

---

## TECNOHOLD DEVELOPMENT TECHNOLOGY

# PROTOCOLO MODBUS - MANUAL DO CLIENTE

### Índice

<b>1. Apresentação e características .....</b>	<b>2</b>
Tabela 1 - Formato do quadro de comunicação .....	2
<b>2. Funções MODBUS.....</b>	<b>3</b>
2.1 - Funções MODBUS para o THMB-01 .....	3
Tabela 2 – Códigos das Funções MODBUS .....	3
2.2 - Bancos de Memória .....	3
2.3 - Read Input Registers (0x04) .....	4
Tabela 3 – Mapeamento da memória acessível pela função .....	4
2.4 – Organização da memória – Read Input Registers: .....	5
2.5 - Exemplo de comunicação com o Conversor – Read Input Registers .....	6
2.5.1 - Leitura de estado dos dispositivos do loop 3 .....	6
<b>3. Exception Response (ERROR) .....</b>	<b>7</b>
Tabela 4 – Código de Respostas de Exceção .....	7
<b>4. Tempos na comunicação MODBUS .....</b>	<b>8</b>
Tabela 5 – Nomenclatura dos tempos na comunicação MODBUS .....	8
Tabela 6 – Tempos padrões do protocolo MODBUS .....	8
Tabela 7 – Tempos adotados no conversor THMB-01 .....	8

## 1. Apresentação e características

O conversor de protocolos THMB-01 com comunicação baseada no protocolo MODBUS possibilita a conversão do protocolo do sistema de detecção e alarme de incêndio SIGMA-485E para sistemas supervisores de terceiros, como PCs, PLCs ou outros tipos de sistemas automatizados.

As especificações técnicas da forma de comunicação do protocolo MODBUS padrão são:

- Taxa transmissão = 9600 bps
- Paridade = nenhuma
- Bits de parada = 2 bits
- Bits de dados = 8 bits

O conversor de protocolos THMB-01 se comunica através do padrão MODBUS “Remote Terminal Unit” (RTU) - Modo de transmissão no qual os dados são transmitidos em oito bits, contendo dois caracteres hexadecimais de quatro bits cada, na estrutura da tabela abaixo:

<b>MASTER (PCs, PLCs ou outros sistemas automatizados) e SLAVE (THMB-01)</b>
Endereço do Slave 8 bits
Código da Função 8 bits
Dados N * 8 bits
Error Check (feito com o polinômio <u>A001h</u> ) <sup>(*)</sup> 16 bits
Intervalo de Silêncio T1 T2 T3

Tabela 1 - Formato do quadro de comunicação.

A interface de comunicação utiliza o driver RS-485 a dois fios com comunicação do tipo half-duplex, ou seja, é possível receber ou transmitir dados, mas não os dois simultaneamente.

Com exceção da taxa de transmissão, os demais parâmetros são fixos. A taxa de transmissão pode ser trocada na especificação do pedido e, uma vez definida, esta não mais poderá ser alterada.

O conversor THMB-01 se comunica no sistema “MASTER-SLAVE” (ou MESTRE-ESCRAVO), onde apenas o “MASTER” pode começar um diálogo com os “SLAVES”. O conversor foi projetado para ser um “SLAVE” do sistema, podendo ou não responder aos questionamentos do “MASTER”.

<sup>(\*)</sup> O pacote transmitido deve ser checado através do método “Cyclical Redundancy Check” (CRC), cálculo polinomial descrito abaixo, em linguagem de programação:

No início da checagem, o registrador “Error Check” é preenchido com FFFFh. Durante a geração do CRC, é feita a operação “Ou exclusiva” com cada dado de 8 bits e o “Error Check”. Em seguida, o resultado é rotacionado uma vez na direção do bit menos significativo (LSB) e deve entrar com um zero para preencher a posição do bit mais significativo (MSB), O LSB é extraído e analisado. Se o LSB foi um (1) então é feito um “Ou exclusivo” com um valor fixo predefinido (no caso deste conversor, A001h), se o LSB for zero (0) nada é feito. Repetir este processo de rotação 8 vezes por byte anexado ao “Error Check” pelo “Ou exclusivo”.

---

## **2. Funções MODBUS**

### **2.1 - Funções MODBUS para o THMB-01**

<b>Código</b>	<b>Função</b>	<b>Descrição</b>
04	Read Input Registers	Ler Registradores de Status do Módulo
19	Reset Comm. Link	Resetar a comunicação com o módulo

Tabela 2 – Códigos das Funções MODBUS.

