during time TDOU (time double), the TSIN (single) output is set to 1 for the duration of a task cycle. If a second pulse occurs after the time TSIN the DOUB output is set to 1 for one task cycle. If two pulses occur during the time TSIN then this is evaluated as one pulse.	I: Input (bit) TSIN: Time in task cycle in which 2 pulses are seen as single click (Word) TDOU: Time in task cycles, after which outputs are set (Word) SING: Output: display single click (bit) DOUB: Output displays double click (bit)	
COUNT	**X**	**X**	Forward/reverse counter with reset input. Only rising edges are counted.	CLK: Edge-tr. input. (BIT) UP: Count direction (bit, 1 = up) RES: RESET (BIT) O: OUTPUT (SINT)	
DELAY	**X**	**X**	The delay module delays the rising and falling edges of input signals by a variable time. If the values on THI (time rising) or TLO (time falling) input change during the delay of output signal than this has no effect on the running delay.	I: Input signal (bit) THI: Delays rising edge (Word) TLO: Delays falling edge (Word) O: Output signal (bit) O\: Inverted output (bit)	
DIV	**X**		DIV divides 1 to 16 divisors from a dividend. Division by zero or overflow generates an error and the overflow Bit is set to 1. Data types: Word, Sint and Value	DIV: Dividend DSOR: Divisors QUOT: Quotient (result ERR: Division error (bit)	
DMFF	**X**	**X**	Dynamic, retriggerable monostable flipflop with variable pulse duration and reset input. With a rising edge on the input a pulse is generated on the output. The duration of the pulse is set on the PW input. If during pulse another rising edge occurs, pulse duration is reset and pulse is retriggered. The RES input always sets output signal to zero.	I: Edge-trigg. input. (bit) PW: Pulse duration in task cycles (Word) RES: Reset O: Non-inv. output (bit) O\: Invert. output (bit)	

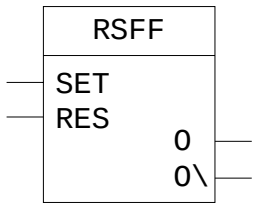
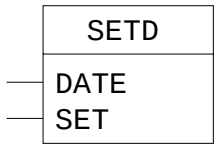
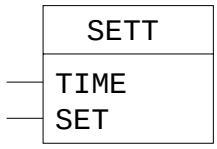
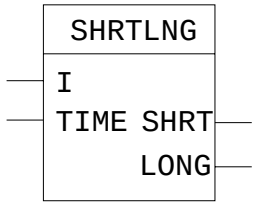
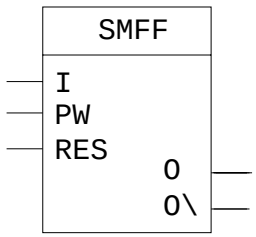
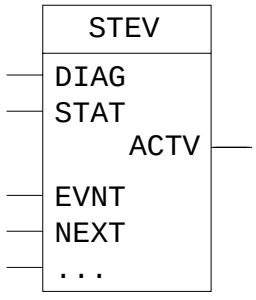
Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
DT1	**X**	**-**	DT1 is a element for regulation tasks. It is based on approximation algorithm from Gauss. Reset sets output to zero. The output value is limited to the value range of Value data type. If T/TC is negative or zero, or FAC is negative the output value does not change.	I: Input signal (Value) FAC: Proportional fact. (Value) T/TC: Time constant (Value) RES: Reset (bit) O: Output signal (Value)	
EDGE	**X**	**X**	EDGE detects positive and negative signal edges. The output is set for one task cycle.	I: Input signal POS: Non-inv. output Rising edge (bit) POS\: Invert. output Rising edge (bit) NEG: Non-inv. output Falling edge NEG\: Invert. output Rising edge (bit)	
EQ	**X**	**X**	If the two inputs are equal, the non-inverted is 1 and the inverted output 0.	I: Inputs for comparison O: Non-inv. output (bit) O\: Invert. output (bit)	
EXP	**X**	**-**	EXP calculates the exponential function with the base e (e=2.71828). Overflow generates an error.	I: Input signal (Value) EXP: Output signal (Value) ERR: Overflow (bit)	
FKTG	**X**	**-**	The function generator calculates the value of a function that is given by X-Y-coordinates. The function Y=f(X) is defined using between 2 and 8 X-Y coordinates. If the input value is outside of defined function, output signal is undefined and error bit is set.	I: Input signal (Value) X: X-coordinate (Value) Y: Y-coordinate (Value) O: Output signal (Value) ERR: Error (bit)	

