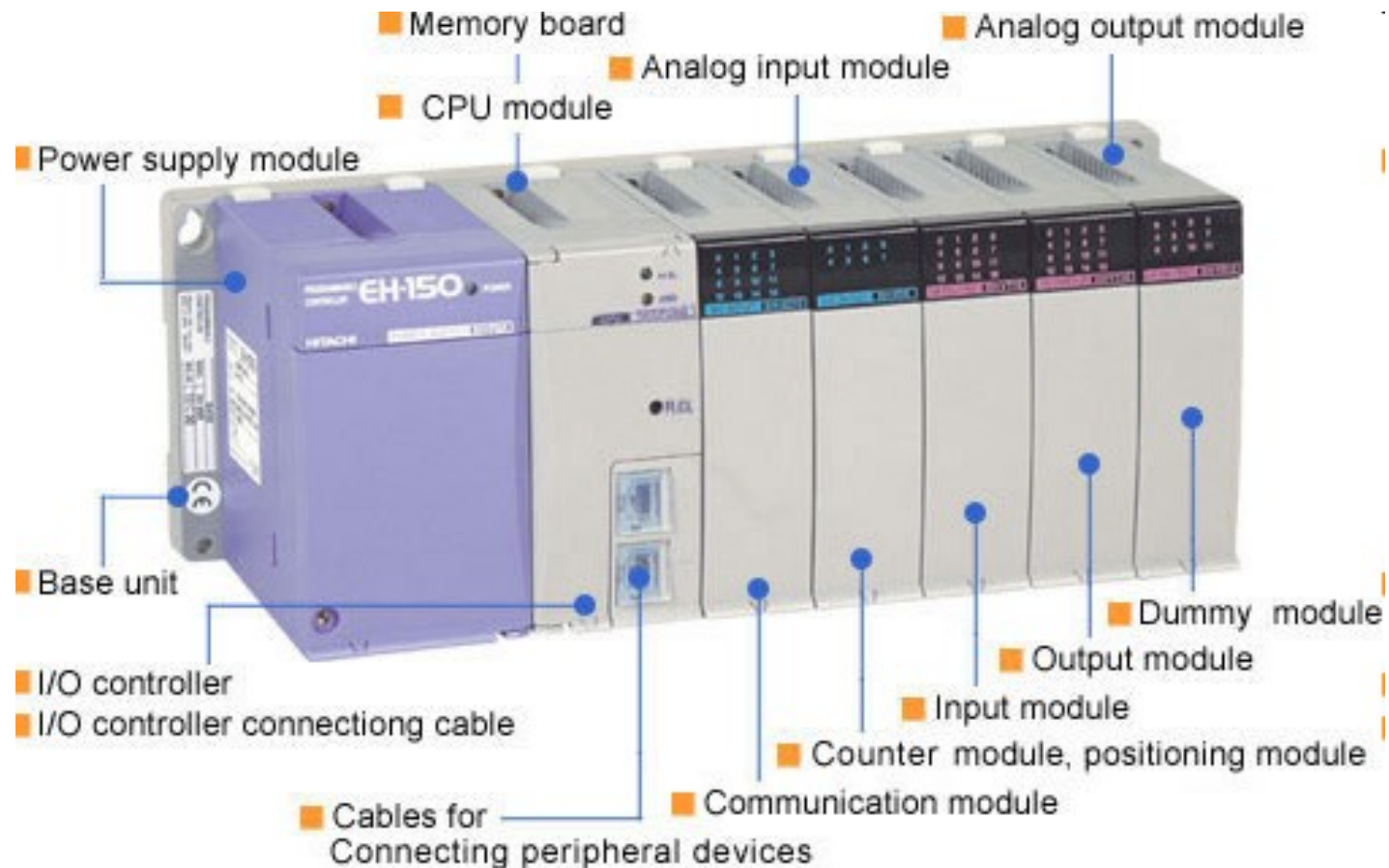


Autómatos Programáveis

Introdução

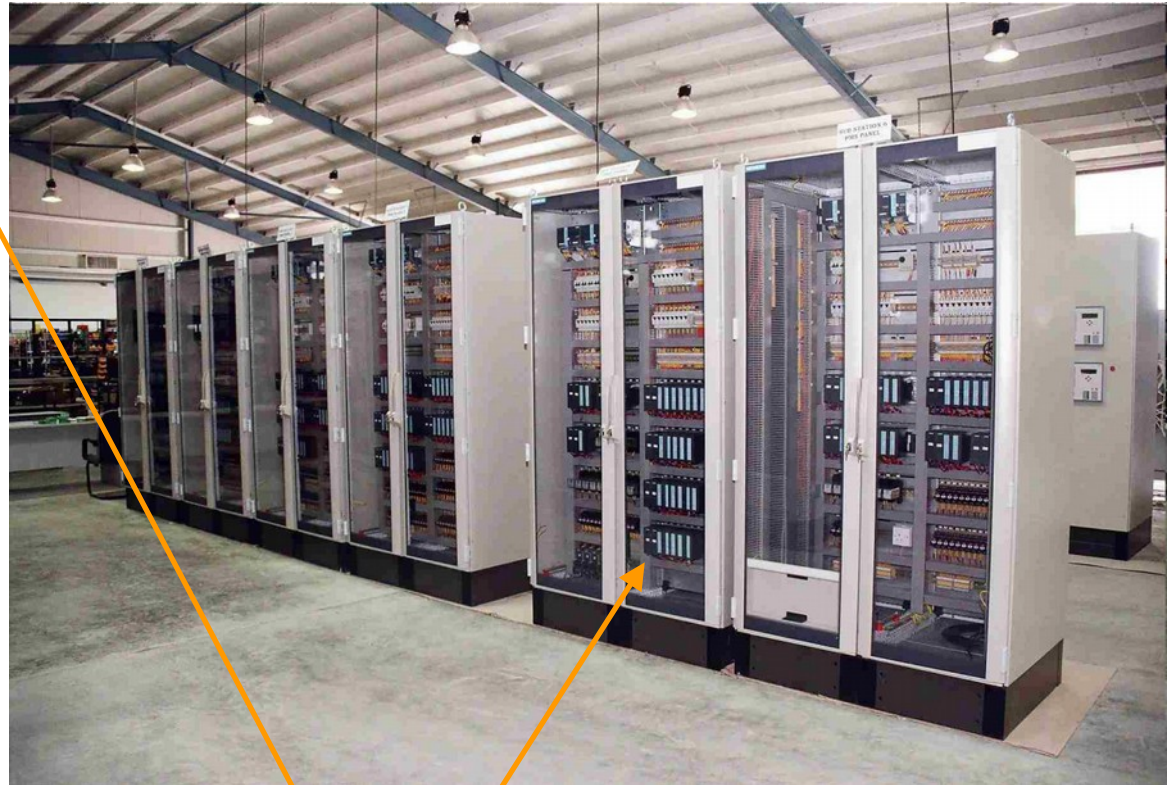
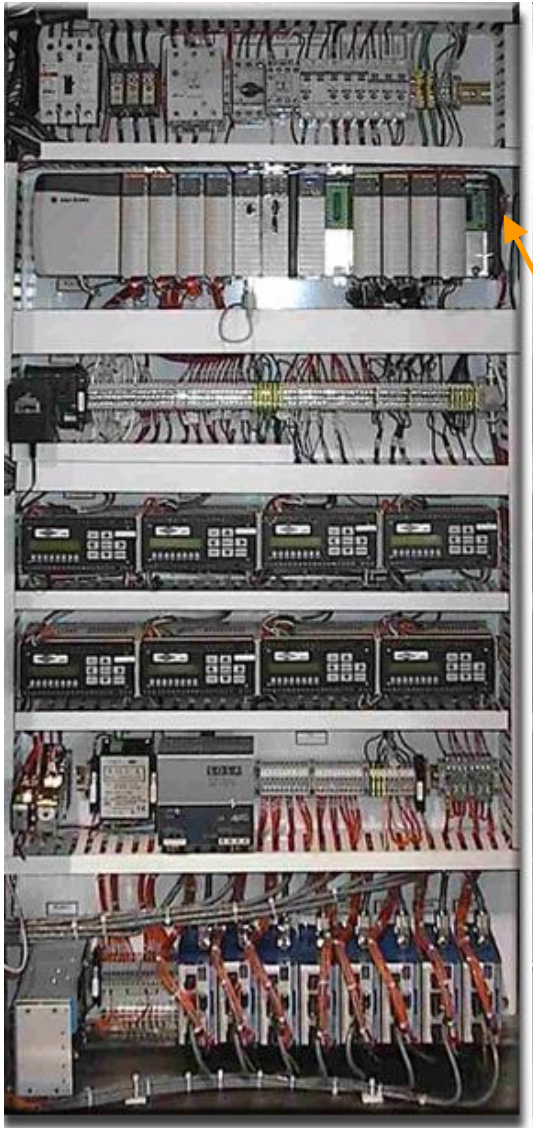
O que é um autómato programável ?



Algumas respostas...

- Sistema computacional (baseado num microprocessador), programável, e especialmente otimizado para tarefas de controlo (de processos industriais)
- Dispõe de interfaces com o processo (entradas/saídas) já incorporadas e normalizadas (ex. para sensores e actuadores)
- Têm uma construção robusta, adaptada às características dos ambientes industriais (temperatura, vibrações, humidade, poeiras, EMI, etc.)
- Muito fiável (típico ~10 anos sem avarias : 87,600h)
- Utilizam uma linguagem de programação bem adaptada às características das tarefas de controlo dos processos;
 - Funções principais: comando (processos discretos) e regulação (processos contínuos);
 - Funções secundárias: protecção, interface homem-máquina, comunicações com outros equipamentos
- Designação na literatura: PLC (Programmable Logic Controller)

Exemplo de instalações



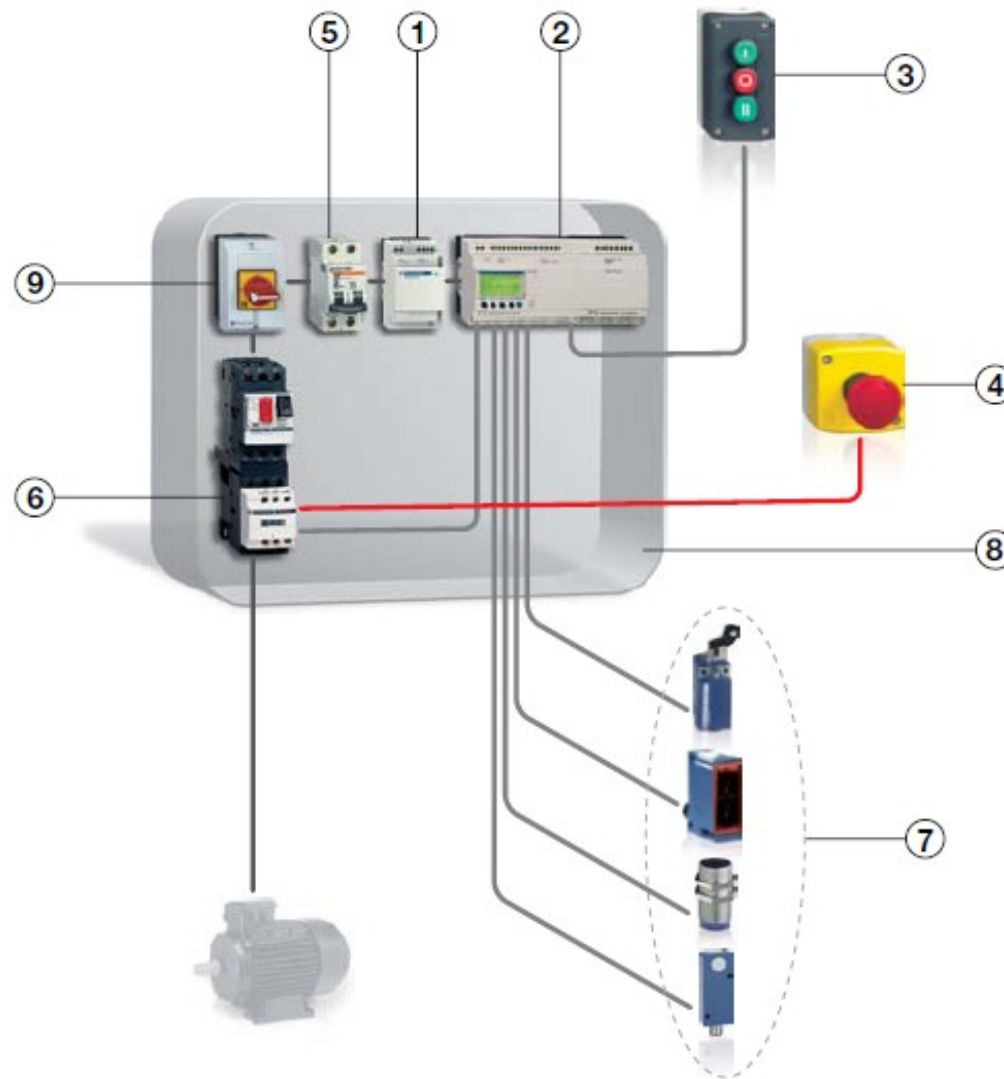
autómato

Autómatos programáveis compactos

- Construção monolítica
- Mono-processador
- Número fixo (e limitado) de entradas/saídas (na sua maioria discretas)
- Capacidade de processamento reduzida
- Capacidade de interligação com redes de campo de baixo custo (em alguns casos também com redes do tipo Ethernet)
- Capacidade limitada (ou nula) de interagir com uma interface homem-máquina
- Produtos típicos: Schneider Micro, Omron CQM1, Siemens S200
- Baixo custo: €250 - €2000



Autómato compacto : exemplo de aplicação

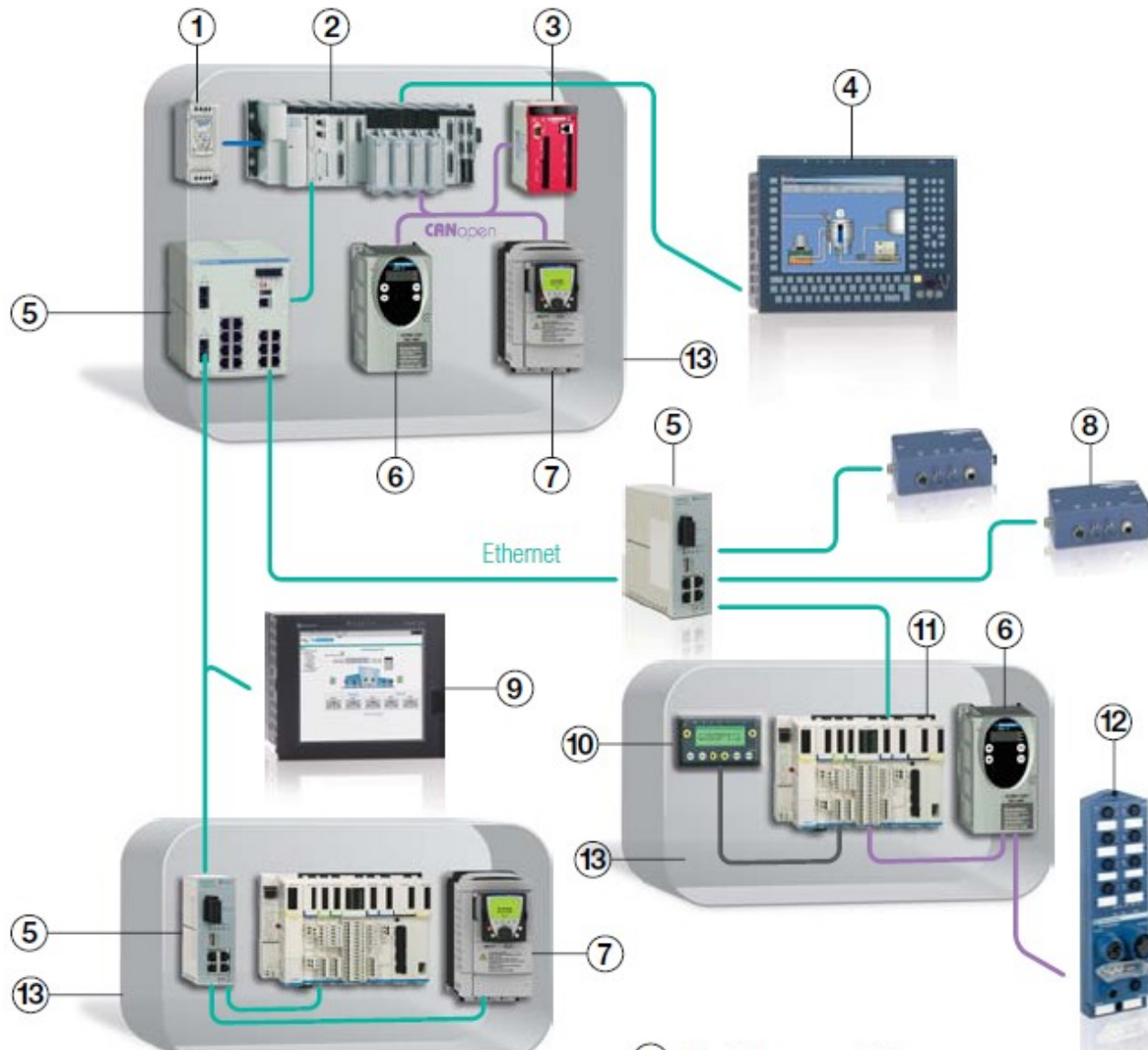


Autómatos programáveis modulares

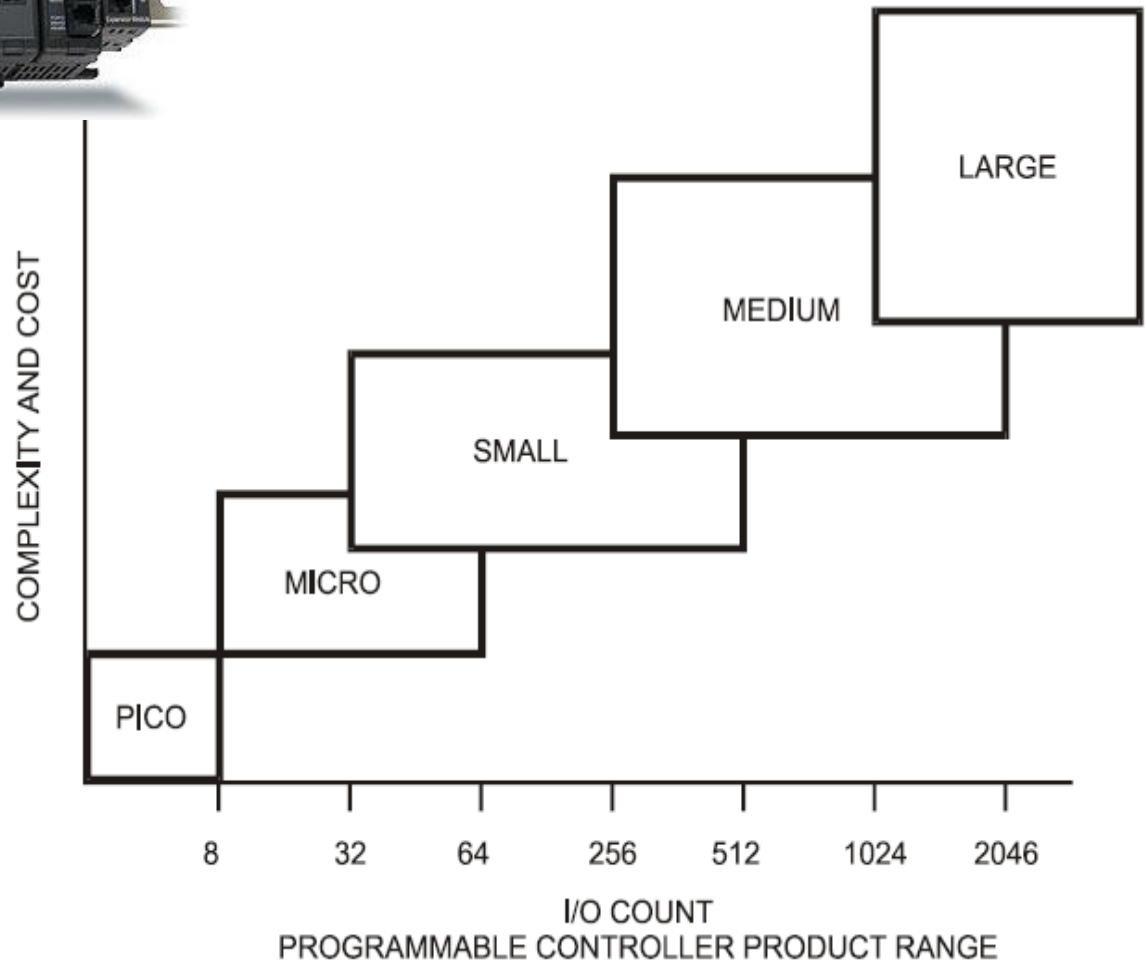
- Construção modular (bastidor)
- Mono ou multi-processador
- Capacidade de processamento elevada
- Grande variedade na escolha de cartas de E/S
- Interface com vários tipos de redes industriais
- Capacidade de interagir com interfaces homem-máquina ou sistemas de supervisão
- Produtos típicos: Scheneider Premium, Omron CS1, Siemens S300
- Médio / alto custo: €1000 - €10,000



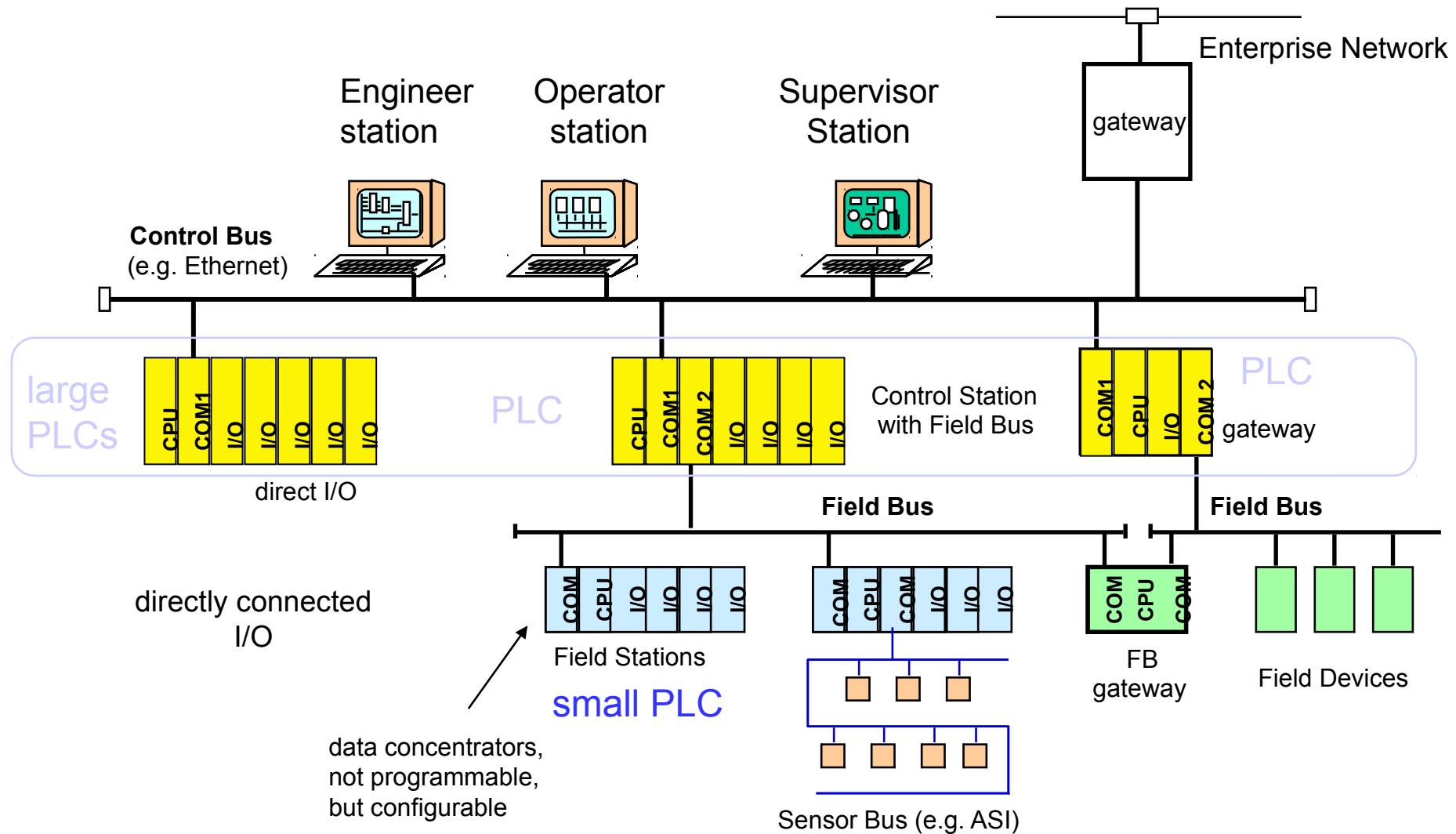
Autómato modular : exemplo de aplicação



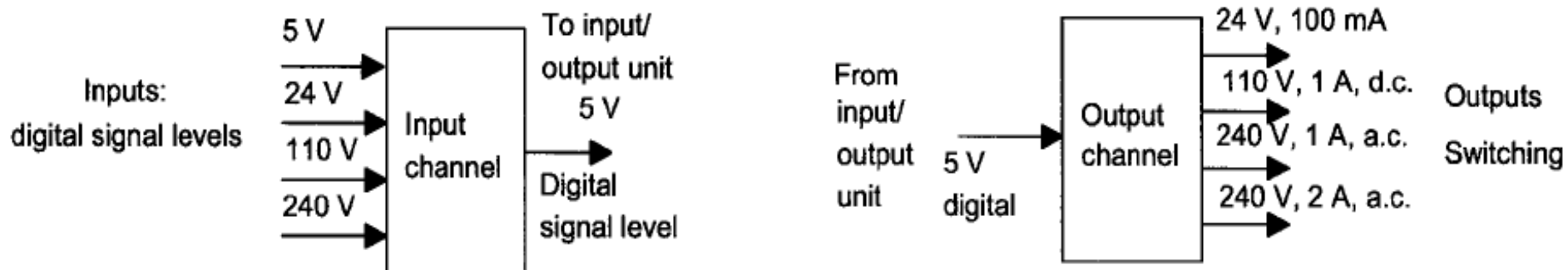
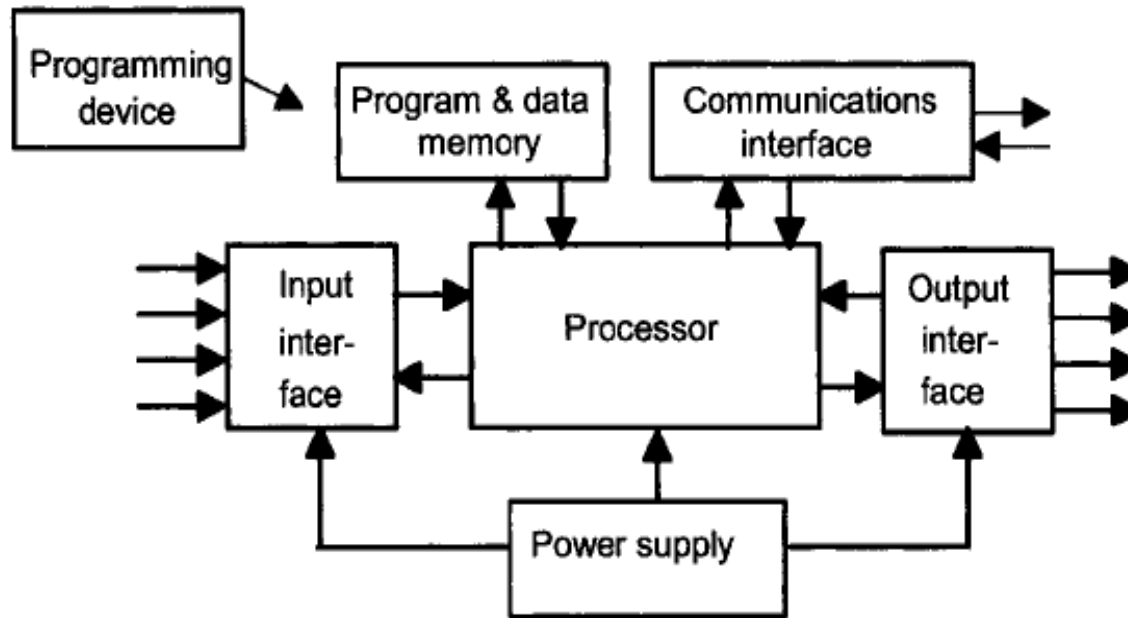
Gamas de aplicação