### **2.2 - Bancos de Memória**

A memória foi organizada por 64 bancos contendo 128 “Words” para armazenar os estados dos dispositivos, 68 “Words” para armazenar o estado dos Loops e painéis e 64 bancos contendo 128 “Words” com o valor analógico de periféricos que possuem tal função.

A interface MODBUS é capaz de supervisionar até 64 Loops de um sistema.

### **2.3 - Read Input Registers (0x04)**

Esta função permite ler os registradores (estado) de cada um dos 8192 endereços de dispositivos do sistema SIGMA sendo limitado a pacotes com no máximo cento e vinte e cinco (125) "Words", conforme o mapa de memória descrito abaixo:

Conjunto	Endereço (Word)	Descrição	Tipo
1	30001~30128	Estado dispositivos Loop 1	128 x Word
2	30129~30256	Estado dispositivos Loop 2	128 x Word
3	30257~30384	Estado dispositivos Loop 3	128 x Word
4	30385~30512	Estado dispositivos Loop 4	128 x Word
...	...	...	...
64	38065~38192	Estado dispositivos Loop 64	128 x Word
1	38193	Estado Loop 1	Word
2	38194	Estado Loop 2	Word
3	38195	Estado Loop 3	Word
4	38196	Estado Loop 4	Word
...	...	...	...
68	38261	Estado Painel Redundante	Word
...	38262	Contador de eventos de Fogo	Word
	38263	Contador de eventos Gerais	Word
...	...	Reservado para uso futuro (Lê sempre 0)	...
1	38321~38448	Valor analógico Dispositivos Loop 1	128 x Word
2	38449~38576	Valor analógico Dispositivos Loop 2	128 x Word
3	38577~30704	Valor analógico Dispositivos Loop 3	128 x Word
4	38705~38832	Valor analógico Dispositivos Loop 4	128 x Word
...	...	...	...
64	46385~46512	Valor analógico Dispositivos Loop 5	128 x Word
...	...	Reservado para uso futuro (Lê sempre 0)	...

Tabela 3 – Mapeamento da memória acessível pela função "Read Input Registers".

Esta função também permite a leitura de sensores de corrente do sistema de alarme de incêndios SIGMA 485-E. Estes sensores são normalmente utilizados para detectar gases explosivos no ambiente supervisionado, a fim de alertar a área quando existe um nível de gás acima do considerado seguro. Os valores dos sensores são convertidos para décimos de miliamperes, para facilitar a programação do usuário e garantir maior precisão do valor lido.

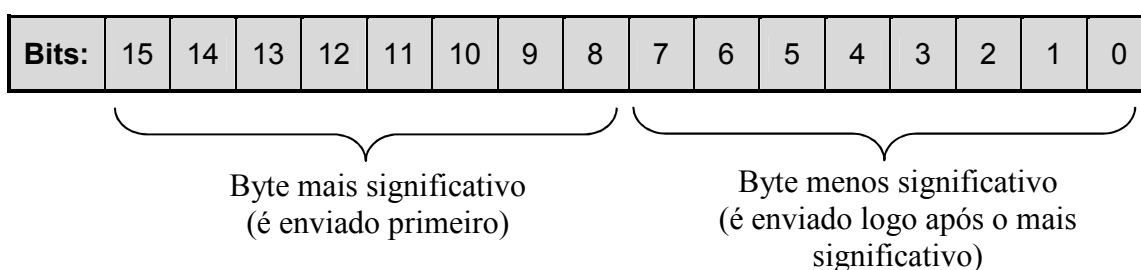
## 2.4 – Organização da memória – Read Input Registers:

Cada conjunto de memória de 128 “Words” representa os estados dos dispositivos de um Loop, ou seja, cada “Word” representa o estado de um dispositivo do loop.

Cada “Word” possui 16 bits, sendo que cada um representa um estado do dispositivo:

Quando o bit de um dispositivo nos registradores está em nível lógico “0” indica que o dispositivo correspondente está em condições normais de funcionamento, já o nível lógico “1” indica que houve um evento.

