SIEMENS

SICAM RTUs

MODBUS

Interoperability

DC0-073-2.05

Document label:
SIC1703-HBIntopMODBUS-ENG_V2.05
Release date:
26.05.14

Preface

This document is applicable to the following product(s):

- SICAM RTUs
  (SICAM AK, SICAM TM, SICAM BC, SICAM EMIC, SICAM CMIC)

Purpose of this manual

This manual describes the interoperability of SICAM RTUs using protocol element according to MODBUS and essentially contains

- Interoperability MODBUS

Target Group

The document you are reading right now is addressed to users, who are in charge of the following tasks:

- Sales engineering and technical clarification
- Conceptual activities, as for example design and configuration

Table of Contents

1. Introduction

Contents

1.1. Area of Application

In this documentation, all definitions are described that are necessary for communication between automation units or between automation and control room process computer systems as per MODBUS protocol specification.

1.2. General Information

Syntax:

☐	implemented by SICAM RTUs
n/i	not implemented by SICAM RTUs
☒	subset selected for the defined project (empty check box to be replaced with this symbol)
■	mandatory
■	implemented by SICAM RTUs, default according MODBUS standard

~~strike-through~~	the text descriptions of parameters which are not applicable to this companion standard

Definition:

MASTER / SLAVE Protocol elements for communication with 3rd party systems using MODBUS protocol according definitions in this interoperability document.

	System Element	Note
	MODBUS Master "serial" (Multipoint Master)	
MODM	SM-2551/MODMA0	SICAM AK, SICAM TM, SICAM BC
	SM-0551/MODMA0	SICAM AK, SICAM TM, SICAM BC
	CP-8000/MODMT0	SICAM CMIC
	CP-6010/MODMT0	SICAM EMIC
	CP-3410/MODMT0	AMIS DC LAN
	CP-3411/MODMT0	AMIS DC seriell
	MODBUS Master "serial"with SICAM FCM Integration (Multipoint Master)	
MODMT2	CP-8000/MODMT2	SICAM CMIC
	MODBUS Slave "serial" (Multipoint Slave)	
MODS	SM-2551/MODSA0	SICAM AK, SICAM TM, SICAM BC
	SM-0551/MODSA0	SICAM AK, SICAM TM, SICAM BC
	CP-8000/MODST0	SICAM CMIC
	CP-6010/MODST0	SICAM EMIC
	CP-3410/MODST0	AMIS DC LAN
	CP-3411/MODST0	AMIS DC seriell
	MODBUS TCP Slave (=Server)	
MODi	SM-2546/MODi00	SICAM AK, SICAM TM, SICAM BC
	SM-2556/MODi00	SICAM AK, SICAM TM, SICAM BC

2. Interoperability of SICAM RTUs using serial MODBUS Master (MODM)

Contents

The MODBUS Interoperability defines presents sets of parameters and alternatives from which subsets have to be selected to implement particular telecontrol systems.
Other parameters, such as the listed set of different MODBUS Function Codes or MODBUS Data Formats in command and in monitor direction allow the specification of the complete set or subsets, as appropriate for given applications. This clause summarizes the parameters of the previous clauses to facilitate a suitable selection for a specific application. If a system is composed of equipment stemming from different manufacturers it is necessary that all partners agree on the selected parameters.

The selected parameters should be crossed in the white boxes.

Note:

In addition, the full specification of a system may require individual selection of certain parameters for certain parts of the system, such as the individual selection of scaling factors for individually addressable measured values.

2.1. Network Configurations

	Configuration	Remark
☐	Point-to-Point	Multipoint-Partyline (half duplex) with one Slave RS232 or RS485 (or RS422)
☐	Multiple Point-to-Point	Only available in SICAM RTUs systems supporting more serial interfaces
☐	Multipoint-Partyline	RS485
☐	Multipoint-Star	
☐	Data Concentrator	
n/i	Multipoint-Ring	
n/i	Dial in	
n/i	Dial out	
n/i	Modem Bank	

2.2. Physical layer

2.2.1. Electrical Interface

	Configuration	Remark
☐	RS232	V.24/V.28 Standard - Point-to-Point (Master with 1 Slave)
☐	RS422	V.11 (4-wire) [1)] - Point-to-Point (Master with 1 Slave)
☐	RS485	V.11 (2-wire) [1)] - Multipoint-Partyline (Master with n-Slaves) - Point-to-Point (Master with 1 Slave)

1) External Converters (V.24/V.28 <-> V.11) required in some cases!

2.2.2. Transmission Speed

Note:
- MODBUS protocol use only unbbalanced communication procedures.
- transmission speed is same for both directions (transmit/receive)

Circuit V.24 / V.28 Standard

	Speed	Remark		Speed	Remark
☐	50 bit/s		☐	1800 bit/s	
☐	75 bit/s		☐	2000 bit/s	
☐	100 bit/s		☐	2400 bit/s	
☐	110 bit/s		☐	4800 bit/s	
☐	134.5 bit/s		☐	9600 bit/s	
☐	150 bit/s		☐	19200 bit/s	
☐	200 bit/s		☐	38400 bit/s	
☐	300 bit/s		☐	56000 bit/s	
☐	600 bit/s		☐	57600 bit/s	
☐	1050 bit/s		☐	64000 bit/s	
☐	1200 bit/s		n/i	115200 bit/s	

2.3. Link Layer

	Transmission Mode
☐	ASCII Mode
☐	RTU Mode

2.3.1. ASCII Mode

Byte asynchronous data transmission is used in in ASCII-Mode (Least Significant Bit sent 1st for each byte).

	Byte Framing	Remark
■	1 Start Bit	
■	7 Data Bits	
■	Parity Bit "even"	
☐	Parity Bit "odd"	
☐	"No" Parity Bit	
■	1 Stop Bit	
☐	1.5 Stop Bits	
☐	2 Stop Bits	

Note:
Byte frame for MODBUS ASCII mode according MODBUS standard: "7E1" (1 start bit, 7 data bits, 1 parity bit „even parity“, 1 stop bit)
For maximum compability with other devices „odd parity, no parity“ and „1.5 stop bits, 2 stop bits“ is also supported.
The use of no parity requires 2 stop bits! **)
**) In old configurations byte frame "7N2" (7 data bits, no parity, 2 stop bits) is used typically for MODBUS ASCII mode.

	Procedure	Remark
■	MODBUS Slave Address - 2 CHARS (8 Bit)	
■	MODBUS Function Code - 2 CHARS (8 Bit)	
■	MODBUS Register Address - 4 CHARS (16 Bit)	Address will address a 16 Bit MODBUS register
■	Longitudinal Redundancy Check (LRC) - 2 CHARS (8 Bit)	

2.3.2. RTU Mode

Byte asynchronous data transmission is used in in RTU-Mode (Least Significant Bit sent 1st for each byte).

	Byte Framing	Remark
■	1 Start Bit	
■	8 Data Bits	
■	Parity Bit "even"	
☐	Parity Bit "odd"	
☐	"No" Parity Bit	
■	1 Stop Bit	
☐	1.5 Stop Bits	
☐	2 Stop Bits	

Note:
Byte frame for MODBUS RTU mode according MODBUS standard: "8E1" (1 start bit, 8 data bits, 1 parity bit „even parity", 1 stop bit)
For maximum compability with other devices „odd parity, no parity" and „1.5 stop bits, 2 stop bits" is also supported.
The use of no parity requires 2 stop bits! **)
**) In old configurations byte frame "8N2" (8 data bits, no parity, 2 stop bits) is used typically for MODBUS RTU mode.