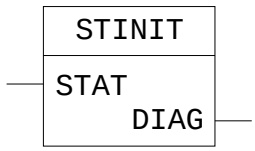
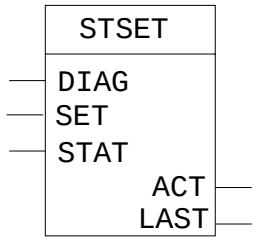
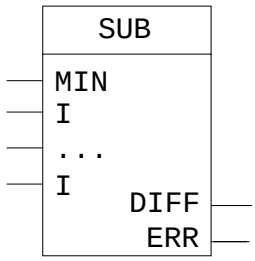
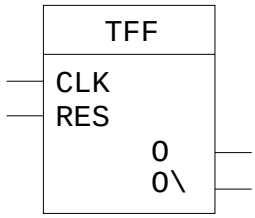
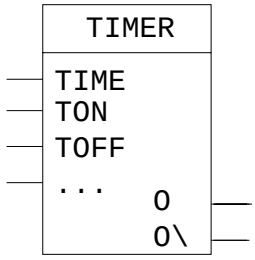
Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
GETD	**X**	**X**	Gives the system date of the function module. This date specifies: day, month and year.	DATE: Date (Date)	
GETT	**X**	**X**	Gives the system time of the function module. The time specifies the day of the week (Monday,…), hours, minutes and seconds.	TIME: Time (Time)	
INTE	**X**	**-**	Integration module with integration constant T/TC, reset and set facility. The integration is based on the Gauss approximation algorithm. With the HALT input integration can be stopped. RES sets the output to zero. If SET is 1, output is set to the value on the SVAL input.	I: Input signal T/TC: Time constant HALT: Stop integration (bit) RES: Reset (bit) SET: Set output signal (bit) SVAL: Set value O: Output signal	
INV	**X**	**X**	INV inverts a binary signal. The number of inputs and outputs is 1 to 8 bits.	I: Input signal (bit) O: Output signal (bit)	
LOG	**X**	**-**	LOG calculates the natural logarithm with the basis e (e=2.71828). Negative input values generate an error.	I: Input signal (Value) LOG: Output signal (Value) ERR: Error (bit)	
LT	**X**	**X**	If the input value is less than the reference, the non-inverted output = 1. Data types: Date, Sint, Time, Value, Word	REF: Reference value I: Value for comparison O: Non-inv. output (bit) O\: Inverted output (bit)	