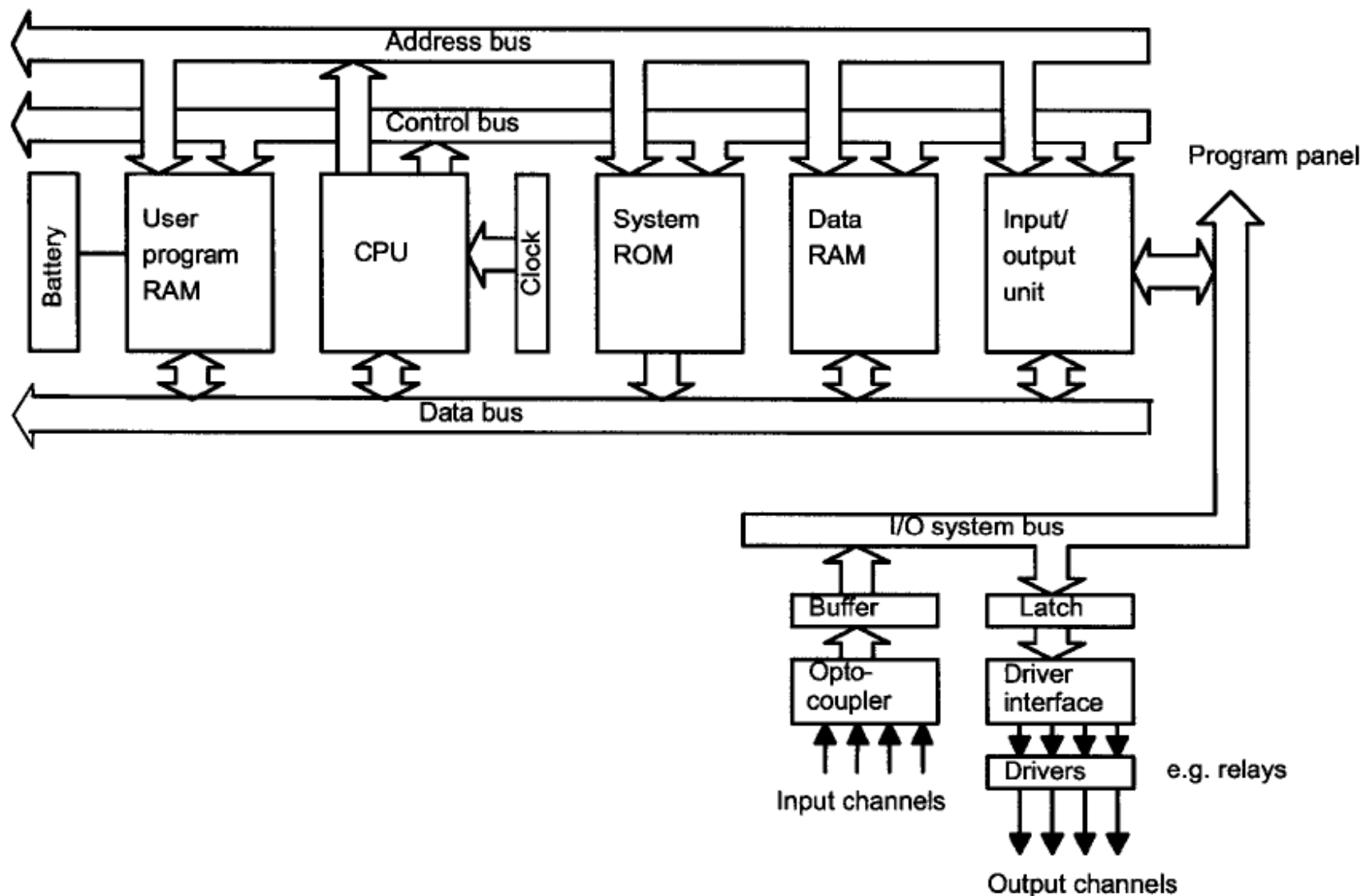
Os autómatos na hierarquia de controlo



Arquitectura geral

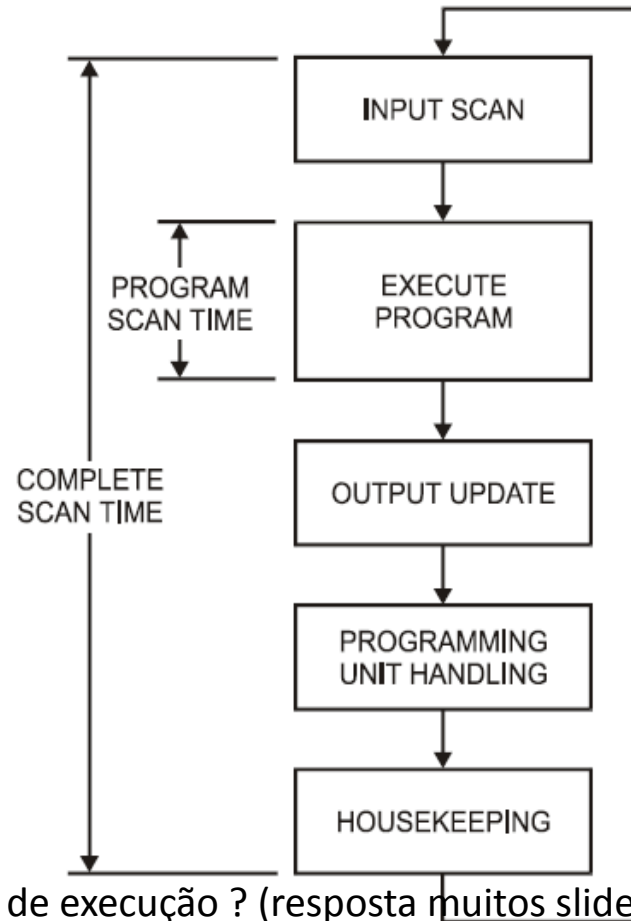


Arquitetura interna



Execução das aplicações

- Modelo de execução cíclico



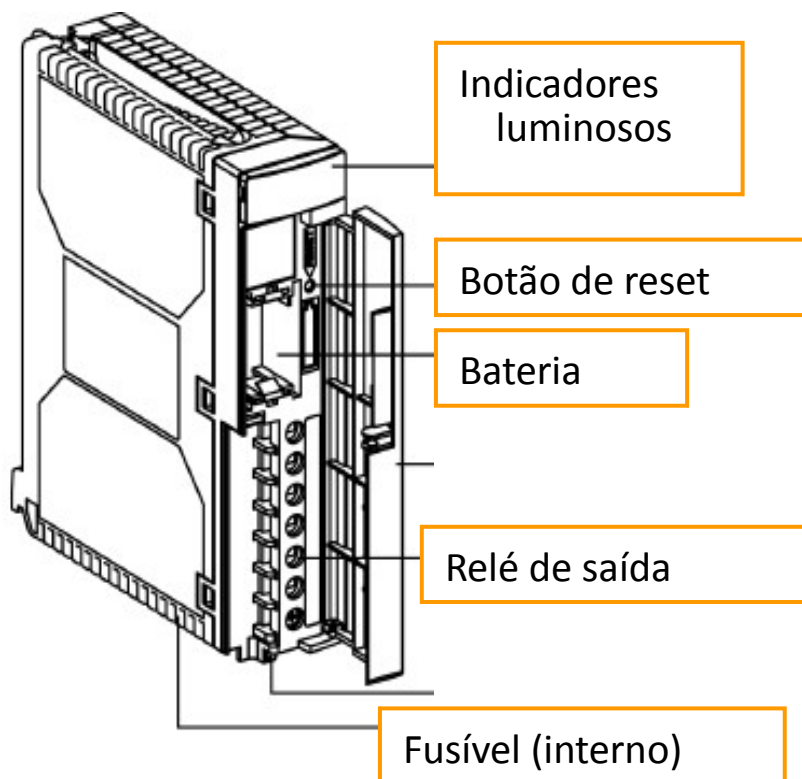
- Porquê adoptar este modelo de execução ? (resposta muitos slides à frente...)

Arquitectura de hardware

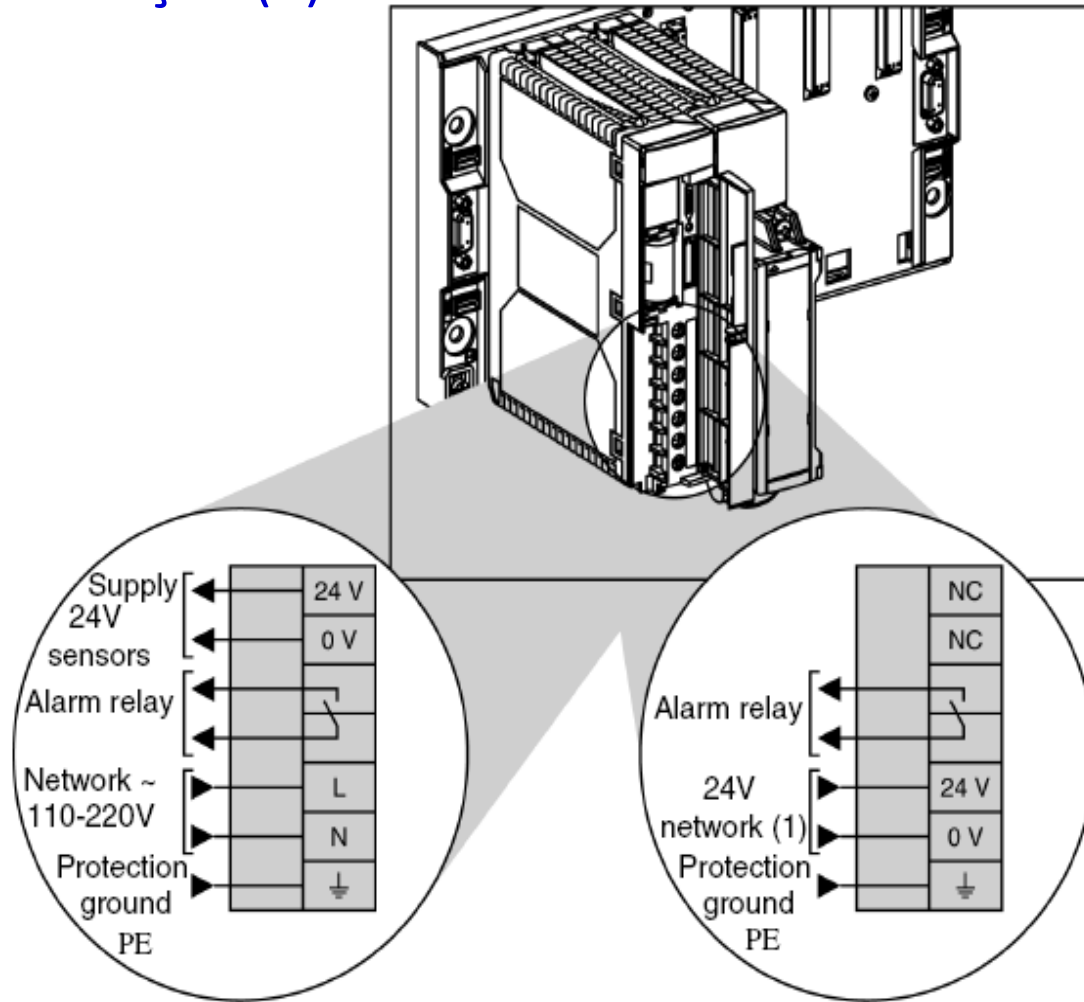
(com ênfase nas características dos autómatos utilizados nas aulas práticas)

Fonte de alimentação

- Alimentação dos vários módulos do autómato (através do bastidor). Tipicamente: +5V.
- Alimentação (externa) dos sensores e pré-actuadores. Tipicamente: +24V.
- Geralmente ocupam o 1º módulo (slot do bastidor)



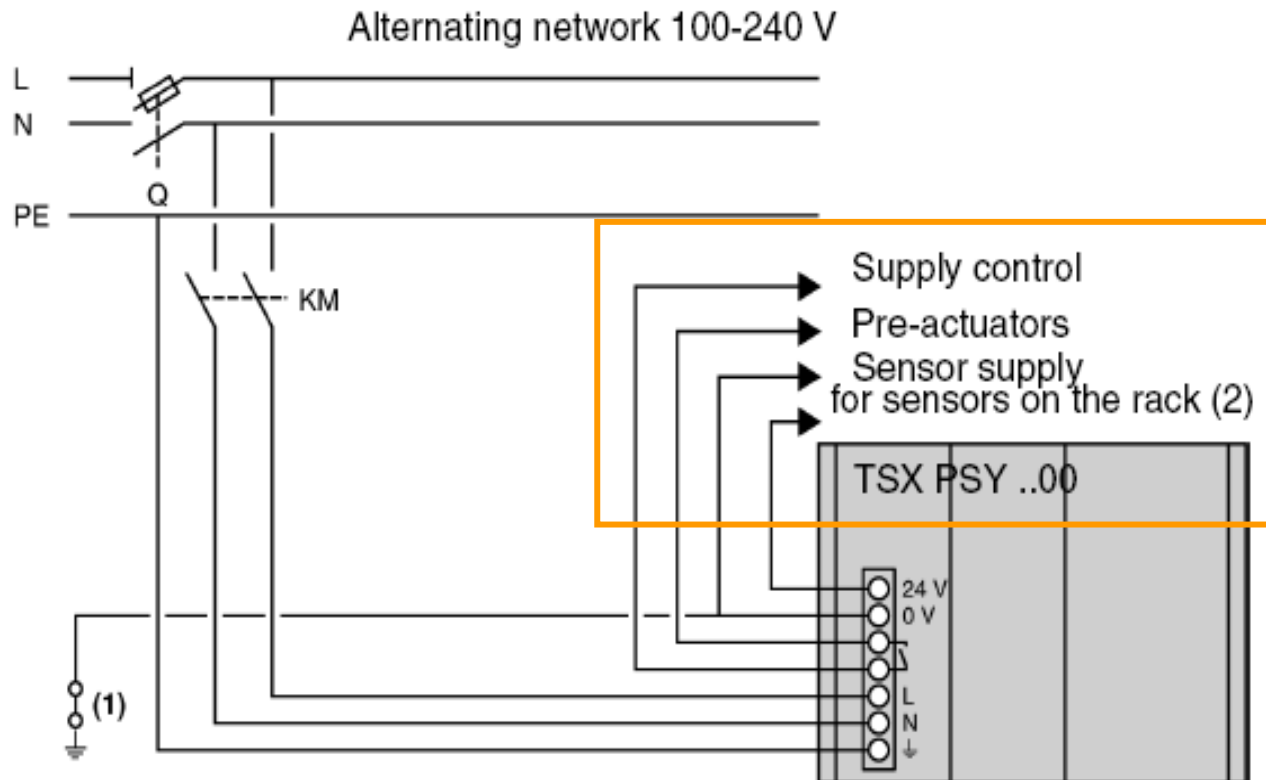
Fonte de alimentação (2)



O relé de saída (*alarm relay*) é activado em situações de avaria específicas que são monitorizadas pelo autómato (pode assim ser utilizado para sinalizar ao operador estas situações)

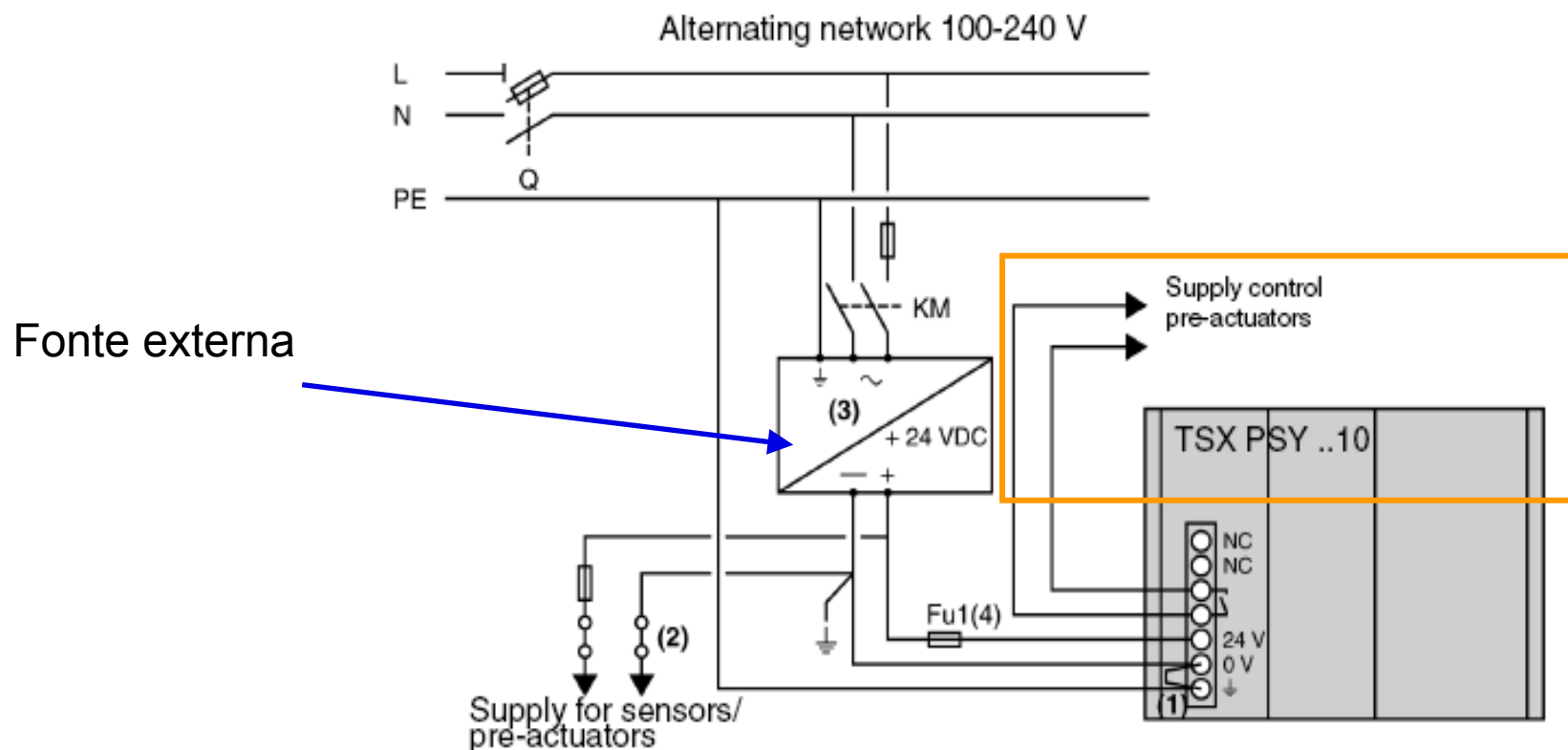
Fonte de alimentação: ligações (1)

- Os sensores e os pré-actuadores são alimentados a partir da fonte de alimentação do autómato através de um circuito externo.



Fonte de alimentação: ligações (2)

- A utilização de uma fonte de alimentação externa pode resultar de:
 - Vários equipamentos partilharem a mesma fonte de alimentação com o objectivo de reduzir custos de instalação
 - Necessidades de satisfazer consumos mais elevados dos sensores e pré-actuadores.



Fontes de alimentação: características

A escolha da fonte depende do calculo (prévio) que deve ser realizado em relação aos consumos envolvidos.

Alimentação dos módulos do autómato (o fabricante fornece o consumo de cada módulo)	Secondary	Total output		26W
		5VDC output	Nominal voltage:	5.1V
			Nominal current	5A
Alimentação dos sensores e pré-actuadores			Power (typical)	25W
		24VR output (24V relay) (2)	Nominal voltage:	24VDC
			Nominal current	0.6A
			Power (typical)	15W
		24VS output (24V sensor)	Nominal voltage:	24VDC
			Nominal current	0.5A
			Power (typical)	12W
		Protection of output from	overloading/short circuits/over-voltages	
Power dissipation				10W

Tempo de armazenamento dos dados com a bateria

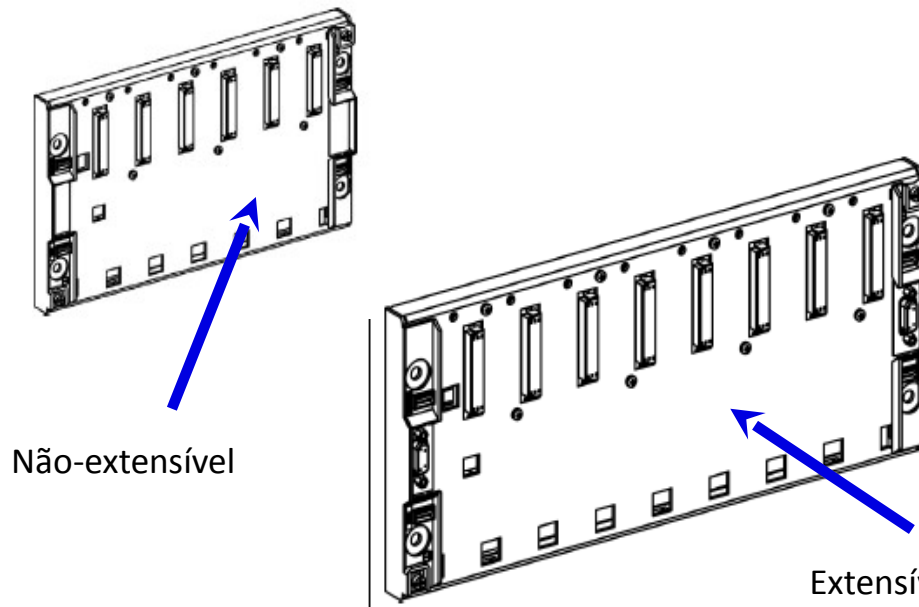
→

Ambient temperature when inoperative		≤ 30°C	40°C	50°C	60°C
Backup time	PLC off for 12 hours per day	5 years	3 years	2 years	1 year
	PLC off for 1 hour per day	5 years	5 years	4.5 years	4 years

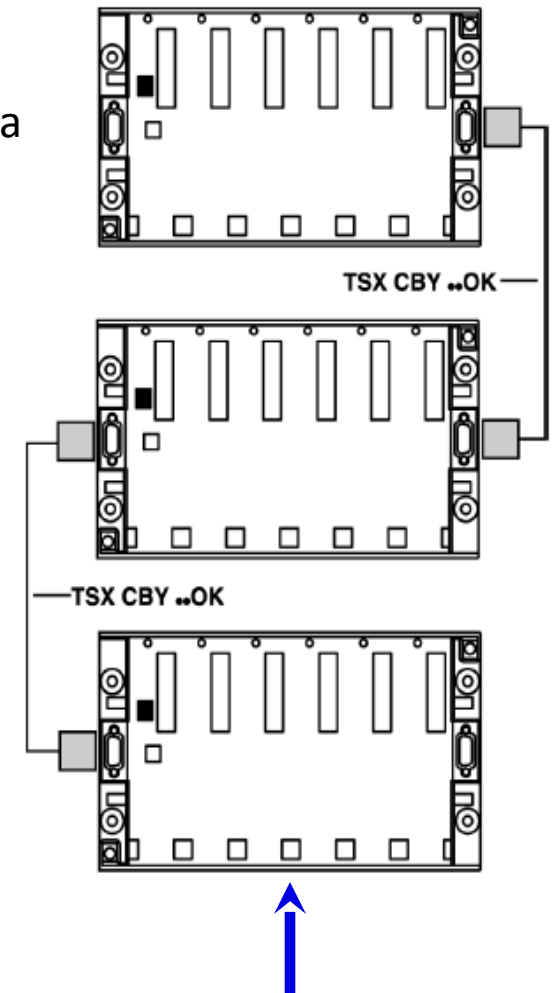
Bastidores (racks)

Funções:

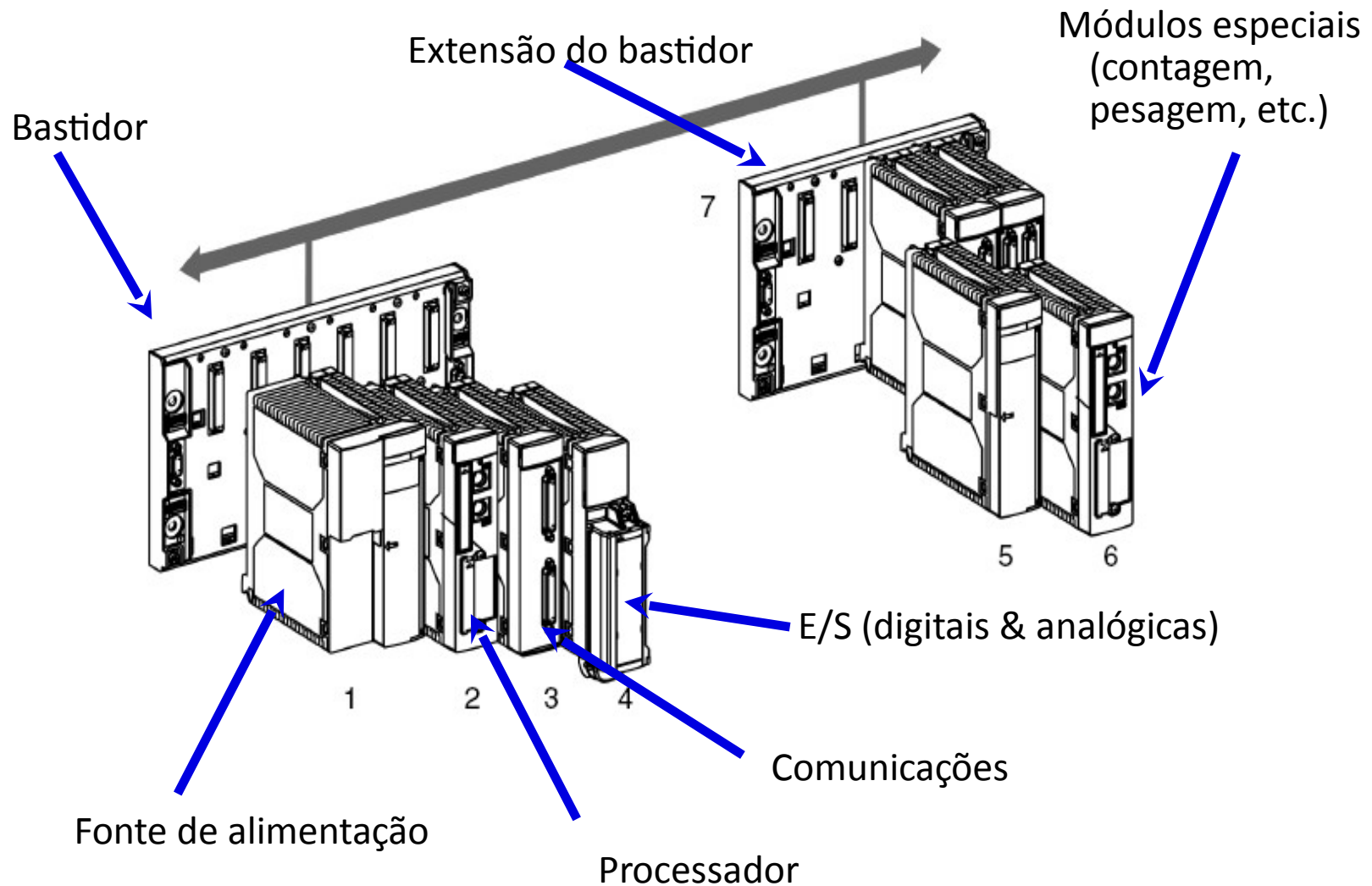
- Mecânicas: utilizados na montagem dos vários módulos do autómato (nº variável de módulos), bem como na sua fixação a uma estrutura (armário)
- Eléctrica: fornecer a alimentação aos vários módulos, bus de controlo, sinais para os bastidores de extensão



Extensível: utilizado quando se pretende estender o número de módulos do autómato

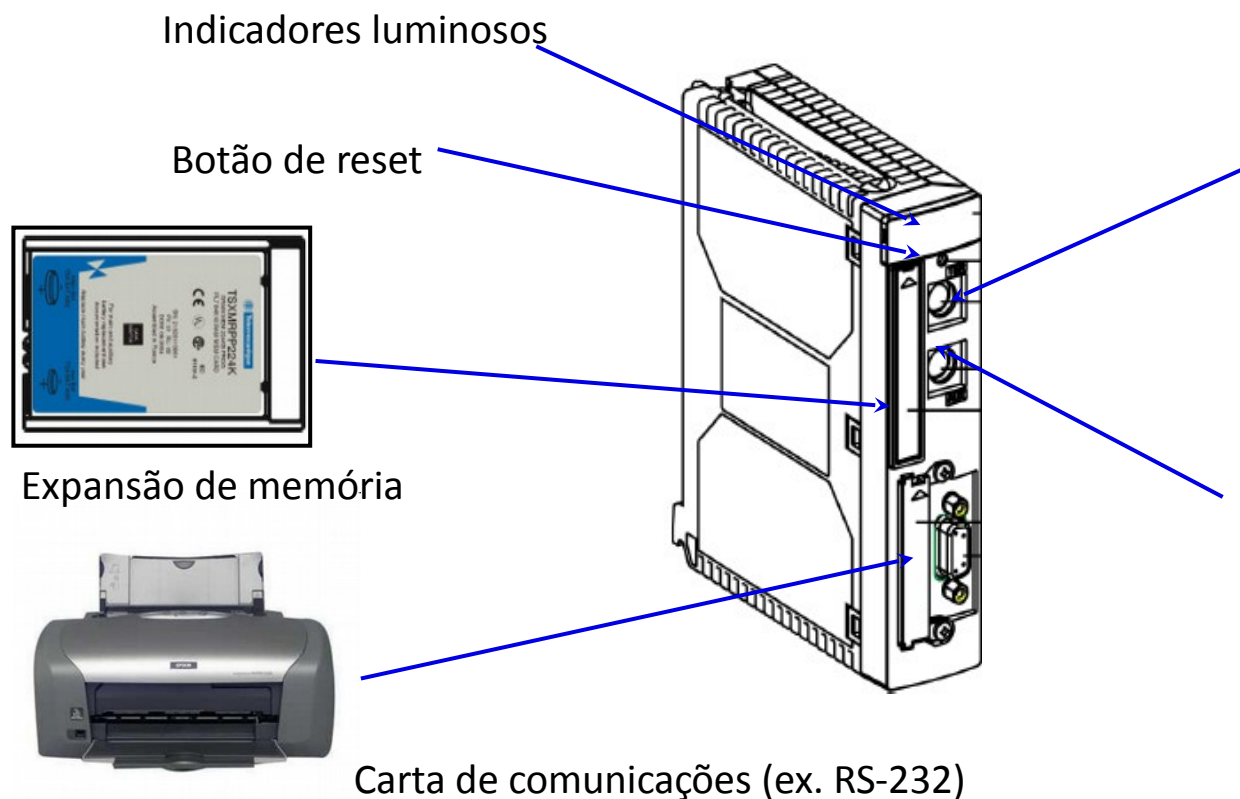


Exemplo de uma solução



Processador

- Execução da aplicação de controlo
- Actualização (leitura/escrita) dos dados nos diferentes módulos
- Funções de comunicação (geralmente interface série)



Características gerais

- A escolha do processador é uma das tarefas mais importantes na selecção de um autómato programável, pois impõe limitações ao restante HW:
 - Cartas: número e tipo
 - Capacidade de expansão: expansível ou não
 - E/S : número e tipo
 - Interfaces de comunicação: número e tipo
 - Memória: de programa, de dados
 - Tempo de ciclo / tempo de execução das instruções: capacidade de controlar +/- rapidamente um processo
 - Linguagens suportadas: facilidade de desenvolver aplicações +/- complexas
 - Capacidade de cálculo: suporte de instruções / funções +/- complexas (aritméticas, de manipulação de dados, etc).
- É necessário considerar bem estas questões antes de proceder a uma escolha.