	Procedure	Remark
■	MODBUS Slave Address (8 Bit)	
■	MODBUS Function Code (8 Bit)	
■	MODBUS Register Address (16 Bit)	Address will address a 16 Bit MODBUS register
■	Cyclical Redundancy Check "CRC" (16 Bit)	

2.3.3. Link Transmission Procedure

	Description	Remark
■	Unbalanced Master / Slave Communication	
■	Unbalanced, Master (half duplex)	
n/i	Unbalanced Slave	See chapter for MODBUS Slave!

2.3.4. Frame Length

	Description	Remark
■	ASCII Mode: Maximum Message length 0 up to 2*252 CHARS (without Start, Address, Function, LRC and End Characters) RTU Mode: Maximum Message length 253 Bytes (without Address and CRC Bytes)	max. frame length is configurable

2.3.5. Address field of the link

	Description	Remark
■	1 octet (8 Bit) … RTU Mode	MODBUS Slave Address (1-247)
■	2 CHAR …..... ASCII Mode	MODBUS Slave Address (1-247)
n/i	BROADCAST Addressing Mode	MODBUS Slave Address (0)

2.4. Application Layer

2.4.1. MODBUS Function Codes

	MODBUS Function Code - Description	Data-Formats
	Data Access (Bit Access)	
☐	01 = READ COILS	0
☐	02 = READ DISCRETE INPUTS	0
☐	05 = WRITE SINGLE COIL	0, 0b, 0c
☐	15 = WRITE MULTIPLE COILS	0, 0b, 0c
	Data Access (16 Bit Access)	
☐	03 = READ HOLDING REGISTERS	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 16, 17
☐	04 = READ INPUT REGISTERS	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 16, 17 18, 19
☐	06 = WRITE SINGLE REGISTER	1, 2, 3, 4, 5, 6, 7, 9
☐	16 = WRITE MULTIPLE REGISTERS	1, 2, 3, 4, 5, 6, 7, 9, 100, 101, 102
n/i	22 = MASK WRITE REGISTER	
n/i	23 = READ / WRITE MULTIPLE REGISTERS	
n/i	24 = READ FIFO QUEUE	
	Data Access (File Record Access)	
n/i	20 = READ FILE RECORD	
n/i	21 = WRITE FILE RECORD	
	Diagnostics	
n/i	07 = READ EXCEPTION STATUS	
☐	08 = DIAGNOSTICS (SUB-Code 00-18,20) [1)]	
n/i	11 = GET COM EVENT COUNTER	
n/i	12 = GET COM EVENT LOG	
n/i	17 = REPORT SLAVE ID	
n/i	43 = READ DEVCE Identification (SUB-Code = 14)	
	Other	
n/i	43 = Encapsulated Interface Transport (SUB-Code = 13,14)	

[1)] only Sub-Code = 00 "Return Query Data" (LOOPBACK CHECK)

2.4.2. MODBUS Exception Status

	MODBUS Exception Code Description	Remark
☐	01 = ILLEGAL FUNCTION	MODBUS Function Code not implemented
☐	02 = ILLEGAL DATA ADDRESS	Requested Data Address not implemented
☐	03 = ILLEGAL DATA VALUE	
n/i	04 = SLAVE DEVICE FAILURE	
n/i	05 = ACKNOWLEDGE	
n/i	06 = SLAVE DEVICE BUSY	
n/i	07 = NEGATIVE ACKNOWLEDGE (NAK)	
n/i	08 = MEMORY PARITY ERROR	

Note: Not supported Exception Codes are handled as no response.

3. Interoperability of SICAM RTUs using serial MODBUS Master (MODMT2)

Contents

The MODBUS Interoperability defines presents sets of parameters and alternatives from which subsets have to be selected to implement particular telecontrol systems.
Other parameters, such as the listed set of different MODBUS Function Codes or MODBUS Data Formats in command and in monitor direction allow the specification of the complete set or subsets, as appropriate for given applications. This clause summarizes the parameters of the previous clauses to facilitate a suitable selection for a specific application. If a system is composed of equipment stemming from different manufacturers it is necessary that all partners agree on the selected parameters.

The selected parameters should be crossed in the white boxes.

Note:

In addition, the full specification of a system may require individual selection of certain parameters for certain parts of the system, such as the individual selection of scaling factors for individually addressable measured values.

3.1. Network Configurations

	Configuration	Remark
☐	Point-to-Point	Multipoint-Partyline (half duplex) with one Slave RS232 or RS485 (or RS422)
☐	Multiple Point-to-Point	Only available in SICAM RTUs systems supporting more serial interfaces
☐	Multipoint-Partyline	RS485
☐	Multipoint-Star	
☐	Data Concentrator	
n/i	Multipoint-Ring	
n/i	Dial in	
n/i	Dial out	
n/i	Modem Bank	

3.2. Physical layer

3.2.1. Electrical Interface

	Configuration	Remark
☐	RS232	V.24/V.28 Standard - Point-to-Point (Master with 1 Slave)
☐	RS422	V.11 (4-wire) [1] - Point-to-Point (Master with 1 Slave)
☐	RS485	V.11 (2-wire) [1] - Multipoint-Partyline (Master with n-Slaves) - Point-to-Point (Master with 1 Slave)

1) External Converters (V.24/V.28 <-> V.11) required in some cases!

3.2.2. Transmission Speed

Note:
- MODBUS protocol use only unbbalanced communication procedures.
- transmission speed is same for both directions (transmit/receive)

Circuit V.24 / V.28 Standard

	Speed	Remark		Speed	Remark
☐	50 bit/s		☐	1800 bit/s	
☐	75 bit/s		☐	2000 bit/s	
☐	100 bit/s		☐	2400 bit/s	
☐	110 bit/s		☐	4800 bit/s	
☐	134.5 bit/s		☐	9600 bit/s	
☐	150 bit/s		☐	19200 bit/s	
☐	200 bit/s		☐	38400 bit/s	
☐	300 bit/s		n/i	56000 bit/s	
☐	600 bit/s		n/i	57600 bit/s	
☐	1050 bit/s		n/i	64000 bit/s	
☐	1200 bit/s		n/i	115200 bit/s	

3.3. Link Layer

	Transmission Mode
n/i	ASCII Mode
☐	RTU Mode

3.3.1. ASCII Mode

Byte asynchronous data transmission is used in in ASCII-Mode (Least Significant Bit sent 1st for each byte).

	Byte Framing	Remark
n/i	1 Start Bit	
n/i	7 Data Bits	
n/i	Parity Bit "even"	
n/i	Parity Bit "odd"	
n/i	"No" Parity Bit	
n/i	1 Stop Bit	
n/i	1.5 Stop Bits	
n/i	2 Stop Bits	

Note:
Byte frame for MODBUS ASCII mode according MODBUS standard: "7E1" (1 start bit, 7 data bits, 1 parity bit „even parity“, 1 stop bit)
For maximum compability with other devices „odd parity, no parity“ and „1.5 stop bits, 2 stop bits“ is also supported.
The use of no parity requires 2 stop bits! **)
**) In old configurations byte frame "7N2" (7 data bits, no parity, 2 stop bits) is used typically for MODBUS ASCII mode.

