

Driver MOELLER Easy

Nome do Arquivo	Easy.dll
Fabricante	Moeller Eaton Industries
Equipamentos	Equipamentos Moeller Easy
Protocolo	Proprietário
Versão	3.0.1
Última Atualização	10/08/2026
Plataforma	Win32
Dependências	IOMKit versão 1.15 ou superior
Superblocos	Não
Nível	0

Introdução

Este Driver permite a comunicação entre sistemas da **Elipse Software** e equipamentos Easy da Moeller Eaton Industries, utilizando um protocolo de comunicação proprietário. A parte da comunicação utiliza um cabo RS-232 normal, fornecido pela Moeller Eaton Industries.

Configuração do Driver

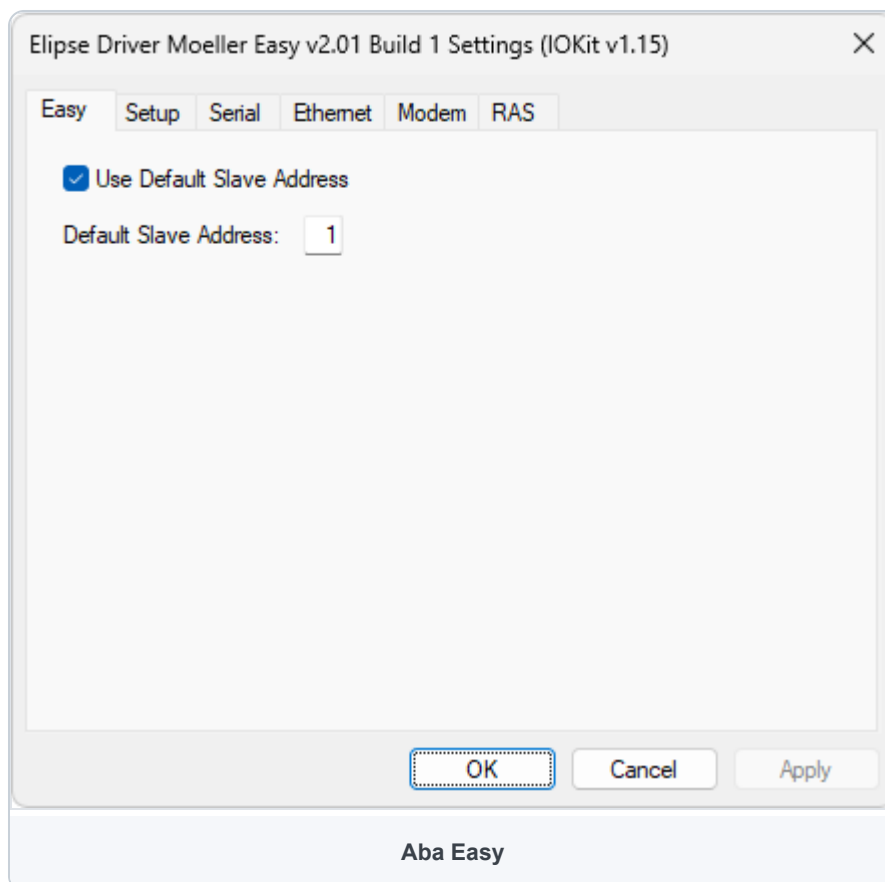
Esta seção contém informações sobre a configuração dos parâmetros **[P]** deste Driver.

Parâmetros [P]

P1	Não utilizado, deixe em 0 (zero)
P2	Não utilizado, deixe em 0 (zero)
P3	Não utilizado, deixe em 0 (zero)

P4	Não utilizado, deixe em 0 (zero)
----	----------------------------------

Todas as configurações são realizadas na aba **Easy** da janela de propriedades deste Driver, mostrada na figura a seguir.



A opção **Use Default Slave Address** permite configurar um valor padrão a ser usado como endereço de um equipamento Easy **Escravo**. Esta opção vem selecionada por padrão e o valor padrão da opção **Default Slave Address** é 1 (um). Se a opção **Use Default Address** está selecionada, qualquer valor usado nos parâmetros *N1* ou *B1* dos Tags deste Driver são desconsiderados.

NOTA

O número máximo de Elementos de um Tag Bloco é 8 (oito, o máximo suportado pelo protocolo). Caso seja criado um Tag Bloco que exceda este limite, os Elementos após este limite são ignorados por este Driver.

Referência de Tags

Esta seção contém informações sobre a configuração dos Tags **[N/B]** deste Driver.

Parâmetros N e B de Endereçamento de Tags

N1 ou B1	Endereço de um equipamento Easy Escravo , entre 1 (um) e 15
----------	--

N2 ou B2	Tipo de objeto para leitura ou escrita ou o número de um comando. Para mais informações, consulte as tabelas Comandos disponíveis e Leitura ou escrita
N3 ou B3	Endereço, usado somente para leitura ou escrita e não para comandos
N4 ou B4	Não usado, deixe em 0 (zero)

Comandos

Comandos disponíveis

N2 OU B2	N3 OU B3	COMANDO
C0h	-	Parar
C1h	-	Executar
C2h	-	Habilitar <i>Write Program</i> (Desabilitar Editor)
C3h	-	Desabilitar <i>Write Program</i> (Verificar EEPROM)
C4h	-	Reiniciar <i>System Test (Watchdog)</i>
C5h	-	Inverter <i>Remanenz On or Off</i>
C6h	-	Inverter <i>Anlauf Stop or Run</i>
C7h	-	Inverter LED
C8h	-	Ligar <i>P-Keys</i>
C9h	-	Desligar <i>P-Keys</i>
CAh	-	Ligar <i>I-Entprellung</i>
CBh	-	Desligar <i>I-Entprellung</i>
CCh	-	Ligar <i>Daylight Savings</i>
CDh	-	Desligar <i>Daylight Savings</i>

N2 OU B2	N3 OU B3	COMANDO
CEh	-	Ligar Modo Remoto
CFh	-	<i>Clear Display in Remote Mode</i>
D0h	-	Desligar Modo Remoto
D1h	-	<i>Hole vierstelligen Wert im Remotemodus</i>
D2h	-	Hablilitar Auftragsbuch Sonder-Objekt
D3h	-	Desabilitar Auftragsbuch Sonder-Objekt

Leitura ou Escrita

Leitura ou escrita

N2 OU B2	N3 OU B3	TIPO DE OBJETO
00h	Endereço	Programa
01h	Endereço	Dados Dinâmicos, Identificação e Status
02h	Endereço	Relógio interno
03h	Endereço	Display, apenas em Modo Remoto
04h	Endereço	Memória especial para Depuração e Monitoramento
05h	Endereço	<i>Auftragsbuch für Sonderspeicher</i>

NOTA

Para uma lista de objetos e respectivos endereços, consulte o tópico Dicas Sucom Easy.