Exemplo de características de vários CPUs

Characteristics and performance

Type of processor			TSX P57 0244M (1)	TSX P57 104M	TSX P57 1634M	TSX P57 154M	TSX P57 204M	TSX P57 2634M	TSX P57 254M
Maximum configuration	No. of racks	4/6/8 slots	1	4			16		
		12 slots	1	2			8		
	Max. no. of slots for modules		12	32			128		
Functions	Max. no. in rack of (3)	Discrete I/O	192/256 (2)	512			1024		
		Analog I/O	12	24			80		
		Process control channels	–				10 (up to 30 simple loops)		
		Application-specific channels (counter, axis, weighing, and serial links) (5)	4	8			24		
	Integrated connections	Ethernet TCP/IP	–		1	–		1	–
		Fipio manager	–			1 (63 agents)	–		1 (127 agents)
		Serial link	1 link with 2 connectors (TER and AUX) 19.2 Kbps						

Parâmetros importantes a observar na escolha do CPU

Exemplo de características de vários CPUs (2)

Integrated connections	Ethernet TCP/IP		–	1	–	1	–
	Fipio manager		–		1 (63 agents)	–	1 (127 agents)
	Serial link		1 link with 2 connectors (TER and AUX) 19.2 Kbps				
Maximum no. of connections	Network (Ethernet TCP/IP, Fipway, Ethway, Modbus Plus)		1	1, none if integrated Ethernet is used	1	1, none if integrated Ethernet is used	1
	AS-Interface bus		1	2		4	
	CANopen or Modbus Plus bus		1 integrated CANopen	1	1 Modbus Plus only	1	
	INTERBUS or Profibus DP bus		–			1	

Memories	Maximum capacity	Without PCMCIA card	Kb	96 prog. + data		160 prog. + data	192 prog. + data
		With PCMCIA card	Kb	128 prog. 96 data	224 prog. 96 data	768 prog. 160 data	768 prog. 192 data
		Data storage	Kb	256		16,384 (limited to 8192 with current PCMCIA cards)	
	Maximum size of object zones	Located internal bits (% of internal memory)	bits	4096		8132	
		Located internal data	Kb	64 for internal words %Moi 64 for constant words %Koi			
		Unlocated internal data	Kb	Unlimited (6)			

Exemplo de características de vários CPUs (3)

Application structure		Master task		1	1	1
		Fast task		1	1	1
		Auxiliary tasks		–	–	–
		Event tasks		32 (1 of which has priority)		64 (1 of which has priority)
Execution time for one instruction	Without PCMCIA card	Boolean	μs	0.19	0.19	0.19
		On word or fixed-point arithmetic	μs	0.25	0.25	0.25
		On floating points	μs	1.75...2.60 (7)	1.75...2.60 (7)	1.75...2.60 (7)
	With PCMCIA card	Boolean	μs	0.25	0.25	0.21
		On word or fixed-point arithmetic	μs	0.50	0.50	0.42
		On floating points	μs	1.75...2.60 (7)	1.75...2.60 (7)	1.75...2.60 (7)
	No. of Kinstructions executed every ms	Without PCMCIA card	100% Boolean	Kinst/ms	4.76	4.76
65% Boolean and 35% fixed arithmetic			Kinst/ms	3.71	3.71	3.71
With PCMCIA card		100% Boolean	Kinst/ms	3.10	3.10	3.70
		65% Boolean and 35% fixed arithmetic	Kinst/ms	2.10	2.10	2.53
System overhead	Master task		ms	1.00	1.00	1.00
	Fast task		ms	0.30	0.30	0.30

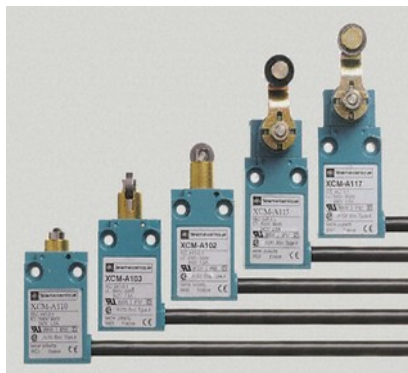
Parâmetros relacionados com o desempenho

Módulos de entradas digitais

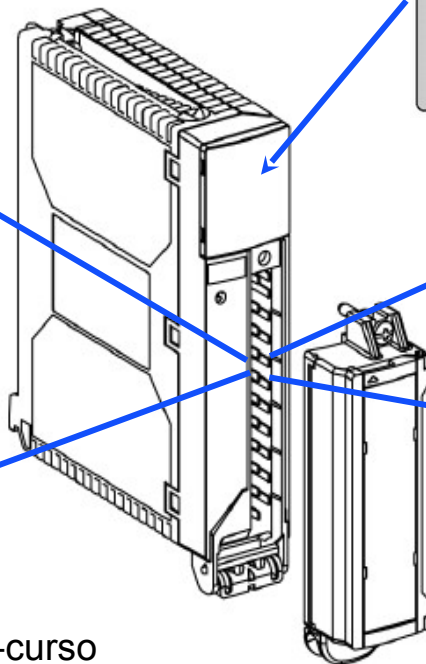
- Número de canais (pontos de E/S) variável: 8, 16, 32, 64
- Gama de sinais de entrada: (DC) 24/48 V; (AC) 24/48/120/240V
- Filtragem dos sinais de entrada
- Isolamento galvânico dos sinais de entrada
- Diagnóstico das entradas e do módulo



barreira óptica



fim-de-curso



RUN ERR	
I/O	
0	8
1	9
2	10
3	11
4	12
5	13
6	14
7	15



sensores de proximidade

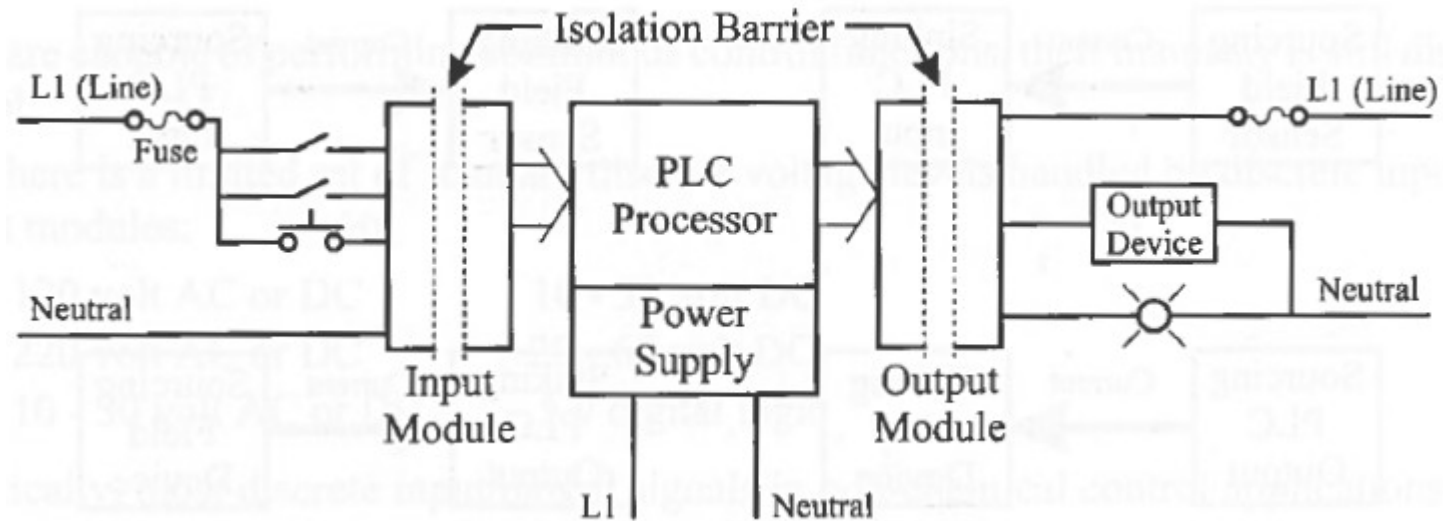


Botões de contacto



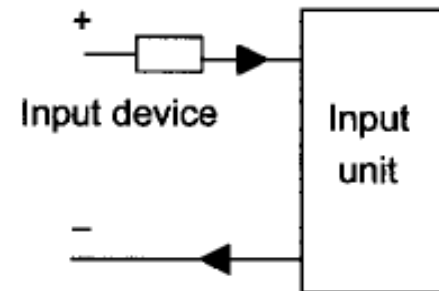
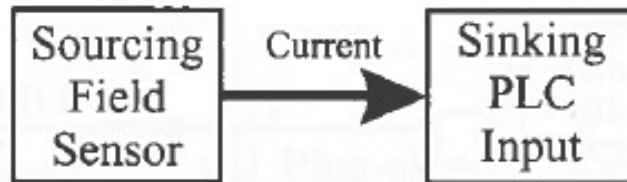
Isolamento de sinais

- Necessidade de isolamento galvânico entre os sinais provenientes dos sensores e actuadores e os sinais internos do autómato:
 - Sobretensões, massas com potenciais diferentes, ruído electromagnético (EMI), etc.

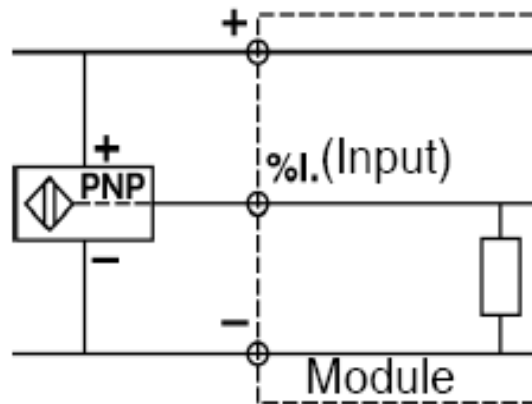


Tipos de interfaces com os sensores

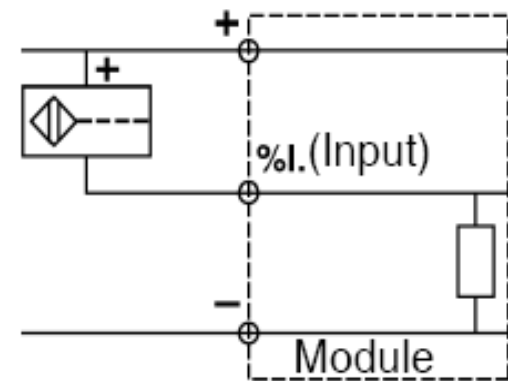
Source: o sensor fornece a corrente ao módulo de entrada (lógica positiva)



Com 3 fios

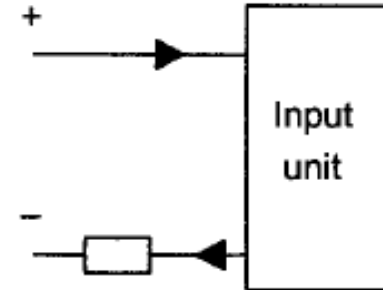
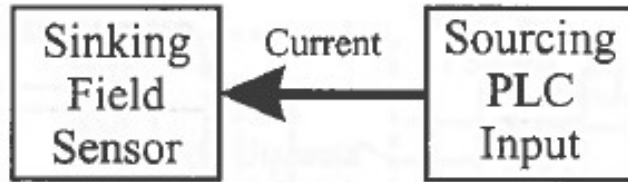


Com de 2 fios

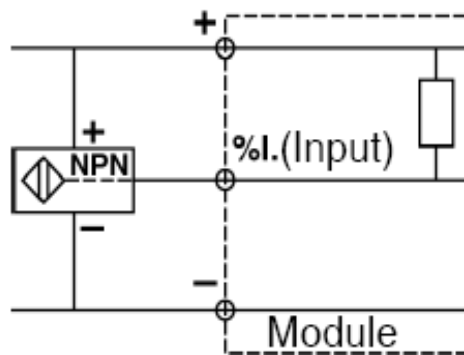


Tipos de interfaces com os sensores (2)

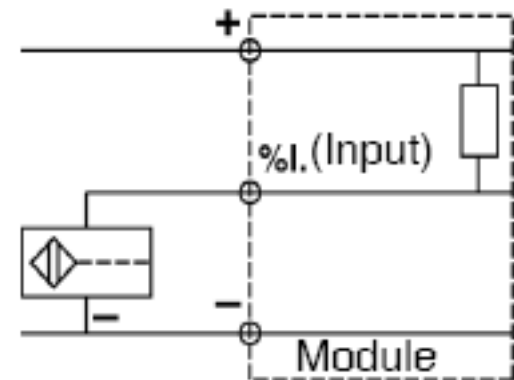
Sink: o módulo de entrada fornece a corrente ao sensor (lógica negativa)



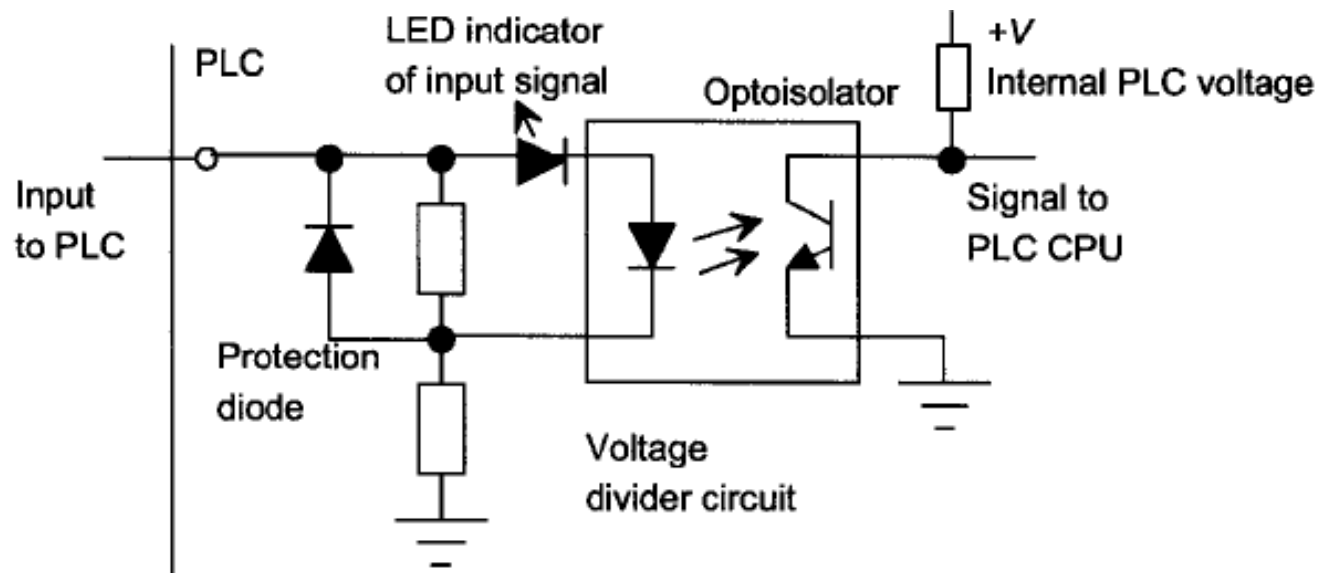
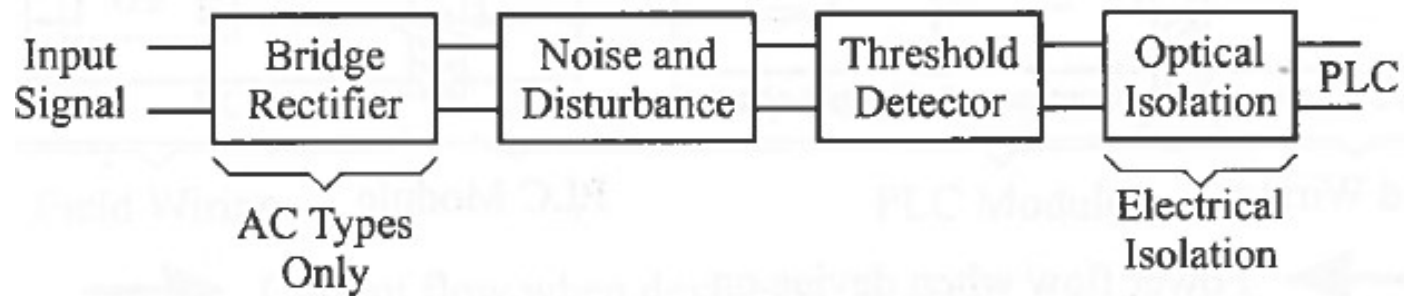
Com 3 fios



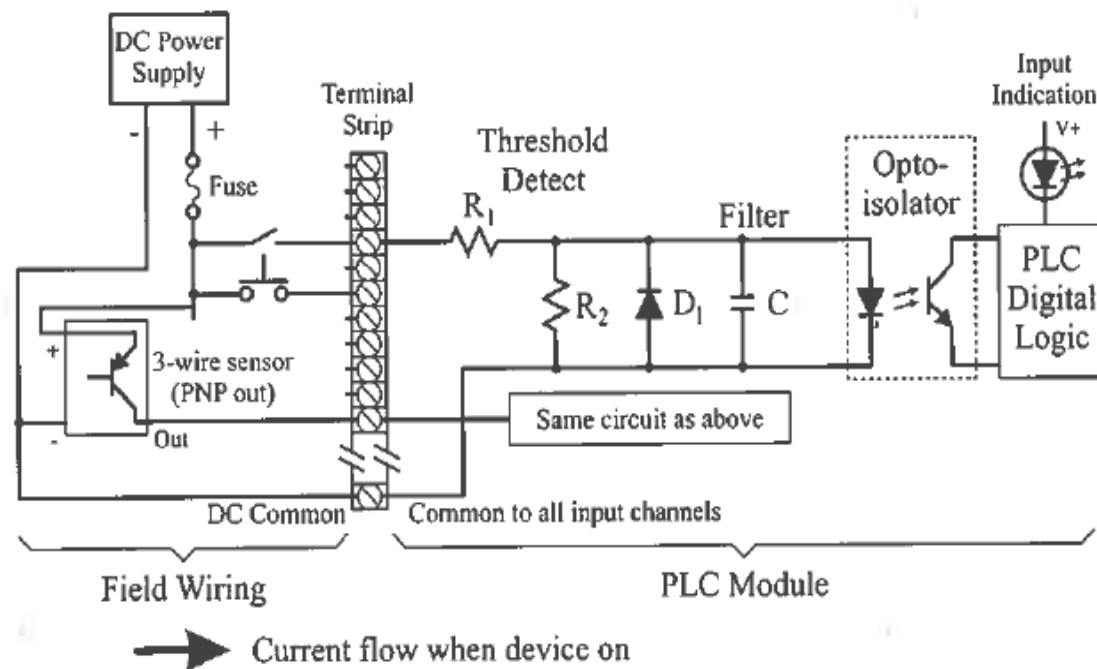
Com de 2 fios



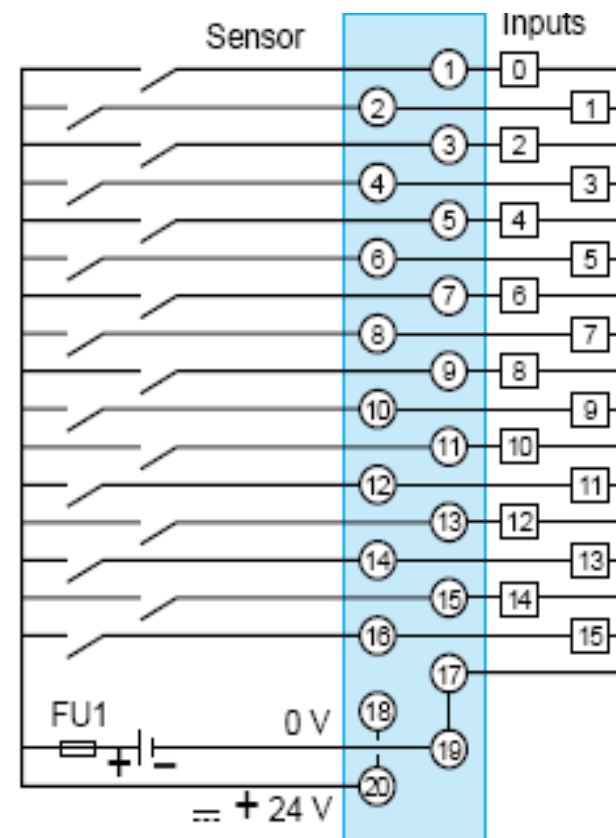
Entradas DC : interface eléctrica



Entradas DC : exemplo de ligação



Ex. carta 16 entradas



Entradas DC : folha de características (1)

Type of module				TSX DEY 08D2/16D2	TSX DEY 16D3	TSX DEY 16A2	TSX DEY 16FK	TSX DEY 32D2K
Number of inputs				8/16	16	16	16	32
Connections				Screw terminal	Screw terminal	Screw terminal	HE 10 connector	HE 10 connector
Nominal input values	Voltage	V	24 (pos. logic)	48 (pos. logic)	24 (neg. logic)	24 (pos. logic) Fast inputs	24 (pos. logic)	24 (pos. logic)
	Current	mA	7	7	16	3.5	3.5	
	Sensor supply (ripple included)	V	19...30	38...60	19...30	19...30	19...30	
Input limit values	At state 1	Voltage	V	≥ 11	≥ 30	≤ Ual-14 V	≥ 11	≥ 11
		Current	mA	≥ 6,5	≥ 6.5	≥ 6.5	≥ 3	≥ 3
	At state 0	Voltage	V	≤ 5	≤ 10	≥ Ual-5	≤ 5	≤ 5
		Current	mA	≤ 2	≤ 2	≤ 2	≤ 1.5	≤ 1.5
Impédance d'entrée à l'état 1			KΩ	4	7	1.6	6.3	6.3

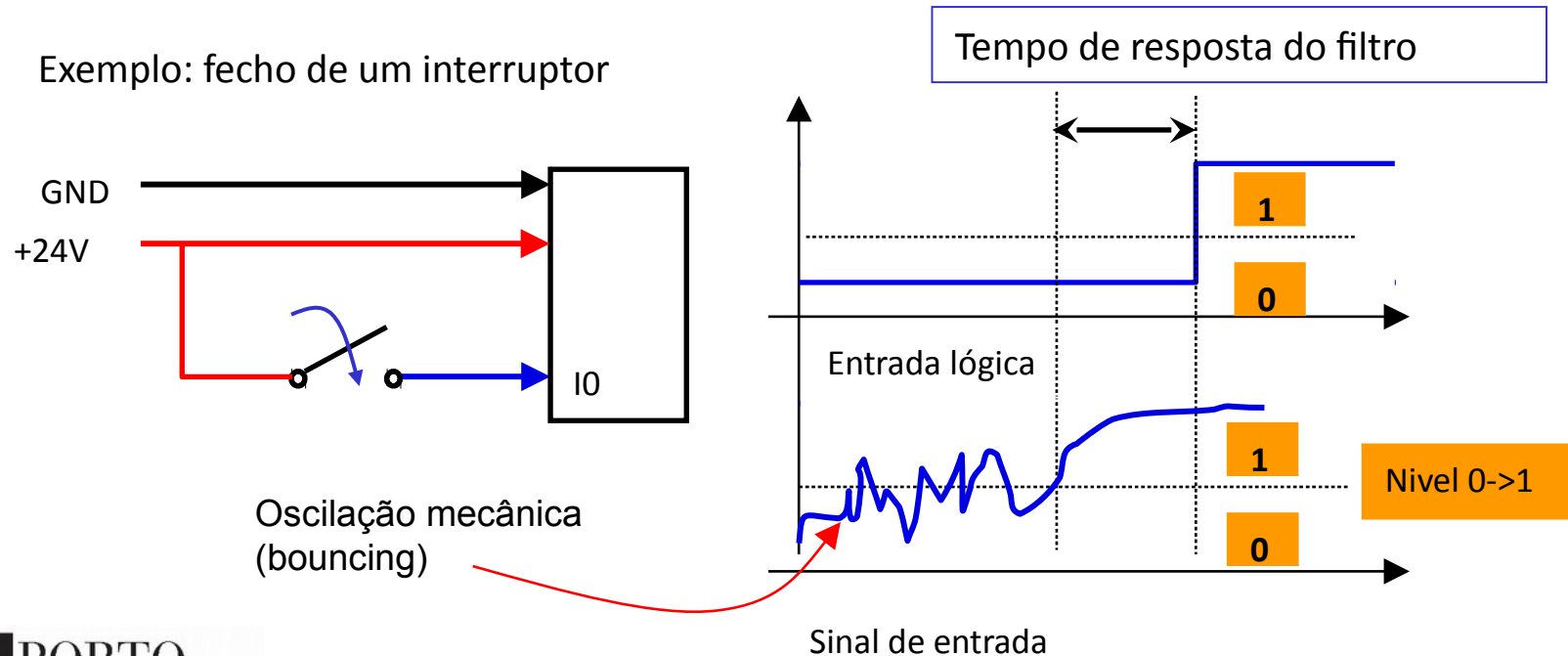
Entradas DC : folha de características (2)

- **Tempo de resposta** (atraso da entrada): tempo necessário que entrada fique ON ou OFF para que o módulo a reconheça com tal
 - Responsável: filtro de entrada e circuito de processamento
 - Dependente do tipo de transição (ON->OFF ou OFF->ON)
 - Limita a máxima frequência que se pode detectar

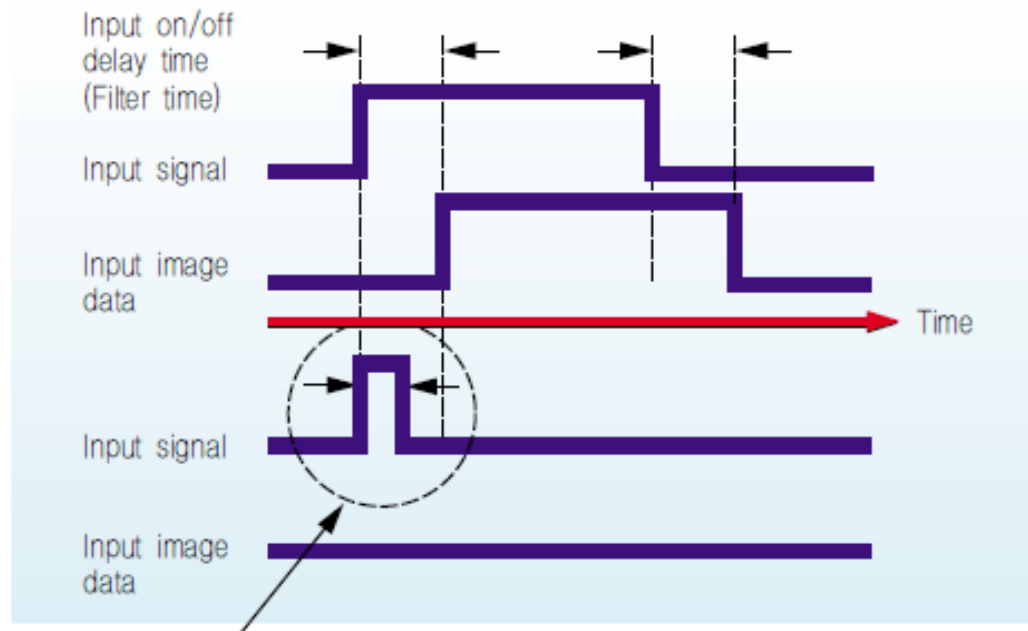
Temps de réponse	Typique	ms	4	4	10	Configurable from 0.1 to	4
	Maximum	ms	7	7	20	7.5	7
Conformité IEC 1131 2			Type 2	Type 2	Type 2	Type 1	Type 1
Compatibility 2-wire/3-wire prox. sensor			IEC 947 5 2	IEC 947 5 2	IEC 947 5 2	See table on page 3/9	

Filtro das entradas

- O objectivo é garantir que “ruído” que exista entrada não possa modificar o valor dessa mesma entrada em termos de interpretação lógica.
- O filtro das entradas é do tipo passa-baixo.
- Geralmente especificado (no autómato) como o tempo de resposta (τ do filtro, em milisegundos).
- Pode ser fixo, ou configurável (depende do tipo de carta).



