

# CONFIGURAÇÃO DOS AUTÓMATOS

## INTRODUÇÃO

Este documento apresenta as características principais dos autómatos que irão ser utilizados no trabalho prático, bem como aspetos relacionados com a respetiva configuração.

## AUTÓMATO PROGRAMÁVEL

Cada grupo dispõe de um autómato do fabricante Schneider Electric dos modelos **Premium TSX P57** ou **M340** com as seguintes características:

- Fonte de alimentação.
- Carta CPU (processador).
- Carta de comunicações *ethernet* incorporada no CPU ou em carta separada.
- Entradas e saídas (E/S), dependendo do modelo do autómato, com a seguinte configuração:
  - 1 carta de 16 saídas digitais, ligadas às luzes indicadoras.
  - 1 carta de 16 entradas digitais, ligadas aos botões de pressão.

OU

- 1 carta com 8 entradas e 8 saídas digitais, ligadas respetivamente aos botões de pressão e luzes indicadoras.

Existem algumas pequenas diferenças entre o hardware dos autómatos utilizados pelos vários grupos, contudo estas não são relevantes para a realização do trabalho.

**Tome nota da referência de todos os equipamentos que possui na sua bancada.** Isto é importante para saber quais os manuais que deve ler/consultar durante a realização do trabalho.

## REPOSITÓRIO DE SOFTWARE E MANUAIS

A primeira tarefa que deve realizar é estudar com detalhe o autómato que vai utilizar. Este estudo implica conhecer o respetivo *hardware* e *software*.

Por forma a facilitar o estudo fora das aulas, são disponibilizados conteúdos que podem ser acedidos a partir do servidor que existe no laboratório 1005. O acesso é realizado através do seguinte link<sup>1</sup>:

- <ftp://labs-automacao.fe.up.pt/1005> (pode ser acedido via *browser*), para o qual não necessita nem de *login* nem de *password*.

Refira-se que **as leituras indicadas devem ser encaradas como necessárias**, pois para além de permitirem conhecer os equipamentos, irão também ser alvo de questões durante os exames.

## HARDWARE

Numa primeira parte deve aprofundar o seu conhecimento sobre o *hardware* do autómato que está a utilizar. Para tal é necessário ler alguns capítulos dos manuais indicados nas subsecções seguintes.

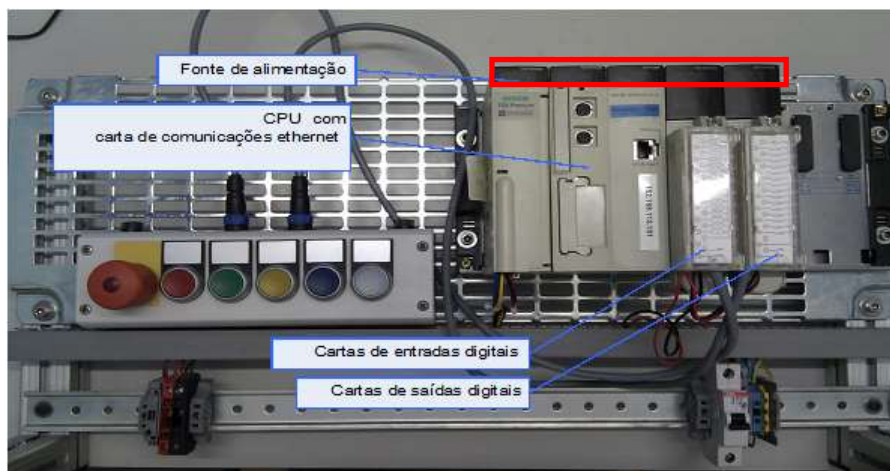
**Leia apenas a secção que diz respeito ao autómato que está a utilizar na sua bancada.**

---

<sup>1</sup> O acesso ao site <ftp://labs-automacao.fe.up.pt> necessita de uma ligação VPN se for realizado fora da FeupNet.

## DOCUMENTAÇÃO - AUTÓMATO TSX PREMIUM

Esta secção deve ser lida apenas pelos grupos cujo autómato seja do tipo **Premium**. A figura seguinte apresenta um exemplo típico da configuração de hardware deste autómato.



Comece por identificar quais os elementos constituintes do autómato que está a utilizar (i.e. quais as respetivas cartas). Cada carta possui um identificador (TSX...) impresso na parte frontal superior da mesma (retângulo vermelho da figura). A informação detalhada de cada carta pode ser encontrada nos seguintes manuais:

- Bastidores (rack), processadores e fontes de alimentação. Manual [Premium and Atrium using Unity Pro, Processors, Racks and Power Supply Modules](#). Ler os seguintes capítulos:
  - Características gerais: capítulo 2
  - Processador: capítulos 6, <capítulo referente ao processador que está instalado>, 26 e 27.
  - Fonte de alimentação: capítulos 34 e 35.
  - Rack: capítulo 49
- Cartas digitais. Manual [Premium and Atrium using Unity Pro, Discrete I/O modules](#). Ler os capítulos: 1, 2, <capítulos referentes às cartas que estão instaladas no autómato> e 34.