Exemplos

Para acessar as saídas digitais de um equipamento Easy, os parâmetros de um Tag PLC devem ser os seguintes:

- **N1:** 1 (um, **Escravo** com identificador um)

- **N2:** 01h (dados dinâmicos, identificação e status)
- **N3:** 16h (para mais informações, consulte o tópico Dicas Sucom Easy)
- **N4:** 0 (zero, não utilizado)

Para acessar os *Markers* de 0 (zero) a 7 (sete), os parâmetros de um Tag PLC devem ser os seguintes:

- **N1:** 1 (um, **Escravo** com identificador um)
- **N2:** 01h (dados dinâmicos, identificação e status)
- **N3:** 14h (para mais informações, consulte o tópico Dicas Sucom Easy)
- **N4:** 0 (zero, não utilizado)

Para ler e ajustar o relógio interno de um equipamento Easy, os parâmetros de um Tag Bloco devem ser os seguintes:

- **B1:** 1 (um, **Escravo** com identificador um)
- **B2:** 02h (relógio interno)
- **B3:** 00h (para mais informações, consulte o tópico Dicas Sucom Easy)
- **B4:** 0 (zero, não utilizado)
- **Size:** 3 (três, para mais informações, consulte o tópico Dicas Sucom Easy)

De acordo com o tópico Dicas Sucom Easy, os Elementos de um Tag Bloco para o relógio interno são os seguintes:

- Dia da semana, entre 0 (zero) e 6 (seis)
- Hora, entre 0 (zero) e 23
- Minuto, entre 0 (zero) e 59

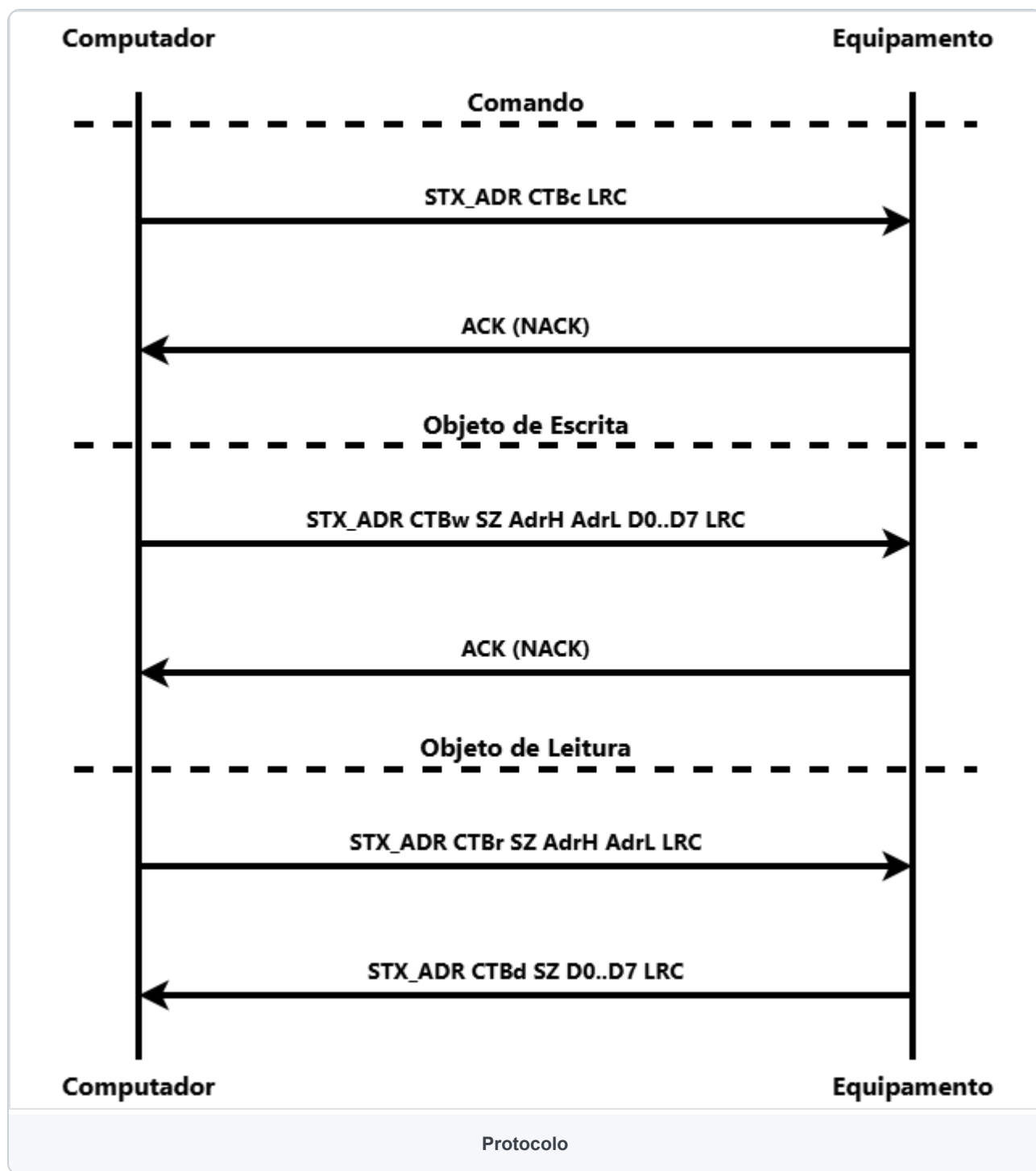
Pode-se ler estes Elementos separadamente mas, para configurar o relógio interno, um Tag Bloco deve ser escrito de uma vez só.

Dicas Sucom Easy

SUCOM-E

Transferência de dados para um computador

O protocolo aqui descrito permite a comunicação entre a porta serial RS-232 de um computador e a interface de um equipamento Easy com uma velocidade de 4800 bauds, 8 (oito) bits de dados e 1 (um) bit de parada, sem checagem de paridade.



STX_ADR = 1100aaaaB Byte de Controle Comando PC -> Easy
 ++++----- Endereçamento (16)
 00000 : Computador
 00001 : Easy

O byte **STX** sempre inicia a comunicação e contém o endereço de um equipamento Easy. Portanto, este protocolo suporta operações *multi-drop*, embora este recurso ainda não seja suportado pela porta serial RS-232.

A interface de programação Easy provê os serviços básicos descritos a seguir, codificados em um byte de controle, que é o primeiro byte a ser transferido.

