

## Driver COUTH MC2000

|                         |                                          |
|-------------------------|------------------------------------------|
| Nome do Arquivo         | MC2000.dll                               |
| Fabricante              | Couth Industrial Marking Systems, S.L.U. |
| Equipamentos            | Série MC2000                             |
| Protocolo               | MC 2000T2                                |
| Versão                  | 2.0.1                                    |
| Última Atualização      | 10/08/2026                               |
| Plataforma              | Win32                                    |
| Dependências            | IOKit versão 1.8 ou superior             |
| Leitura com Superblocos | Não                                      |
| Nível                   | 0                                        |

## Introdução

O Driver COUTH MC2000 deve ser utilizado para realizar a comunicação entre aplicações da **Elipse Software** e os marcadores por micropontos da série MC 2000 T<sup>2</sup> da Couth Industrial Marking Systems, S.L.U.

## Configuração do Driver

Utilize a interface da biblioteca **IOKit** para configurar a comunicação com um equipamento. Para mais informações, consulte o tópico Documentação das Interfaces de Comunicação.

## Referência de Tags

Todos os Tags deste Driver são configurados com valores numéricos através dos parâmetros **[N/B]**.

## Estado de um Marcador

### Somente Leitura

|    |                                          |
|----|------------------------------------------|
| N1 | Endereço do identificador de um marcador |
|----|------------------------------------------|

|    |               |
|----|---------------|
| N2 | 2 (dois)      |
| N3 | Não utilizado |
| N4 | Não utilizado |

Tag PLC para leitura do estado atual de um marcador. Os valores retornados por este Tag ao consultar o estado de um marcador são os seguintes:

- **0:** A unidade de controle encontra-se em repouso, pronta para realizar uma nova marcação
- **1:** A unidade de controle está realizando uma marcação
- **2:** A unidade de controle está realizando uma marcação, que neste momento se encontra em pausa ou à espera de receber mais dados
- **3:** A última marcação foi cancelada pelo operador
- **4:** Ocorreu um erro durante a última marcação, pois um texto a ser marcado excede os limites da marcação
- **5:** Ocorreu um erro durante a última marcação, pois um logotipo a ser marcado excede os limites da marcação
- **6:** Ocorreu um erro durante a última marcação, pois um código *DataMatrix* a ser marcado excede os limites da marcação
- **7:** Ocorreu um erro durante a última marcação, pois não foi detectada a unidade marcadora ao dirigi-la à posição de origem
- **8:** Ocorreu um erro durante a última marcação, pois não foi possível encontrar o tipo de letra selecionado
- **9:** Ocorreu um erro durante a última marcação, pois não foi possível encontrar o logotipo selecionado
- **10:** Ocorreu um erro durante a última marcação, pois não foi possível encontrar o arquivo selecionado
- **11:** Ocorreu um erro durante a última marcação, pois não foi possível acessar a memória interna

## Marcar Texto Sem Resposta

### Somente Escrita

|    |                                          |
|----|------------------------------------------|
| N1 | Endereço do identificador de um marcador |
| N2 | 57                                       |
| N3 | Não utilizado                            |
| N4 | Não utilizado                            |

Tag PLC para envio de uma mensagem para marcação de uma controladora, em formato **Texto**. Para saber se a mensagem enviada já foi marcada ou não por uma controladora, utilize o Tag Estado de um Marcador.

As controladoras suportam a impressão de caracteres ASCII que variam do código 20 até 7F, em hexadecimal. Além destes caracteres ASCII, o usuário também pode inserir códigos especiais para formatar as mensagens, como por exemplo usar o código 0A em hexadecimal para inserir uma quebra de linha, ou *line feed*, em uma mensagem.

## Exemplo de Script no Elipse SCADA

```
tag001 = "Texto Linha 1" + Chr(0Ah) + "Texto Linha 2"
tag001.Write()
```

## Exemplo de Script no E3, Power ou Water

```
Application.GetObject("Driver.tag001").Value = "Texto Linha 1" & Chr(&H0A) & "Texto Linha 2"
Application.GetObject("Driver.tag001").Write(EWriteSyncMode)
```

Para usar outros códigos especiais de formatação em uma mensagem, consulte o manual do usuário da controladora.

# Documentação das Interfaces de Comunicação

Esta seção contém a documentação das Interfaces de Comunicação referentes ao Driver **MC 2000 T²**.

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                                                                                                    |
|--------|------------|-----------|----------------------------------------------------------------------------------------------------------------|
| 2.0.1  | 10/08/2026 | M. Ludwig | • Driver atualizado para a biblioteca <b>IOKit</b> versão <b>3</b> e Visual Studio 2022 ( <i>Case 38773</i> ). |
| 1.0.1  | 12/04/2012 | C. Mello  | • Versão inicial deste Driver ( <i>Case 12671</i> ).                                                           |