## DOCUMENTAÇÃO - AUTÓMATO M340

Esta secção deve ser lida apenas pelos grupos cujo autómato seja do tipo **M340**. A figura seguinte apresenta um exemplo típico da configuração de hardware deste autómato.



Comece por identificar quais os elementos constituintes do autómato que está a utilizar (i.e. quais as respetivas cartas). Cada carta possui um identificador (P...) impresso na parte frontal superior da mesma (retângulo vermelho da figura). A informação detalhada de cada carta pode ser encontrada nos seguintes manuais:

- Bastidores (rack), processadores e fontes de alimentação. Manual M340 - Processors, Racks and Power Supply Modules, Unity Pro. Ler os seguintes capítulos:
  - Características gerais: capítulo **2**
  - Processador: capítulos **5**, <capítulo referente ao processador que está instalado> e **9**.
  - Fonte de alimentação: capítulos **10** e **11**.
  - Rack: capítulos **15**
- Cartas digitais. Manual M340 - Discrete Input/Output Modules, Unity Pro. Ler os capítulos: **1**, **2**, <capítulos referentes às cartas que estão instaladas no autómato> e **26**.

## SOFTWARE - UNITYPRO

Numa segunda parte deve aprofundar o seu conhecimento sobre o *software* de configuração e programação do autómato, que neste caso é o **UnityPro**. Este estudo deve ser realizado com bastante cuidado, pois a maioria das tarefas do trabalho estão ligadas à programação do autómato.

No repositório de software é disponibilizada uma versão [DEMO do UnityPro](#), totalmente funcional, que pode ser utilizada durante 30 dias. O UnityPro executa num ambiente do tipo WinXX [xx=7..10]. Para ambientes de base de outro tipo (ex. MacOS) recomenda-se a sua instalação numa máquina virtual.

Para garantir que pode utilizar este software durante todo o semestre deve gerir a respetiva instalação de uma das seguintes formas:

- Articular com o colega de grupo uma 2ª instalação que será realizada num outro computador após a licença da 1ª instalação ter expirado.
- Instalar o SW numa máquina virtual que terá que ser reinstalada após a licença da 1ª instalação do ter expirado.

Este *software* tem uma funcionalidade importante que permite implementar e testar uma aplicação sem necessidade de estar ligado fisicamente ao autómato, utilizando para isso um emulador do autómato. Esta funcionalidade deve ser utilizada para preparar e testar os programas desenvolvidos fora das aulas.

## UNITYPRO - DOCUMENTAÇÃO

A documentação está organizada de forma a existirem vários manuais que abordam temas complementares, existindo também algumas sobreposições de conteúdos. Por forma a facilitar esta tarefa selecionaram-se os manuais mais importantes (que também estão disponível no **Help do UnityPro**):

- O [Unity Pro Operating Modes](#) é um manual que descreve essencialmente a ferramenta UnityPro. Dever ser utilizado quando se tem dúvidas em utilizar a ferramenta de programação. Sugere-se a leitura dos capítulos: **1**, **2**, **8** e **9**, por forma a facilitar a tarefa de programação.
- O [Unity Pro Languages](#) and [Program Structure Reference Manual](#) descreve a arquitetura de programação do autómato e as respetivas linguagens de programação. Este manual é muito extenso e deve ser lido de forma diferenciada e em alturas diferentes do trabalho:
  - Inicialmente devem ser lidos os capítulos: **1** a **4**, e **7** a **10**.
  - Posteriormente, e em função das linguagens de programação utilizadas, devem ser lidos os capítulos da **Parte IV**.

- Numa fase mais adiantada do trabalho deve ser lidos os capítulos da **Parte V**.
- Por fim devem ser lidos os capítulos **5** e **6**.
- O manual [Standard Block Library](#) contém uma descrição detalhada de todos os **Function Block** (FB) mais importantes, nomeadamente Contadores, Temporizadores, Detecção de Flancos, funções Aritméticas e Lógicas, bem como a sua utilização nas várias linguagens.

## TUTORIAL

Inicie o seu contacto com o UnityPro visualizando o seguinte [tutorial](#).