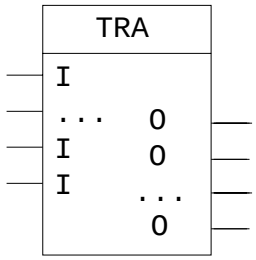
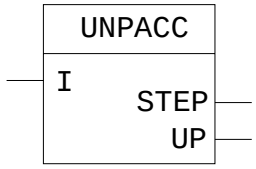
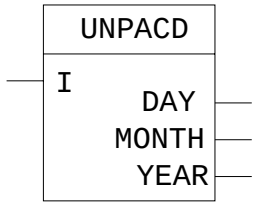
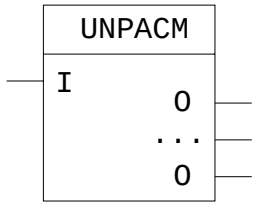
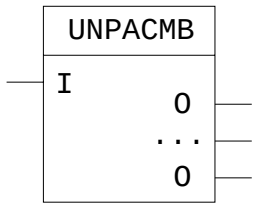
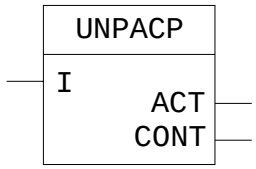
Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
MUL	**X**	**-**	MUL multiplies 2 to 16 factors with one another. Overflow generates an error.	I: Factors PROD: Product ERR: Error, overflow (bit)	
OR	**X**	**X**	Logical OR operation with 2 to 16 inputs and non-inverted and inverted output.	I: Input signal (bit) O: Output signal (bit) O\: Inverted output (bit)	
OSCI	**X**	**X**	Oscillator with programmable cycle time and pulse width. A change of input value on the CYCL or PW input, first has an affect at the beginning of next period.	CYCL: Cycle time in task cycles (Word) PW: Pulse width in task cycles (Word) O: Output signal (bit) O\: Inverted output bit)	
PACBM	**X**		PACBM packs a number of bytes into a type of max. The range is from 1 to 14 bytes.	I: Inputs (byte) O: Output (Max)	
PACBW	**X**	**X**	PACBW packs a low-value byte and a high-value byte into a word.	NB: Low-value byte HB: High-value byte A: Output (Word)	

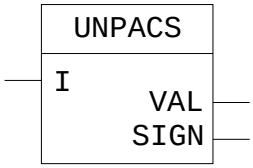
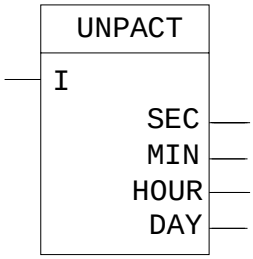
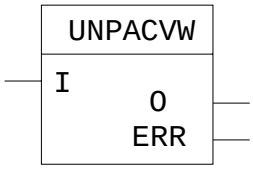
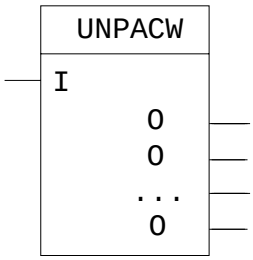
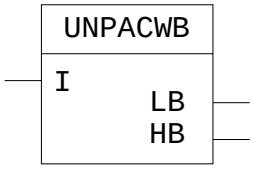
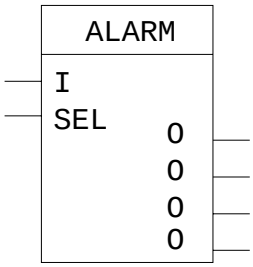
Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
PACC	**X**	**X**	PACC packs a step value (0..7) and a direction bit (0=DOWN/1=UP) in a signal of data type Control.	STEP: Step size (byte) UP: Direction (bit) O: Output (4 bit control)	
PACD	**X**	**X**	PACD forms a data type signal Date from day, month, year signals. Incorrect input values generate an error.	DAY: Day, Value range: byte 1..31 MON: Month, Value range: byte 1..12 YEAR: Year, Value range: byte 0..99 DATE: Date (DATE) ERR: Error (bit)	
PACM	**X**	**-**	PACM packs 1 to 7 signals with data type Word to the data type Max. The low-order byte of input 1 is the first byte on the output.	I: Inputs (Word) O: Output (Max)	
PACP	**X**	**X**	PACP packs a control bit and an active bit to a signal of data type Pcontrol.	KONT: Control bit, bit 1 on bus AKT: Action bit, bit 0 on bus A: Output (Pcontrol 2 bit)	
PACS	**X**	**-**	PACS converts a value of data type Word and a sign bit into a signal of data type Sint. 0 means a positive sign and 1 a negative. Overflow generates an error.	VAL: positive value (word) SIGN: Sign (bit 0/+;1/-) O: Output (Sint) ERR: Overrange (bit)	
PACSV	**X**	**-**	PACSV converts a signal of data type Sint into a Value.	I: Input (Sint) O: Output (Value)	