Comandos em que o código de controle começa com o valor C0h (são permitidos até 32 comandos)

```
CTBc = 11000000B Byte de Controle Comando PC -> Easy
|||+++++--Código do Comando (32)
|||      00000: Parar
|||      00001: Iniciar
|||      00010: Habilitar programa de ESCRITA (-> Desabilita Editor)
|||      00011: Desabilitar programa de ESCRITA (Verifica+->EEPROM)
|||      00100: Reiniciar teste do sistema (Watchdog)
|||      00101: Retenção ON/OFF
|||      00110: Execução STOP/RUN
|||      00111: Alternar estado do LED em 4xx (não implementado em 6xx)
|||      01000: Pushbuttons ON
|||      01001: Pushbuttons OFF
|||      01010: Debouncing ON
|||      01011: Debouncing OFF
|||      01100: Horário de Verão ON
|||      01101: Horário de Verão OFF
|||      01110: Ativar Modo Remoto
|||      01111: Limpar display no Modo Remoto
|||      10000: Desativar Modo Remoto
|||      10001: Buscar valor de 4 dígitos no Modo Remoto
|||      10010: Habilitar objeto especial de Order Book
|||      10011: Desabilitar objeto especial de Order Book
+++----- Comando
```

Um computador deve enviar um comando **EASY HALT** através da sequência a seguir.

```
STX_ADR (C1h), CTBc (C0h), LRC (C0h)
```

O byte **LRC** é igual a soma dos dados entre os bytes **STX_ADR** e **LRC**.

Objeto de memória contendo a leitura, um código de controle endereçado a partir de 80h, ou a escrita, um código de controle endereçado a partir de A0h (podem ser endereçados até 32 objetos)

```
CTBr = 10000000B Byte de Controle PC -> Easy "Leitura de objeto a partir de ENDEREÇO"
CTBw = 10100000B Byte de Controle PC -> Easy "Escrita de objeto a partir de ENDEREÇO"
|||+++++-- Código de Objetos (32 objetos)
|||      00000: Programa
|||      00001: Imagens (dados dinâmicos + ID, status)
|||      00010: Definir relógio
|||      00011: Display (somente no Modo Remoto)
|||      00100: Memória especial para debugging/visualização
|||      00101: Order book para armazenamento especial
+++----- 0 = Objeto de leitura
            1 = Objeto de escrita
```

A outra estrutura da mensagem consiste, primariamente, de 3 (três) bytes de controle. Em **SZ**, o número de bytes de dados é indicado e pode chegar a 8 (oito). Um ponteiro para um endereço de 16 bits é adicionado ao início de um objeto. Cada objeto pode ter um tamanho de até 64 Kb.

```
SZ   = número de bytes de dados 1-8 (8 bytes)
AdrH = endereço inicial relativo ao início do objeto (High)
AdrL = endereço inicial relativo ao início do objeto (Low)
```

A isto se sucede uma mensagem de dados (*letter* para um equipamento Easy) de até 8 (oito) bytes de dados e um *check sum* (**LRC**), que contém a soma a partir de **AdrL**. Um equipamento Easy responde aos comandos com um byte de **ACK** (*Acknowledge*), **NACK** (*NoAcknowledge*) ou **REJ** (*Rejeição*).

```
REJ  01010000b ; REJECT + Adr.PC
NACK 00110000b ; NACK + Adr.PC
ACK  01100000b ; ACK + Adr.PC
```

Se um pacote de dados é solicitado, o protocolo responde no formato a seguir.

```
CTBd 00010100b ; 14h: Easy -> PC "Dados Recebidos";
D0.D7 com até 8 (oito) bytes de dados; LRC
```

Composição de objetos e formatos

O cabeçalho de um objeto programa contém, entre outros dados, os valores de *set point* dos blocos de sistema.

Offset	Descrição
=====	
Cabeçalho do Programa	
0000h	2 Bytes Comprimento do programa (00 = Programa deletado) Low, High
0002h	1 Byte Comprimento dos parâmetros (094h = 148 bytes)
0003h	1 Byte Senha do programa
0004h	1 Byte Easy hardware mínimo requerido pelo programa
	00000000
	+-- Analógica: 1=True 0=False
	+--- Hora: 1=True 0=False
	+---- I16_Q8: 1=True 0=False
	+----- S_R_Data: 1=True 0=False
	+----- Pushbuttons usados no programa
	+----- Flag de Retenção
	+----- 0=Pushbuttons: 0=Desabilitados 1=Habilitados
	+----- 0=Debouncing: 0=Sim 1=Não
0005h	8 bytes Nome do programa
000Dh	1 byte Cabeçalho do Check sum (os primeiros 16 bytes 0-f ADDC)
000Eh	1 byte Check sum do programa
000Fh	1 byte Check sum do campo de parâmetros
Parâmetros (set point)	
0010h	1 byte Byte de controle do módulo de timer T0
	00000000
	+++-- 000 = X Atraso de elevação (on delay)
	001 = ° Atraso de soltura (off delay)
	010 = ?X Aleatório - Atraso de elevação
	011 = ?° Aleatório - Atraso de soltura
	100 = _U_ Gerador de Clock
	+----- Base de tempo:
	00 = 10 ms
	01 = Segundos
	10 = Minutos
	11 = Não implementado
	+----- Não implementado
	+----- 0= - Remove do menu de parâmetros
	1= + Adiciona ao menu de parâmetros
	+----- 0= Parado
	1= Em progresso
0011h	2 bytes T0: Set point em 10 ms: S: (S*100+ms)LOW,HIGH
	T0: Set point em segundos: M:S (M*60+S) LOW,HIGH
	T0: Set point in minutos: H:M (H*60+M) LOW,HIGH
0013h	Parâmetros T2
0016h	Parâmetros T3
0019h	Parâmetros T4
001Ch	Parâmetros T5
001Fh	Parâmetros T6
0022h	Parâmetros T7
0025h	Parâmetros T8
0028h	1 byte Byte de controle do contador C0

```

00000000
|+----- 0= - Remove do menu de parâmetros
|          1= + Adiciona ao menu de parâmetros
+----- 0= Parado
          1= Em progresso
0029h    2 Bytes C0: Set point: Low,High (0-9999)
002Bh    Parâmetros C2
002Eh    Parâmetros C3
0031h    Parâmetros C4
0034h    Parâmetros C5
0037h    Parâmetros C6
003Ah    Parâmetros C7
003Dh    Parâmetros C8
0040h    1 byte Byte de controle W1
00000000
+----- 0= Parado
          1= Em progresso
0041h    W1 1 byte Byte de controle do Canal 1, Dia da Semana Inicial-Final
00000000
|||||+++-- Data Inicial
|||||      000 = Canal: não editado
|||||      001 = Se
|||||      010 = Te
|||||      011 = Qu
|||||      100 = Qu
|||||      101 = Se
|||||      110 = Sa
|||||      111 = Do
|||+----- Data Final
|||      000 = Desabilitar
|||      001 = Se
|||      010 = Te
|||      011 = Qu
|||      100 = Qu
|||      101 = Se
|||      110 = Sa
|||      111 = Do
|||
|+----- 1= Ton>Toff
|          0= Ton<=Toff ou Ton ou Toff
|          "Off" = (Bit.15=1)
+----- 0= - Remover do menu de parâmetros
          1= + Adicionar ao menu de parâmetros
0042h    W1 2 bytes Canal 1, Hora/Minuto On (Low,High) Formato: Inteiro "HH*60+MM"
          "Off" = Bit.15=1 e 10-Date=3
0044h    W1 2 bytes Canal 1, Hora/Minuto Off (Low,High.7=1:Off) Formato: Inteiro "HH*60+MM"
          "Off" = Bit.15=1 e 10-Date=3
0046h    W1 1 byte Byte de controle do Canal 1, Dia da Semana Inicial-Final
0047h    W1 2 byte Canal 2, Hora/Minuto On
0049h    W1 2 byte Canal 2, Hora/Minuto Off
004Bh    W1 1 byte Byte de controle do Canal 2, Dia da Semana Inicial-Final
004Ch    W1 2 byte Canal 3, Hora/Minuto On