- Tutoriais na Web (Youtube: search UnityPro)

## CONFIGURAÇÃO DO AUTÔMATO

Nesta seção é apresentado um procedimento simples que permite realizar uma configuração mínima do hardware do autômato. Mais tarde pode alterar esta configuração por forma a incluir outras opções.

Proceda da seguinte forma:

1. Identifique qual o tipo de autômato presente na sua bancada (Premium ou M340).
2. Identifique qual a versão do CPU do autômato. Esta informação está impressa na parte frontal superior da carta do CPU (ver figura seguinte).

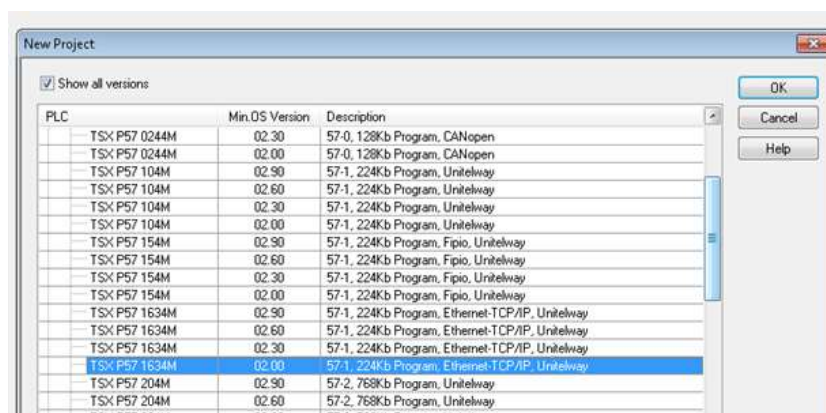


Premium



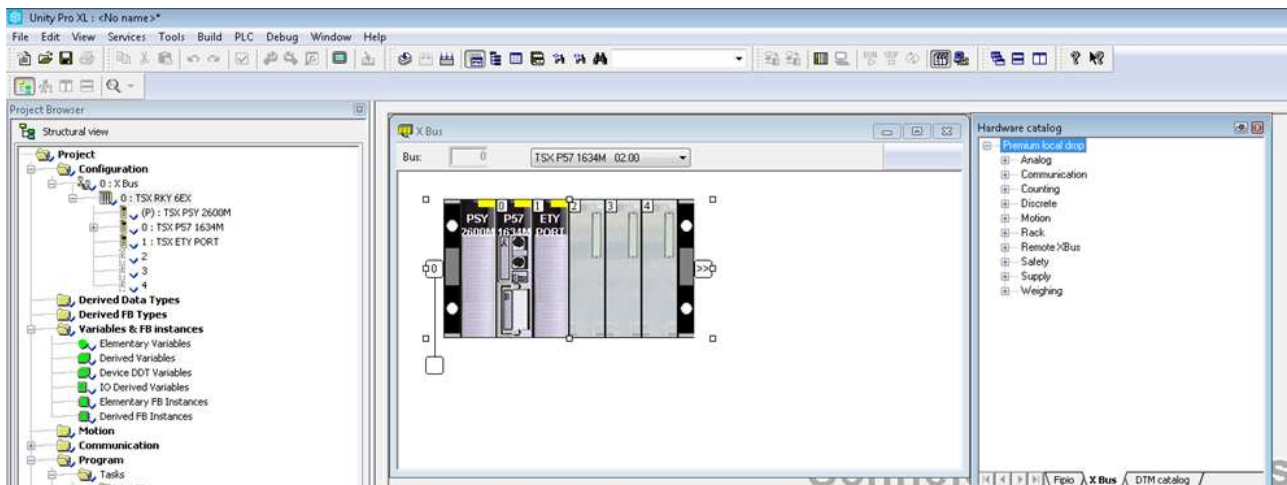
M340

3. Arranque com o UnityPro e arranque com um novo projeto.
4. Na janela *New Project* ative a opção **Show All Versions** e selecione na opção **Min.OS Version** a versão do sistema operativo mais antiga do autômato (o número mais baixo).



5. Crie o respetivo projeto.
6. Selecione a janela do *Project Browser* e abra sucessivamente a árvore **Configuration**→**X Bus**→**TSX RK**. Nesta última pasta faça *double click* e será aberta a janela **X Bus**. Esta janela permite fazer a configuração do restante hardware do autômato.
7. Identifique as cartas de E/S do seu autômato. O identificador de cada carta está impresso na parte frontal e superior da carta (tal como no CPU). Cada autômato tem tipicamente uma carta de entradas digitais e outra de saídas digitais.
8. Adicione as cartas de E/S ao seu projeto. Para isso selecione o menu **Tools**→**Hardware Catalog**, que abrirá o painel do Catálogo de Hardware (*Hardware Catalog*). Escolha o submenu **Discrete** e selecione a carta pretendida. Com o rato pressionado sobre a referência da carta, arraste esta referência até ao slot respetivo do autômato na janela **Xbus**. Uma outra alternativa é selecionar o slot do autômato e com o botão diereito do rato (**New Device**) introduzir a respetiva carta.