	Procedure	Remark
n/i	MODBUS Slave Address - 2 CHARS (8 Bit)	
n/i	MODBUS Function Code - 2 CHARS (8 Bit)	
n/i	MODBUS Register Address - 4 CHARS (16 Bit)	Address will address a 16 Bit MODBUS register
n/i	Longitudinal Redundancy Check (LRC) - 2 CHARS (8 Bit)	

3.3.2. RTU Mode

Byte asynchronous data transmission is used in in RTU-Mode (Least Significant Bit sent 1st for each byte).

	Byte Framing	Remark
■	1 Start Bit	
■	8 Data Bits	
■	Parity Bit "even"	
☐	Parity Bit "odd"	
☐	"No" Parity Bit	
■	1 Stop Bit	
☐	1.5 Stop Bits	
☐	2 Stop Bits	

Note:
Byte frame for MODBUS RTU mode according MODBUS standard: "8E1" (1 start bit, 8 data bits, 1 parity bit „even parity", 1 stop bit)
For maximum compability with other devices „odd parity, no parity" and „1.5 stop bits, 2 stop bits" is also supported.
The use of no parity requires 2 stop bits! **)
**) In old configurations byte frame "8N2" (8 data bits, no parity, 2 stop bits) is used typically for MODBUS RTU mode.

	Procedure	Remark
■	MODBUS Slave Address (8 Bit)	
■	MODBUS Function Code (8 Bit)	
■	MODBUS Register Address (16 Bit)	Address will address a 16 Bit MODBUS register
■	Cyclical Redundancy Check "CRC" (16 Bit)	

3.3.3. Link Transmission Procedure

	Description	Remark
■	Unbalanced Master / Slave Communication	
■	Unbalanced, Master (half duplex)	
n/i	Unbalanced Slave	See chapter for MODBUS Slave!

3.3.4. Frame Length

	Description	Remark
n/i	ASCII Mode: Maximum Message length 0 up to 2*252 CHARS (without Start, Address, Function, LRC and End Characters)	max. frame length is configurable
■	RTU Mode: Maximum Message length 253 Bytes (without Address and CRC Bytes)	

3.3.5. Address field of the link

	Description	Remark
■	1 octet (8 Bit) … RTU Mode	MODBUS Slave Address (1-247)
n/i	2 CHAR ..... ASCII Mode	MODBUS Slave Address (1-247)
n/i	BROADCAST Addressing Mode	MODBUS Slave Address (0)

3.4. Application Layer

3.4.1. MODBUS Function Codes

	MODBUS Function Code - Description	Data-Formats
	Data Access (Bit Access)	
☐	01 = READ COILS	0, 0a
☐	02 = READ DISCRETE INPUTS	0
☐	05 = WRITE SINGLE COIL	0, 0b, 0c
☐	15 = WRITE MULTIPLE COILS	0, 0b, 0c
	Data Access (16 Bit Access)	
☐	03 = READ HOLDING REGISTERS	1, 2, 3, 4, 5, 6, 7a, 7b, 16
☐	04 = READ INPUT REGISTERS	1, 2, 3, 4, 5, 6, 7a, 7b, 16
☐	06 = WRITE SINGLE REGISTER	1, 2
☐	16 = WRITE MULTIPLE REGISTERS	1, 2, 3, 4, 5, 6, 7a, 7b, 104
n/i	22 = MASK WRITE REGISTER	
n/i	23 = READ / WRITE MULTIPLE REGISTERS	
n/i	24 = READ FIFO QUEUE	
	Data Access (File Record Access)	
n/i	20 = READ FILE RECORD	
n/i	21 = WRITE FILE RECORD	
	Diagnostics	
n/i	07 = READ EXCEPTION STATUS	
n/i	08 = DIAGNOSTICS (SUB-Code 00-18,20)	
n/i	11 = GET COM EVENT COUNTER	
n/i	12 = GET COM EVENT LOG	
n/i	17 = REPORT SLAVE ID	
n/i	43 = READ DEVCE Identification (SUB-Code = 14)	
	Other	
n/i	43 = Encapsulated Interface Transport (SUB-Code = 13,14)	

3.4.2. MODBUS Exception Status

	MODBUS Exception Code Description	Remark
☐	01 = ILLEGAL FUNCTION	MODBUS Function Code not implemented
☐	02 = ILLEGAL DATA ADDRESS	Requested Data Address not implemented
☐	03 = ILLEGAL DATA VALUE	
n/i	04 = SLAVE DEVICE FAILURE	
n/i	05 = ACKNOWLEDGE	
n/i	06 = SLAVE DEVICE BUSY	
n/i	07 = NEGATIVE ACKNOWLEDGE (NAK)	
n/i	08 = MEMORY PARITY ERROR	

Note: Not supported Exception Codes are handled as no response.

4. Interoperability of SICAM RTUs using serial MODBUS Slave (MODS)

Contents

The companion standard defined presents sets of parameters and alternatives from which subsets have to be selected to implement particular telecontrol systems. Certain parameter values, represent mutually exclusive alternatives. This means that only one value of the defined parameters is admitted per system. Other parameters, such as the listed set of different process information in command and in Monitor direction allow the specification of the complete set or subsets, as appropriate for given applications. This clause summarizes the parameters of the previous clauses to facilitate a suitable selection for a specific application. If a system is composed of equipment stemming from different manufacturers it is necessary that all partners agree on the selected parameters.

The selected parameters should be crossed in the white boxes.

Note:

In addition, the full specification of a system may require individual selection of certain parameters for certain parts of the system, such as the individual selection of scaling factors for individually addressable measured values.

4.1. Network Configurations

	Configuration	Remark
☐	Point-to-Point	Multipoint-Partyline (half duplex) Master with 1 Slave RS232 or RS485 (or RS422)
☐	Multiple Point-to-Point	Only available in SICAM RTUs systems supporting more serial interfaces
☐	Multipoint-Partyline	RS485
☐	Multipoint-Star	
☐	Data Concentrator	
n/i	Multipoint-Ring	
n/i	Dial in	
n/i	Dial out	
n/i	Modem Bank	

4.2. Physical layer

4.2.1. Electrical Interface

	Configuration	Remark
☐	RS232	V.24/V.28 Standard - Point-to-Point (Master with 1 Slave)
☐	RS422	V.11 (4-wire) [1)] - Point-to-Point (Master with 1 Slave)
☐	RS485	V.11 (2-wire) [1)] - Multipoint-Partyline (Master with n-Slaves) - Point-to-Point (Master with 1 Slave)

1) External Converters (V.24/V.28 <-> V.11) required in some cases!

4.2.2. Transmission Speed

Note:
- MODBUS protocol use only unbbalanced communication procedures.
- transmission speed is same for both directions (transmit/receive)

Circuit V.24 / V.28 Standard

	Speed	Remark		Speed	Remark
☐	50 bit/s		☐	1800 bit/s	
☐	75 bit/s		☐	2000 bit/s	
☐	100 bit/s		☐	2400 bit/s	
☐	110 bit/s		☐	4800 bit/s	
☐	134.5 bit/s		☐	9600 bit/s	
☐	150 bit/s		☐	19200 bit/s	
☐	200 bit/s		☐	38400 bit/s	
☐	300 bit/s		☐	56000 bit/s	
☐	600 bit/s		☐	57600 bit/s	
☐	1050 bit/s		☐	64000 bit/s	
☐	1200 bit/s		n/i	115200 bit/s	

4.3. Link Layer

	Transmission Mode
☐	ASCII Mode
☐	RTU Mode

4.3.1. ASCII Mode

Byte asynchronous data transmission is used in in ASCII-Mode (Least Significant Bit sent 1st for each byte).