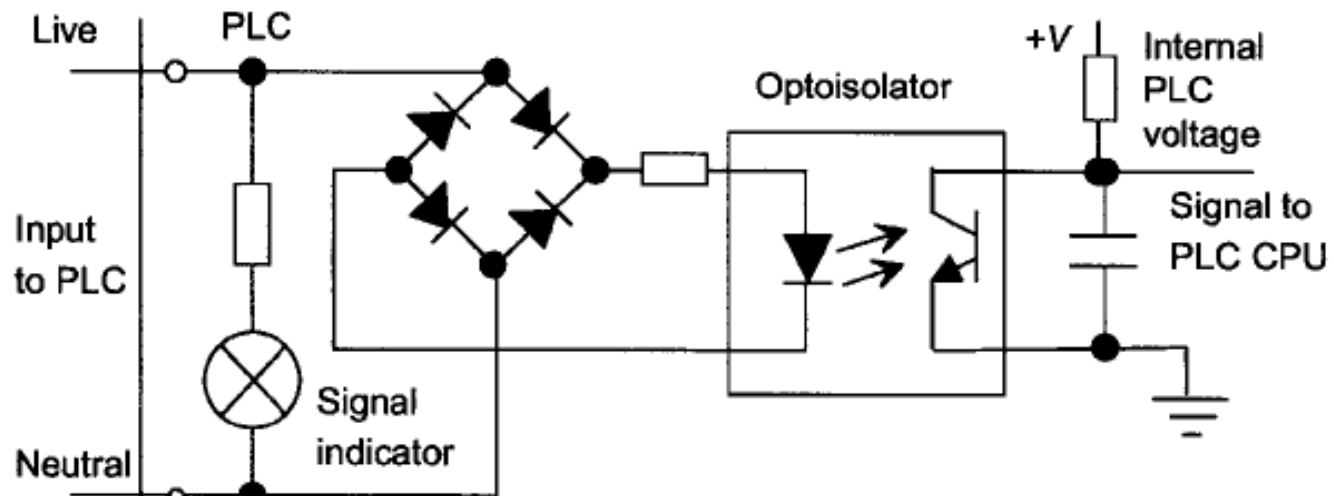
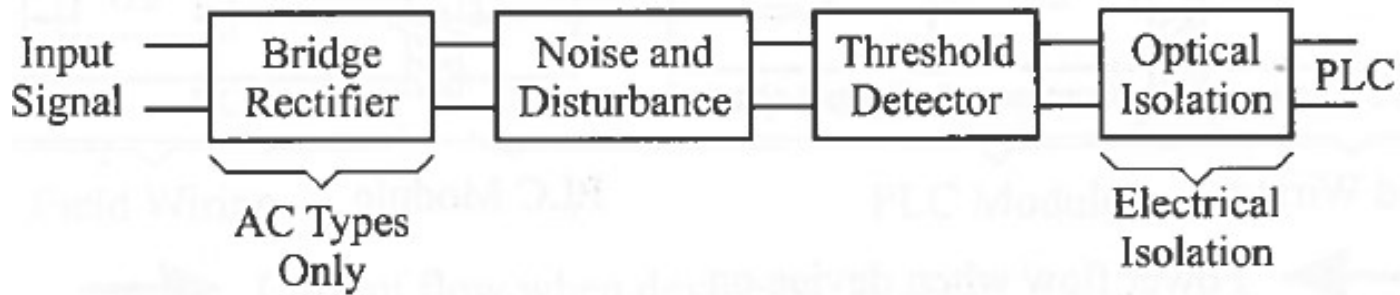
Filtro das entradas



Any input signal shorter than assigned on/off filter time from KGLWIN is ignored.

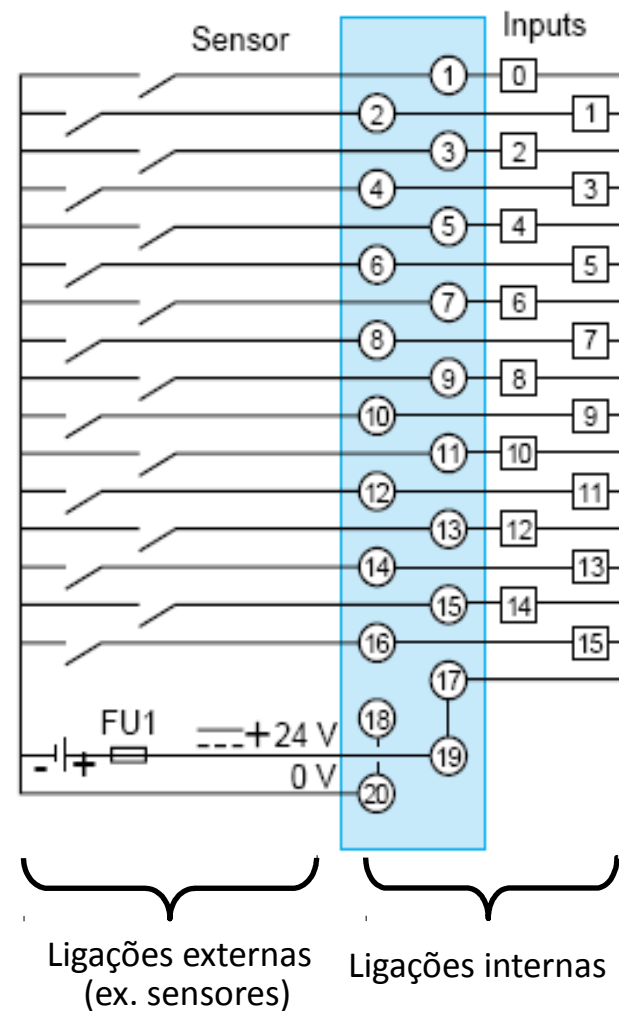
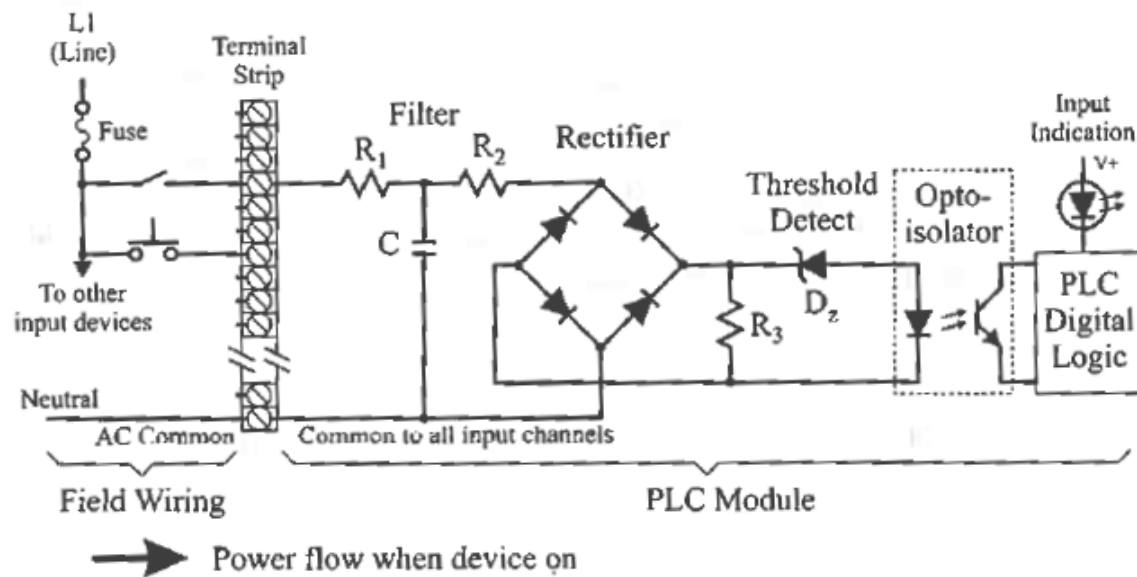
Analogia com tempo de resposta do filtro RC

Entradas AC : interface eléctrica



Entradas AC : exemplo de ligação

Ex. carta 16 entradas

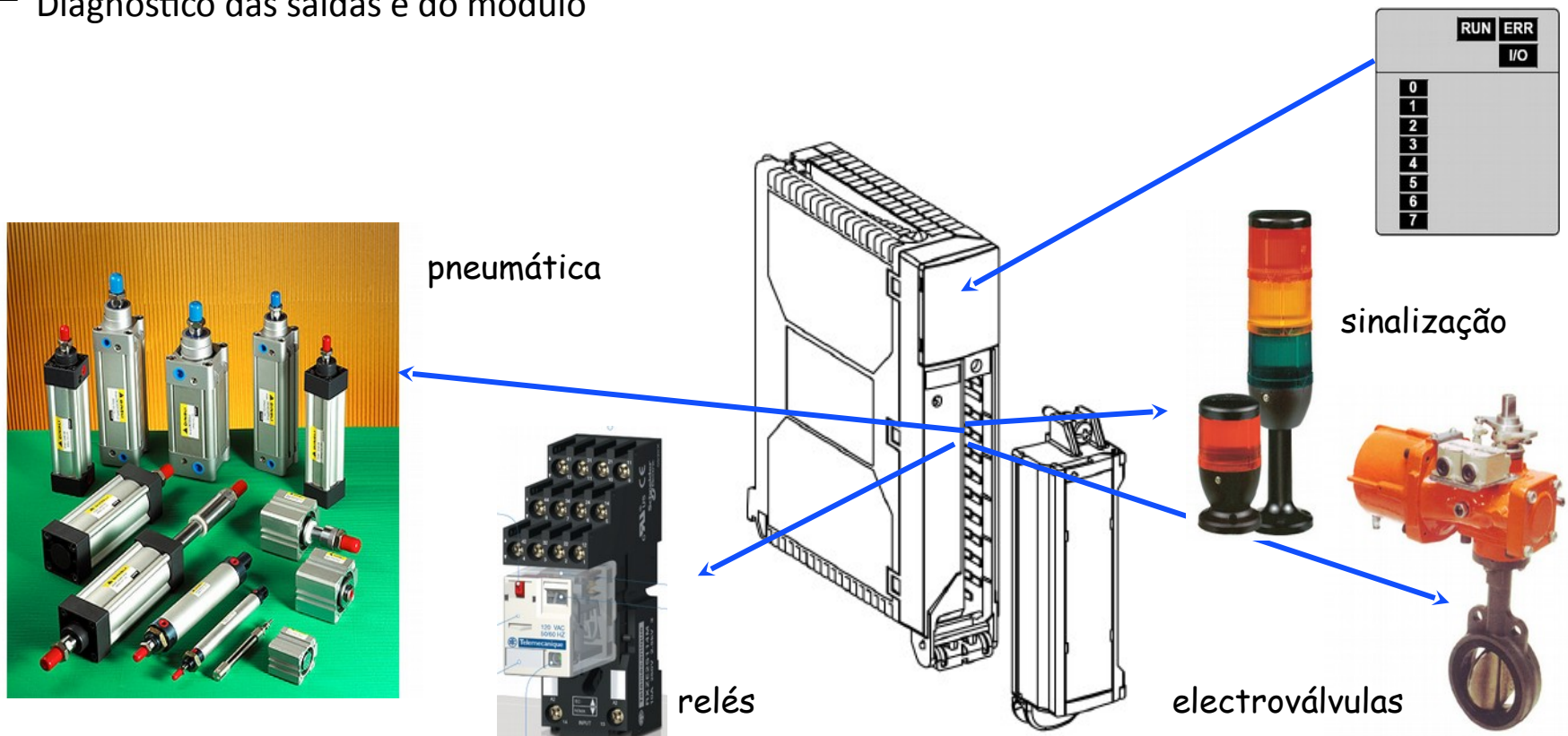


Entradas AC : folha de características

Type of module			TSX DEY 16A2	TSX DEY 16A3	TSX DEY 16A4	TSX DEY 16A5	
Number of inputs				16	16	16	16
Nominal input values	Voltage	V	~ 24	~ 48	~ 110	~ 220	
	Current	mA	15	16	12	15	
	Frequency	Hz	47...63	47...63	47...63	47...63	
	Sensor supply (ripple included)	V	20...26	40...52	85...132	170...264	
Input limit values	At state 1	Voltage	V	10	29	74	159
		Current	mA	6	6	6	6
	At state 0	Voltage	V	5	10	20	40
		Current	mA	4	4	4	4
Input impedance at state 1 for 24 V			KΩ	1.6	3.2	9.2	20
Response time	Typical	ms	15	10	10	10	
	Maximum	ms	20	20	20	20	
IEC 1131 2 conformity				Type 2	Type 2	Type 2	Type 2
Compatibility 2-wire/3-wire prox. sensor				IEC 947 5 2			
Isolation résistance			MΩ	> 10 at --- 500 V			
Dialectic strength			Veff	1500 - 50/60 Hz for 1 minute			
Type of input				Resistive	Capacitive		

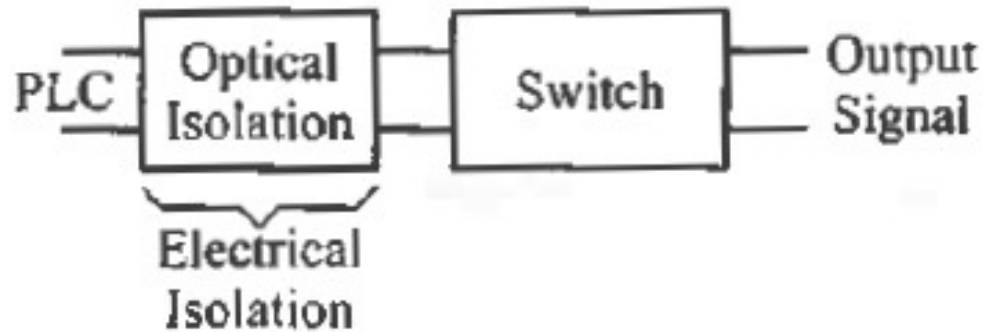
Módulos de saídas digitais

- Número de canais (pontos de E/S) variável: 8, 16, 32, 64
- Gama de sinais de entrada: (DC) 24/48 V; (AC) 24/48/120/240V
- Protecção das saídas: curto-circuito, *fallback*
- Diagnóstico das saídas e do módulo



Tipos de interfaces com os actuadores (1)

- Conceito geral



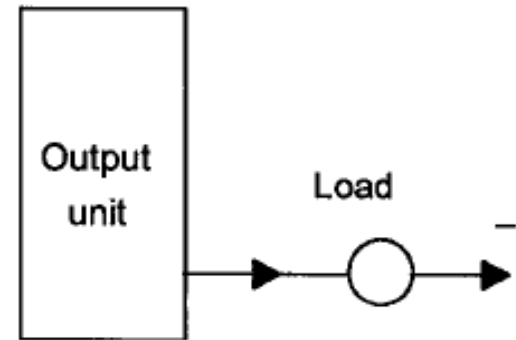
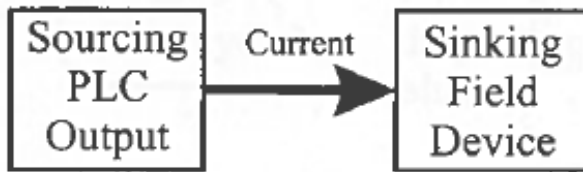
- Tipos:

- Transistor: cargas DC, tipicamente não indutivas (ou ligeiramente)
- Triac: cargas AC
- Relé: cargas DC ou AC com potência significativa. Existem questões mecânicas que devem ser avaliadas (número máximo de ciclos de abertura/fecho do relé)

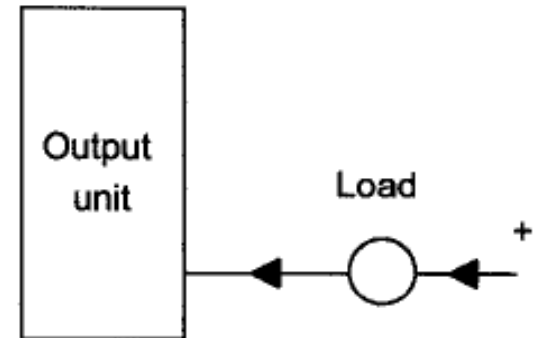
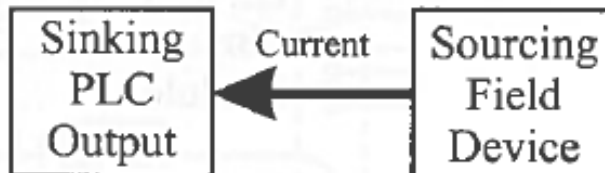
Tipos de interfaces com os actuadores (1)

- Lógica

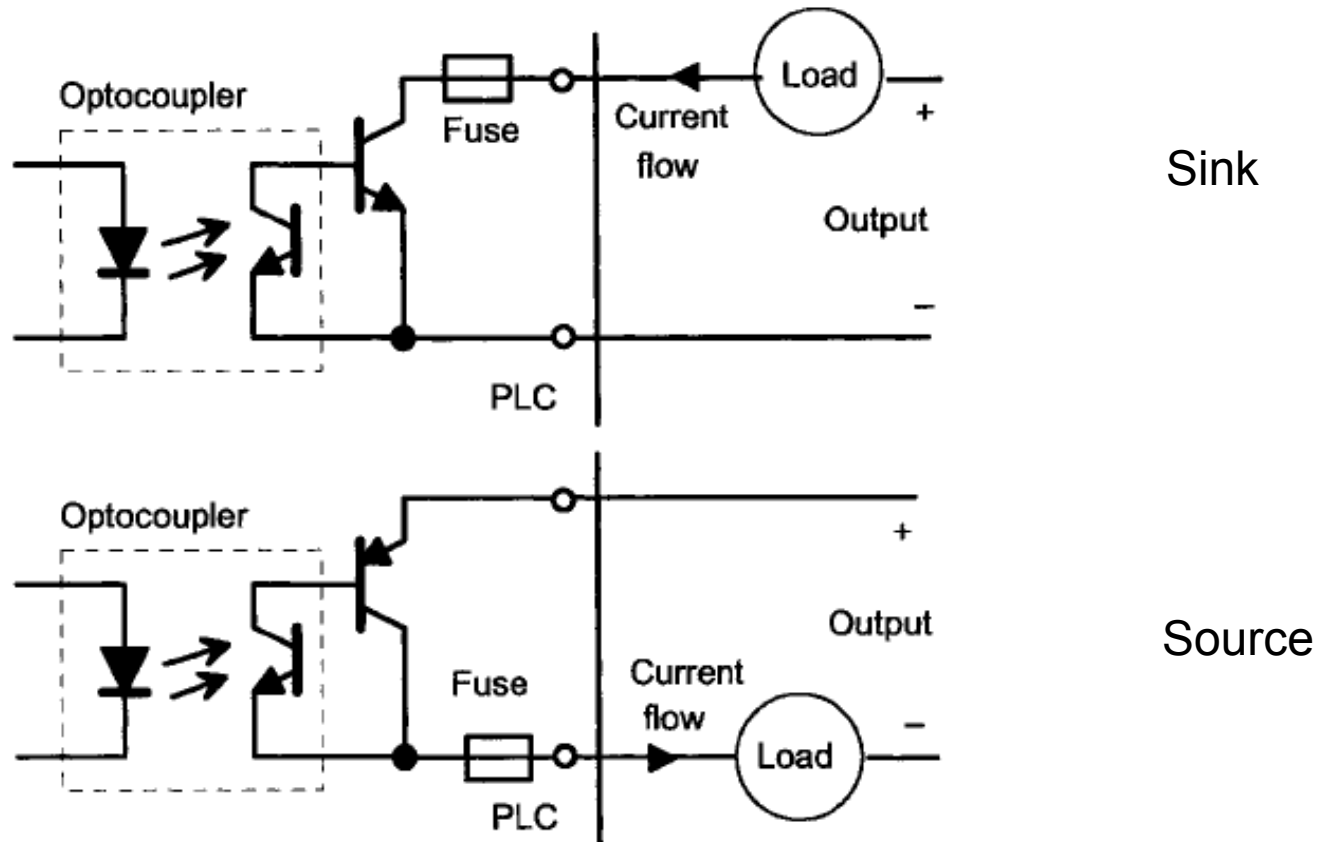
- Source : activas ao nível alto (ex. +24V)



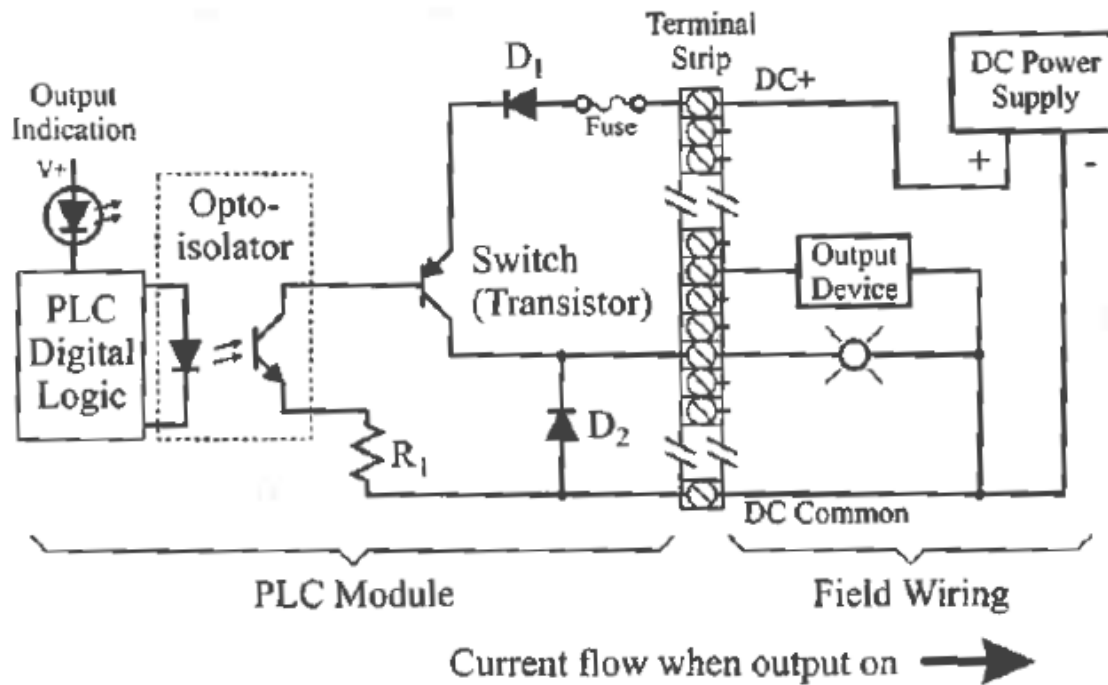
- Sink : activas ao nível baixo (ex. 0V)



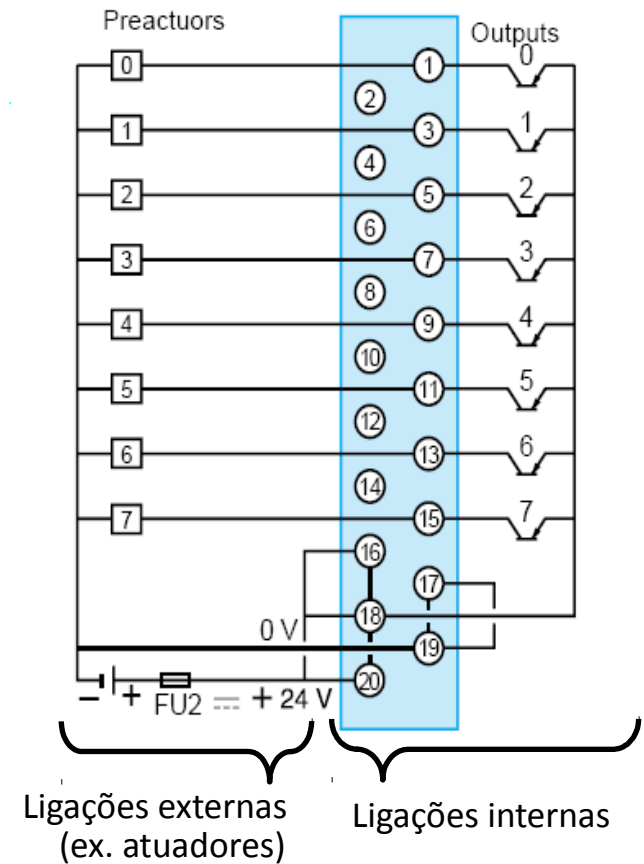
Saída a transístor



Saída a transistor : exemplo de ligação



Ex. carta 8 saídas

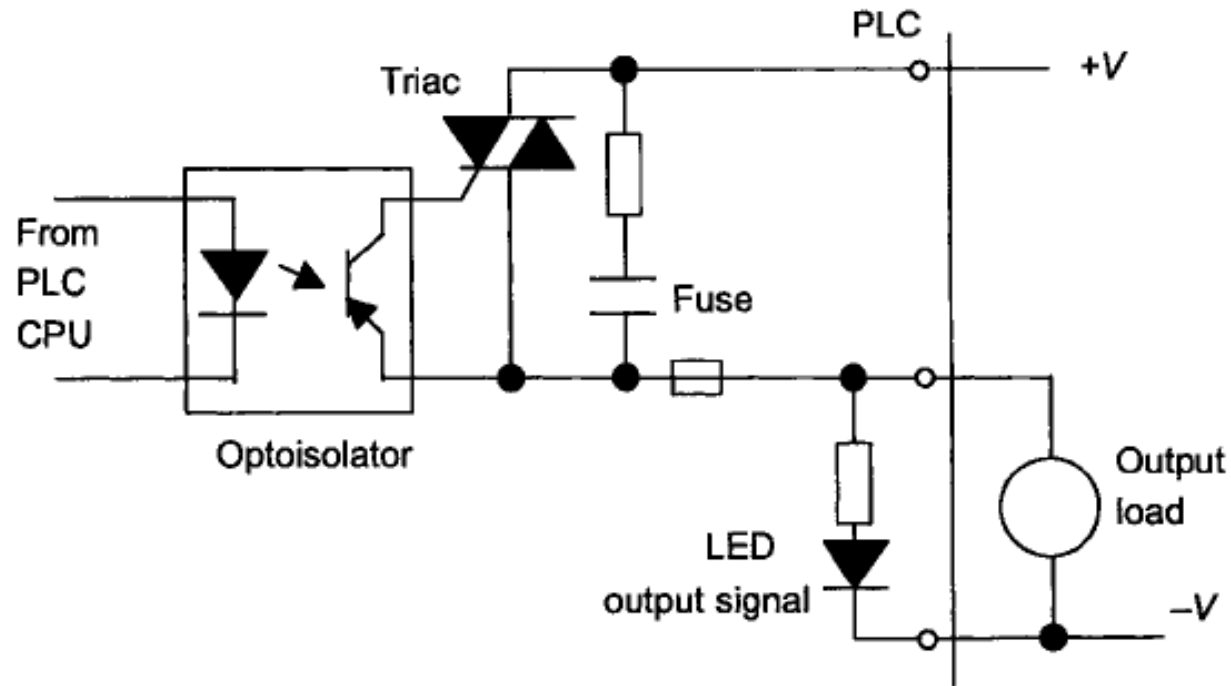


É necessário tomar especial atenção aos pontos de ligação que são comuns entre as várias saídas (ou entradas, no caso dos sensores). Na prática, representam o “circuito de retorno” dos vários actuadores. Podem existir problemas quando os vários actuadores não partilham um ponto de referência comum (ex. massa).