```

```

004Eh      W1 2 byte Canal 3, Hora/Minuto Off
0050h      W1 1 byte Byte de controle do Canal 3, Dia da Semana Inicial-Final
0051h      W1 2 byte Canal 4, Hora/Minuto On
0053h      W1 2 byte Canal 4, Hora/Minuto Off
0055h      Parâmetro W2
006Ah      Parâmetro W3
007Fh      Parâmetro W4
0094h      1 byte Byte de controle do comparador analógico A1
00000000
|||||+-- 0 = >=
|||||    1 = <=
|||||+--- Modo de Comparação
|||||    00 = IA1 : IA2
|||||    01 = IA1 : Const
|||||    10 = IA2 : Const
|||||    11 =
|+----- 0= - Remover do menu de parâmetros
|          1= + Adicionar ao menu de parâmetros
+----- 0= Parado
          1= Em progresso
0095h      1 byte A1: Constante: (Formato: 0-99)
0096h      Parâmetro A2
0098h      Parâmetro A3
009Ah      Parâmetro A4
009Ch      Parâmetro A5
009Eh      Parâmetro A6
00A0h      Parâmetro A7
00A2h      Parâmetro A8
00A4h      4 bytes Compensação/Offset para o limite de paginação (16 Bytes) de EEPROMs seriais
Programa do usuário (início do diagrama)
00A8h      8 bytes por linha * 121 linhas (968 bytes)
0470h      e outras 2 linhas reservas
MMI Módulos de Texto
0480h      8 textos com 80 bytes de códigos de controle e texto (640 bytes)
06FFh      Final do objeto programa
Escrita nos endereços listados não é bloqueada.
Deste modo, podem ocorrer travamentos durante esta operação

```

Um objeto **imagem** contém os dados atuais de entradas, saídas e *flags*, entre outros, e também os valores de tempo real dos módulos de sistema.

Offset	Descrição
=====	
	Configuração Atual (determinada pelo auto-teste)
	Recursos providos pelo equipamento
0000h	1 byte Configuração do Equipamento / identificador de tipo
	00000000
	+-- Analógico: 1=True 0=False
	+--- Relógio: 1=True 0=False
	+---- I16_Q8: 1=True 0=False
	+----- S_R_Data: 1=True 0=False
	+----- Início: 1=True 0=False
	+----- 1=1. Linguagem de consulta de tempo
	+----- 1=DST ID
	+----- AC/DC: 1=AC 0=DC
	Tipo Código Tipo Código
	=====
	412-DC-R 01h 412-AC-R 80h
	412-DC-RC 03h 412-AC-RC 82h
	412-DC-TCX 03h 412-AC-RCX 82h
	620-DC-TC 0Fh 618-AC-RC 8Eh
	EASY Status Equipamento
0001h	1 byte informação de status
	00000000
	+-- Estado: 1=Executando 0=Parado
	+--- Senha: 1=Senha ativa
	+---- Programa: 1=Esquemático disponível
	+----- Teste: 1=Teste de hardware ativado
	+----- Cartão: 1=Cartão de memória plugado
	+----- I2C: 1=I2C ocupado, acesso EEPROM
	+----- Salvar Par. 1=Salvar parâmetros (EEPROM)
	+----- Salvar Pro. 1=Escrever programa (EEPROM) 620-DC-TCX 0Fh 618-AC-RCX 8Eh
	EASY Erros/Configuração
0002h	1 byte Flags de Erros e parte 2 da configuração
	00000000
	+-- EEPROM: Defeito na EEPROM
	+--- EEAck: EEPROM não responde
	+---- Relógio: Defeito no relógio
	+----- LCD: Defeito no display ou display inexistente
	+----- ACLow: Nível AC abaixo do necessário
	+----- Rede: Desconectado/indisponível
	+----- X-Type: 1=Tipo sem display (s. o.)
	+----- InithW: 1=Inicialização do hardware
	Configuração de Referência (requerida pelo programa)
	Recursos requeridos pelo programa do equipamento
0003h	1 byte Flags de Configuração
	00000000
	+-- Analógico: 1=Dispositivos analógicos em uso
	+--- Relógio: 1=Relógio em uso
	+---- I16Q8: 1=I>8 programa em uso
	+----- SR-Operando: 1=Envio/Recebimento em uso
	+----- Push Buttons: 1=Push Buttons em uso

```

    ||+----- Retenção:      1=Retenção habilitada
    |+----- Push buttons: 1=Push Buttons habilitados
    +----- Debouncing:      1=Debouncing desabilitado

Versão / Linguagem
0004h    1 byte Versão do S0 (10h = V1.0)
0005h    1 byte Linguagem de exibição
          0: Inglês, 1: Alemão, 2: Francês, 3: Espanhol, 4: Italiano
          5: Sueco, 6: Holandês, 7: Polonês, 8: Português, 9; Turco

Dados de imagem
0006h          3 bytes Referência de relógio
              WD, Dia da semana 0-6
              HH, Hora 0-23
              MM, Minuto 0-59

0009h          2 bytes Referência das entradas analógicas I7,I8 (Formato: 0-99)
000Bh          1 byte Constant Analógica 0-k0 / N * k0 = FF (deve ser "0")
000Ch          2 bytes Imagem das entradas I0 até I16
000Eh          1 byte livre
000Fh          1 byte Push buttons

00000000
|||||||
|||||||+----- Tesquerda
|||||||+----- Tcima
|||||+----- Tbaixo
|||+----- Tdireita
||+----- Tesc
||+----- Tok
|+----- Tdel
+----- Tpen

0010h          1 byte Imagem de saída do Timer
              OutputTimer: 0-7 (Bit 0 = T1 etc.)
0011h          1 byte Imagem de saída do Counter
              OutputCounter: 0-7
0012h          1 byte Imagem de saída do módulo de relógio W
              OutputClockWatch: 0-3
0013h          1 byte Imagem de saída do bloco analógico A
              OutputIAnalogComp: 0-7

Mais dados de imagem
0014h          Image_M0: Flag: 0-7
0015h          Image_M8: Flag: 8-F
0016h          Image_Q0: Saída Digital: 0-7
0017h          Image_S0: Enviar comunicação de saída: 0-7
0018h          Image_D0: Flag de texto: 0-7
0019h          Image_R0: Receber comunicação: 0-7
001Ah          Image_R8: Receber comunicação: 8-F

Dados de imagem para módulos de Timer
001Bh          T1: Flags de Status
              00000000 Byte de Controle
              |||||||
              |||||++-- 000 = X Atraso de Elevação.(ON delay)
              |||||    001 = ° Atraso de soltura.(OFF delay)
              |||||    010 = _|- Atraso de elevação aleatório
              |||||    011 = _U_ Gerador de clock