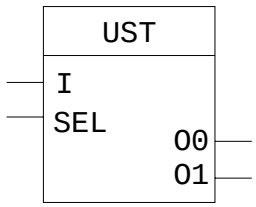
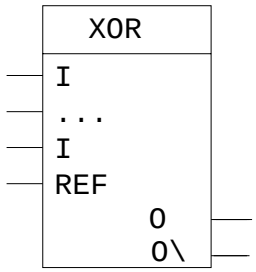
Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
PACT	**X**	**X**	PACT forms a signal of data type Time from seconds, minutes, hours and weekday. Incorrect input values generate an error.	SEC: 0...59 (byte) MIN: 0...59 (byte) HOUR: 0...23 (byte) DAY: 1...7, 0 neutral, (byte) TIME: (Time) ERR: Error (bit)	
PACVS	**X**	**-**	PACVS converts a signal of data type Value to Sint. It is always rounded off. Overflow generates an error.	I: Input (Value; -32768 - +32767) A: Output (Sint) ERR: Overrange (bit)	
PACW	**X**	**X**	PACW packs several Bits into a Word. The number of inputs is variable between 1 to 16. The first bit is the lowest value bit.	I: Inputs (bit) O: Output (Word)	
PACWV	**X**	**-**	PACWV converts a signal of data type Word into a Value.	I: Inputs (Word) O: Output (Value)	
PI	**X**	**-**	PI element for regulation tasks. It consists of a proportional factor and an integrator section. With reset the integrator section is set to zero. A time constant and reset input are available.	I: Input signal (Value) FAC: Proportional factor (Value) T/TC: Time constant /Value) RES: Reset (bit) O: Output signal (Value)	
PT1	**X**	**-**	PT1 element is for regulation tasks. It is based on the approximation algoritm from Gauss. A time constant and set input are available. With SET the output is set to the input value.	I: Input signal (Value) FAC: Proportional factor (Value) T/TC: Time constant /Value) SET: SET (bit) O: Output signal (Value)	

Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
RSFF	**X**	**X**	Reset/set flipflop. If the SET input is 1, the O output is set to 1. If the reset input is 1 the O output is set to 0. Reset has priority. After a reset the O output is 0.	SET: Set input (bit) RES: Reset input (bit) O: Output signal (bit) O\: Inverted output (bit)	
SETD	**X**	**X**	Sets the system date for the function modules to the preset value if SET=1.	DAT: DATE (Date) SET: Set (bit)	
SETT	**X**	**X**	Sets the system time for the function modules to the preset value if SET=1.	TIME: TIME (Time) SET: Set (bit)	
SHRT LNG	**X**	**X**	SHRTLNG detects whether a signal is applied for less or longer than a specified time. Occurs the signal I longer then specified by TIME, so is output LONG otherwise output SHORT for just one task cycle set.	I: Input (bit) TIME: Time (Word) SHRT: Output: displays short Input signal (bit) LONG: Output: displays long Input signal (bit)	
SMFF	**X**	**X**	Static monostable flipflop. It shortens a pulse on the input to an adjustable pulse width with variable pulses. The RES input always sets the output signal to zero.	I: Input signal (bit) PW: Pulse duration (Word) RES: Reset (bit) O: Output signal (bit) O\: Inverted output (bit)	
STEV	**X**	**-**	Status block for status-event diagrams. Please see user manual.	DIAG: Diagram signal (MAX) STAT: Status number (WORD) AKTV: Active (BIT) EVNT: Event inputs (BIT) NEXT: Next status after event (WORD)	