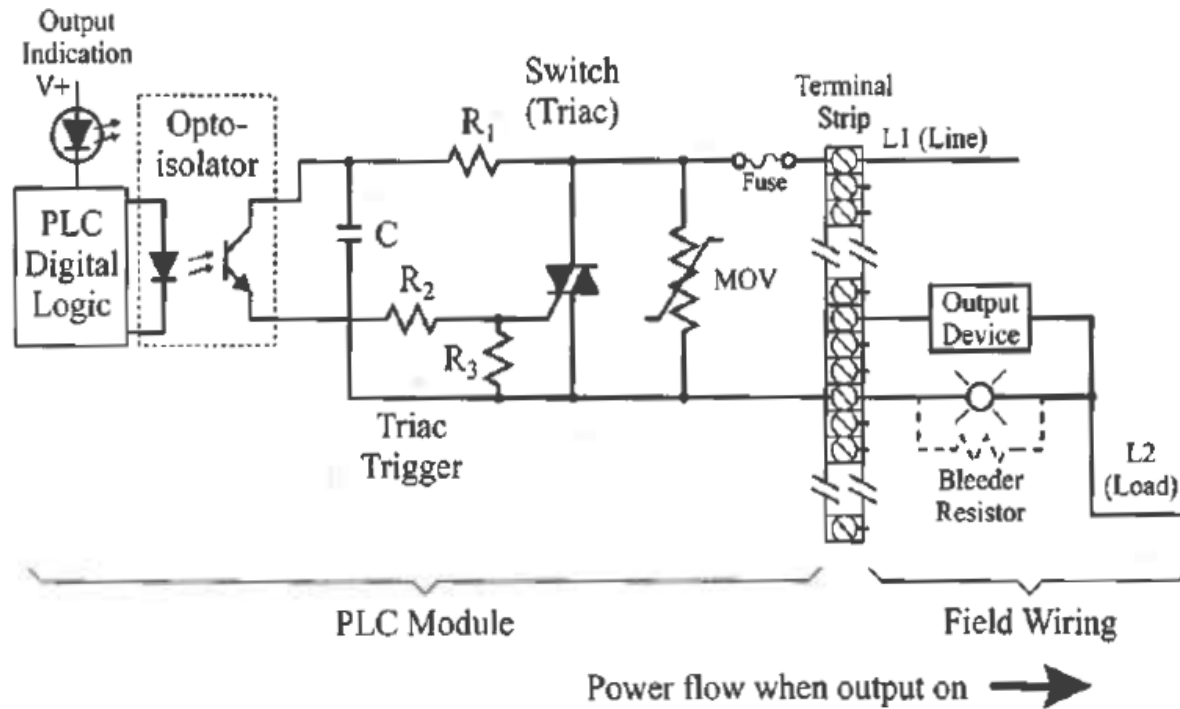
Saída a transístor : folha de características

Type of module			TSX DSY 08T2/16T2	TSX DSY 08T22	TSX DSY 08T31	TSX DSY 16T3
Output nominal values	Voltage	V	≐ 24	≐ 24	≐ 48	≐ 48
	Current	A	0.5	2	1	0.250
Output limit values	Voltage	V	19...30	19...30	38...60	38...60
	Current/channel	A	0.625	2.5	1.25	0.31
	Current/module	A	4/7	14	7	4
Leakage current	At state 0	mA	< 0.5	< 1	< 1	< 0.5
Residual voltage		V	< 1.2	< 0.5	< 1	< 1.5
Min. load impedance		W	48	12	48	192
Response time			1.2 ms	200 μs	200 μs	1.2 ms
Switching frequency on inductive load		Hz	0.5/LI2			
Built-in protection	Against overvoltages		Yes, by Transil diode			
	Against inversions		Yes, by reverse mounted diode, use a fuse on the + 24 V or + 48 V of the preactuators			
	Against short-circuits and overloads		Electronic tripping on reactivation (automatic or via program)			
Preactuator voltage detection threshold		V	16		34	
Isolation resistance		MΩ	> 10 sous ≐ 500 V			

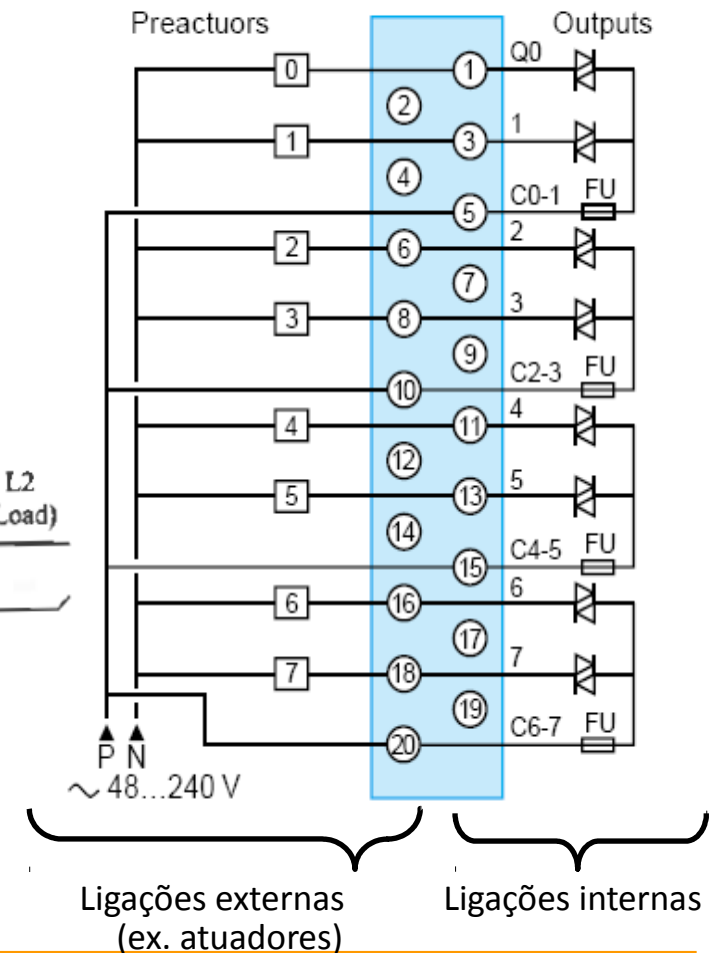
Saída a triac



Saída a triac : exemplo de ligação



Ex. carta 8 saídas

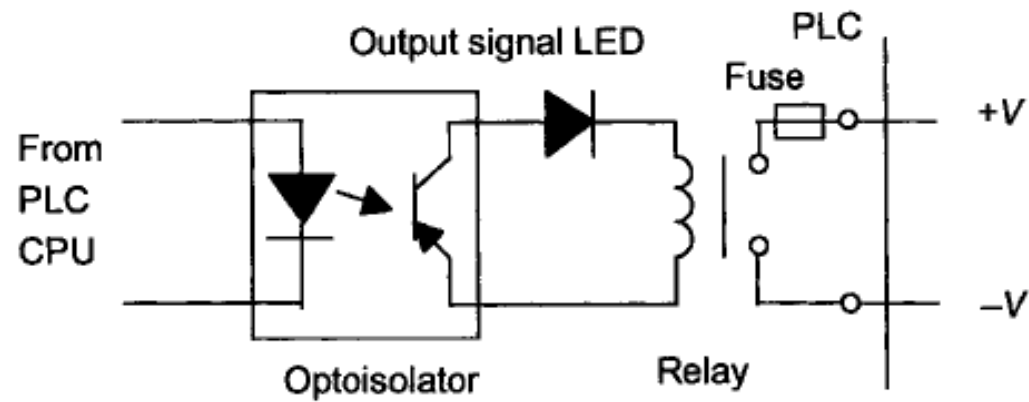


A existência de pontos comuns independentes permite ter tensões de alimentação independentes para cada saída

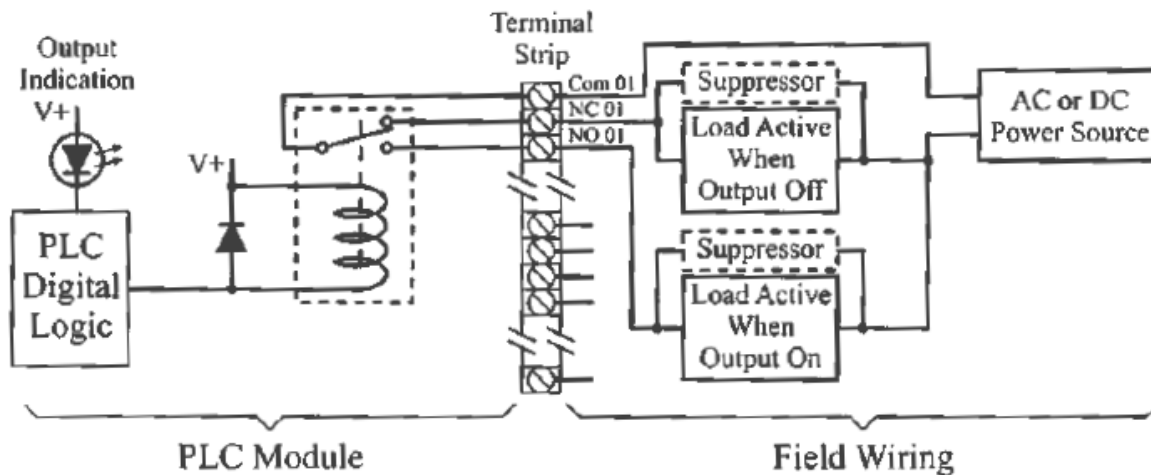
Saída a triac : folha de características

Type of module				TSX DSY 08S5	TSX DSY 16S5	TSX DSY 16S4
Operating voltage	a.c.	Nominal	V	~ 48...240		~ 24...120
		Limit	V	~ 41...264		~ 20...132
Permissible current			A	2 A per channel - 12 A per module	1 A per channel - 12 A per module	1 A per channel - 12 A per module
Response time	Activation		ms	≤ 10		
	Deactivation		ms	≤ 10		
Built-in protection	Against overvoltages			Ge-Mov		
	Against averloads and short-circuits			Fast blow fuse per common ≤ 5 A		Non interchangeable fireproof protection per common, 10 A

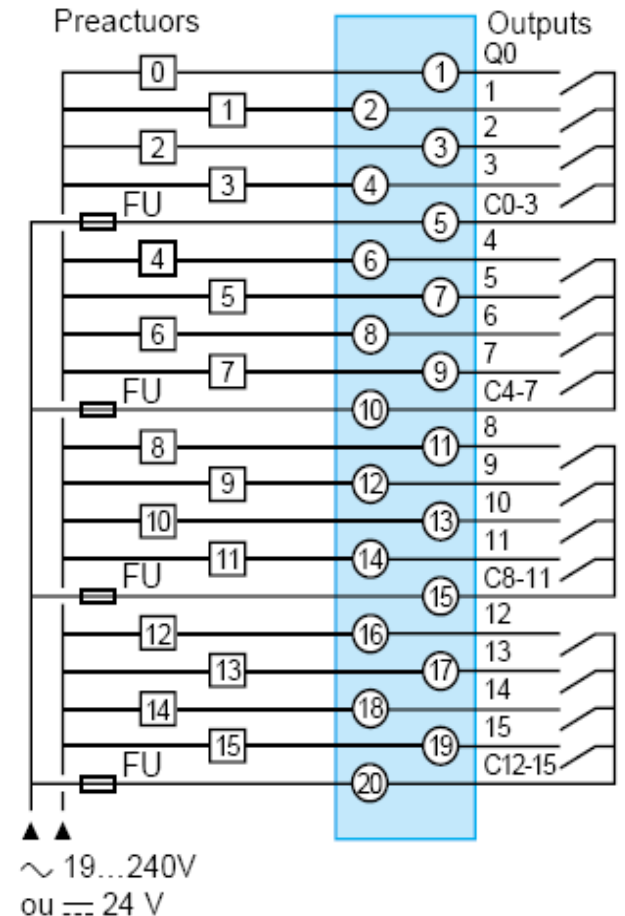
Saída a relé



Saída a relé : exemplo de ligação



Ex. carta 16 saídas



Ligações externas
(ex. atuadores)

Ligações internas

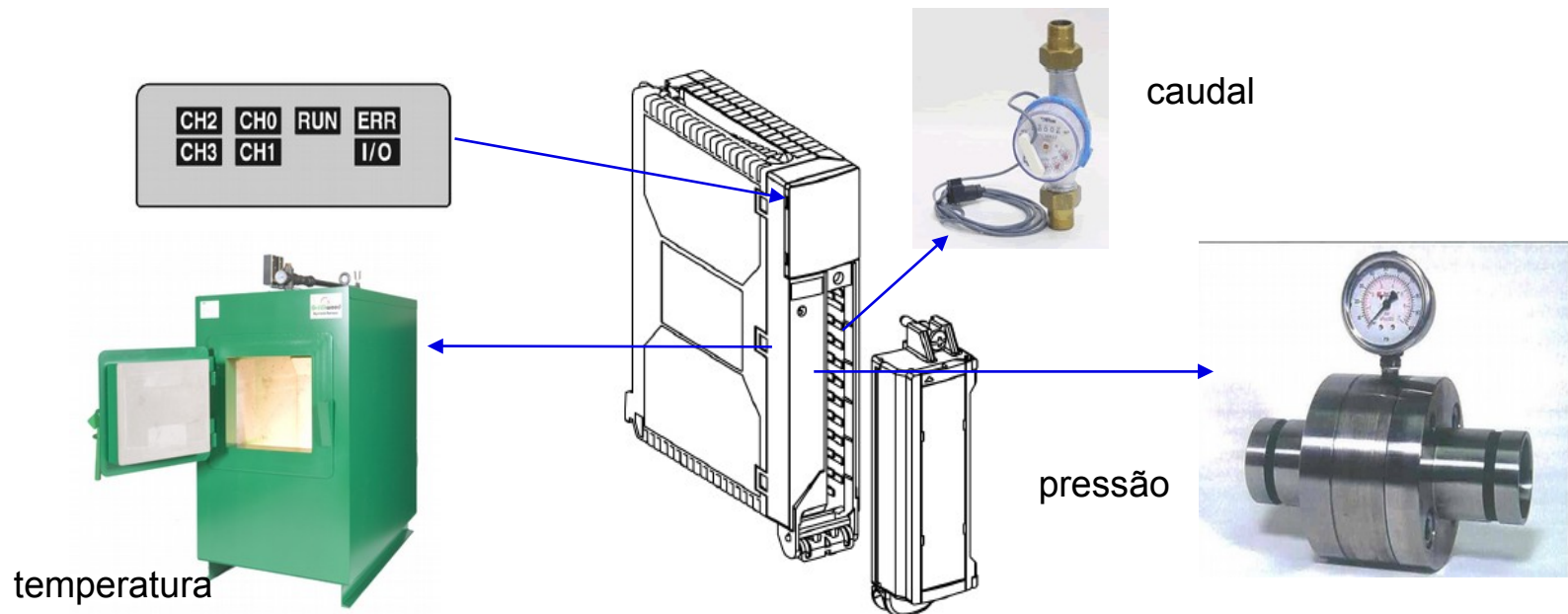
Saída a relé : folha de características

Characteristics of 50 VA relay output modules

Type of module				TSX DSY 08R5/16R5				
Voltage	a.c.	Nominal	V	~ 24...240				
		Limit	V	~ 20...264				
	d.c.	Nominal	V	= 12...24				
		Limit	V	= 10...34				
Thermal current			A	3				
Voltage			V	24	48	110	220	
a.c. load	AC-12 duty, resistive	Power	VA	50 (5)	50 (6), 110 (4)	110 (6), 220 (4)	220 (6)	
	AC-14 and AC-15 duty, inductive	Power	VA	24 (4)	10 (10), 24 (8)	10 (11), 50 (7), 110 (2)	10 (11), 50 (9) 110 (6), 220 (1)	
d.c. load	DC-12 duty, resistive	Power	W	24 (6), 40 (3)	–			
	DC-3 duty, inductive	Power	W	10 (8), 24 (6)	–			
Response time	Activation		ms	< 8				
	Deactivation		ms	< 10				
Type of contact				Normally open				
Built-in protection	Against overloads and short-circuits			None, each channel or group of channels must have a fast blow fuse				
	Against a.c. inductive overvoltages			None, an RC circuit MOV (ZNO) peak limiter circuit appropriate to the voltage must be mounted in parallel across the terminals of each preactuator				
	Against d.c. inductive overvoltages			None, a discharge diode must be fitted across the terminals of each preactuator				
Isolation resistance			MΩ	> 10 at = 500 V				

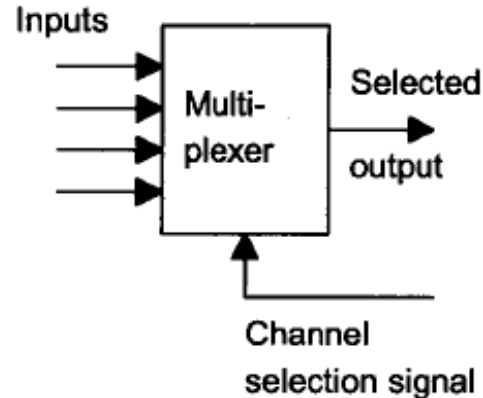
Módulos de entradas analógicas

- Conversão dos sinais analógicos em de dados digitais (memória do autómato)
- Número de canais (pontos de I/O) variável: 2, 4 ou 8
- Entradas em tensão (+/- 5/10 V), corrente (0/4-20mA), termopares e RTD
- Funções de processamento (ex. média de valores)
- Protecção das entradas
- Diagnóstico das entradas e do módulo
- A configuração (ex. tipos de entradas) e parametrização (ex. escalas) do módulo é efectuada por SW



Características gerais (1)

- Princípio de funcionamento: multiplexagem + conversor A/D

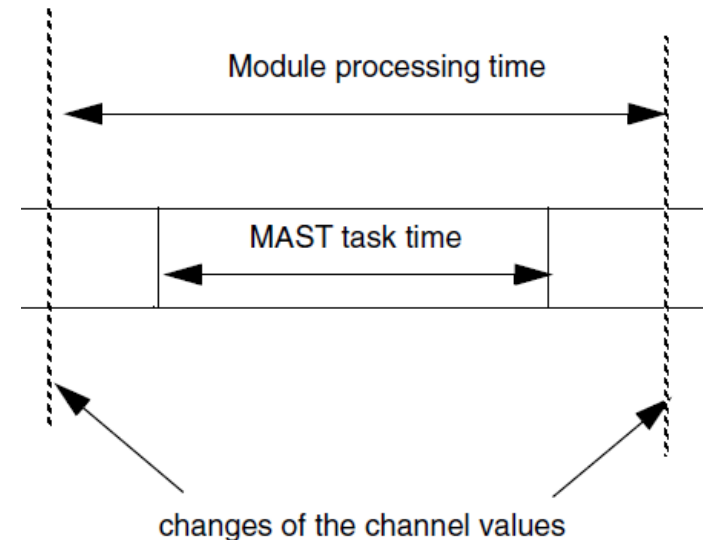
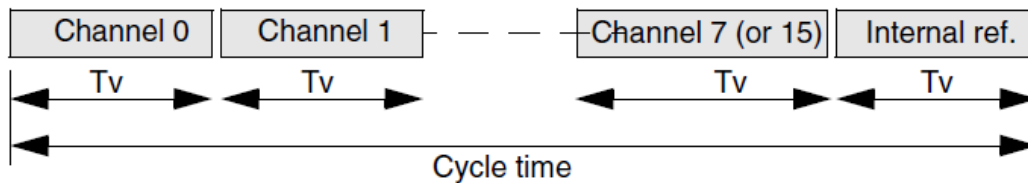


- Resolução: o nº de bits do conversor A/D limita a resolução da medida (i.e. a menor diferença que pode ser medida)
- Sinais em tensão (*single* e *diferencial*) e em corrente com gamas normalizadas

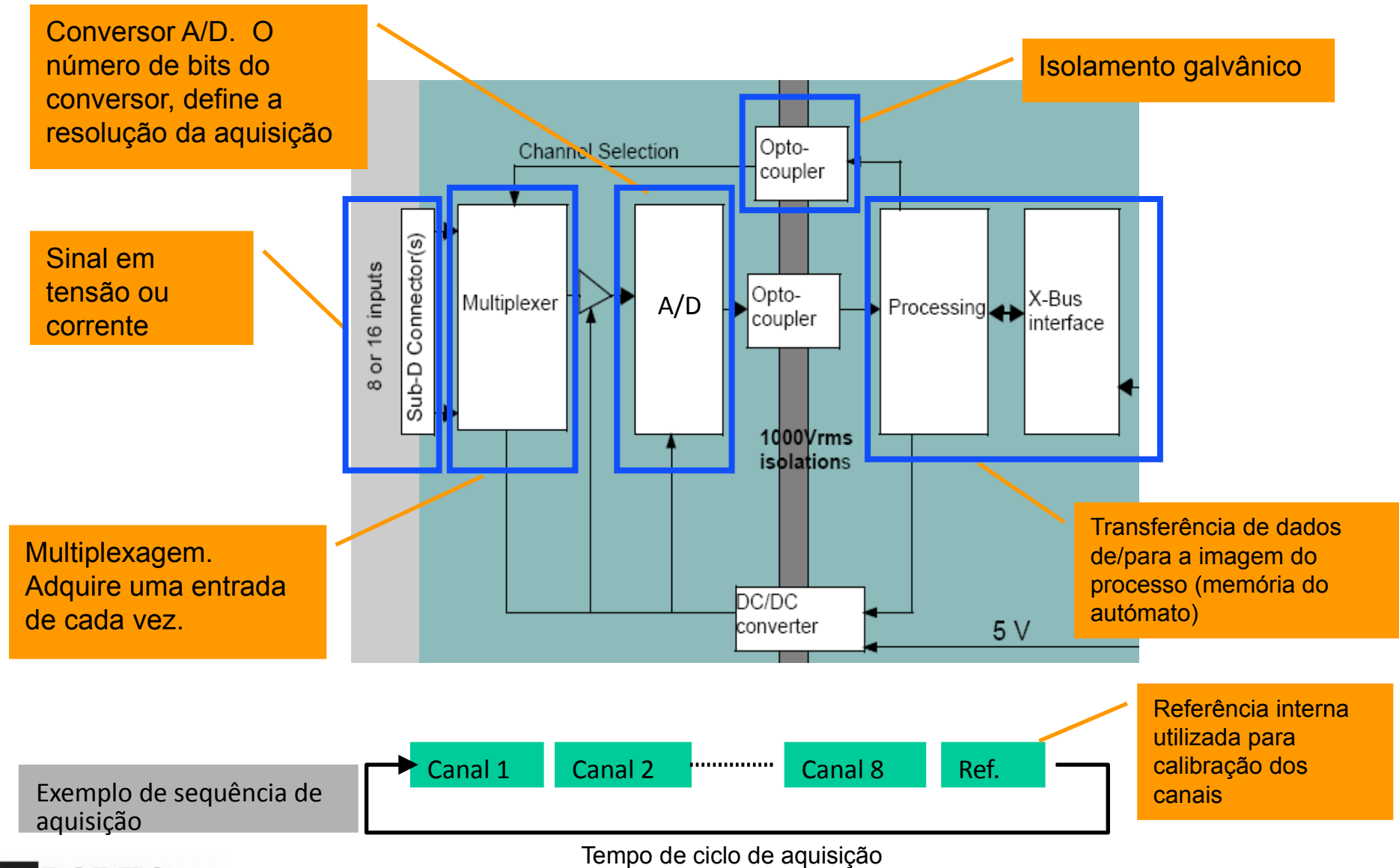
Analog Signal	Analog Signal
1 to 5 volt DC	4 to 20 mA
0 to 5 volt DC	0 to 20 mA
-5 to +5 volt DC	-20 to +20 mA
-10 to +10 volt DC	0 to 10 volt DC

Características gerais (2)

- Entradas dedicadas a sensores especiais (ex: RTD, termopares)
- Isolamento : nem sempre existe, é preciso ter especial cuidado com os sinais que vêm do campo
- A calibração (do sistema de aquisição) é geralmente realizada por SW
- As entradas não são amostradas em cada ciclo do autómato (podem demorar vários ciclos)

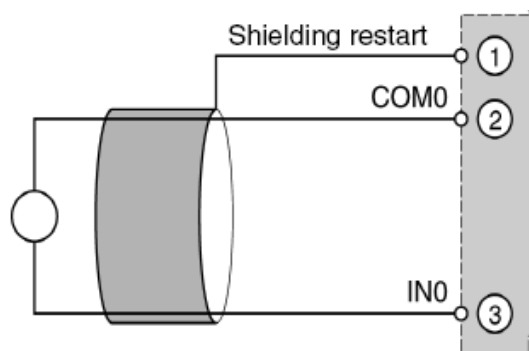


Exemplo : esquema interno do módulo

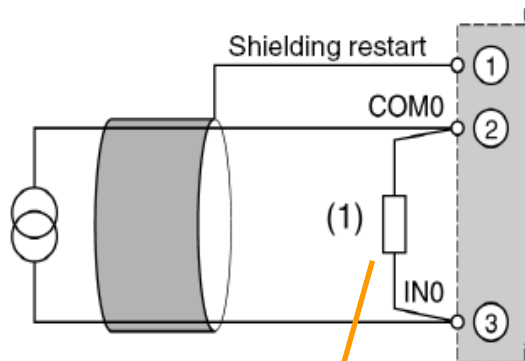


Exemplo de ligações (1)

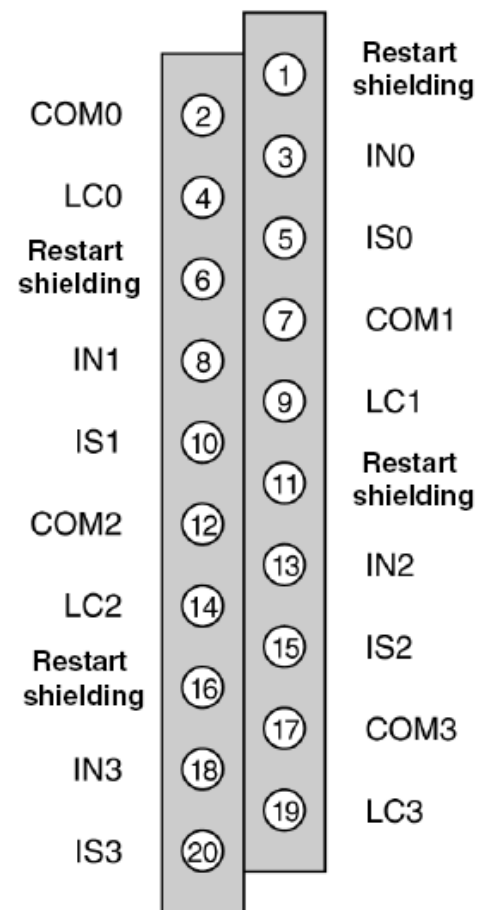
Entrada em tensão



Entrada em corrente (0/4..20mA)



Resistência de 250 Ω , para conversão tensão-corrente
 $U = I * R$



INx + Pole input of channel x

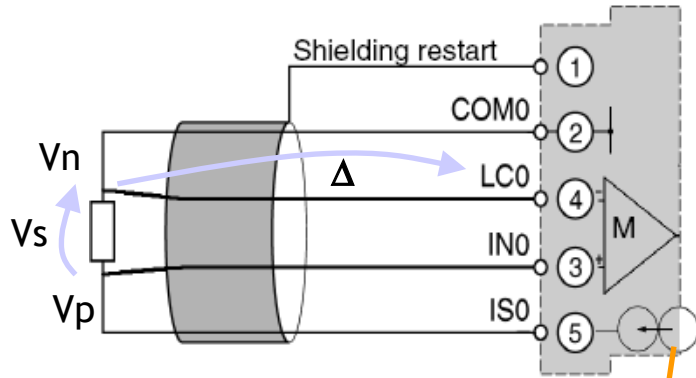
COMx - Pole input of channel x

ISx + Pole supply of the probe

LCx Line compensation

Exemplo de ligações

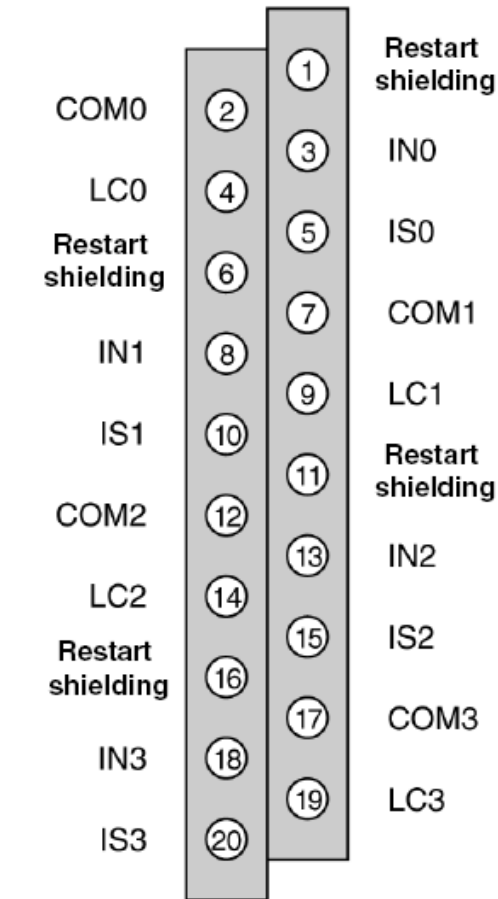
Ligação com 4 fios



$$IN0-LC0 = (Vp-\Delta) - (Vn-\Delta) = Vp-Vn = Vs$$

Permite a transmissão de um sinal em tensão, minimizando o problema da distância.