```

```

|||||
|||||+---- 0= MS
|||||      1= Segundos
|||||+---- Reset-SET
|||+----- "SET"-Estado = SET-ALT
|+----- "RES"-Estado
|+----- "Tx"-Estado
+----- 0= Parado
          1= Incremento de INTERRUPT-TIMER

001Ch      2 bytes T1: Valor instantâneo do timer
001Eh      1 byte  T1: Valor aleatório
001Fh      Timer 2 w.o.
0023h      Timer 3
0027h      Timer 4
002Bh      Timer 5
002Fh      Timer 6
0033h      Timer 7
0037h      Timer 8

  Dados de imagem para módulos contadores
003Bh      1 byte C1: Flag de Status
          2 byte C1: Valor do Contador
003Eh      Contador 2 w.o.
0041h      Contador 3
0044h      Contador 4
0047h      Contador 5
004Ah      Contador 6
004Dh      Contador 7
0050h      Contador 8

  Dados internos
0053h      1 byte FB Watch: Compara minutos
0054h      1 byte NetIdentCode: Identificação da extensão conectada
          0 Erro, 1 Expensão AC, 2 Expensão DC, 3 Easy200Easy, 4 Participantes da rede
0054h      Final do objeto imagem
          Escritas acima deste limite levam a um comportamento indefinido do computador
O objeto Uhr (relógio, em alemão) contém o relógio de tempo real
Offset      Descrição
=====
0000h      3 bytes Referência de relógio
          WD, Dia da semana 0-6
          HH, Hora 0-23
          MM, Minuto 0-59

A escrita deste objeto realiza a definição do horário
Com a seguinte sequência de comandos, definimos o horário atual para segunda, 12:00:
STX_ADR (C1h), CTBw Clock-Object (A2h), Number (03h), OffsetH (0), OffsetL (0),
WK (0=Monday), HH (12d), MM (0), LRC (B1h)
O Easy devolve um ACK (B0h) quando o comando é executado

```

Um objeto **memória especial** é utilizado para otimizar a transferência de dados para visualização.

Offset	Descrição
=====	
Array de bytes, onde cada byte representa uma área de dados de uma entrada correspondente no order book	
0000h	Tamanho do array
0001h	Dados relacionados ao primeiro array no order book
0002h	Dados para o segundo array, etc.
A memória é gerenciada dinamicamente	

O objeto **memória especial para order book** controla as entradas que um computador escreve no objeto **memória especial**.

Offset	Descrição
=====	
É manipulado usando um array de words	
0000h	Primeiro elemento de visualização especial de memória
0002h	Segundo... etc.
Format:	
aaaa aaaa-aaao oooo	
+-++++- Código de Objeto	
++++-++++-+++----- 12 bit Endereço Offset	
A memória é gerenciada dinamicamente. O código 1Fh marca o final do objeto.	
Enviar o comando e-SUCOM "Habilita Objeto Especial Order Book (10010)" irá sobrescrever este código.	
Códigos (bytes) dos objetos de visualização no order book:	
00000: Programa	
00001: Imagens (dados dinâmicos + ID, status)	
00010: Definir relógio	
00011: Display (somente no Modo Remoto)	
00100: Memória especial para debugging/visualização	
00101: Order book para armazenamento especial	

NOTA

Formatos e conteúdos de dados inválidos enviados para um equipamento Easy podem resultar em travamento deste equipamento ou comportamentos indesejados, como por exemplo atuação de saídas. Isto pode causar acidentes e levar à morte.

Documentação das Interfaces de Comunicação

Esta seção contém a documentação das Interfaces de Comunicação referentes ao Driver **Easy**.

Esta documentação é compartilhada e mantida no manual do IOKit:

- Configurações de um Driver
- Configurações Gerais
- Configuração de Estatísticas
- Configuração da Interface Ethernet
- Configuração da Interface Modem
- Configuração da Interface RAS

- Configuração da Interface Serial

Histórico de Revisões do Driver

VERSÃO	DATA	AUTOR	COMENTÁRIOS
3.0.1	10/08/2026	M. Ludwig	• Driver atualizado para a biblioteca IOKit versão 3 e Visual Studio 2022 (<i>Case 38767</i>).
2.1.1	16/08/2013	G. Taschetto	• Traduzido o tópico Dicas Sucom Easy do alemão para o português (<i>Case 2845</i>).
2.0.1	23/04/2009	M. Zani M. Ludwig	• Driver portado para a biblioteca IOKit (<i>Case 6815</i>). • Driver portado para o Windows CE (<i>Case 10331</i>).
1.1.1	29/03/2004	C. Mello	• Corrigido um problema de instabilidade de comunicação, mantendo o sinal RTS sempre em nível alto.
1.0.1	15/08/2001	R. Haetinger	• Versão inicial deste Driver.

Tópicos Adicionais

Topics not linked from the main table of contents:

- Sucom Easy Hinweis

Sucom Easy Hinweis

SUCOM-E

Datenübertragung zum PC

Folgende Protokolle werden über die UART des PC und die Programmiergeräteschnittstelle des EASY mit 4800 Baud 8 Datenbit / 1 Stopbit / ohne Parity abgewickelt:

```

Protokoll          PC          EASY
=====
Commando:  ----- STX_ADR STBc LRC ----->
           <----- ACK (NACK) -----
           <----- ACK (NACK) -----

Write_Objekt: ----- STX_ADR STBw ANZ AdrH AdrL D0..D7 LRC ----->
           <----- ACK (NACK) -----

Read_Objekt: ----- STX_ADR STBr ANZ AdrH AdrL LRC ----->
           <----- STX_ADR STBd ANZ D0..D7 LRC -----

STX_ADR = 1100aaaaB Steuerbyte PC->Easy "Comando"
++++----- Codierung der Adresse (16)
           00000 : PC
           00001 : EASY

```

Das StartOfText-Byte beginnt immer die Kommunikation und enthält die Adresse des EASY-Geräts das angesprochen werden soll. Die Schnittstelle ist so softwareseitig für Multidropbetrieb ausgelegt, dies wird aber z. Zt. von der Hardware (RS232) nicht unterstützt.

Die Programmiergeräteschnittstelle des Easy kennt grundsätzlich drei Dienste, die im Steuerbyte, dem ersten Byte der Übertragung codiert sind.