Word (palavra):

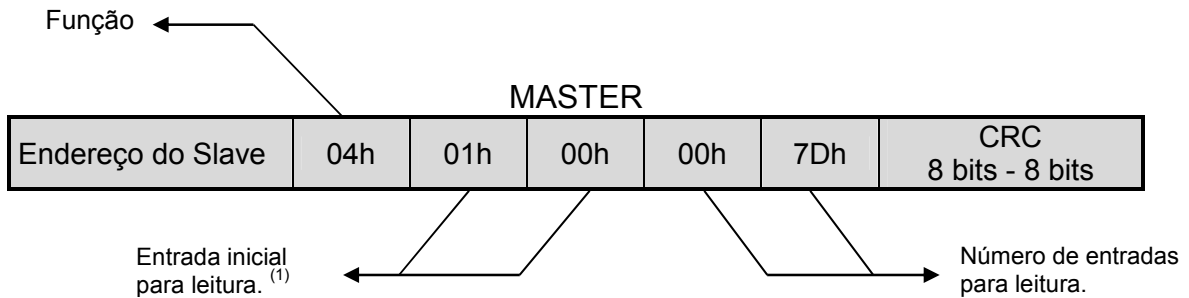


- Bit 0: Indica que o dispositivo gerou primeiro alarme de Fogo.
- Bit 1: Indica que o dispositivo gerou confirmação do primeiro alarme de fogo.
- Bit 2: Indica falha no funcionamento do dispositivo.
- Bit 3: Não utilizado.
- Bit 4: Indica que o dispositivo está bloqueando o disparo do gás de combate à incêndio.
- Bit 5: Indica que o piro disparou o gás no setor programado.
- Bit 6: Não utilizado.
- Bit 7: Não utilizado.
- Bit 8: Indica que o dispositivo está inibido (inoperante).
- Bit 9: Indica falha de comunicação com o dispositivo.
- Bit 10: Indica um alerta de aviso no dispositivo.
- Bit 11: Indica dispositivo em modo de Manutenção (loops e painéis).
- Bit 12: Indica Convocação de brigada de incêndio no setor do dispositivo.
- Bit 13: Indica Autorização de combate de gás no setor do dispositivo.
- Bit 14: Não utilizado.
- Bit 15: Não utilizado.

## 2.5 - Exemplo de comunicação com o Conversor – Read Input Registers:

### 2.4.1 - Leitura de estado dos dispositivos do loop 3:

Para ler o estado dos dispositivos do Loop três, deve-se enviar o quadro solicitando leitura de 125 “Words” a partir da posição 30257:



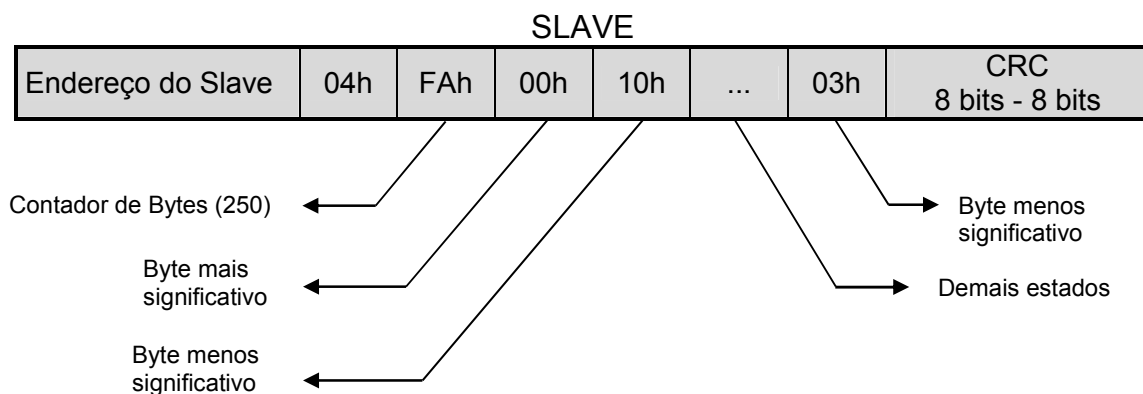
<sup>(1)</sup> A entrada inicial para leitura é obtida pelo calculo:

$$END = (LP-1) * 128 + 30001$$

Onde: END => Endereço da memória; LP => Endereço do Loop.

**Exemplo:** Loop três => END = (3-1) \* 128 + 30001 = 30257.

Logo depois, remove-se o primeiro dígito (número 3) e subtrai do resultado o valor um (1). No exemplo, a entrada 30257 (decimal) é requisitada como 0x0100 (hexadecimal): 30257 = 257 = (257 - 1) = 256 = 0x0100 hexadecimal.