	Byte Framing	Remark
■	1 Start Bit	
■	7 Data Bits	
■	Parity Bit "even"	
☐	Parity Bit "odd"	
☐	"No" Parity Bit	
■	1 Stop Bit	
☐	1.5 Stop Bits	
☐	2 Stop Bits	

Note:
Byte frame for MODBUS ASCII mode according MODBUS standard: "7E1" (1 start bit, 7 data bits, 1 parity bit „even parity“, 1 stop bit)
For maximum compability with other devices „odd parity, no parity“ and „1.5 stop bits, 2 stop bits“ is also supported.
The use of no parity requires 2 stop bits! **)
**) In old configurations byte frame "7N2" (7 data bits, no parity, 2 stop bits) is used typically for MODBUS ASCII mode.

	Procedure	Remark
■	MODBUS Slave Address - 2 CHARS (8 Bit)	
■	MODBUS Function Code - 2 CHARS (8 Bit)	
■	MODBUS Register Address - 4 CHARS (8 Bit)	Address will address a 16 Bit MODBUS register
■	Longitudinal Redundancy Check (LRC) - 2 CHARS (8 Bit)	

4.3.2. RTU Mode

Byte asynchronous data transmission is used in in RTU-Mode (Least Significant Bit sent 1st for each byte).

	Byte Framing	Remark
■	1 Start Bit	
■	8 Data Bits	
■	Parity Bit "even"	
☐	Parity Bit "odd"	
☐	"No" Parity Bit	
■	1 Stop Bit	
☐	1.5 Stop Bits	
☐	2 Stop Bits	

Note:
Byte frame for MODBUS RTU mode according MODBUS standard: "8E1" (1 start bit, 8 data bits, 1 parity bit „even parity“, 1 stop bit)
For maximum compability with other devices „odd parity, no parity“ and „1.5 stop bits, 2 stop bits“ is also supported.
The use of no parity requires 2 stop bits! **)
**) In old configurations byte frame "8N2" (8 data bits, no parity, 2 stop bits) is used typically for MODBUS RTU mode.

	Procedure	Remark
■	MODBUS Slave Address (8 Bit)	
■	MODBUS Function Code (8 Bit)	
■	MODBUS Register Address (16 Bit)	Address will address a 16 Bit MODBUS register
■	Cyclical Redundancy Check "CRC" (16 Bit)	

4.3.3. Link Transmission Procedure

	Description	Remark
■	Unbalanced Master / Slave Communication	
n/i	Unbalanced, Master (half duplex)	See chapter for MODBUS Master!
■	Unbalanced Slave	

4.3.4. Frame Length

	Description	Remark
■	ASCII Mode: Maximum Message length 0 up to 2*252 CHARS without Start, Address, Function, LRC and End Characters RTU Mode: Maximum Message length 253 Bytes without Address and CRC Bytes	max. frame length is configurable

4.3.5. Address field of the link

	Description	Remark
■	1 octet (8 Bit) … RTU Mode	MODBUS Slave Address (1-247)
■	2 CHAR …..... ASCII Mode	MODBUS Slave Address (1-247)
■	BROADCAST Addressing Mode	MODBUS Slave Address (0)

4.4. Application Layer

4.4.1. MODBUS Function Codes

	MODBUS Function Code - Description	Data-Formats
	Data Access (Bit Access)	
☐	01 = READ COILS	0
☐	02 = READ DISCRETE INPUTS	0
☐	05 = WRITE SINGLE COIL	0
☐	15 = WRITE MULTIPLE COILS	0
	Data Access (16 Bit Access)	
☐	03 = READ HOLDING REGISTERS	1, 2, 3, 4, 5, 6, 7, 12, 13, 120, 121
☐	04 = READ INPUT REGISTERS	1, 2, 3, 4, 5, 6, 7, 12, 13, 120, 121 18, 19
☐	06 = WRITE SINGLE REGISTER	1, 2, 3, 4, 5, 6, 7, 12, 13
☐	16 = WRITE MULTIPLE REGISTERS	1, 2, 3, 4, 5, 6, 7, 12, 13, 103
n/i	22 = MASK WRITE REGISTER	
n/i	23 = READ / WRITE MULTIPLE REGISTERS	
n/i	24 = READ FIFO QUEUE	
	Data Access (File Record Access)	
n/i	20 = READ FILE RECORD	
n/i	21 = WRITE FILE RECORD	
	Diagnostics	
n/i	07 = READ EXCEPTION STATUS	
☐	08 = DIAGNOSTICS (SUB-Code 00-18,20) [1)]	
n/i	11 = GET COM EVENT COUNTER	
n/i	12 = GET COM EVENT LOG	
n/i	17 = REPORT SLAVE ID	
n/i	43 = READ DEVCE Identification (SUB-Code = 14)	
	Other	
n/i	43 = Encapsulated Interface Transport (SUB-Code = 13,14)	

[1)] only Sub-Code = 00 "Return Query Data" (LOOPBACK CHECK)

4.4.2. MODBUS Exception Status

	MODBUS Exception Code Description	Data-Formats
☐	01 = ILLEGAL FUNCTION	MODBUS Function Code not implemented
☐	02 = ILLEGAL DATA ADDRESS	Requested Data Address not implemented
☐	03 = ILLEGAL DATA VALUE	
n/i	04 = SLAVE DEVICE FAILURE	
n/i	05 = ACKNOWLEDGE	
n/i	06 = SLAVE DEVICE BUSY	
n/i	07 = NEGATIVE ACKNOWLEDGE (NAK)	
n/i	08 = MEMORY PARITY ERROR	

5. Interoperability of SICAM RTUs using MODBUS/TCP Slave “Server” (MODi)

Contents

The companion standard defined presents sets of parameters and alternatives from which subsets have to be selected to implement particular telecontrol systems. Certain parameter values, represent mutually exclusive alternatives. This means that only one value of the defined parameters is admitted per system. Other parameters, such as the listed set of different process information in command and in Monitor direction allow the specification of the complete set or subsets, as appropriate for given applications. This clause summarizes the parameters of the previous clauses to facilitate a suitable selection for a specific application. If a system is composed of equipment stemming from different manufacturers it is necessary that all partners agree on the selected parameters.

The selected parameters should be crossed in the white boxes.

Note:

In addition, the full specification of a system may require individual selection of certain parameters for certain parts of the system, such as the individual selection of scaling factors for individually addressable measured values.

5.1. Network Configurations

	Configuration	Remark
⊟	~~Point-to-Point~~	~~Multipoint-Partyline (half duplex)~~ ~~Master with 1 Slave~~ ~~RS232 or RS485 (or RS422)~~
⊟	~~Multiple Point-to-Point~~	~~Only available in SICAM RTUs systems supporting more serial interfaces~~
⊟	~~Multipoint-Partyline~~	~~RS485~~
⊟	~~Multipoint-Star~~	
⊟	~~Data Concentrator~~	
~~n/i~~	~~Multipoint-Ring~~	
~~n/i~~	~~Dial-in~~	
~~n/i~~	~~Dial-out~~	
~~n/i~~	~~Modem-Bank~~	
■	LAN / WAN	

5.2. Physical layer

5.2.1. Electrical Interface

	Configuration	Remark
⊟	~~RS232~~	~~V.24/V.28 Standard~~ ~~- Point-to-Point (Master with 1 Slave)~~
⊟	~~RS422~~	~~V.11 (4-wire)~~ [1] ~~- Point-to-Point (Master with 1 Slave)~~
⊟	~~RS485~~	~~V.11 (2-wire)~~ [1] ~~- Multipoint-Partyline (Master with n Slaves)~~ ~~- Point-to-Point (Master with 1 Slave)~~
☐	Ethernet (electrical)	
☐	Ethernet (optical)	

1) External Converters (V.24/V.28 <-> V.11) required in some cases!