1. Kommandos mit dem Steuercode beginnend bei C0h. Es können bis zu 32 Kommandos implementiert werden.

```

STBc = 11000000BSteuerbyte PC->Easy "Comando"
|||+++++-- Codierung der Comandos (32)
|||      00000 : Halt
|||      00001 : Start
|||      00010 : Enable WRITE Programm(-> Disable Editor)
|||      00011 : Disable WRITE Programm (Check+-->EEPROM)
|||      00100 : Reset Systemtest ( Watchdog )
|||      00101 : Toggle Remanenz ON/OFF
|||      00110 : Toggle Anlauf STOP/RUN
|||      00111 : bei 4xx Toggle LED, bei 6xx nicht implementiert
|||      01000 : P-Tasten ein
|||      01001 : P-Tasten aus
|||      01010 : I-Entprellung ein
|||      01011 : I-Entprellung aus
|||      01100 : Sommerzeit ein
|||      01101 : Sommerzeit aus
|||      01110 : Remotemodus einschalten
|||      01111 : Display im Remotemodus loeschen
|||      10000 : Remotemodus ausschalten
|||      10001 : Hole vierstelligen Wert im Remotemodus
|||      10010 : Enable Auftragsbuch Sonder-Objekt
|||      10011 : Disable Auftragsbuch Sonder-Objekt
+++----- Kommando

```

Der PC muss für das Kommando ‚EASY HALT‘ folgenden seriellen Strom übertragen:

STX_ADR (C1h), STBc (C0h), LRC (C0h) das LRC ist gleich der Summe der Daten zwischen STX_ADR und LRC.

2. und 3. Objekt aus Speicher lesen mit Steuercode ab 80h und in Speicher schreiben mit Steuercode ab A0h. Es sind bis zu 32 Objekte ansprechbar.

```
STBr = 10000000B Steuerbyte PC->Easy "Read-Objekt ab ADRESS "
STBw = 10100000B Steuerbyte PC->Easy "Write-Objekt ab ADRESS "
|||+++++-- Objekt-Codierung (32 Objekte)
|||      00000 : Programm
|||      00001 : Abbilder (dynamische Daten+Ident,Status)
|||      00010 : Setze-Uhr
|||      00011 : Display ( nur im Remotemodus )
|||      00100 : Sonderspeicher für Debugging/Visualisierung
|||      00101 : Auftragsbuch fuer Sonderspeicher
+++----- 0=Read Objekt
            1=Write Objekt
```

Der weitere Telegrammaufbau besteht zunächst aus drei weiteren Steuerbytes. Im Anzahl-Byte wird die Anzahl der angehängten Datenbytes mitgeteilt. Dies können bis zu 8 sein. Ein 16-Bit-Adresszeiger wird dem Objektbeginn hinzuaddiert um an die gewünschte Information in diesem Objekt zu gelangen. Ein Objekt kann daher bis zu 64k groß sein.

```
ANZ  = Anzahl der Datenbytes 1-8 (8 Bytes)
AdrH = Relative Anfangsadresse zum Objekt-Begin (High)
AdrL = Relative Anfangsadresse zum Objekt-Begin (Low)
```

Darauf folgen bei einem Datentelegramm (Schreiben in das Easy) bis zu 8 Datenbytes und ein Checksummenbyte (LRC) das die Inhaltssumme ab AdrL enthält.

Das EASY antwortet auf die Befehle mit einem Byte als Acknowledge, NoAcknowledge oder einem Reject.

```
REJ  01010000b ; REJECT+ Adr.PC
NACK 00110000b ; NACK+ Adr.PC
ACK  01100000b ; ACK + Adr.PC
```

Wird ein Datenpaket angefordert setzt sich das Antwortprotokoll wie folgt zusammen:

```
STBd 00010100b ; 14h: Easy->PC "Quittung DATEN";
D0.D7 bis zu 8 Bytes mit Daten; LRC
```

Objektzusammensetzung und Formate

Das Objekt Programm enthält im sogenannten Header unter anderem die Sollwerte der Systembausteine.

OffsetBeschreibung

=====

Programmheader

```
0000h      2 Bytes Programmlaenge (00=Progr.geloescht) LOW,HIGH
0002h      1 Byte Parameterfeldlaenge (094h = 148 Byts)
0003h      1 Byte Programm-Password
0004h      1 Byte minimale EASY-Hardware-Anforderung vom Programm
            00000000
            |||||+-- Analog:      1=Da  0=nicht
            |||||+--- Uhr:       1=Da  0=nicht
            |||||+---- I16_Q8:   1=Da  0=nicht
            |||+----- S_R_Data: 1=Da  0=nicht
            ||+----- P-Tasten im Programm genutzt
            |+----- Flag_Remanenz
            |+----- 0=P-Tasten:      0= gesperrt 1=aktiv
            +----- 0=I-Entprellung: 0=Ja      1=nein
0005h      8 Bytes Programmname
000Dh      1 Byte Checksumme Header ( die ersten 16 Bytes 0-f ADDC )
000Eh      1 Byte Checksumme Programm
000Fh      1 Byte Cheksumme Parameterfeld
```

Parameterfeld (Sollwerte)

```
0010h      1 Byte Steuerbyte Timerbaustein T0
0
            |||||+++- 000 = X  Anzugsverz.(EinschaltVerz.)
            |||||      001 = °  Abfallverz.(AusschaltVerz.)
            |||||      010 = ?X Zufall-Anzugsverz.
            |||||      011 = ?° Zufall-Abfallverz.
            |||||      100 = _U_ TaktGenerator
            |||||
            |||+----- Zeitbasis:
            |||      00= 10-ms
            |||      01= SEKUNDE
            |||      10= MINUTE
            |||      11= nicht implementiert
            |||
            |+----- nicht implementiert
            +----- 0= - nicht im Parameter-Menue
            |      1= + im Parameter-Menue
            +----- 0= nicht in Bearbeitung
            |      1= in Bearbeitung
0011h      2 Bytes T0: Soll_Wert in 10.ms: S: (S*100+ms)LOW,HIGH
            T0: Soll_Wert in Sek : M:S (M*60+S) LOW,HIGH
            T0: Soll_Wert in Min : H:M (H*60+M) LOW,HIGH
0013h      Parameterfeld T2
0016h      Parameterfeld T3
0019h      Parameterfeld T4
001Ch      Parameterfeld T5
```

```

001Fh      Parameterfeld T6
0022h      Parameterfeld T7
0025h      Parameterfeld T8
0028h      1 Byte Steuerbyte Counter C0
           00000000
           |+----- 0= - nicht im Parameter-Menue
           |           1= + im Parameter-Menue
           +----- 0= nicht in Bearbeitung
           |           1= in Bearbeitung

0029h      2 Bytes C0: Soll_Wert: LOW,HIGH (0-9999)
002Bh      Parameterfeld C2
002Eh      Parameterfeld C3
0031h      Parameterfeld C4
0034h      Parameterfeld C5
0037h      Parameterfeld C6
003Ah      Parameterfeld C7
003Dh      Parameterfeld C8
0040h      1 Byte Steuerbyte W1
           00000000
           +----- 0= nicht in Bearbeitung
           |           1= in Bearbeitung

0041h      W1 1 Byte Steuerbyte für Channel 1, Wochentag Begin-End
           00000000
           |||||++--- Tag-Begin
           |||||      000 = Channel: NICHT-Editiert
           |||||      001 = Mo
           |||||      010 = Tu
           |||||      011 = We
           |||||      100 = Th
           |||||      101 = Fr
           |||||      110 = Sa
           |||||      111 = Su
           ||++----- Tag-End
           ||          000 = disable
           ||          001 = Mo
           ||          010 = Tu
           ||          011 = We
           ||          100 = Th
           ||          101 = Fr
           ||          110 = Sa
           ||          111 = Su
           ||
           |+----- 1= Ton>Toff
           |           0= Ton<=Toff oder Ton oder Toff
           |           "Off" ist (Bit.15=1))
           +----- 0= - nicht im Parameter-Menue
           |           1= + im Parameter-Menue

0042h      W1 2 Byte Channel 1, Stunde/Minute ON (Low,High) Format:
Integer "HH*60+MM"
           "Off" ist Bit.15=1 und Zehner-Tag =3

0044h      W1 2 Byte Channel 1, Stunde/Minute OFF (Low,High.7=1:0ff)
Format: Integer "HH*60+MM"