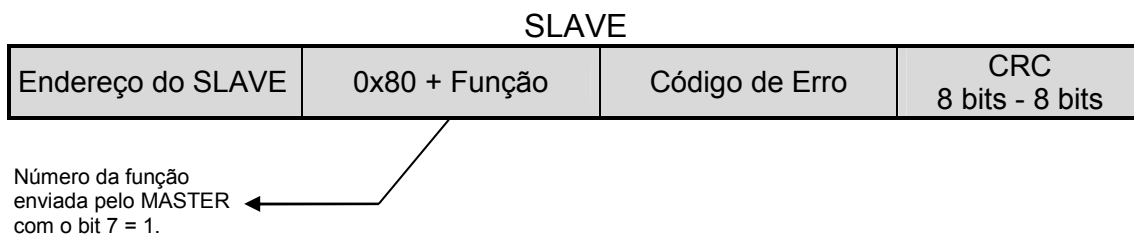
O registro “byte count” é igual ao total de entradas multiplicada por dois (2), pois cada Word possui 2 bytes (entradas).

Os endereços de cada dispositivo correspondem à ordem das Words recebidas, ou seja, a primeira Word deste quadro indica o estado do dispositivo 1, representados pelos bits 0 ao 15, respectivamente, a Word 2 corresponde ao estado do dispositivo 2, etc., até a última Word que representa o dispositivo 125.

### 3. Exception Response (ERROR)

Os erros de comunicação detectados pelo slave são manipulados de duas maneiras:

1. “No Reply” (sem resposta): Caso o slave detecte um erro no formato dos dados, Erro na checagem do CRC, etc., ele ignorará a mensagem.
2. Exception Response (erro de exceção): Ocorre caso a checagem dos dados forem válidos, mas a informação presente no pacote não corresponder a alguma condição do conversor. O SLAVE enviará um frame com um código de erro, na formatação abaixo:



Os possíveis códigos de erro são:

Exception Responses		
Código	Nome	Descrição
01	ILLEGAL FUNCTION	Função não suportada pelo Módulo
02	ILLEGAL DATA ADDRESS	Violação do espaço de memória estabelecido pela função
03	ILLEGAL DATA VALUE	Valor especificado não suportado pela função
04	SLAVE DEVICE FAILURE	Erro de leitura de memória do dispositivo
06	SLAVE DEVICE BUSY	Módulo ocupado e executando uma função
07	NEGATIVE ACKNOWLEDGE	A função é válida, mas o dispositivo não pode executá-la.

Tabela 4 – Código de Respostas de Exceção.

O conversor MODBUS enviará o código 4 – SLAVE DEVICE FAILURE caso o módulo tenha perdido a comunicação com o painel, ou caso não consiga acessar a memória interna.

## 4. Tempos na comunicação MODBUS

Os seguintes tempos devem ser respeitados na comunicação entre o MASTER e o SLAVE, segundo protocolo MODBUS:

Ação	Equivalente
Questionar um mesmo Slave (caso ocupado)	DELAY_RETRY_SEND_FRAME
Tempo para iniciar a transmissão de um frame	DELAY_SEND_FRAME
Aguardar a recepção de um frame	TIMEOUT_RECEIVE_FRAME
Aguardar a recepção de um byte do frame	TIMEOUT_RECEIVE_BYTE
Tempo para confirmar silêncio na rede	CONFIRM_SILENT

Tabela 5 – Nomenclatura dos tempos na comunicação MODBUS.

Independente da velocidade de comunicação, os seguintes tempos do máster são sempre os mesmos:

Ação	Tempo
DELAY_RETRY_SEND_FRAME	1 segundo
DELAY_SEND_FRAME	100 ms
TIMEOUT_RECEIVE_FRAME	2 segundos

Tabela 6 – Tempos padrões do protocolo MODBUS.

Para 9600 bps, o MASTER e o SLAVE devem possuir os seguintes tempos:

DELAY	MASTER	SLAVE
DELAY_SEND_FRAME	100 ms	8 ms
TIMEOUT_RECEIVE_FRAME	2 segundos	∞
TIMEOUT_RECEIVE_BYTE	4 ms	4 ms
CONFIRM_SILENT	-	4 ms

Tabela 7 – Tempos adotados no conversor THMB-01.