9. O passo seguinte consiste em configurar a rede *Ethernet* do autómato. Os autómatos têm uma interface Ethernet instalada no CPU ou numa carta em separado, dependendo do modelo. Identifique qual a sua configuração. Se se tratar de uma carta em separado terá que a adicionar ao hardware do autómato tal como fez para as cartas de E/S. A carta está no submenu *Communications* do Catálogo de Hardware.

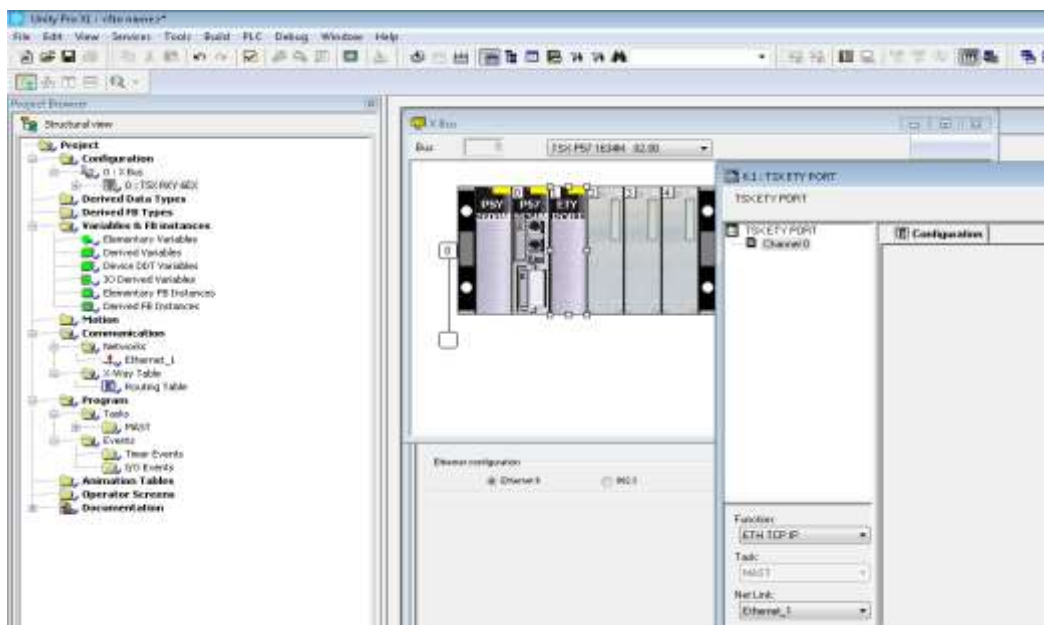


10. Selecione a janela do *Project Browser* e abra a pasta **Communication**. Selecione a entrada *Networks*. Com o botão direito do rato, selecione *New Network* e crie uma rede do tipo *Ethernet* e atribua-lhe o nome **Rede**. Faça agora *double click* sobre esta nova rede que acabou de criar para que a possa configurar.



11. Na opção *Model Family* selecione o local onde está instalada a interface Ethernet. Há duas opções: no CPU ou numa carta em separado. Faça a escolha em função das características do seu autómato (tenha em atenção às versões da carta – se tiver dúvidas contacte o docente).
12. Configure o endereço IP da carta (*IP address*) com o valor que se encontra escrito na parte frontal do autómato (é um número do tipo **192.168.113.xxx**, em que **xxx** é um valor diferente para cada autómato).
13. Configure a *subnet mask* e a *default gateway* com os seguintes valores: **255.255.255.0** e **192.168.113.254**, respetivamente.
14. Selecione a janela do *Project Browser* e abra *Configuration* → *X Bus*. Faça *double-click* sobre a carta *Ethernet* (ou selecione a interface ethernet no caso desta estar no CPU). Selecione o canal (*channel*), a função (*Function: ETH TCP IP*) e em **Net Link** escolha a rede que acabou de configurar.





## TESTE DA CONFIGURAÇÃO

O autômato está agora configurado e pronto a ser programado. Para descarregar o programa para o autômato possui duas alternativas configuráveis no menu **PLC**→**Set Address**:

1. Utilizando uma ligação série. Neste caso o computador e o autômato têm de estar ligados por um cabo série (RS-232 ou USB). Configure a ligação da seguinte forma:

- **Address** = SYS, **Media** = UNTLW

Esta é uma solução lenta a descarregar o programa, mas é garantido que funciona em todas as situações.