```

```

"Off" ist Bit.15=1 und Zehner-Tag =3
0046h    W1 1 Byte Steuerbyte für Channel 1, Wochentag Begin-End
0047h    W1 2 Byte Channel 2, Stunde/Minute ON
0049h    W1 2 Byte Channel 2, Stunde/Minute OFF
004Bh    W1 1 Byte Steuerbyte für Channel 2, Wochentag Begin-End
004Ch    W1 2 Byte Channel 3, Stunde/Minute ON
004Eh    W1 2 Byte Channel 3, Stunde/Minute OFF
0050h    W1 1 Byte Steuerbyte für Channel 3, Wochentag Begin-End
0051h    W1 2 Byte Channel 4, Stunde/Minute ON
0053h    W1 2 Byte Channel 4, Stunde/Minute OFF
0055h    Parameterfeld W2
006Ah    Parameterfeld W3
007Fh    Parameterfeld W4
0094h    1 Byte Steuerbyte Analovergleicher A1
          00000000
          |||||+-- 0 = >=
          |||||    1 = <=
          |||||++--- Mode Vergleicher
          |||||    00 = IA1 : IA2
          |||||    01 = IA1 : Const
          |||||    10 = IA2 : Const
          |||||    11 =
          |+----- 0= - nicht im Parameter-Menue
          |          1= + im Parameter-Menue
          +----- 0= nicht in Bearbeitung
          1= in Bearbeitung
0095h    1 Byte A1: Constante: (Format: 0-99)
0096h    Parameterfeld A2
0098h    Parameterfeld A3
009Ah    Parameterfeld A4
009Ch    Parameterfeld A5
009Eh    Parameterfeld A6
00A0h    Parameterfeld A7
00A2h    Parameterfeld A8
00A4h    4 Bytes Ausgleich für 16Byte-Pagegrenze bei seriellen EEPROMs

```

Anwenderprogramm (Schaltplanbeginn)

```

00A8h    8 Bytes je Zeile * 121 Zeilen ( 968 Bytes )
0470h    und noch mal 2 Zeilen reserve

```

MMI-Textbausteine

```

0480h    8 Texte mit je 80 Bytes Steuercodes und Text ( 640 Bytes )

```

```

06FFh    Programm-Objekt Ende
Ein Schreiben auf die folgenden Adressen wird nicht verhindert,
sodaß es zu Abstürzen kommen kann wenn dies gemacht wird.

```

Das Objekt Abbild enthält die aktuellen Daten von Eingängen, Ausgängen, Merkern etc. und auch die Istwerte der Systembausteine.

OffsetBeschreibung

=====

Istkonfiguration (ermittelt aus Selbsttest)

Ressourcen die das Gerät zur Verfügung stellt

0000h 1 Byte Gerätekonfiguration / Typenkennung

00000000

|||||+-- Analog: 1=Da0=nicht
 |||||+--- Uhr: 1=Da 0=nicht
 ||||+---- I16_Q8: 1=Da 0=nicht
 |||+----- S_R_Data: 1=Da 0=nicht
 ||+----- Anlauf: 1=Stop0=Run
 |+----- 1=1. Mal Sprachabfrage
 +----- 1=Kennung Sommerzeit
 +----- AC/DC:1=AC0=DC

Type	Code	Type	Code
=====			

412-DC-R	01h	412-AC-R	80h
412-DC-RC	03h	412-AC-RC	82h
412-DC-TCX	03h	412-AC-RCX	82h
620-DC-TC	0Fh	618-AC-RC	8Eh
620-DC-TCX	0Fh	618-AC-RCX	8Eh

EASY-Gerätestatus

0001h 1 Byte Statusinformationen

00000000

|||||+-- Zustand: 1=RUN 0=HALT
 |||||+--- Passwort: 1=Passwort aktiviert
 ||||+---- Programm: 1=Schalplan vorhanden
 |||+----- Test: 1=Hardwaretest aktiv
 ||+----- Card: 1=Memory Card gesteckt
 |+----- I2C: 1=I2C busy, EEPROM-Zugriff
 +----- Save Param 1=Parameter werden geschrieben (EEPROM)
 +----- Save Prog 1=Programm wird geschrieben (EEPROM)

EASY-Fehlermeldungen / Konfiguration

0002h 1 Byte Fehlerflags und Konfiguration Teil 2

0

|||||+-- EEPROM: EEPROM defekt
 |||||+--- EEAck: EEPROM antwortet nicht
 ||||+---- Uhr: Uhr defekt
 |||+----- LCD: Display defekt/nicht vorhanden
 ||+----- ACLow: AC-Pegel nicht ok

```

    ||+----- Network: Teilnehmer abgekoppelt/nicht vorhanden
|+----- X-Type: 1=Type ohne Display ( s. o. )
    +----- InitHW: 1=Hardwareinitialisierung läuft
    Sollkonfiguration ( vom Programm angefordert )
    Ressourcen die das Programm vom Gerät benötigt

```

0003h 1 Byte Konfigflags

```

0
    |||||+-- Analog: 1=Analogbausteine genutzt
    |||||+--- Uhr: 1=Uhr verwendet
    ||||+---- I16Q8: 1=I>8 vom Programm genutzt
    |||+----- SR-Operand: 1=Send/Receive genutzt
    ||+----- PTasten: 1=PTasten genutzt
    |+----- Remanenz: 1=Remanenz eingeschaltet
    +----- Ptasten ein: 1=Ptasten eingeschaltet
    +----- Entprellung: 1=Entprellung ausgeschaltet

```

Version / Sprache

```

0004h 1 Byte Version OS ( 10h = V1.0 )
0005h 1 Byte eingestellte Display-Sprache
    0,engl. 1,deutsch 2,frz. 3,span. 4,ital. 5,schwed.
    6,nl. 7,poln. 8,port. 9,türk.