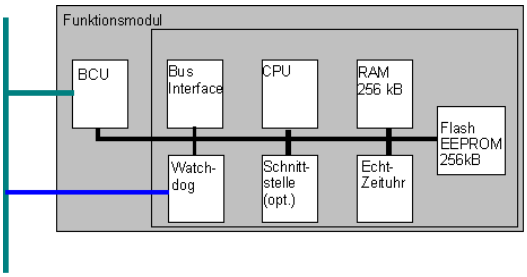
Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
STINIT	**X**	**-**	Initialises a status-event diagram. Please see user manual.	STAT: Initializing status (WORD) DIAG: Diagram signal (MAX)	
STSET	**X**	**-**	Sets a status-event diagram into a defined status. Please see user manual.	DIAG: Diagram signal (MAX) SET: Set a status STAT: Status to be set ACT: Current status (WORD) LAST: Last status (WORD)	
SUB	**X**	**-**	SUB subtracts 1 to 16 subtrahends from a minuend. Overflow generates an error. Data types: Sint, Value, Word DIFF=MIN – I1 – I2 - … - In	MIN: Minuend (sint, value, word) I: Subtrahend (sint, value, word) DIFF: Difference (sint, value, word) ERR: Overflow (bit)	
TFF	**X**	**X**	Toggle flipflop with reset input. Switches the output value over on each rising edge. Reset always sets the O output to 0.	CLK: Edge-triggered input (BIT) RES: Reset (BIT) O: Output signal (bit) O\: Inverted output (bit)	
TIMER	**X**	**X**	The time switch switches on, if the time is between TON and TOFF. Weekly programming possible. The current time must be provided on the TIME input.	TIME: Input (TIME) TON: Switch on time (TIME) TOFF: Switch off time (TIME) O: Output signal (bit) O\: Inverted output (bit)	

Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
TRA	**X**	**X**	The transfer module passes the value of the input to the output.	I: Input O: Outputs	
UN PACC	**X**	**X**	UNPACC divides input data type Control into a step value (0..7) and a direction bit (0=DOWN/1=UP).	I: Input (4 bit control) STEP: Increment (byte) UP: Direction (bit) 0 = downwards 1 = upwards	
UN PACD	**X**	**X**	UNPACD divides an input data type Date into day, month, year.	I: Input (DATE) DAY: Day (BYTE) MON: Month (BYTE) YEAR: Year (byte, 0..99)	
UN PACM	**X**	**-**	UNPACM divides an input data type Max into 1 to 7 Words.	I: Input (Max) O: Outputs (Word)	
UN PACMB	**X**	**-**	UNPACMB divides an input data type Max into 1 to 14 byte(s).	I: Input (Max) O: Outputs (byte)	
UN PACP	**X**	**X**	UNPACP divides an input data type Pcontrol into an active bit and control bit.	I: Input (2 bit Pcontrol) ACT: Action bit, bit 0 from Input CONT: Control bit, bit 1 from Input	

Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
UN PACS	**X**		UNPACS converts an input data type Sint into a Word and a sign bit.	I: Input (SINT) VAL: Absolute value (WORD) SIGN: Sign (BIT)	
UN PACT	**X**	**X**	UNPACT divides an input data type Time into seconds, minutes, hours and day of week.	I: Input (time) SEC: Seconds (BYTE) MIN: Minutes (BYTE) HOUR: Hour (BYTE) DAY: Day (BYTE, 0 = no day)	
UN PACVW	**X**		UNPACVW converts an input data type Value into Word. Overflow generates an error. In this case the output value is zero.	I: Input (Value) O: Output (Word) ERR: Error (bit)	
UN PACW	**X**	**X**	UNPACW divides an input data type Word into 1 to 16 bits. The first bit is the one with the lowest value.	I: Input (Word) O: Outputs (BIT)	
UN PACWB	**X**	**X**	UNPACWB divides an input data type Word into a low-value and a high-value byte.	E: Input (Word) NB: Low-value byte HB: High-value byte	
USM	**X**	**X**	The output selection switch connects one of the 1 - 16 outputs with the input.	I: Input signal SEL: Selection selects the Output (WORD) O: Output signal	