2. Utilizando a rede *Ethernet*. Neste caso o computador e o autômato têm de estar ligados à rede Ethernet do laboratório. Configure a ligação da seguinte forma:

- **Address** = Endereço IP do autômato, **Media** = TCPIP

Esta é uma solução rápida a descarregar o programa. Contudo existem algumas situações que não funcionam bem (alguns modos de *debug* das cartas). Nestes casos deve utilizar a solução anterior.

Descarregue a configuração para o autômato. Para isso selecione o menu **PLC** e verifique que tem a opção *Standard Mode* ativada. Conecte-se ao autômato (**PLC**→**Connect**) e descarregue a configuração para o autômato (**PLC**→**Transfer Project to PLC**).



## AMBIENTE DE EXECUÇÃO E TESTE DAS APLICAÇÕES

Nesta seção são descritos os procedimentos para realizar a execução e o teste das aplicações desenvolvidas. Foi desenvolvida uma solução flexível que permite que estas tarefas possam ser realizadas quer laboratório, quer fora do laboratório (i.e. em casa) sem recorrer aos equipamentos físicos (autómatos e kit da fábrica) existentes.

### EXECUÇÃO DAS APLICAÇÕES

As aplicações podem ser executadas quer no autómato, quer no emulador (simulador) do autómato disponível no UnityPro. O emulador do autómato permite executar as aplicações tal como se as executasse no autómato real (i.e. comportam-se de forma igual).

Para utilizar o modo de execução no autómato selecione no menu PLC a opção **Standard Mode**.

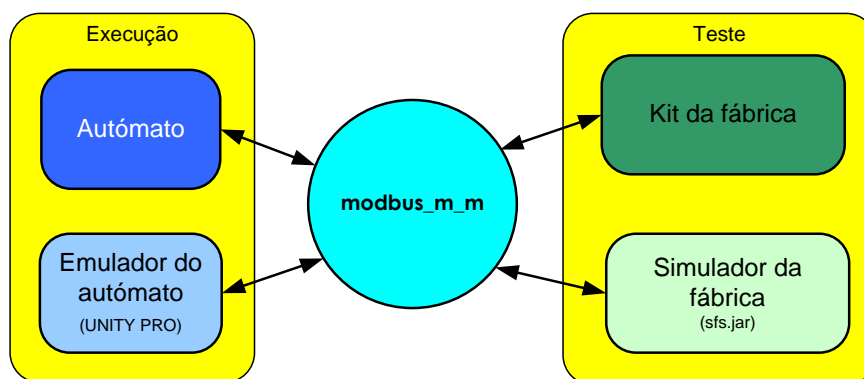
Para utilizar o modo de simulação selecione no menu PLC a opção **Simulation Mode**. Há, no entanto, uma restrição importante quando se utiliza este modo: não é possível ter acesso às E/S ligadas às cartas digitais. Pode, no entanto, solucionar esta questão ativando ou analisando os respetivos endereços dessas cartas numa **Tabela de Animação (Animation Table)**.

Em quaisquer dos modos a aplicação só pode ser executada depois de ser transferida para o autómato ou para o simulador.

### TESTE DAS APLICAÇÕES

As aplicações podem ser testadas no kit da fábrica (i.e. na fábrica real) ou utilizando o respetivo simulador (sfs.jar). Esta última opção é a mais desejável pois permite realizar o trabalho fora do laboratório.

Para interligar o ambiente onde as aplicações são executadas (autómato ou respetivo emulador) com o ambiente onde são testadas (kit da fábrica ou respetivo simulador) é necessário utilizar uma aplicação intermédia, denominada **modbus\_m\_m**, tal como é apresentado na figura seguinte:



Através desta aplicação é possível dispor de várias combinações diferentes de teste e execução sem ter que realizar qualquer alteração no programa implementado no autómato (nem trocar cabos), como por exemplo:

- Emulador do autómato ↔ Simulador da fábrica: ideal para testes fora do laboratório.
- Autómato ↔ Kit da fábrica: ideal para testes 'reais'.
- Autómato ↔ Simulador da fábrica: quando o kit não está disponível (ex. ocupado por outro grupo).
- Emulador do autómato ↔ Kit da fábrica: quando o autómato não está disponível.

## REALIZAR A INTERLIGAÇÃO

Dependendo do ambiente onde pretende executar as aplicações (autómato ou emulador) verifique se o autómato está ligado ou se o emulador do autómato está a executar (pequeno bloco verde na barra inferior do Windows). Coloque o autómato ou o emulador em modo RUN.

Se pretender utilizar o simulador da fábrica (**sfs.jar**), garanta que este está a ser executado.