```

Abbilddaten

```

0006h 3 Bytes Uhrenreferenz
    W0, Wochentag 0-6
    HH, Stunde 0-23
    MM, Minute 0-59
0009h 2 Bytes Referenz Analogeingänge I7,I8 (Format: 0-99)
000Bh 1 Byte Analogkonstante 0-k0 / N * k0 = FF ( muss "0" sein )
000Ch 2 Bytes Eingangabbild I0 bis I16
000Eh 1 Byte frei
000Fh 1 Byte Abbild P-Tasten 00000000B
    |||||
    |||||+----- Tleft
    |||||+----- Tup
    ||||+----- Tdown
    |||+----- Tright
    ||+----- Tesc
    |+----- Tok
    +----- Tdel
    +----- Tpen

0010h 1 Byte Ausgangsabbild Timer
OutputTimer: 0-7 ( Bit 0 = T1 usw. )
0011h 1 Byte Ausgangsabbild Counter
OutputCounter: 0-7

```

```

0012h      1 Byte Ausgangsabbild Uhrenbaustein W
           OutputClockWatch: 0-3
0013h      1 Byte Ausgangsabbild Analogbaustein A
           OutputIAnalogComp: 0-7

```

Weitere Abbilddaten

```

0014h      Abbild_M0: Merker: 0-7
0015h      Abbild_M8: Merker: 8-F
0016h      Abbild_Q0: Output-Digital: 0-7
0017h      Abbild_S0: Output-SendKommunikation: 0-7
0018h      Abbild_D0: Textmerker: 0-7
0019h      Abbild_R0: ReciveKommunikation: 0-7
001Ah      Abbild_R8: ReciveKommunikation: 8-F

```

Abbilddaten für Timerbausteine

```

001Bh      T1: Zustands-Merker

           00000000 SteuerByte
           |||||
           |||||++-- 000 = X Anzugsverz.(EinschaltVerz.)
           |||||    001 = ° Abfallverz.(AusschaltVerz.)
           |||||    010 = _|- Zufall-Anzugsverz.
           |||||    011 = _U_ TaktGenerator
           |||||
           |||||+---- 0= MS
           |||||    1= SEC
           |||+----- Reset-SET
           ||+----- "SET"-Zustand = SET-ALT
           |+----- "RES"-Zustand
           |+----- "Tx"-Zustand
           +----- 0= nicht in Bearbeitung
                   1= INTERRUPT-TIMER increment

001Ch      2 Bytes T1: Timer-Istwert
001Eh      1 Byte T1: Random Wert
001Fh      Timer 2 w.o.
0023h      Timer 3
0027h      Timer 4
002Bh      Timer 5
002Fh      Timer 6
0033h      Timer 7
0037h      Timer 8

```

Abbilddaten für Counterbausteine

```

003Bh      1 Byte C1: Zustands-Merker
           2 Byte C1: Counter-Istwert
003Eh      Counter 2 w.o.
0041h      Counter 3
0044h      Counter 4

```

```
0047h      Counter 5
004Ah      Counter 6
004Dh      Counter 7
0050h      Counter 8
```

Interne Daten

```
0053h      1 Byte FB Watch: Compare-Minute
0054h      1 Byte NetIdentCode: Kennung der angeschlossenen Erweiterung
           0,Fehler 1,AC-Erweiterung 2,DC-Erweiterung, 3,Easy200Easy
           4,Netzwerkteilnehmer ( DP.. )
```

```
0054h      Abbild-Objekt Ende
           Das Schreiben über diese Grenze führt zu nicht vorhersehbarem
Verhalten der CPU
```

Das Objekt Uhr enthält die Echtzeit

```
Offset      Beschreibung
=====
```

```
0000h      3 Bytes Uhrenreferenz
           WO, Wochentag 0-6
           HH, Stunde 0-23
           MM, Minute 0-59
```

Das Schreiben des Objekts führt zum Setzen der Uhrzeit.

Mit dem folgenden Befehlestrom setzen wir die Uhrzeit auf 12:00 am Montag:

```
STX_ADR ( C1h ), STBw Uhrenobjekt ( A2h ), Anzahl ( 03h ), OffsetH ( 0 ), OffsetL ( 0 ), WO ( 0=Montag ),
HH ( 12d ), MM ( 0 ), LRC ( B1h )
```

Das Easy gibt ein ACK (B0h) zurück wenn der Befehl ausgeführt ist.

Das Objekt "Sonderspeicher" dient zunächst zur Optimierung bei der Übertragung von Visualisierungsdaten

```
Offset      Beschreibung
=====
```

Es handelt sich um ein Bytearray. Jeder Eintrag repräsentiert den Zieldatenbereich eines korrespondierenden Eintrag im Auftragsbuch.

```
0000h      Arraygröße
0001h      Daten, die zum ersten Vektor im Auftragsbuch gehören
0002h      ..., zum 2. Vektor gehören usw.
```

Der Speicher wird dynamisch verwaltet.

Das Objekt "Auftragsbuch Sonderspeicher" steuert die Einträge, die die CPU in das Objekt "Sonderspeicher" schreibt

Offset	Beschreibung
=====	

Es handelt sich um ein Wortarray.

0000h	Erstes Element zur Visualisierung im Sonderspeicher
0002h	Zweites ... usw.

Format:

```

aaaa aaaa-aaa0 0000
|||| |||| |||| ||||
|||| |||| |||+---- Objektcode
++++-++++-+++----- 12 Bit Offsetadresse zum Objektbeginn

```

Der Speicher wird dynamisch verwaltet. Als Endekennung wird das Objekt mit dem Code 1Fh definiert. Nach dem Senden des SUCOM-E-Befehls "Enable Auftragsbuch Sonder-Objekt (10010)" wird die Endekennung überschrieben.

Codes der Visualisierungsobjekte (Bytes) im Auftragsbuch:

```

00000 : Programm
00001 : Abbilder (dynamische Daten+Ident,Status)
00010 : Setze-Uhr
00011 : Display ( nur im Remotemodus )
00100 : Sonderspeicher für Debugging/Visualisierung
00101 : Auftragsbuch fuer Sonderspeicher

```

HINWEIS:

Fehlerhafte Datenformate bzw. Dateninhalte die zum EASY übertragen werden, können das Gerät zum Absturz bringen oder ungewollte Reaktionen, wie z.B. das Setzen von Ausgängen usw. hervorrufen. Dadurch kann es zu Personenschäden bzw. tödlichen Unfällen kommen.