Funct. block	Funct. module RMD	Mini-funct. module RMD	Brief description	Inputs/outputs	Function block diagramm
UST	**X**	**X**	The double output selection switch connects one of the two outputs with the input. It has the data type Bit. If SEL is 0, output O0 is selected otherwise O1 is. The value of the non-selected output does not change.	I: Input signal (free) SEL: Selection selects the Output (WORD) O0: Output 0 O1: Output 1	
XOR	**X**	**X**	XOR with several inputs. If the sum of the inputs is between 1 and REF, the output is 1.	I: Input signal (BIT) REF: Reference value (Word) O: Output signal (bit) O\: Inverted output (bit)	

General product information:



Appliance structure:
BCU: interface to EIB
Bus interface: output/input telegram memory
CPU: central unit for processing the application
Watchdog: among others, monitors system voltages
Interface: for downloading the application
RAM: volatile memory for, e.g. input/output signals, calculation variables, etc.
Real-time clock: for forming the time functions
Flash EEPROM: among others, application, constants

Determining the size of the application: when the application size of the **function module RMD (75720010)** in a project is calculated, not only the RAM memory capacity but also that of the EEPROM must be taken into account. Each input and output signal, variables and the function modules that are used load the RAM memory and the EEPROM memory with different factors. The following table shows the links in the form of an example:

1.) RAM (Arbeitsspeicher) mit 65536 Byte:

Anzahl Funktionsblöcke x 15
+ Anzahl Eingänge x 22
+ Anzahl Ausgänge x 44
= genutzter Speicherbedarf RAM

2.) EEPROM (Programm) mit 65536 Byte

Anzahl Funktionsblöcke x 20
+ Anzahl Eingänge x 6
+ Anzahl Ausgänge x 10
+ Anzahl der Konstanten x 2
= genutzter Speicherbedarf EPROM

Beispiel:

Kriterium	Anzahl	Faktor RAM	Summe RAM	Faktor EEPROM	Summe EEPROM
Funktionsblöcke	1900	15	28500	20	38000
Eingänge	600	22	13200	6	3600
Ausgänge	540	44	23760	10	5400
Konstanten	200	////	////	2	400
SUMMEN:			65460		47400

Working method and task cycle times: The function modules work on the IPO principle (Input Processing Output). **Input and output buffers** (50/100 telegrams) ensure that the telegrams are processed correctly. The application in the Fm consists of a group of function modules, which are themselves integrated in individual, logically linked segments. These segments are assigned individually or in groups of so-called tasks.

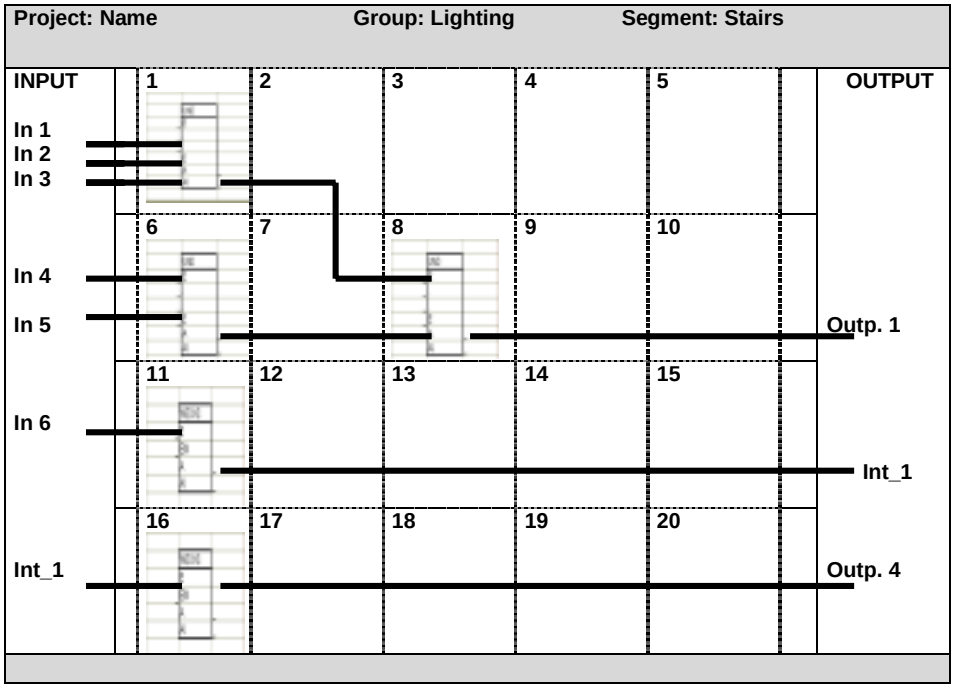
Tasks are given a cycle time that ensures the functions are processed within this defined time in accordance with the IPO principle. This cycle time can be set **(function module RMD 75720010 only)**. The processing times for the function blocks must be taken into account for the correct setting. On average, processing a function block requires about 50 us. If a cycle time of 100 ms is selected, for example, about 2000 function blocks can be processed correctly. Up to 16 tasks with different cycle times between 200 ms and approx. 21 minutes can be set with the FM. The reason for this procedure (multi-tasking process) lies in the nature of the application: tasks relevant to safety require faster processing and function triggering of the EIB components (e.g. signal lamp via actuator) as slow regulation of the room temperature. In principle a task should be carried out as quickly as necessary and as slowly as possible.