A fonte de corrente **IS0** alimenta o sensor, e o sinal proveniente do mesmo é medido sob a forma diferencial (**INO-LC0**), o que anula os efeitos da atenuação com a distância.



INx + Pole input of channel x
COMx - Pole input of channel x
ISx + Pole supply of the probe
LCx Line compensation

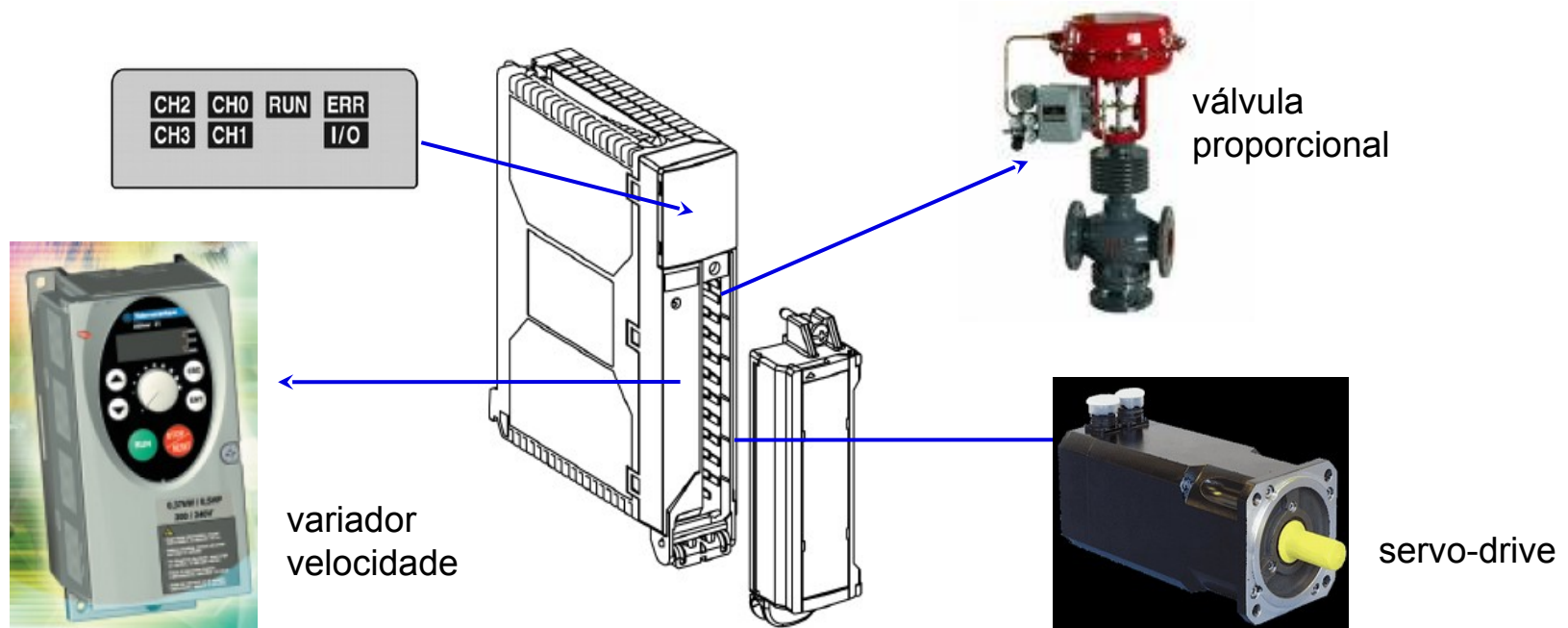
Características eléctricas das entradas: exemplo

Characteristics of analog input modules

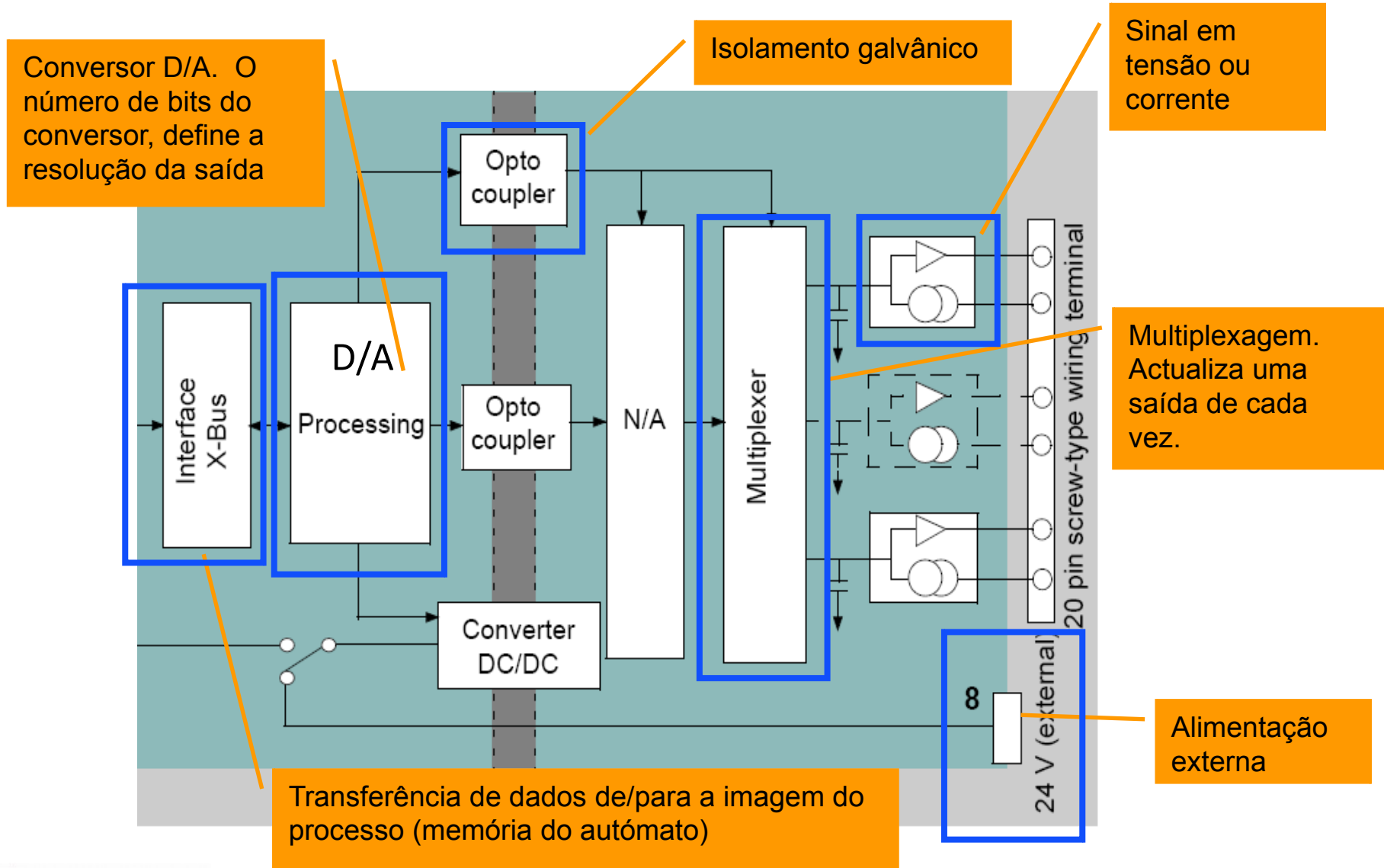
Type of input module			TSX AEY 800		TSX AEY 1600		TSX AEY 810			TSX AEY 420										
Number of channels			8		16		8			4										
Input range			± 10 V, 0...10 V, 0...5 V, 1...5 V, 0...20 mA, 4...20 mA																	
Analog/digital conversion			12 bits					16 bits												
Read time	Normal scan	ms	27		51		29.7			1										
	Fast scan	ms	3 x (no. of channels used + 1)					3.3 x (no. of channels used + 1)			–									
Max.error	Adquire apenas um conjunto específico de canais (indicado pelo utilizador)		± 10 V 0...10 V		0...5 V 1...5 V		0...20 mA 4...20 mA		± 10 V 0...10 V		0...5 V 1...5 V		0...20 mA 4...20 mA							
			%FS		0.19/0.1 (1)		0.25/0.16 (1)		0.244		0.13		0.142		0.1		0.2		0.2	
			%FS		0.22/0.13 (1)		0.22/0.13 (1)		0.41/0.32 (1)		0.305		0.191		0.12		0.2		0.4	
Isolation	Betw. ch. and bus	V rms	1000																	
	Betw. ch. and earth	V rms	1000																	
	Between channels	≡ V	Common point					± 200			Common point									
Common mode between channels			None					± 200			None									
Max. overvoltage/overcurrent on the inputs			± 30 V voltage ± 30 mA en current																	

Módulos de saídas analógicas

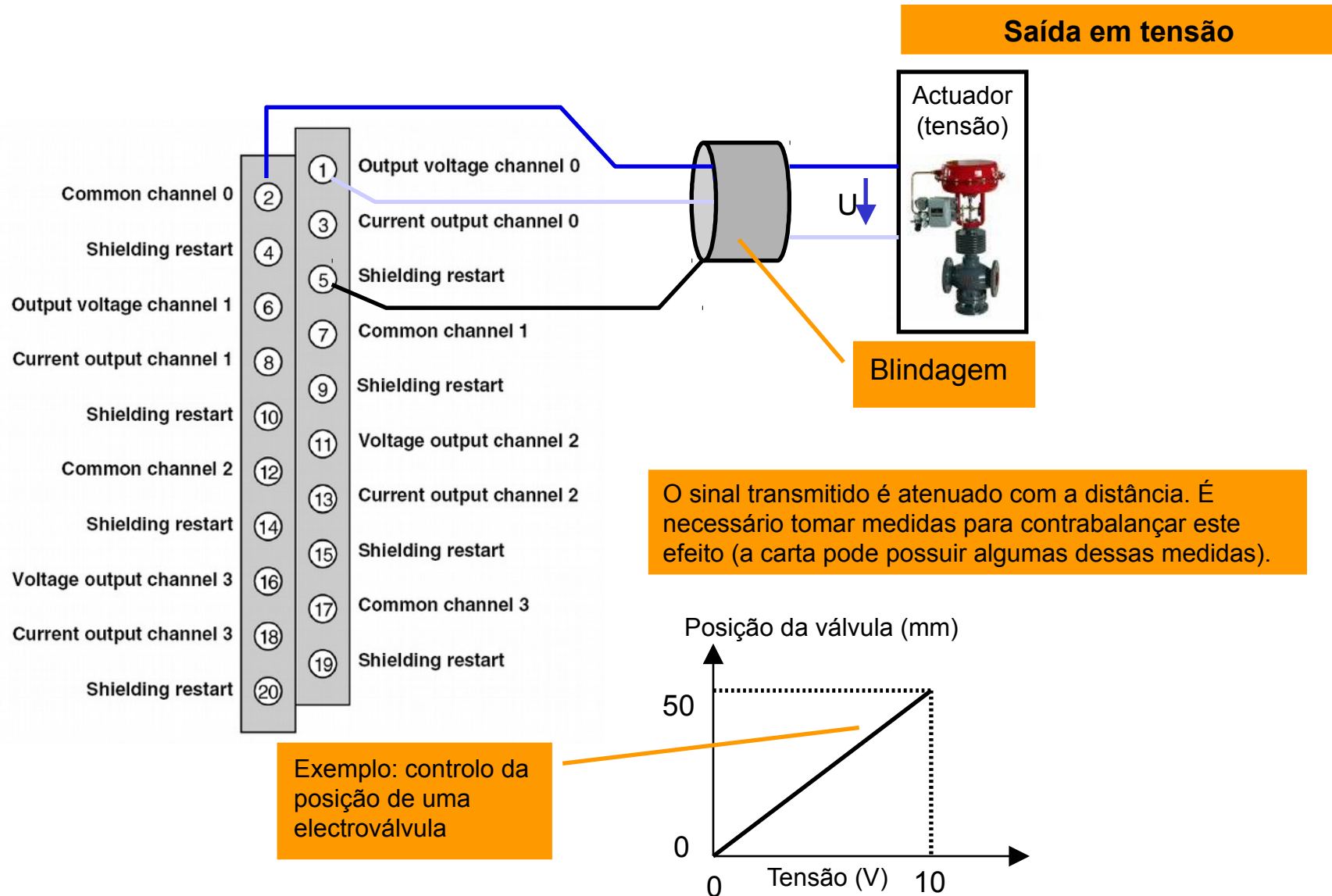
- Conversão de dados digitais (memória do autómato) em sinais analógicos
- Número de canais (pontos de E/S) variável: 2, 4 ou 8
- Resoluções : 10, 12 e 16 bits.
- Saídas em tensão (+/- 5/10V) ou corrente (0/4-20mA)
- Protecção das saídas: curto-circuito, *fallback*
- Diagnóstico das entradas e do módulo
- A configuração (ex. tipos de saídas) e parametrização (ex. escalas) do módulo é efectuada por SW



Exemplo : esquema interno do módulo

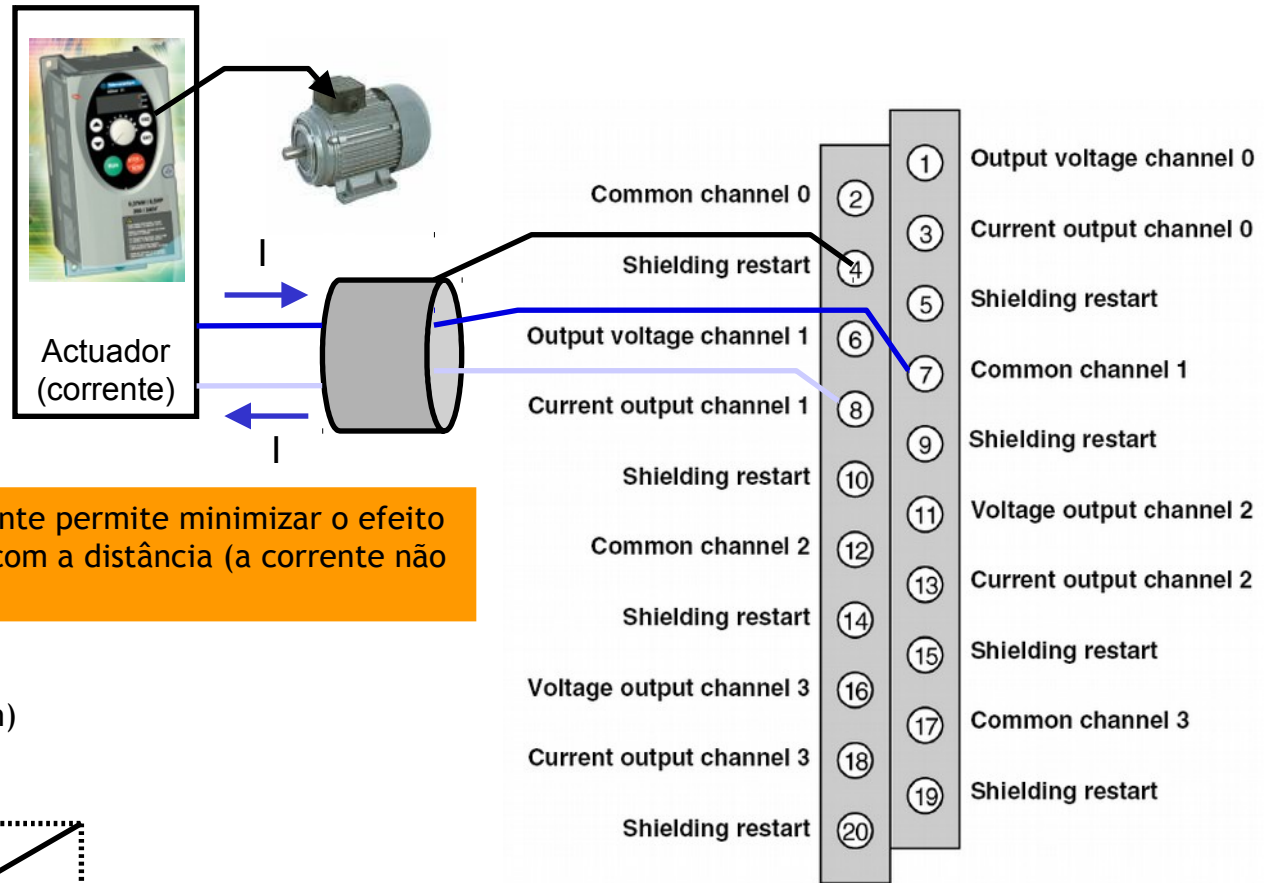


Exemplo de ligações (1)

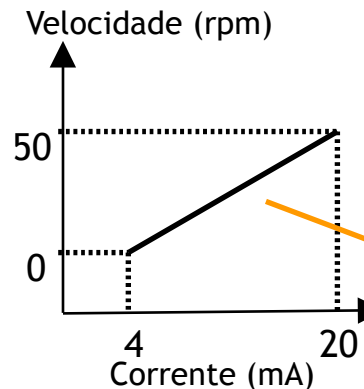


Exemplo de ligações (2)

Saída em corrente



A transmissão em corrente permite minimizar o efeito da atenuação do sinal com a distância (a corrente não depende da distância)



Exemplo: controlo velocidade de um motor

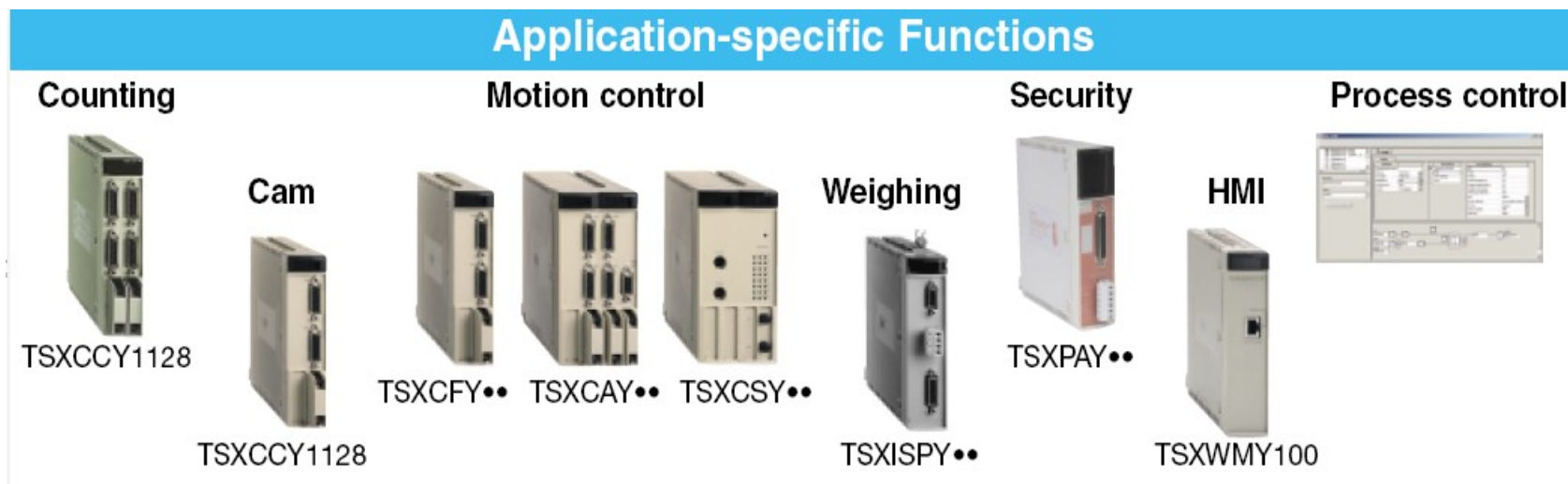
Características eléctricas das saídas: exemplo

Characteristics of analog output modules

Type of output module		TSX ASY 410	TSX ASY 800
Number of channels		4	8
Output range		± 10 V, 0...20 mA and 4...20 mA, output supplied by PLC (or 24 V SELV external on TSX ASY 800 , see page 8/31)	
Analog/digital conversion		11 bits + sign	13 bits + sign (voltage), 13 bits current
Conversion time	ms	2,5	5
Maximum resolution		Voltage output 5.12 mV (5), current output 10.25 μ A (6)	Voltage output 1.28 mV, current output 2.56 μ A
Output load		Voltage output, impedance > 1 k Ω , load < 0.1 μ F, current output, impedance < 600 Ω , load < 300 μ H	
Measurement error as a % of FS			
Voltage output, FS = 10 V	%FS	0.45 to 25 °C, 0.75 from 0 to 60 °C	± 0.14 to 25 °C, ± 0.28 from 0 to 60 °C
Current output, FS = 20 mA	%FS	0.52 to 25 °C, 0.98 from 0 to 60 °C	± 0.21 to 25 °C, ± 0.52 from 0 to 60 °C
Isolation between channels and bus	V rms	1500	1000
Isolation between channels and earth		--- 500 V	1000 V rms
Isolation between channels	V rms	1500	Common point
Type of protection		Short-circuits and overload	

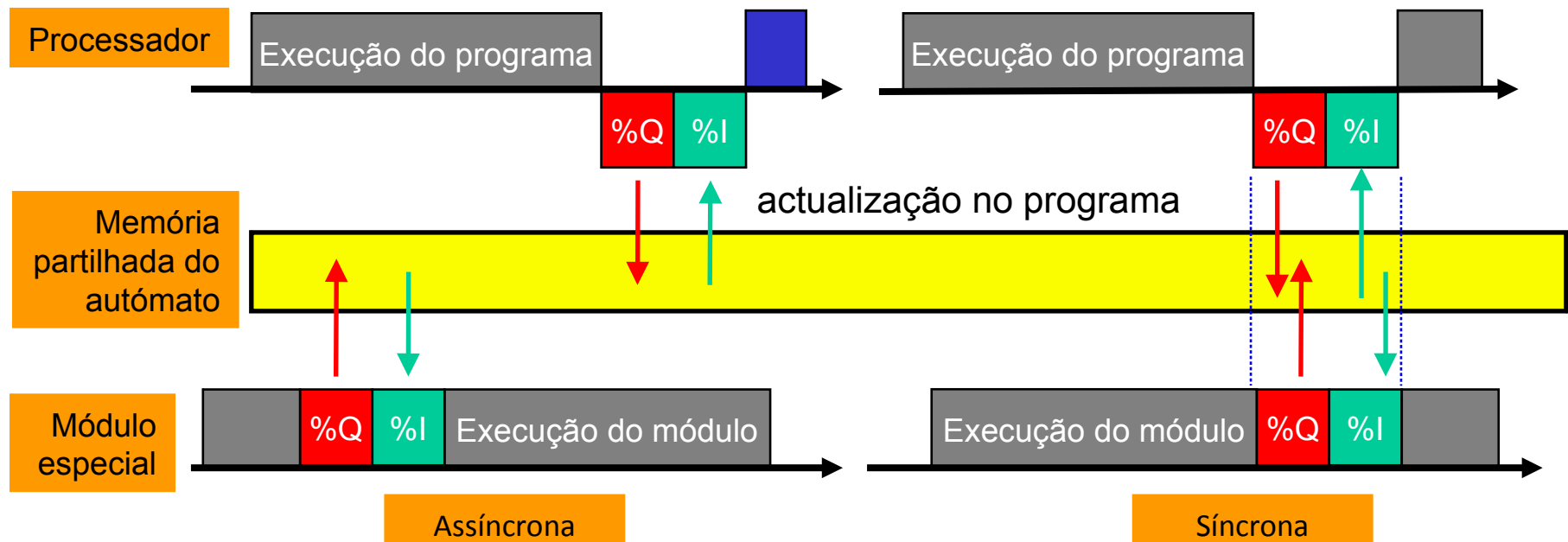
Módulos especiais

- Disponibilizam à aplicação um conjunto de funções pré-definidas e configuráveis
- O objectivo é “aliviar” a aplicação (no processador) de executar funções muito complexas que, geralmente, envolvem cálculos complexos e muito demorados.
- Tipos: Contagem, Pesagem, Accionamentos, Lógica difusa, Controlo de processos (PID), Interface homem-máquina



Modelo de execução dos módulos especiais

- O módulo executa uma aplicação “paralelo” com a aplicação que é executada no processador.
- Na fase de refrescamento dos I/Os, dados e comandos são trocados entre ambos os módulos através de uma zona de memória partilhada (com acesso mutuamente exclusivo).
- A troca de dados na zona de memória partilhada pode ser:
 - Assíncrona : a execução das aplicações não estão sincronizadas;
 - Síncrona: a execução das aplicações estão sincronizadas.