5.2.2. Transmission Speed (common for both directions)

Note:
- MODBUS protocol use only unbbalanced communication procedures.
- same transmission speed in transmit and receive direction

~~Circuit V.24 / V.28 Standard~~

	Speed	Remark		Speed	Remark
⊟	~~50 bit/s~~		⊟	~~1800 bit/s~~	
⊟	~~75 bit/s~~		⊟	~~2000 bit/s~~	
⊟	~~100 bit/s~~		⊟	~~2400 bit/s~~	
⊟	~~110 bit/s~~		⊟	~~4800 bit/s~~	
⊟	~~134.5 bit/s~~		⊟	~~9600 bit/s~~	
⊟	~~150 bit/s~~		⊟	~~19200 bit/s~~	
⊟	~~200 bit/s~~		⊟	~~38400 bit/s~~	
⊟	~~300 bit/s~~		⊟	~~56000 bit/s~~	
⊟	~~600 bit/s~~		⊟	~~57600 bit/s~~	
⊟	~~1050 bit/s~~		⊟	~~64000 bit/s~~	
⊟	~~1200 bit/s~~		~~n/i~~	~~115200 bit/s~~	
☐	10 mbit/s		☐	100 mbit/s	

5.2.3. TCP-Port

	TCP-Port	Remark
■	502	registered to MODBUS/TCP

5.2.4. Connections

	Number of Connections	Remark
■	max. 100	

Note: each connection can be used as "SERVER" with individual data base, unique MODBUS-Slave Address and unique TCP/IP-address of the Client.

5.3. Link Layer (MODBUS)

	Transmission Mode
☒	~~ASCII-Mode~~
☒	~~RTU-Mode~~
☐	TCP-Mode

~~5.3.1. ASCII Mode~~

~~Byte asynchronous data transmission is used in in ASCII-Mode (Least Significant Bit sent 1st for each byte).~~

	Byte Framing	Remark
■	~~1 Start Bit~~	
☒	~~7 Data Bits~~	
■	~~8 Data Bits~~	
☒	~~Parity Bit "even"~~	
☒	~~Parity Bit "odd"~~	
☒	~~"No" Parity Bit~~	
☒	~~1 Stop Bit~~	
☒	~~1.5 Stop Bits~~	
☒	~~2 Stop Bits~~	

Note: Byteframe "7N2" (7 data bits, no parity, 2 stop bits) is used typically for MODBUS ASCII-Mode.

	Procedure	Remark
■	~~MODBUS Slave Address - 2 CHARS (8 Bit)~~	
■	~~MODBUS Function Code - 2 CHARS (8 Bit)~~	
■	~~MODBUS Register Address - 4 CHARS (8 Bit)~~	~~Address will address a 16 Bit MODBUS register~~
■	~~Longitudinal Redundancy Check (LRC) 2 CHARS (8 Bit)~~	

~~5.3.2. RTU Mode~~

~~Byte asynchronous data transmission is used in in RTU-Mode (Least Significant Bit sent 1st for each byte).~~

	Byte Framing	Remark
■	~~1 Start Bit~~	
■	~~8 Data Bits~~	
⊟	~~Parity Bit "even"~~	
⊟	~~Parity Bit "odd"~~	
⊟	~~"No" Parity Bit~~	
⊟	~~1 Stop Bit~~	
⊟	~~1.5 Stop Bits~~	
⊟	~~2 Stop Bits~~	

~~Note: Byteframe "8N2" (8 data bits, no parity, 2 stop bits) is used typically for MODBUS RTU-Mode.~~

	Procedure	Remark
■	~~MODBUS Slave Address (8 Bit)~~	
■	~~MODBUS Function Code (8 Bit)~~	
■	~~MODBUS Register Address (16 Bit)~~	~~Address will address a 16 Bit MODBUS register~~
■	~~Cyclical Redundancy Check "CRC" (16 Bit)~~	

5.3.3. TCP Mode

~~Byte asynchronous data transmission is used in in RTU-Mode (Least Significant Bit sent 1st for each byte).~~

	Byte Framing	Remark
■	8 Data Bits	

	Procedure	Remark
■	MODBUS Transaction Number (16 Bit)	MBAP Header
■	Protocol Identifier = "0" (16 Bit)	MBAP Header
■	Length (16 Bit)	MBAP Header
■	Unit Identifier (8 Bit)	MBAP Header
■	MODBUS Slave Address (8 Bit)	
■	MODBUS Function Code (8 Bit)	
■	MODBUS Register Address (16 Bit)	Address will address a 16 Bit MODBUS register
⊟	~~Cyclical Redundancy Check "CRC" (16 Bit)~~	CRC is not used by MODBUS/TCP (TCP-Checksum used only)

MBAP Header … MODBUS Application Protocol Header

5.3.4. Link Transmission Procedure

	Description	Remark
n/i	Unbalanced Master / Slave Communication	
n/i	Unbalanced, Master (half duplex)	
■	Unbalanced Slave	

5.3.5. Frame Length

	Description	Remark
■	~~ASCII-Mode:~~ ~~Maximum Message length 0 up to 2*252 CHARS without Start, Address, Function, LRC and End Characters~~ RTU-Mode, TCP-Mode: Maximum Message length 253 Bytes without Address and CRC Bytes	max. frame length is configurable

5.3.6. Address field of the link

	Description	Remark
■	1 octet (8 Bit) … RTU-Mode, TCP-Mode	MODBUS Slave Address (1-247)
~~■~~	~~2 CHAR …… ASCII Mode~~	~~MODBUS Slave Address (1-247)~~
n/i	BROADCAST Addressing Mode	MODBUS Slave Address (0)

5.4. Application Layer

5.4.1. MODBUS Function Codes

	MODBUS Function Code - Description	Data-Formats
	Data Access (Bit Access)	
☐	01 = READ COILS	0
☐	02 = READ DISCRETE INPUTS	0
☐	05 = WRITE SINGLE COIL	0
☐	15 = WRITE MULTIPLE COILS	0
	Data Access (16 Bit Access)	
☐	03 = READ HOLDING REGISTERS	1, 2, 3, 4, 5, 6, 7, 12, 13, 14, 15, 16, 120, 121
☐	04 = READ INPUT REGISTERS	1, 2, 3, 4, 5, 6, 7, 12, 13, 14, 15, 16, 120, 121
☐	06 = WRITE SINGLE REGISTER	1, 2, 3, 4, 5, 6, 7, 12, 13, 14, 15, 16
☐	16 = WRITE MULTIPLE REGISTERS	1, 2, 3, 4, 5, 6, 7, 12, 13, 14, 15, 16
n/i	22 = MASK WRITE REGISTER	
n/i	23 = READ / WRITE MULTIPLE REGISTERS	
n/i	24 = READ FIFO QUEUE	
	Data Access (File Record Access)	
n/i	20 = READ FILE RECORD	
n/i	21 = WRITE FILE RECORD	
	Diagnostics	
n/i	07 = READ EXCEPTION STATUS	
☐	08 = DIAGNOSTICS (SUB-Code 00-18,20) [1)]	
n/i	11 = GET COM EVENT COUNTER	
n/i	12 = GET COM EVENT LOG	
n/i	17 = REPORT SLAVE ID	
n/i	43 = READ DEVCE Identification (SUB-Code = 14)	
	Other	
n/i	43 = Encapsulated Interface Transport (SUB-Code = 13,14)	