Along with these "normal" tasks the **function module RMD (75720010)** has two additional tasks available:
Power-fail task: this function is used in order to create defined statuses in the system in the case of bus voltage failure.
Power-up task: after initializing (voltage return) this task can be used, e.g. to set constants or define general statuses (behaviour on power return). After this task has been processed the appliance moves into normal operating mode,
the segments with the function modules are processed cyclically.

Structure of a segment: the logic emerging from the supported function is simulated with the help of suitable function blocks. Function blocks are placed on the work interface (segment) with the help of the graphics editor of the software FM tool / FM mini tool and linked with the corresponding signals.
A segment has the structure of an A4 page and may have up to 20 function blocks. However, we recommend that no more than 8 function blocks are placed in a segment, to improve the overview. The logic emerging from the supported function is simulated with the help of suitable function blocks. Function blocks are placed on the work interface (segment) with the help of the graphics editor of the software FM tool / FM mini tool and linked with the corresponding signals.
A segment has the structure of an A4 page and may have up to 20 function blocks. However, we recommend that no more than 8 function blocks are placed in a segment, to improve the overview**.** Function blocks are always processed from top left to bottom right (field 1 – 20).

Header

Input range



Output range

Signals of the function modules: the function modules differentiate between four different signal types
1.) Constants that can be set at the inputs of the function blocks (e.g. 21.8°C)
2.) Local signals that are used to link the function blocks **within a segment** (see above).
3.) Internal signals that are used to link the function blocks **of several segments** (example Int_1)
4.) External signals in connection with an EIB telegram (input and output signals, control signals).

Control signals are a special form of external signals. Control signals are created automatically by the software when the input and output signals are created. **Input control signals** are always set to 1 for the duration of the cycle, if any type of telegram was received with the corresponding group address. This internal signal can be used any number of times for creating links
Transmission of the output signals on the bus only takes place if the value of the signal has changed. However, if a function is to be sent cyclically on the bus, the function of the **output control signal** can be used: transmission of the current signal value is forced through the award of an internally generated cyclical impulse (e.g. through the function block oscillator every 10 seconds) onto the output control signal.

Signal rules: the following rules must be complied with when a project is created to ensure system-compatible handling:

1.) Processing rule: when the function blocks are placed, signal processing must be observed (top left to bottom right). Signals that are processed later must be created before.
2.) Link rule: more than one input can be connected to a single output. A single input must not be connected to more than one outputs.
3.) External signals may only have a single group address.
4.) External signals can either transmit or receive.
5.) Non-initialized signals always have the value 0
6.) Names of internal and external signals must be different. Control signals are marked automatically.

Additional information and tips can be found in the on-line help system for the FM tool software.