Contagem de impulsos

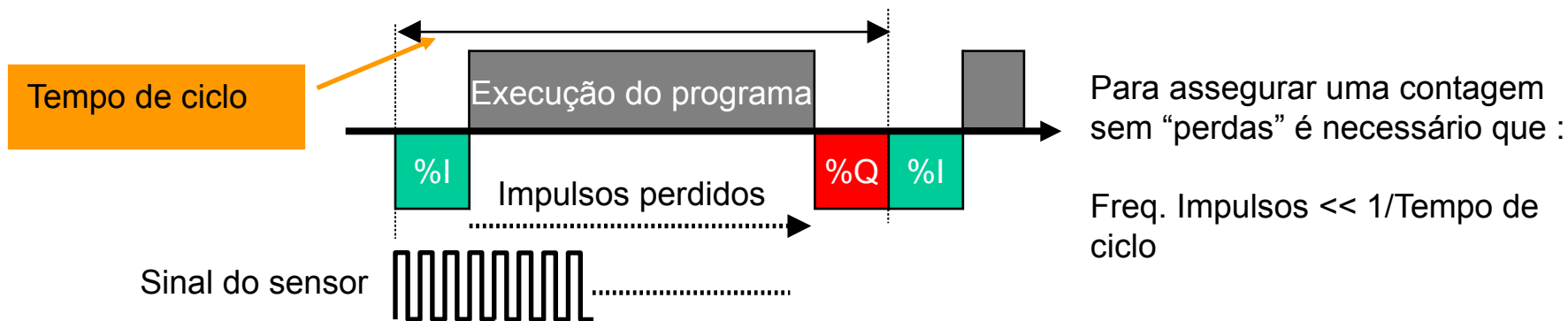
- Existem muitas aplicações industriais em que é necessário realizar operações de contagem de média-alta frequência:

ex. controlo da velocidade de um motor



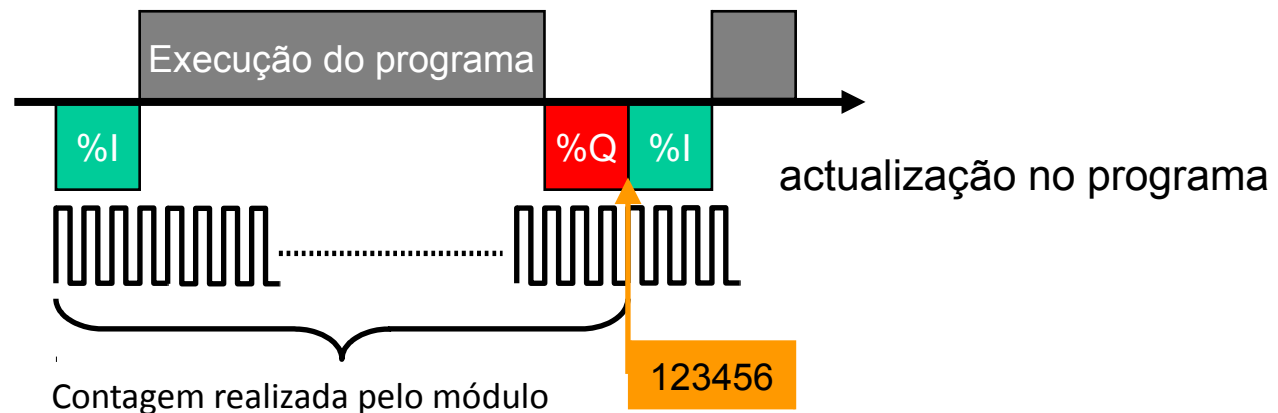
ex. contagem de garrafas

- As entradas digitais (standard) têm limitações no que diz respeito à sua utilização como entradas de contagem:
 - Ex. Tempo resposta $\sim 10\text{ms}$ \Rightarrow frequência máxima $\sim 50\text{ Hz}$ ($=100/2$)
- Além disso, devido ao carácter cíclico da execução do programa do autómato, entre aquisições consecutivas das entradas os impulsos não são contados pelo programa que é executado:



Módulos de contagem

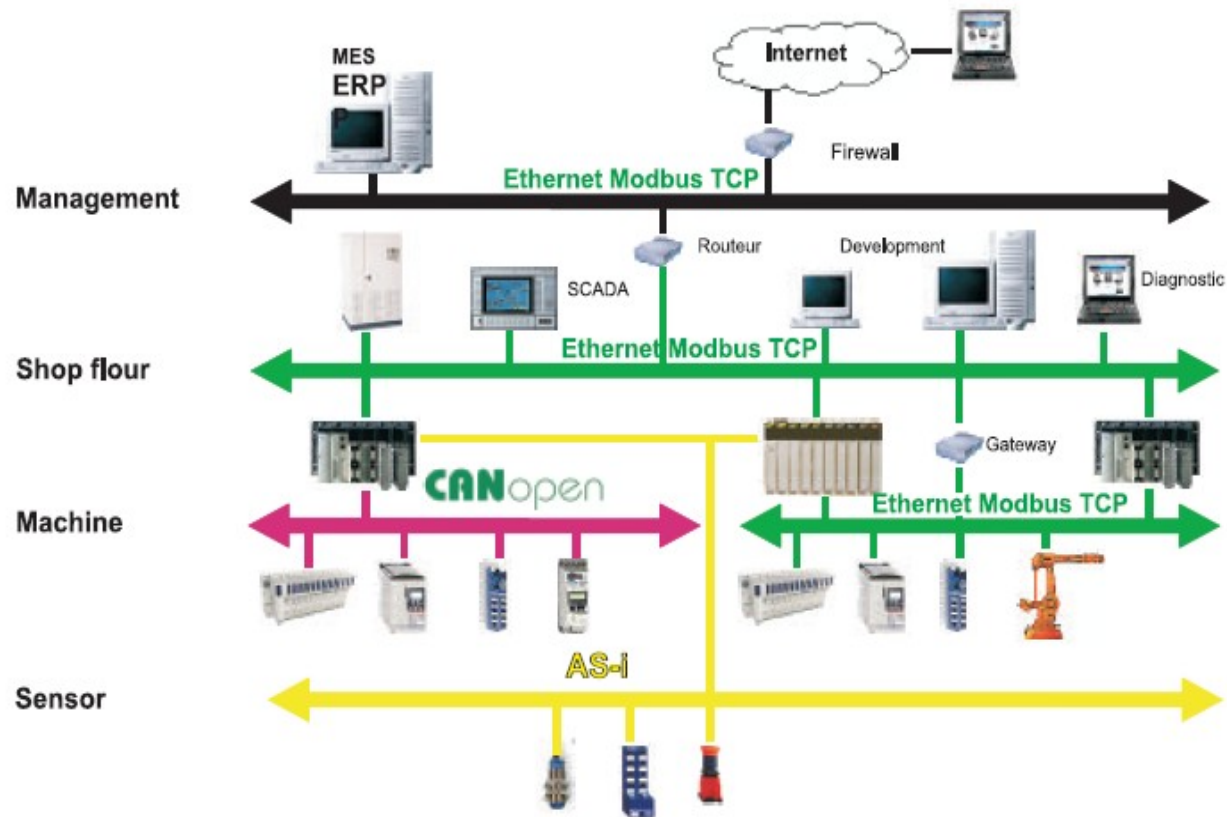
- A utilização de módulos de contagem resolve este problema:
 - Funcionam “em paralelo” com o processador, “enviando” o valor da contagem no momento da aquisição das entradas (i.e. não há perda de ‘informação’ durante a execução do programa no CPU)



- Possuem entradas adaptadas aos sensores mais utilizados nestas áreas:
 - Codificadores incrementais
 - Codificadores absolutos
- Vários tipos de contagem: up, down, up-down

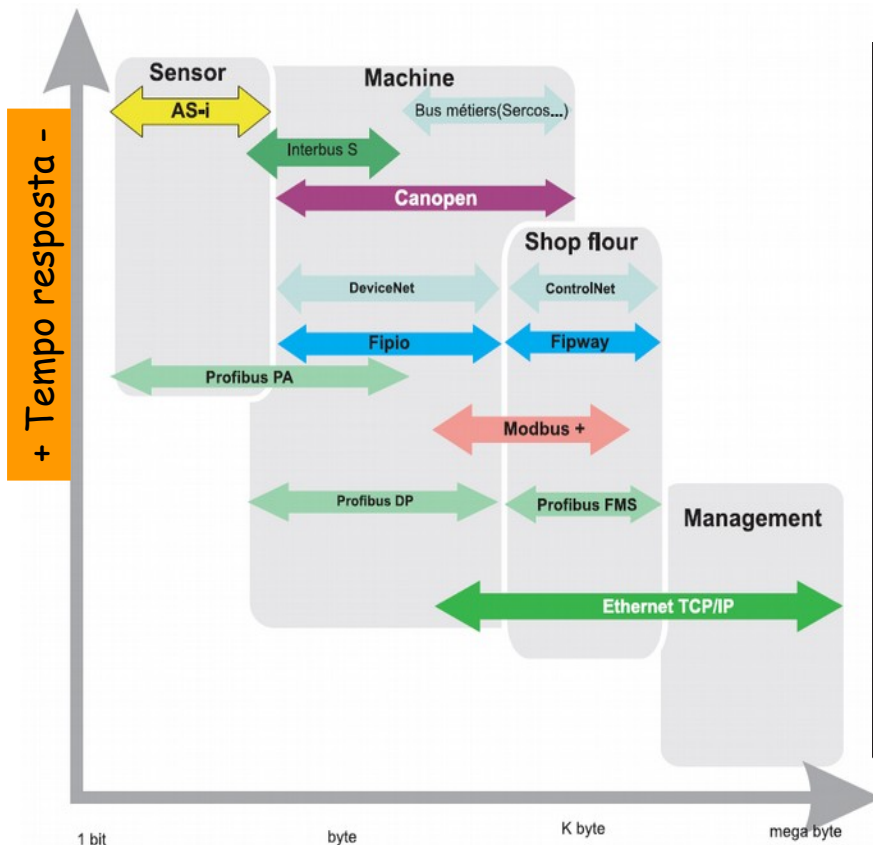
Módulos de comunicações

- Permitem que o autómato possa trocar dados (informações / comandos) com outros equipamentos:
 - Impressoras - Outros autómatos - Scadas
 - Leitores de códigos de barras - PCs - HMI
 - Variadores de velocidade - Módulos de I/Os - Robots
 - Etc.



Módulos de comunicações : soluções

- Existe uma grande variedade de soluções, cada uma adaptada aos níveis de controlo, aos requisitos dos fluxos de dados existentes e aos tipos de equipamentos



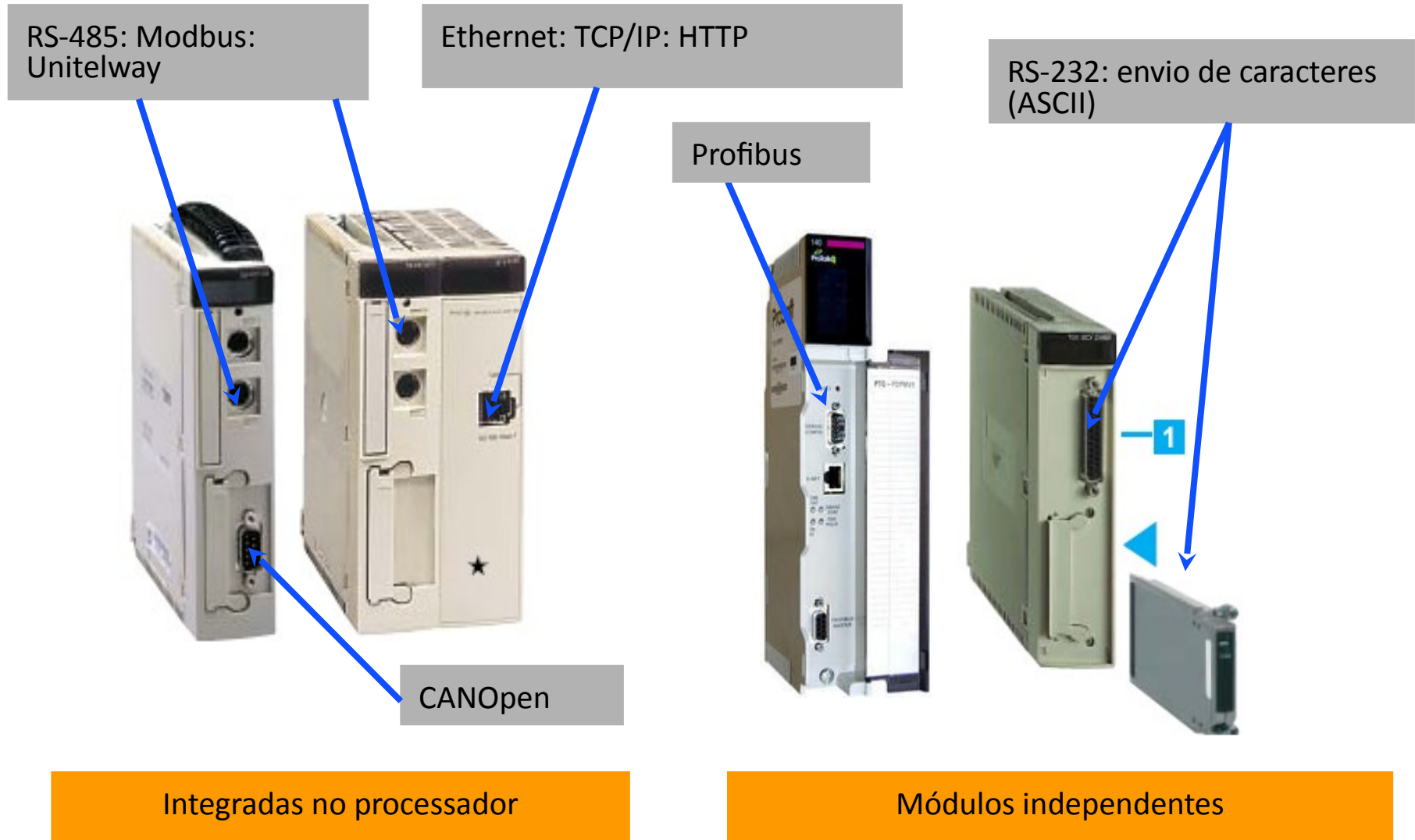
Level	Requirement	Volume to data to transmit	Response time	Distance
Management	Data exchange. Computer security . Standards between software packages.	Files Mbits	1 min	World
Shop floor	Synchronisation of PLC's in the same data exchange automation cell in client/server mode with the control tools (HMI, supervision). Real-time performances.	Data Kbits	50-500 ms	2-40 Km
Machine	Distributed architecture. Embedded functions and exchange. Transparency. Topology and connection costs.	Data Kbits	5-100 ms (PLC cycle)	10 m to 1K m
Sensor	Simplification of distribution wiring for power supply to sensors and actuators. Optimised wiring costs.	Data Bits		1- 100m

- Volume de dados +

Módulos de comunicações : características gerais

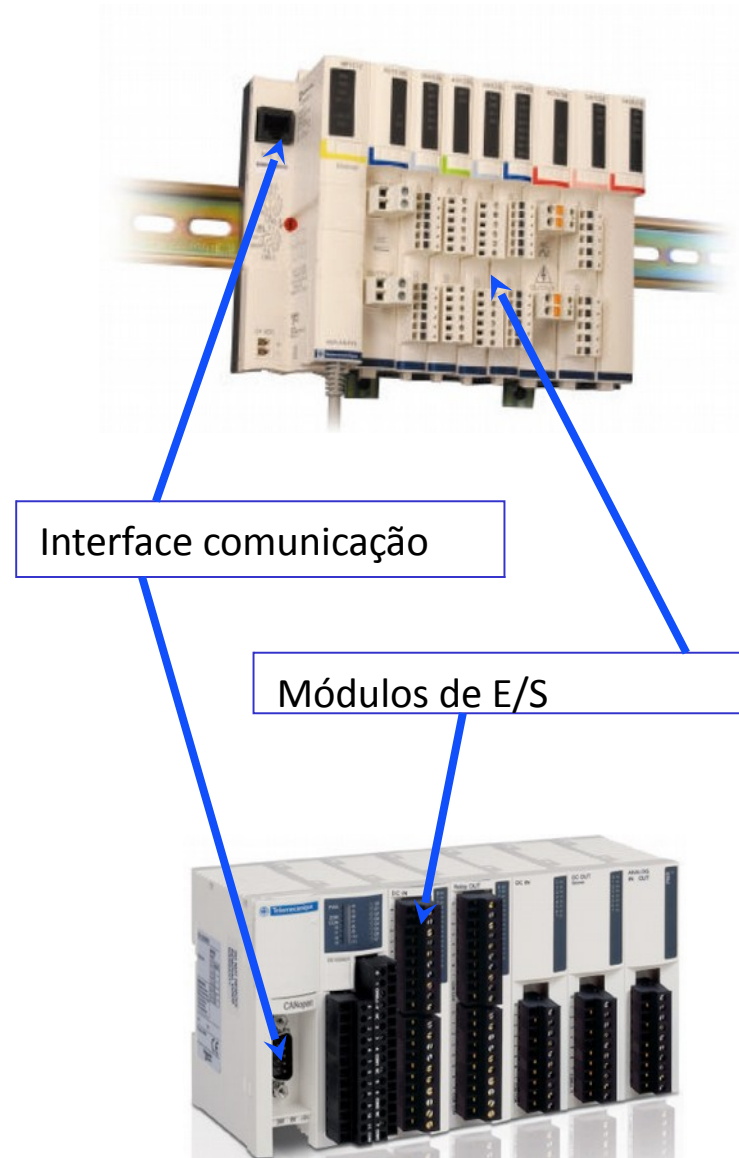
- Soluções abertas (não dominadas por um fabricante) e fechadas (dominadas por um fabricante)
- Funções de comunicação de 2 tipos:
 - Troca periódica de dados de I/Os entre o autómato e os restantes equipamentos utilizada para actualizar a imagem do processo;
 - Troca esporádica de dados: configuração, alarmes, etc.
- A execução dos serviços de comunicação (execução do protocolo) é uma tarefa realizada pelo módulo, sem intervenção (processamento) do processador. A troca de dados entre ambos é realizada numa zona de memória partilhada por ambas as cartas (imagem do processo): processador e módulo de comunicações.
- Interfaces mais usuais: RS-232, RS-485, Ethernet
- Protocolos mais usuais: Modbus, Profibus, DeviceNet, TCP/IP, HTTP

Módulos de comunicações: exemplos

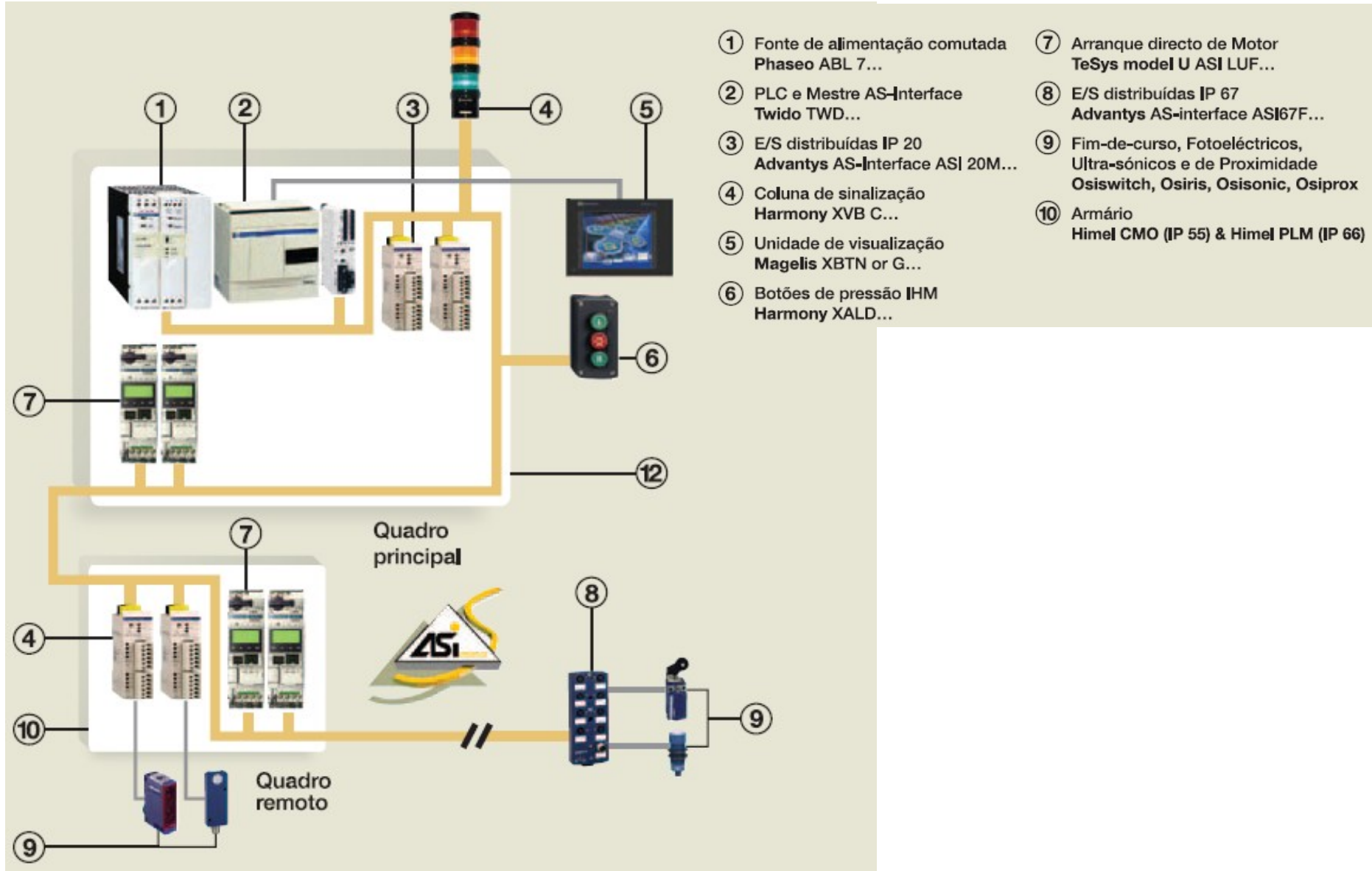


Módulos de E/S distribuídas

- Permitem deslocalizar os módulos de interface com os E/S do autómato para uma localização mais próxima dos sensores & actuadores
- Redução do custo da cablagem
- Diagnóstico e detecção falhas, isolamento dos equipamentos em caso de avaria
- A interligação ao autómato é realizada através de uma rede proprietária, ou de um rede aberta (DeviceNet, Profibus, Ethernet, etc.)
- Funcionalidades semelhantes aos módulos locais do autómato: E/S digitais, analógicas, contagem, etc.
- Acesso ao dados mais lento (ao nível da aplicação que é executada no autómato) em comparação com os módulos locais.
- Apresentam uma boa relação custo/função
- Algumas soluções disponibilizam a possibilidade de aplicações muito simples serem executadas nos módulos.

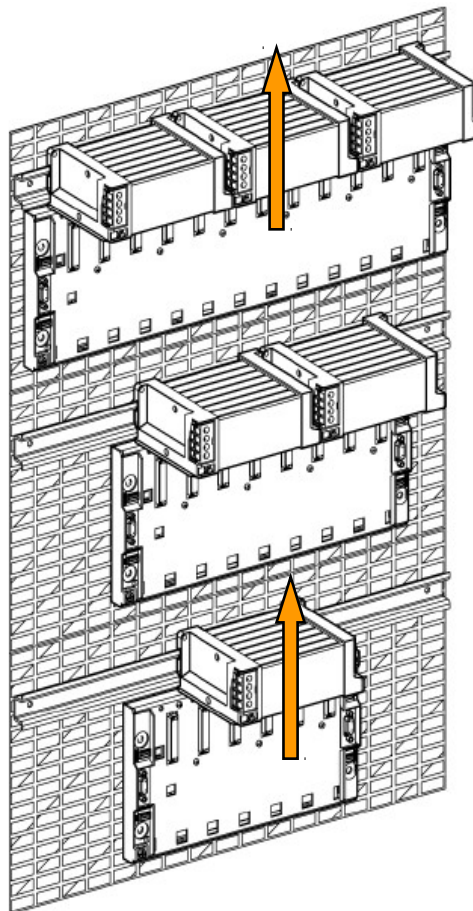


Exemplo de ligação

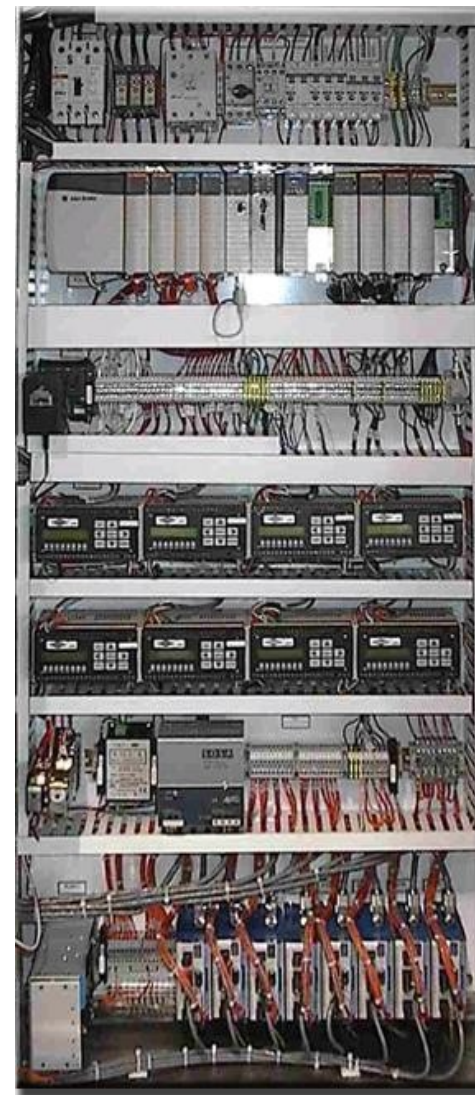


Módulos de ventilação

- Os autómatos são ventilados através de convecção natural
- No entanto podem existir situações em que é necessário introduzir ventilação forçada:
- Armários fechados
- Temperatura ambiente elevada
- Em situações extremas é comum utilizar-se sistemas de ar-condicionado



Armário de equipamentos

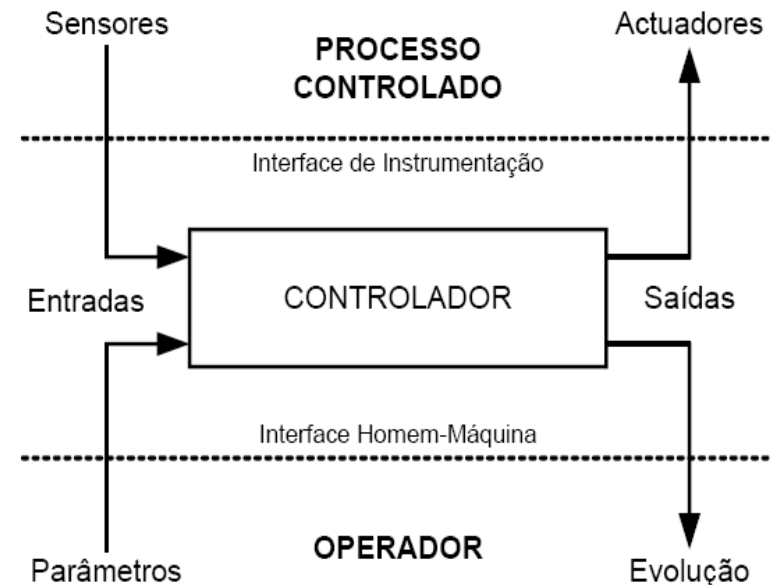


Arquitectura de software

(com ênfase nas características dos autómatos Premium da Schneider)

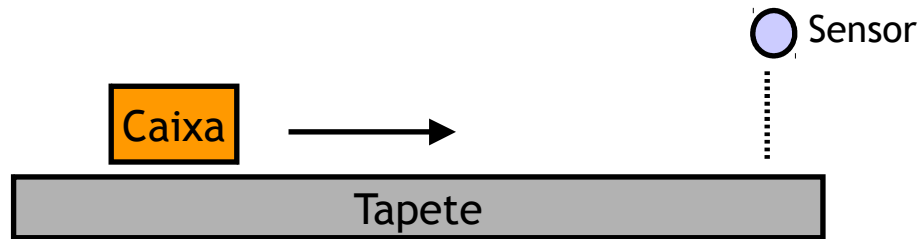
Modelos de execução dos algoritmos de controlo

- O processo controlado gera eventos sobre os quais o algoritmo de controlo reage de forma apropriada;
- Os eventos ocorrem em instantes não-previsíveis, e existe concorrência entre eventos (i.e podem ocorrer “simultaneamente”)
- A execução correcta do algoritmo de controlo tem que ser vista de 2 perspectivas:
 - **Lógica:** se produz os resultados correctos para as entradas existentes
 - **Temporal:** se produz resultados correctos dentro de um intervalo de tempo pré-especificado (resultados “fora-de-tempo” = errados).
- Soluções para a execução do algoritmo de controlo:
 - Guiado por eventos (Event-triggered)
 - Guiado pelo tempo (Time-triggered)



Exemplo: execução lógica vs. temporal

Problema: uma caixa é transportada pelo tapete T. Quando o sensor S for activado, o tapete deve parar em $< 20\text{ms}$, caso contrário a caixa “cai” para fora do tapete.



Programa:

Se $S == \text{True}$ Então

$T := \text{False};$

Senão

$T := \text{True};$

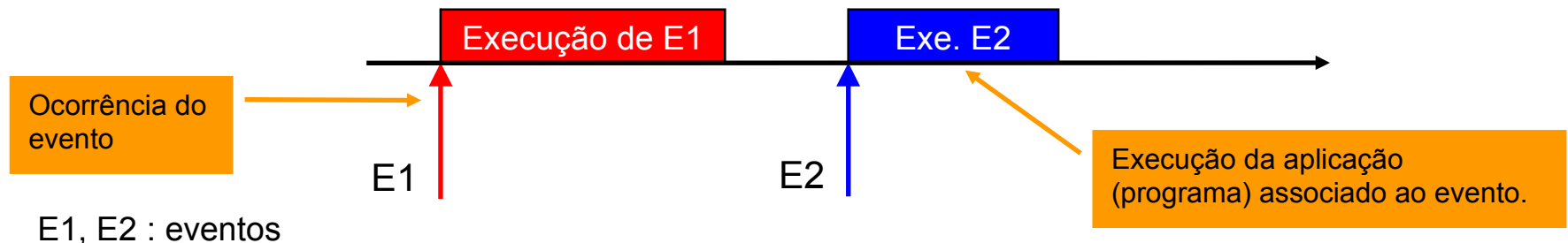
O programa está logicamente correcto.

No entanto, se a desactivação do tapete não for efectuada $< 20\text{ms}$ depois da detecção da caixa pelo sensor, o programa não cumpre as especificações. Logo está temporalmente errado.