[1)] only Sub-Code = 00 "Return Query Data" (LOOPBACK CHECK)

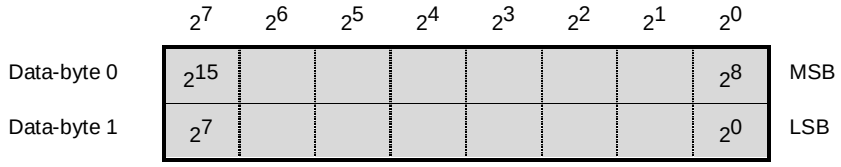
5.4.2. MODBUS Exception Status

	MODBUS Exception Code Description	Data-Formats
☐	01 = ILLEGAL FUNCTION	MODBUS Function Code not implemented
☐	02 = ILLEGAL DATA ADDRESS	Requested Data Address not implemented
☐	03 = ILLEGAL DATA VALUE	
n/i	04 = SLAVE DEVICE FAILURE	
n/i	05 = ACKNOWLEDGE	
n/i	06 = SLAVE DEVICE BUSY	
n/i	07 = NEGATIVE ACKNOWLEDGE (NAK)	
n/i	08 = MEMORY PARITY ERROR	

6. MODBUS Data Formats

Data formats which need more than 1 MODBUS register (a 16 Bit) for representation will be sent "HIGH Order before LOW Order". Negative values will be sent in two's complement, the most significant bit of the specific data format will be used as sign.

Data in MODBUS Register:



Most significant byte (MSB) is sent first!

Following data formats will be supported by the protocol element for MODBUS:

Format-0: 1 Bit

Range of value: 0 .. 1
Note: This Format is used for coils only!

Format-0a: 3 Bit Indication (Siemens MCU "Motor Control Unit")

3 neighbouring coils (coil address n, n+1, n+2) will be processed as "3 Bit Indication".
Range of value: 0 .. 7
Note: This format is used for coils only!

	OFF (n+2)	**Grounding Connector** (n+1)	**Disconnector** (n)	**Disconnector** [IEC DPI]	**Grounding Connector** [IEC DPI]
0	0	0	0	DIFF	DIFF
1	**0**	**0**	**1**	**ON**	**OFF**
2	**0**	**1**	**0**	**OFF**	**ON**
3	0	1	1	ERR	ERR
4	1	0	0	**OFF**	**OFF**
5	1	0	1	ERR	ERR
6	1	1	0	ERR	ERR
7	1	1	1	ERR	ERR

Disconnector [IEC DPI] ..... Disconnector as IEC60870-5-101/-104 double point information
Grounding Connector [IEC DPI] ..... Grounding Connector as IEC60870-5-101/-104 double point information
DIFF ..... indeterminate or intermediate state
ERR ..... indeterminate state

Format-0b: Single command

The command state of a single command can be sent with Coil address (n) or with a pulse.

Transmission Mode for Singel Command:

Command	Command Type	Command Transmission	MODMA0	MODMT0	MODMT2
Single Command “ON“	persistent	COIL (n) = “ON“	✓	✓	✓
Single Command “OFF“	persistent	COIL (n) = “OFF“	✓	✓	✓
Single Command “ON“	pulse	COIL (n) = pulse	✓	✓	✓
Single Command “OFF“	pulse				

Format-0c: Double Command

The command state of a double command can be sent with coil address (n) or with a pulse on 2 neighbouring coils (n, n+1).

Transmission Mode for Double Command:

Command *)	Command Type *)	Command Transmission	MODMA0	MODMT0	MODMT2
Double Command “ON“	persistent	COIL (n) = “ON“	✓	✓	✓
Double Command “OFF“	persistent	COIL (n) = “OFF“	✓	✓	✓
Double Command “ON“	pulse	COIL (n) = pulse or COIL (n+1) = pulse	✓	✓	✓ ✓
Double Command “OFF“ “	pulse	COIL (n+1) = pulse or COIL (n) = pulse			✓

*) The command type and the transmissioin oft he command state ON/OFF with Coil address (n) or (n+1) can be parameterized.

Format-1: 16 Bit Signed Integer

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	VZ	2^{14}						2^8
Data-byte 1	2^7							2^0

VZ: "sign" (0="+", 1="-")
range of value: -32768 ... 0 ... +32767
Note: negative values will be stored in two's complement.

Format-2: 16 Bit Unsigned Integer

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	2^{15}							2^8
Data-byte 1	2^7							2^0

range of value: 0 ... 65535

Format-3: 32 Bit Signed Integer (HIGH before LOW)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	VZ	2^{30}						2^{24}
Data-byte 1	2^{23}							2^{16}
Data-byte 2	2^{15}							2^8
Data-byte 3	2^7							2^0

VZ: "sign" (0="+", 1="-")
range of value: -2 147 483 648 ... 0 ... +2 147 483 647
Note: negative values will be stored in two's complement.

Format-4: 32 Bit unsigned Integer (HIGH before LOW)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	2^{31}							2^{24}
Data-byte 1	2^{23}							2^{16}
Data-byte 2	2^{15}							2^8
Data-byte 3	2^7							2^0

range of value: 0 ... 4 294 967 295

Format-5: 32 Bit Signed Integer (LOW before HIGH)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	2^{15}							2^8
Data-byte 1	2^7							2^0
Data-byte 2	VZ	2^{30}						2^{24}
Data-byte 3	2^{23}							2^{16}

VZ: "sign" (0="+", 1="-")
range of value: -2 147 483 648 ... 0 ... +2 147 483 647
Note: negative values will be stored in two's complement.

Format-6: 32 Bit Unsigned Integer (LOW before HIGH)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	2^{15}							2^8
Data-byte 1	2^7							2^0
Data-byte 2	2^{31}							2^{24}
Data-byte 3	2^{23}							2^{16}

range of value: 0 ... 4 294 967 295

Format-7: IEEE Floating Point Number (IEEE 754)
Short Floating Point Number IEEE 754 (Real)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	S	2^7						2^1	exponent, S (=sign)
Data-byte 1	2^0	2^{22}						2^{16}	mantissa
Data-byte 2	2^{15}							2^8	mantissa
Data-byte 3	2^7							2^0	mantissa

range of value: $-2^{128} + 2^{104}$ … $+2^{128} - 2^{104}$
$-3{,}4.10^{38}$ … $+3{,}4.10^{38}$
smallest negative number: $-2^{-149} = -1{,}4.10^{-45}$
smallest positive number: $+2^{-149} = +1{,}4.10^{-45}$
s (sign): <0> = "+"; <1> = "-"
Decimal places: 7-8