Abra a pasta onde a aplicação **modbus\_m\_m** está colocada. Nessa janela pressione simultaneamente a tecla **SHIFT** e o botão do lado direito do rato. No menu que foi aberto selecione a opção '*Open command window here*'.

A aplicação **modbus\_m\_m** tem 3 parâmetros, com a seguinte ordem: **<IP> <Célula> <Modo>**, em que

- o parâmetro **<IP>** deve ser substituído pelo **endereço IP** do autómato ou do emulador.
- o parâmetro **<Célula>** deve ser substituído pelo identificador da célula do seu trabalho: **C1** ou **C2**.
- o parâmetro **<Modo>** indica se a interligação deve ser feita com o kit da fábrica ou com o simulador. No primeiro caso o parâmetro deve ser substituído pelo identificador **kit** e no segundo caso por **sim**.

Exemplos:

- Utilizar no simulador da fábrica a célula C2, controlada pela aplicação a executar no emulador do autómato:

**modbus\_m\_m 127.0.0.1 C2 sim**

- Utilizar no kit da fábrica a célula C1, controlada pela aplicação a executar no autómato real:

**modbus\_m\_m 192.168.113.24 C1 kit**

Nas secções seguintes são descritas as configurações que deve realizar ao nível dos equipamentos para que possa realizar esta interligação.

---

## UTILIZAR O AUTÓMATO PARA EXECUTAR AS APLICAÇÕES

Neste modo de execução basta que a carta *ethernet* do autómato esteja configurada com o endereço IP correto.

Para realizar a ligação com o **modbus\_m\_m** o parâmetro **endereço\_IP** deve ser substituído pelo endereço IP que utilizou na configuração da carta *ethernet* do autómato.

---

## UTILIZAR O EMULADOR DO AUTÓMATO PARA EXECUTAR AS APLICAÇÕES

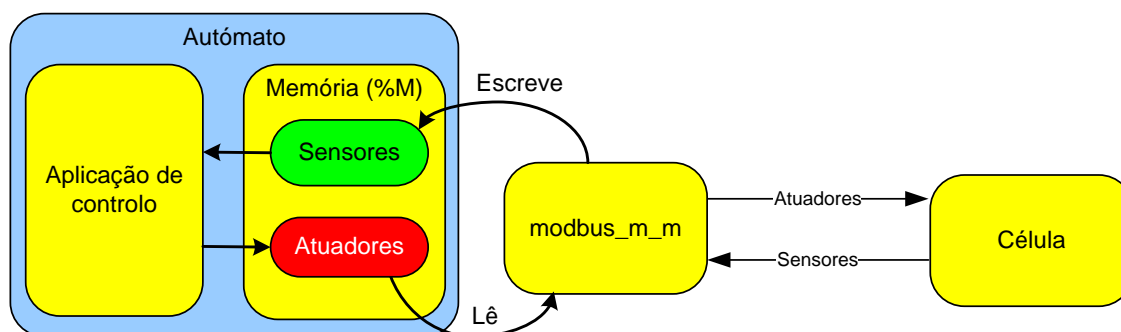
Para utilizar o emulador do autómato selecione no menu *PLC* a opção *Simulation Mode*. Para ligar o emulador ao autómato não necessita de configurar a carta de rede *ethernet*. No entanto pode (e deve) fazê-lo. Assim quando utilizar o seu programa no laboratório não necessita de voltar a configurar a rede.

Para realizar a ligação com o **modbus\_m\_m** o parâmetro **endereço\_IP** deve ser substituído pelo endereço IP do computador onde está a executar o UnityPro ou pelo endereço **127.0.0.1** (são equivalentes).

## MAPEAMENTO DOS SENSORES E ATUADORES

Por forma a utilizar o kit da fábrica ou o respetivo simulador de forma transparente, o **modbus\_m\_m** foi desenvolvido da seguinte forma:

- Escreve ciclicamente numa zona de memória do autómato os valores associados aos sensores físicos do kit da fábrica (ou do simulador).
- Lê ciclicamente os valores associados aos atuadores físicos de uma zona de memória do autómato e envia-os para o kit da fábrica (ou para o simulador).



Desta forma, a aplicação desenvolvida no autómato tem apenas que ler uma zona de memória específica para ter acesso aos valores dos sensores ou escrever numa zona de memória específica para atuar nos atuadores. Não é assim necessário realizar qualquer alteração na aplicação que irá desenvolver sempre que alterna entre o kit e o simulador (ou entre o autómato e o emulador).

Nos conteúdos da disciplina são disponibilizados 2 ficheiros:

- [lista de entradas](#)
- [lista de saídas](#)

que indicam as posições de memória (variáveis) atribuídas aos sensores e aos atuadores. Não é necessário declarar todas as variáveis. É **obrigatório** utilizar estas posições de memória **sem qualquer modificação** caso contrário o seu programa não irá funcionar. Declare apenas as variáveis que estão atribuídas à sua célula (estão assinaladas com a cor respetiva). O nome das variáveis pode ser escolhido livremente.

Estas variáveis devem ser declaradas na zona de memória do tipo *memory bits (%M)* e são do tipo **EBOOL**. Este tipo é específico do UnityPro que é equivalente a um **Bool** (no IEC61131-3). Internamente correspondem a 3 bits: um para guardar o valor corrente da variável, outro para guardar o valor da variável no ciclo anterior e um outro para indicar se está ou não forçada. Este tipo de dados permite detetar flancos recorrendo às funções tradicionais **RE** e **FE** (i.e. sem usar Function Blocks).

## MAPEAMENTO DO CÓDIGO DE BARRAS

O código de barras é lido também pelo **modbus\_m\_m**, sendo tratado como uma entrada que está mapeada na posição de memória **%MWO** (ver ficheiro *lista\_de\_entradas*). O valor do código de barras é um número inteiro sem sinal (**UINT16**).

Para simular a leitura do código de barras fora do laboratório pode-se utilizar as seguintes alternativas:

- Escrever diretamente o valor do código na posição de memória **%MWO**
- Utilizar uma aplicação que simula um leitor de código de barras e que envia o respetivo valor para o **modbus\_m\_m**. Na seção seguinte é apresentada uma aplicação deste tipo e respetiva configuração.

## CONFIGURAÇÃO DO WIRELESS BARCODE-SCANNER FOR ANDROID

O leitor de código de barras pode ser simulado por uma aplicação Android denominada **Wireless Barcode-Scanner for Android** (da TEC-IT). A aplicação pode ser utilizada em qualquer telemóvel Android que disponha de uma camera. Depois de instalada (descarregar na PlayStore), transforma um telemóvel num leitor de código de barras.

Este leitor permite ler códigos de barras (1D e 2D) de vários tipos e enviar o respetivo valor através de uma rede (WiFi ou Bluetooth) para um servidor remoto.

Antes de avançar, é importante ressaltar que a aplicação gera periodicamente e de forma aleatória alguns códigos de barras “fora do contexto” (trata-se de um demo....). Estes códigos podem ser facilmente ignorados pois não cumprem o padrão esperado.

Por fim note que para receber códigos de barras do telemóvel é necessário que o telemóvel tenha uma ligação WiFi que permita aceder à rede onde está o PC que executa o **modbus\_m\_m**.

Após descarregar a aplicação é necessário realizar uma curta configuração. Proceda da seguinte forma:

1. Inicie a aplicação.
2. Selecione **Transfer Data** para indicar que pretende que o código seja enviado via rede.
3. Selecione a conexão **TCP**
4. Selecione a opção **Client**
5. Configure o **IP Address** com o endereço IP do computador onde o **modbus\_m\_m** está a executar. Pode obter este endereço nas conexões de rede do Windows.
6. Configure o **TCP Port** com o valor **5600**
7. Configure o **Suffix** com : \n
8. Continue e ative a opção que permite guardar a configuração localmente (para não estar sempre a configurar o leitor).
9. A aplicação está pronta a ler códigos de barras e a envia-los para um servidor remoto. Os códigos podem também ser inseridos diretamente a partir de um teclado disponibilizado pela aplicação.

Existem muitas aplicações que permitem gerar códigos de barras. Uma forma simples de realizar esta tarefa é utilizar a aplicação on-line do site da TEC-IT, acessível no seguinte [link](#):

- No lado direito do ecrã, selecione o código que pretende (neste projeto utilize Code-39, mas pode experimentar com outros).
- Na parte central do ecrã existe uma janela (Data) onde pode escrever o valor a codificar.
- No lado direito, irá surgir o código de barras respetivo.
- Pode descarregar o código, ou ler diretamente do ecrã (basta apontar o telemóvel)

## SIMULAÇÃO DA BOTONEIRA

Na sua bancada de trabalho existe uma botoneira com indicadores luminosos e botões que estão ligadas às cartas de E/S do autómato. As respetivas ligações estão descritas no seguinte [documento](#).