Format-7a: 32 Bit Floating Point
Short Floating Point Number IEEE 754 (Real, Short)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	2^{15}							2^8	mantissa
Data-byte 1	2^7							2^0	mantissa
Data-byte 2	S	2^7						2^1	exponent, S (=sign)
Data-byte 3	2^0	2^{22}						2^{16}	mantissa

range of value: $-2^{128} + 2^{104}$ … $+2^{128} - 2^{104}$
$-3{,}4.10^{38}$ … $+3{,}4.10^{38}$
smallest negative number: $-2^{-149} = -1{,}4.10^{-45}$
smallest positive number: $+2^{-149} = +1{,}4.10^{-45}$
s (sign): <0> = "+"; <1> = "-"
Decimal places: 7-8

Format-7b: 32 Bit Floating Point "swapped"
Short Floating Point Number IEEE 754 (Real, Short)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	S	2^7						2^1	exponent, S (=sign)
Data-byte 1	2^0	2^{22}						2^{16}	mantissa
Data-byte 2	2^{15}							2^8	mantissa
Data-byte 3	2^7							2^0	mantissa

range of value: $-2^{128} + 2^{104}$ … $+2^{128} - 2^{104}$
$-3{,}4.10^{38}$ … $+3{,}4.10^{38}$
smallest negative number: $-2^{-149} = -1{,}4.10^{-45}$
smallest positive number: $+2^{-149} = +1{,}4.10^{-45}$
s (sign): <0> = "+"; <1> = "-"
Decimal places: 7-8

Format-8: **32 Bit Transparent**

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	2^{31}							2^{24}
Data-byte 1	2^{23}							2^{16}
Data-byte 2	2^{15}							2^8
Data-byte 3	2^7							2^0

range of value: 0 ... 4 294 967 295

Format-9: **Protronic Controller Format**
(Customer specific data format)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	2^{15}							2^8	value
Data-byte 1	2^7							2^0	value
Data-byte 2	2^{15}							2^8	comma information
Data-byte 3	2^7							2^0	comma information

range of value: 0 ... 65535
comma information:
0 ... value_final = value x 10^0 (x 1)
1 ... value_final = value x 10^{-1} (x 0,1)
2 ... value_final = value x 10^{-2} (x 0,01)
3 ... value_final = value x 10^{-3} (x 0,001)
4 ... 65534: not used!

Example: 13 83 00 02 = 49,95 %

Format-10: **Unsigned 32 Bit (8 Bit Exponent + 24 Bit Value)**
(Customer specific data format)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	VZ	2^6						2^0	exponent, VZ= sign exponent
Data-byte 1	2^{23}							2^{16}	value
Data-byte 2	2^{15}							2^8	value
Data-byte 3	2^7							2^0	value

VZ: "sign"
range of value:

Format-11: **Signed 32 Bit (8 Bit Exponent + 24 Bit Value)**
(Customer specific data format)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	VZ	2^6						2^0	exponent, VZ= sign exponent
Data-byte 1	VZ	2^{22}						2^{16}	value
Data-byte 2	2^{15}							2^8	value
Data-byte 3	2^7							2^0	value

VZ: "sign"
range of value:
Note: negative values will be stored in two's complement.

Format-12: **OMV Counter Value Format**
(Customer specific data format)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	I	T	T	I	VZ	2^{26}	2^{25}	2^{24}	VZ (=sign)
Data-byte 1	2^{23}							2^{16}	
Data-byte 2	2^{15}							2^8	
Data-byte 3	2^7							2^0	

range of value:
I …… Invalid-Bit
T……... Tendency-Bit (not supported)
VZ …….. sign (not supported)

Format-13: **HARTMANN & BRAUN Measurand Format (0 – 4000)**
(Customer specific data format)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	0	I	I	0	2^{11}			2^8
Data-byte 1	2^7							2^0

I……..... Invalid-Bit
0 = Value "VALID"
1 = Value "INVALID"
range of value: 0 ... 4000
0% = 0
100% = 4000

Format-14: Bitstring of 32 Bit (LOW before HIGH)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	2^{15}							2^8
Data-byte 1	2^7							2^0
Data-byte 2	2^{31}							2^{24}
Data-byte 3	2^{23}							2^{16}

range of value: 0 ... 4 294 967 295

Format-15: Bitstring of 32 Bit (HIGH before LOW)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	2^{31}							2^{24}
Data-byte 1	2^{23}							2^{16}
Data-byte 2	2^{15}							2^8
Data-byte 3	2^7							2^0

range of value: 0 ... 4 294 967 295

Format-16: Bitstring of 16 Bit (Bit Pattern of 16 Bit)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Data-byte 0	2^{15}							2^8
Data-byte 1	2^7							2^0

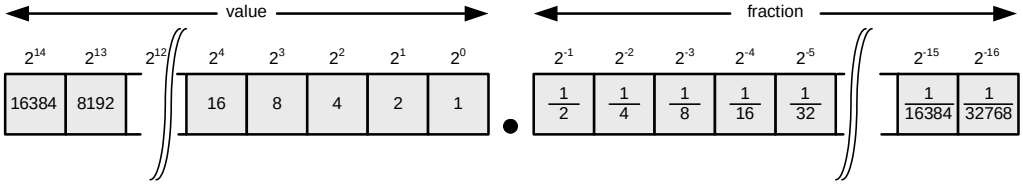
range of value: 0 ... 65535

Format-17: Real (Fixpoint 16.16)

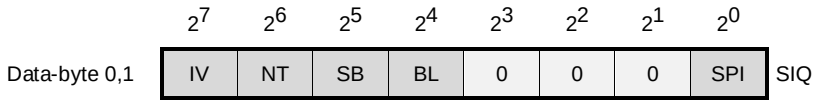
	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	VZ	2^{14}						2^8	value , VZ (=sign)
Data-byte 1	2^7							2^0	value
Data-byte 2	2^{-1}							2^{-8}	fraction
Data-byte 3	2^{-9}							2^{-16}	fraction

Value -32768 ... + 32767
fraction 0 .. 1
VZ: VZ ……"sign" (0="+", 1="-")

Example:



Format-18: IEC 60870-5-101 Single-Point Information with Qualifier



SIQ … Single-Point Information with Qualifier

SPI … Single-Point Information
<0> := OFF
<1> := ON

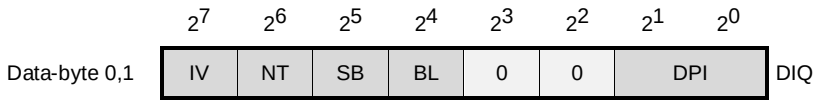
BL … Blocked
<0> := not blocked
<1> := blocked

SB … Substituted
<0> := not substituted
<1> := substituted

NT … Not Topical
<0> := topical
<1> := not topical

IV … Invalid
<0> := valid
<1> := invalid

Format-19: IEC 60870-5-101 Double-Point Information with Qualifier



DIQ … Double-Point Information with Qualifier

DPI … Double-Point Information
<0> := indeterminate or intermediate state
<1> := OFF
<2> := ON
<3> := indeterminate state

BL … Blocked
<0> := not blocked
<1> := blocked

SB … Substituted
<0> := not substituted
<1> := substituted

NT … Not Topical
<0> := topical
<1> := not topical

IV … Invalid
<0> := valid
<1> := invalid

Format-100: Time (Milliseconds n*10ms)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	0	0	0	2^4				2^0	Hour (0-23)
Data-byte 1	0	0	2^5					2^0	Minutes (0-59)
Data-byte 2	0	0	2^5					2^0	Seconds (0-59)
Data-byte 3	2^7							2^0	Milliseconds (0-99) n*10ms

Note:
- e.g. Time Set Format (MASTER → SLAVE) GAZNAT

Byte sending order:
Databyte 0 (MSB of 1st MODBUS register) is sent 1st.
Databyte 1 (LSB of 1st MODBUS register) is sent 2nd.
Databyte 2 (MSB of 2nd MODBUS register) is sent 3rd.
Databyte 3 (LSB of 2nd MODBUS register) is sent 4th.