Contudo, como este dispositivo está fisicamente ligado ao autómato não é possível utilizar este equipamento fora do laboratório. Para resolver este problema há duas opções possíveis:

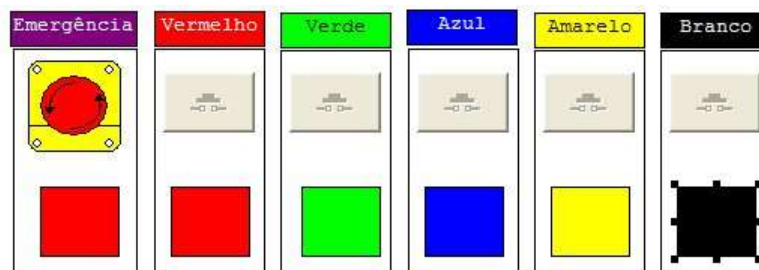
- Utilizar uma **Tabela de Animação (Animation Table)** no UnityPro para observar os valores das variáveis associadas aos indicadores ou forçar as variáveis associadas aos botões.

- Utilizar um *Operator Screen* que simula estes elementos. Esta é opção mais desejável pois permite ter acesso aos indicadores e botões de forma intuitiva. Assim, o restante desta secção é destinado a explicar esta opção.

O *Operator Screens* é uma ferramenta disponibilizada no UnityPro que permite criar interfaces homem-máquina muito simples e que podem interagir com o programa que executa no autómato (ou no emulador). Neste contexto foi criado um *Operator Screen*, disponibilizado nos conteúdos da disciplina, para simular a interface local com o operador.

Para o utilizar este *Operator Screen* este proceda da seguinte forma:

1. Descarregue o ficheiro [PAINEL.XCR](#) dos conteúdos da disciplina para uma pasta à sua escolha.
2. Dentro do UnityPro, procure na árvore de configuração do projeto (*Project Browser*) a sub-pasta *Operator Screens*.
3. Coloque o rato nesta sub-pasta e com o botão do lado direito escolha a opção *Import*. Escolha o ficheiro anterior para importar.
4. Faça *double-click* sobre o *screen* que foi importado. Vai abrir uma janela com o seguinte aspeto:



Neste momento já dispõe da interface, faltando apenas associar os objetos gráficos com as variáveis que está a usar no seu programa. Neste ecrã só há dois tipos de objetos: botões e luzes. Assim, vamos descrever como se realiza a associação com cada um deles, sendo a descrição igual para os restantes.

Nota: antes de alterar o painel verifique se este se encontra já configurado. Neste caso não será necessário realizar nenhum procedimento.

**Exemplo:** associar a variável do botão vermelho ao respetivo objeto gráfico

1. Vamos admitir que a variável que criou no seu programa para este botão chama-se **B\_vrm** (pode ser um nome qualquer).
2. O objeto gráfico é um botão cujo centro contém o texto Vermelho. Faça *double-click* sobre este objeto.
3. Na janela de propriedades do botão escolha a *tab Control*. Este *tab* vai permitir definir qual a variável que é modificada quando se clica no botão. Na opção *Variable* e utilize o botão à direita (ou escreva) para selecionar a variável **B\_vrm**.
4. O botão vai funcionar da seguinte forma. Quando clicar no botão, a variável **B\_vrm** toma o valor **True**. Este valor mantém-se enquanto estiver a pressionar o botão. Quando liberta o botão a variável toma o valor **False**.

**Exemplo:** associar a variável do indicar vermelho ao respetivo objeto gráfico

1. Vamos admitir que a variável que criou no seu programa para este indicador chama-se **L\_vrm** (pode ser outro nome qualquer).
2. O objeto gráfico é um círculo pintado de vermelho. Faça *double-click* sobre este objeto.

3. Na janela de propriedades do botão escolha a *tab Animation*. Esta *tab* vai permitir definir a forma como é feita a animação do círculo. Selecione a *check-box Animated Object*. Na opção *Variable* e utilize o botão à direita (ou escreva) para seleccionar a variável **L\_vrm**. Na opção *Display Condition* escolha a opção *Bit=1*.
4. O indicador vai funcionar da seguinte forma. Quando a variável **L\_vrm** tomar o valor TRUE o círculo vai aparecer. Quando tomar o valor FALSE o círculo não aparece.

Para aceder ao ecrã de controlo durante a execução do programa (no autómato) proceda da seguinte forma:

1. Nas opções do projecto *Tools*→*Options* seleccione o menu *Operator Screens*. Desactive todos os *check-box* na opção *Online Mode*. Isto vai inibir a edição do painel durante a sua utilização. Só é necessário realizar esta operação uma única vez (fica gravada no projeto).
2. Transfira a aplicação para o autómato.
3. Na árvore de configuração *Project*→*Operator Screens* faça *double-click* sobre o painel. Este vai abrir.
4. Para poder utilizar os botões é necessário ativar a opção *Enable Variable Modification*. Isto pode ser realizado pressionado o botão F7 quando o rato está sobre a janela do painel.

---

FIM