Format-101: Time (Milliseconds n*100ms)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	0	0	0	2^4				2^0	Hour (0-23)
Data-byte 1	0	0	2^5					2^0	Minutes (0-59)
Data-byte 2	0	0	2^5					2^0	Seconds (0-59)
Data-byte 3	2^7							2^0	Milliseconds (0-9) n*100ms

Note:
- e.g. Time Set Format (MASTER → SLAVE) ABB M102

Byte sending order:
Databyte 0 (MSB of 1st MODBUS register) is sent 1st.
Databyte 1 (LSB of 1st MODBUS register) is sent 2nd.
Databyte 2 (MSB of 2nd MODBUS register) is sent 3rd.
Databyte 3 (LSB of 2nd MODBUS register) is sent 4th.

Format-102: Date + Time

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	0								
Data-byte 1	0	2^6						2^0	Year (0-99)
Data-byte 2	0	0	0	0	2^3			2^0	Month (1-12)
Data-byte 3	0	0	0	2^4				2^0	Day (1-31)
Data-byte 4	0	0	0	2^4				2^0	Hour (0-23)
Data-byte 5	0	0	2^5					2^0	Minute (0-59)
Data-byte 6	2^{15}							2^8	Milliseconds (0-59999) n*1ms
Data-byte 7	2^7							2^0	

Note:
- e.g. Time Set Format (MASTER → SLAVE) for Sepam Series 80, Alarm Panel Hathaway

Byte sending order:
Databyte 0 (MSB of 1st MODBUS register) is sent 1st.
Databyte 1 (LSB of 1st MODBUS register) is sent 2nd.
Databyte 2 (MSB of 2nd MODBUS register) is sent 3rd.
:
Databyte 7 (LSB of 4th MODBUS register) is sent last.

Year: 0-99 = 2000-2099

Notes:
- Time + Date = Binary coded
- for time synchronization only these MODBUS registers has to be included in MODBUS message format.

Format-103: Date + Time (Yokogawa Format)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	2^{15}							2^8	
Data-byte 1	2^7							2^0	Year (2001-2099)
Data-byte 2	0								
Data-byte 3	0	0	0	0	2^3			2^0	Month (1-12)
Data-byte 4	0								
Data-byte 5	0	0	0	2^4				2^0	Day (1-31)
Data-byte 6	0								
Data-byte 7	0	0	0	2^4				2^0	Hour (0-23)
Data-byte 8	0								
Data-byte 9	0	0	2^5					2^0	Minute (0-59)
Data-byte 10	0								
Data-byte 11	0	0	2^5					2^0	Seconds (0-59)

Note:
- Time Set Format (MASTER → SLAVE) for e.g. Yokogawa

Byte sending order:
Databyte 0 (MSB of 1st MODBUS register) is sent 1st.
Databyte 1 (LSB of 1st MODBUS register) is sent 2nd.
Databyte 2 (MSB of 2nd MODBUS register) is sent 3rd.
:
Databyte 11 (LSB of 6th MODBUS register) is sent last.

Year: valid range: 2001-2099

Notes:
- Time + Date = Binary coded
- For time synchronization only these MODBUS registers has to be included in MODMUS message format.

Format-104: Date + Time (SICAM FCM)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	2^{15}							2^8	Milliseconds (0-59999) n*1ms HIGH
Data-byte 1	2^7							2^0	Milliseconds (0-59999) n*1ms LOW
Data-byte 2	0	0	0	2^4				2^0	Hour (0-23)
Data-byte 3	0	0	2^5					2^0	Minute (0-59)
Data-byte 4	0	0	0	0	2^3			2^0	Month (1-12)
Data-byte 5	2^2	WOT	2^0	2^4		Tag		2^0	Day of Week + Day
Data-byte 6	0								Year (0) HIGH
Data-byte 7	2^7							2^0	Year (12-99) LOW

Note:
- Time Set Format (MASTER → SLAVE) e.g. for SICAM FCM

Byte sending order:
Databyte 0 (MSB of 1st MODBUS register) is sent 1st.
Databyte 1 (LSB of 1st MODBUS register) is sent 2nd.
Databyte 2 (MSB of 2nd MODBUS register) is sent 3rd.
:
Databyte 7 (LSB of 4th MODBUS register) is sent last.

Year: 12-99 = 2012-2099

Note: Time + Date = Binary Coded

Format-120: Date + Time (BCD)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	=0								1/100 Seconds (0-99)
Data-byte 1	0-5				0-9				Seconds (0-99)
Data-byte 2	0-5				0-9				Minutes (0-59)
Data-byte 3	0-2				0-9				Hour (0-23)
Data-byte 4	0-3				0-9				Day (1-31)
Data-byte 5	0,1				0-9				Month (1-12)
Data-byte 6	0-9				0-9				Year (0-99)
Data-byte 7	0				1-7				Day of Week (1-7)

Note:
- e.g. Time Set Format (SLAVE → MASTER) OMV

Byte sending order:
Databyte 0 (MSB of 1st MODBUS register) is sent 1st.
Databyte 1 (LSB of 1st MODBUS register) is sent 2nd.
Databyte 2 (MSB of 2nd MODBUS register) is sent 3rd.
:
Databyte 7 (LSB of 4th MODBUS register) is sent last.

Day of Week: 1= Monday, 2 = Tuesday, … 7 = Sunday
Year: 0-99 = 2000-2099

Note: Time + Date = BCD coded

Format-121: Date + Time (BCD reverse)

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
Data-byte 0	0				1-7				Day of Week (1-7)
Data-byte 1	0-9				0-9				Year (0-99)
Data-byte 2	0,1				0-9				Month (1-12)
Data-byte 3	0-3				0-9				Day (1-31)
Data-byte 4	0-2				0-9				Hour (0-23)
Data-byte 5	0-5				0-9				Minutes (0-59)
Data-byte 6	0-5				0-9				Seconds (0-99)
Data-byte 7	=0								1/100 Seconds (0-99)

Note:
- e.g. Time Set Format (SLAVE → MASTER) OMV

Byte sending order:
Databyte 0 (MSB of 1st MODBUS register) is sent 1st.
Databyte 1 (LSB of 1st MODBUS register) is sent 2nd.
Databyte 2 (MSB of 2nd MODBUS register) is sent 3rd.
:
Databyte 7 (LSB of 4th MODBUS register) is sent last.

Day of Week: 1= Monday, 2 = Tuesday, … 7 = Sunday
Year: 0-99 = 2000-2099

Note: Time + Date = BCD coded

Literature

SICAM RTUs **.** Ax 1703 Common Functions MODBUS	DC0-088-2
SICAM RTUs Common Functions System- and Basic System Elements	DC0-015-2

International Standards

Modicon MODBUS Protocol Reference Guide	PI-MBUS-300 (Rev. J)
Modbus Application Protocol Specification, V 1.1b	
Modbus Messaging on TCP Implementation Guide, Rev. 1.0b	