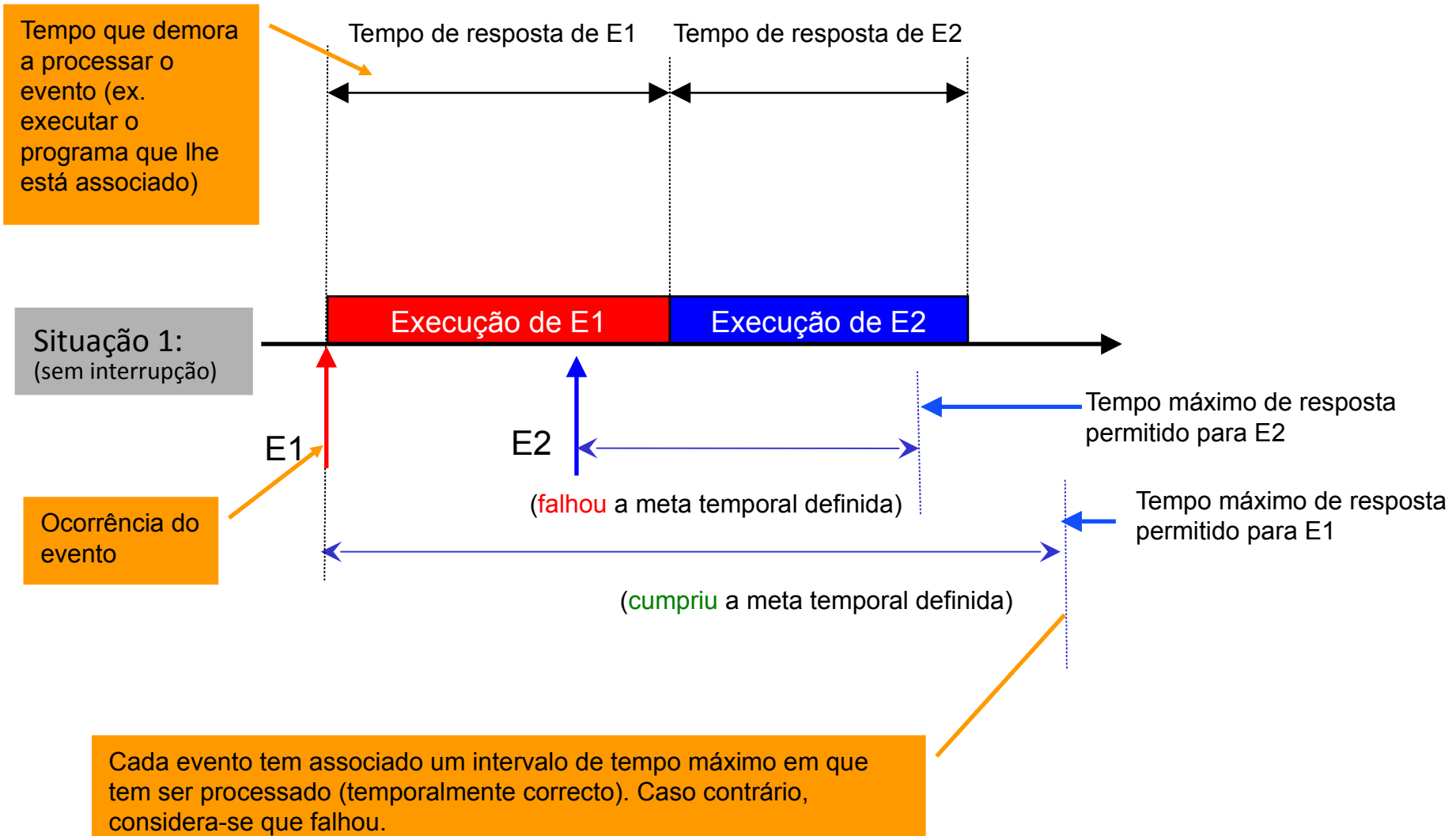
Conclusão: o algoritmo de controlo está a ser executado de forma incorrecta.

Guiado por eventos

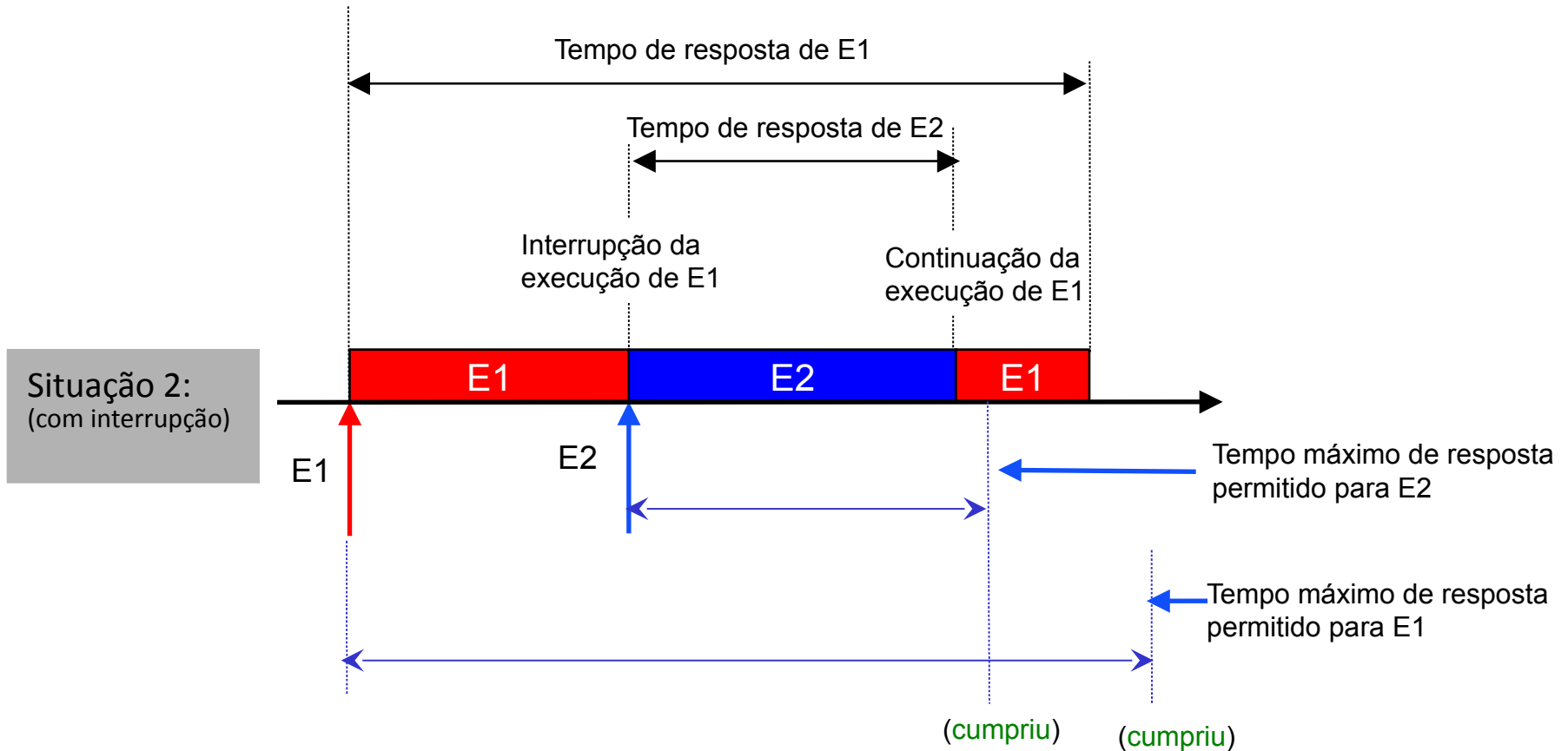
- **Reagir a cada evento quando este ocorre**
- Necessidade de estabelecer prioridades entre os eventos
- É necessário escalonar a execução (da reacção) dos eventos
- Vantagens
 - Flexibilidade: a resposta aos eventos está de acordo com a sua importância
- Desvantagens
 - Implementação complexa
 - Dificuldade em analisar a resposta a eventos “simultâneos”
 - Difícil de prever o tempo de reacção (a resposta) de um evento devido a ser muito variável



Guiado por eventos: exemplo 1



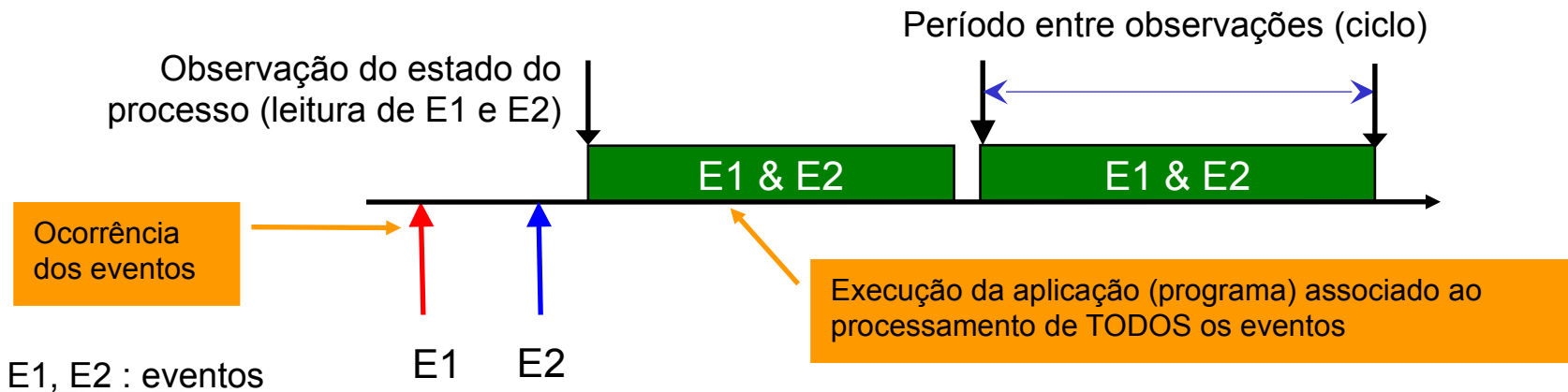
Guiado por eventos: exemplo 2



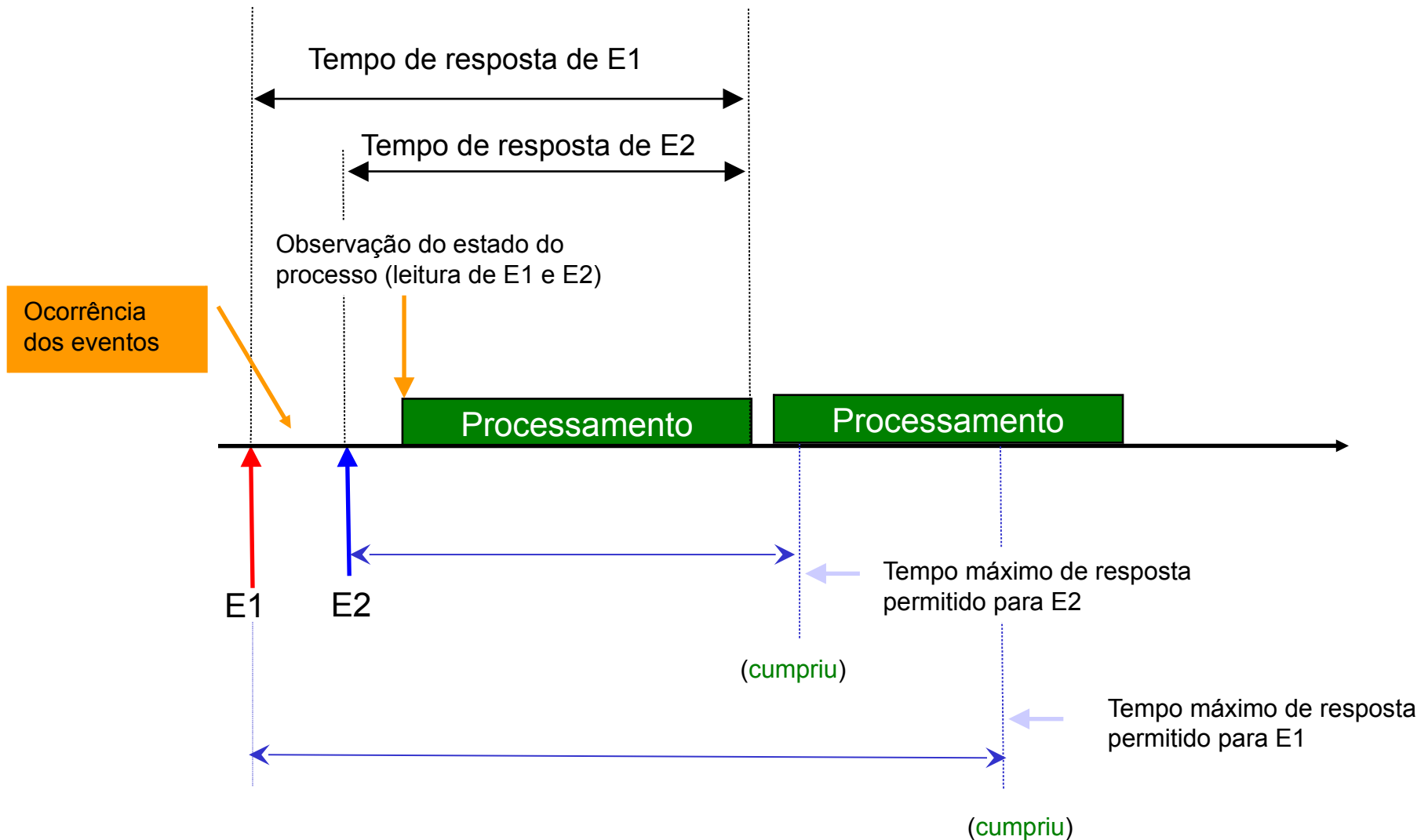
Conclusão: as formas de organizar (escalonar) a execução das respostas aos eventos influenciam o cumprimento, ou não, das suas metas temporais. É difícil definir uma organização que garanta que as metas são sempre cumpridas. É também difícil estimar os piores casos para os tempos de resposta.

Guiado pelo tempo

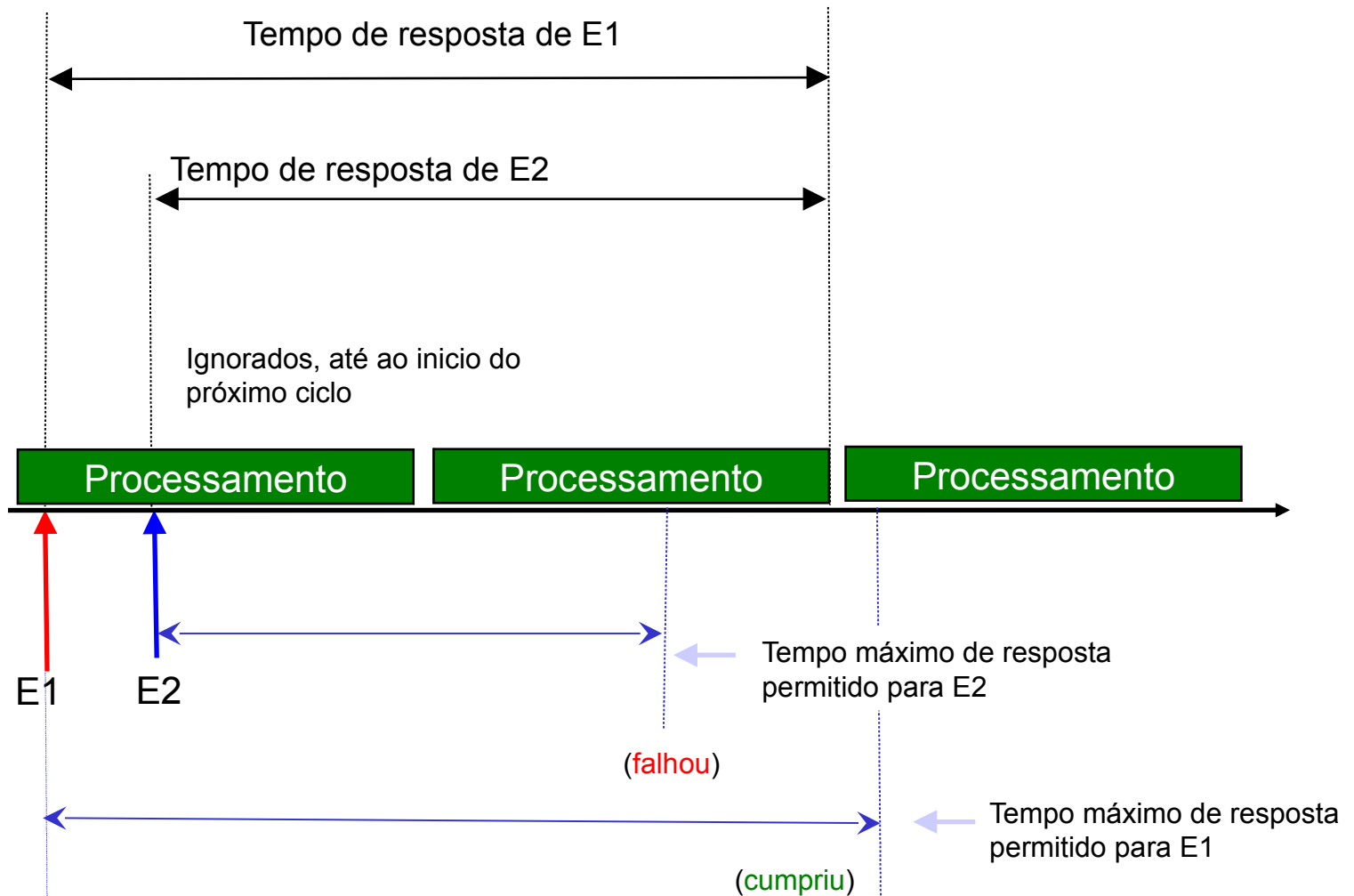
- Observar periodicamente o estado do sistema
- Vantagens
 - Simples de implementar: execução cíclica
 - Comportamento previsível: é (quase sempre) possível estimar o tempo máximo de reacção do sistema. O pior caso do tempo de resposta pode ser obtido com facilidade ($\sim 2 \times$ ciclo)
- Desvantagens
 - Entre 2 observações consecutivas podem perder-se eventos: a qualidade do controlo depende do período da observação
 - Necessidade de ajustar o período às características dos eventos
- Solução empregue pelos autómatos programáveis



Guiado pelo tempo: exemplo 1



Guiado pelo tempo: exemplo 2

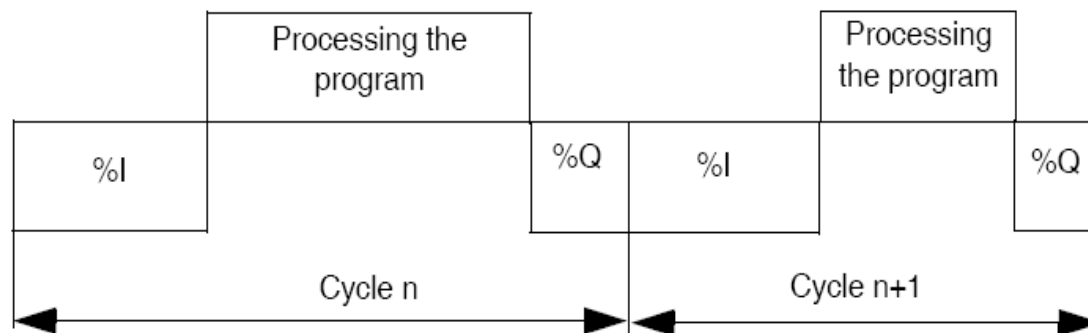
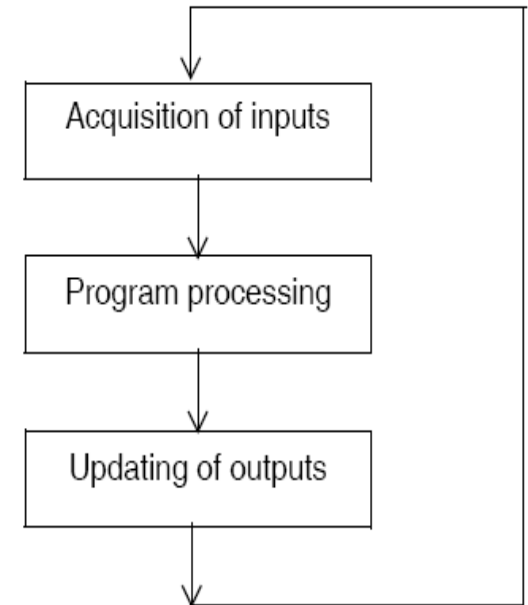


Conclusão: conhecendo os tempos de processamento e os intervalos de ocorrência dos eventos, é simples verificar se as metas temporais são, ou não, cumpridas. É também simples estimar os piores casos para os tempos de resposta.

Modelo básico de execução de um autómato

Modelo cíclico:

1. Aquisição das entradas para uma zona de memória denominada **imagem do processo** (imagem das entradas)
2. Execução do programa (processamento). Qualquer modificação das entradas é, em princípio, ignorada (podem existir variantes em relação a este aspecto). Os valores das saídas são escritas também na **imagem do processo** (imagem das saídas)
3. Actualização das saídas físicas a partir da imagem das saídas



%I Reading of inputs
%Q Writing of outputs

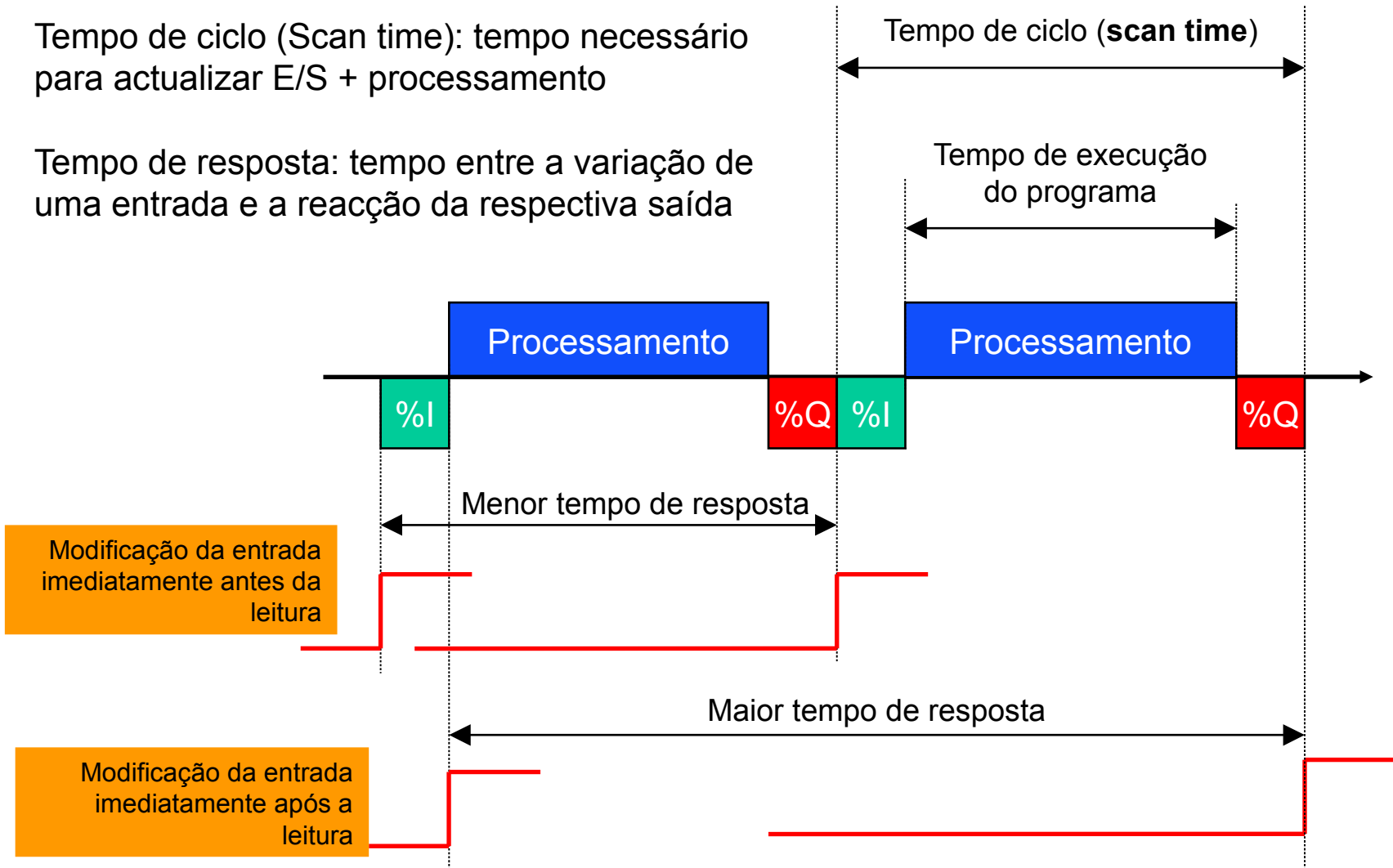
Modelo de execução: algumas notas

- Um evento que ocorre durante a execução de um ciclo é em princípio ignorado. No entanto existem alguns autómatos que permitem processar, no momento, eventos que ocorrem durante a execução de um ciclo.
- O tempo de ciclo limita a dinâmica de controlo do processo. Não é possível considerar eventos cuja periodicidade é inferior ao tempo de ciclo (ver excepções adiante).
- É possível prever os melhores e piores casos para o tempo de ciclo (ver adiante).
- A actualização de todas as saídas no mesmo instante permite assegurar a sua consistência temporal e evita comportamentos imprevisíveis.
- O tempo que demora a ler a imagem das entradas é função do número de entradas (+entradas => +tempo) e do tipo (ex. analógicas > digitais). Um raciocínio semelhante também se aplica às saídas.
- O tempo de execução do programa é função do número de instruções (+instruções => +tempo), do tipo de instruções (ex. booleanas < aritméticas) e da estrutura do programa (ex. ciclos, saltos)
- Durante a actualização da imagem do processo (E/S) são também realizadas outras operações internas ao funcionamento do autómato. Estas operações podem ser realizadas em paralelo com a actualização ou em série. Neste último caso é “consumido” tempo adicional.

Tempo de ciclo vs. tempo de resposta

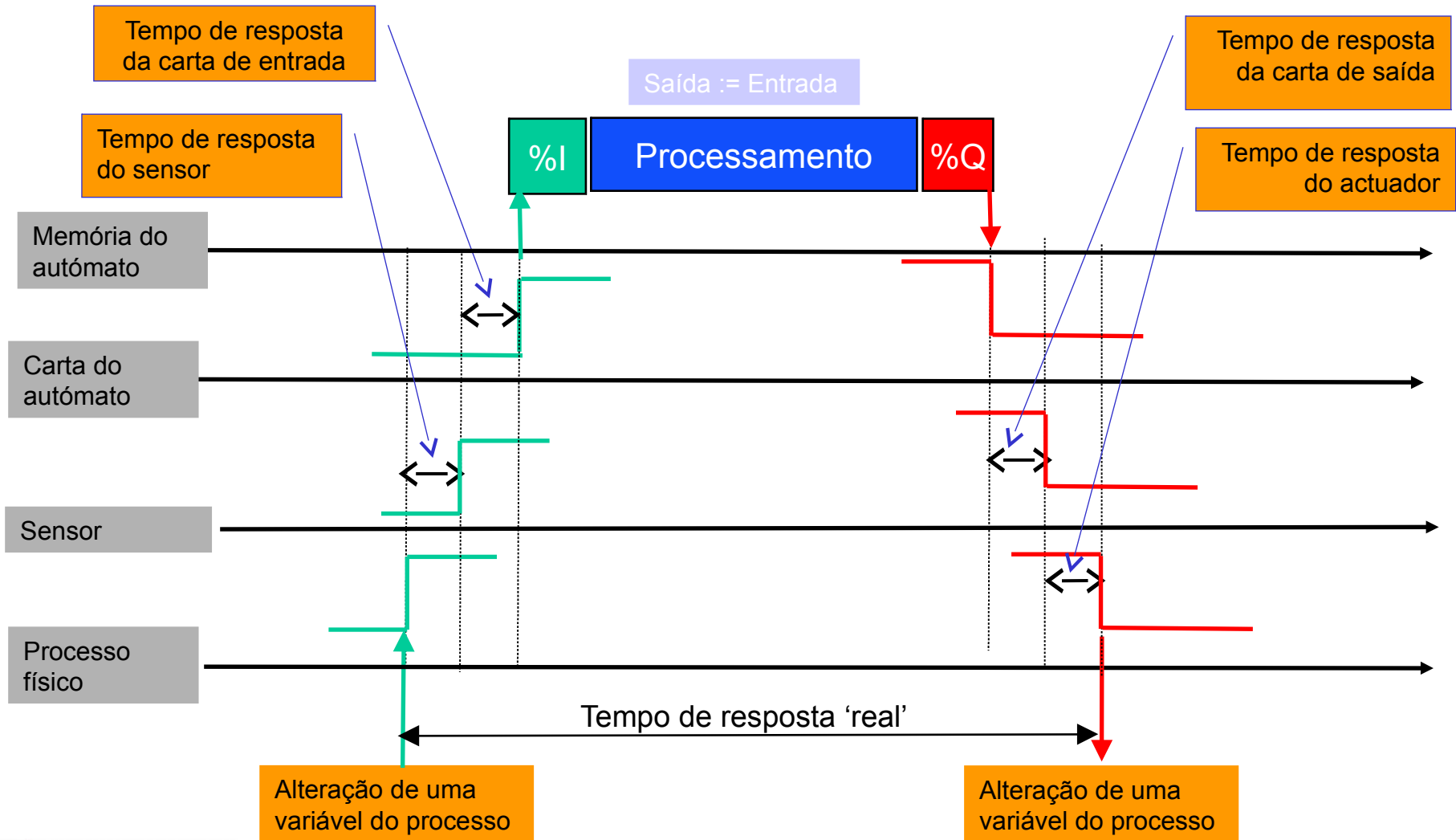
Tempo de ciclo (Scan time): tempo necessário para actualizar E/S + processamento

Tempo de resposta: tempo entre a variação de uma entrada e a reacção da respectiva saída

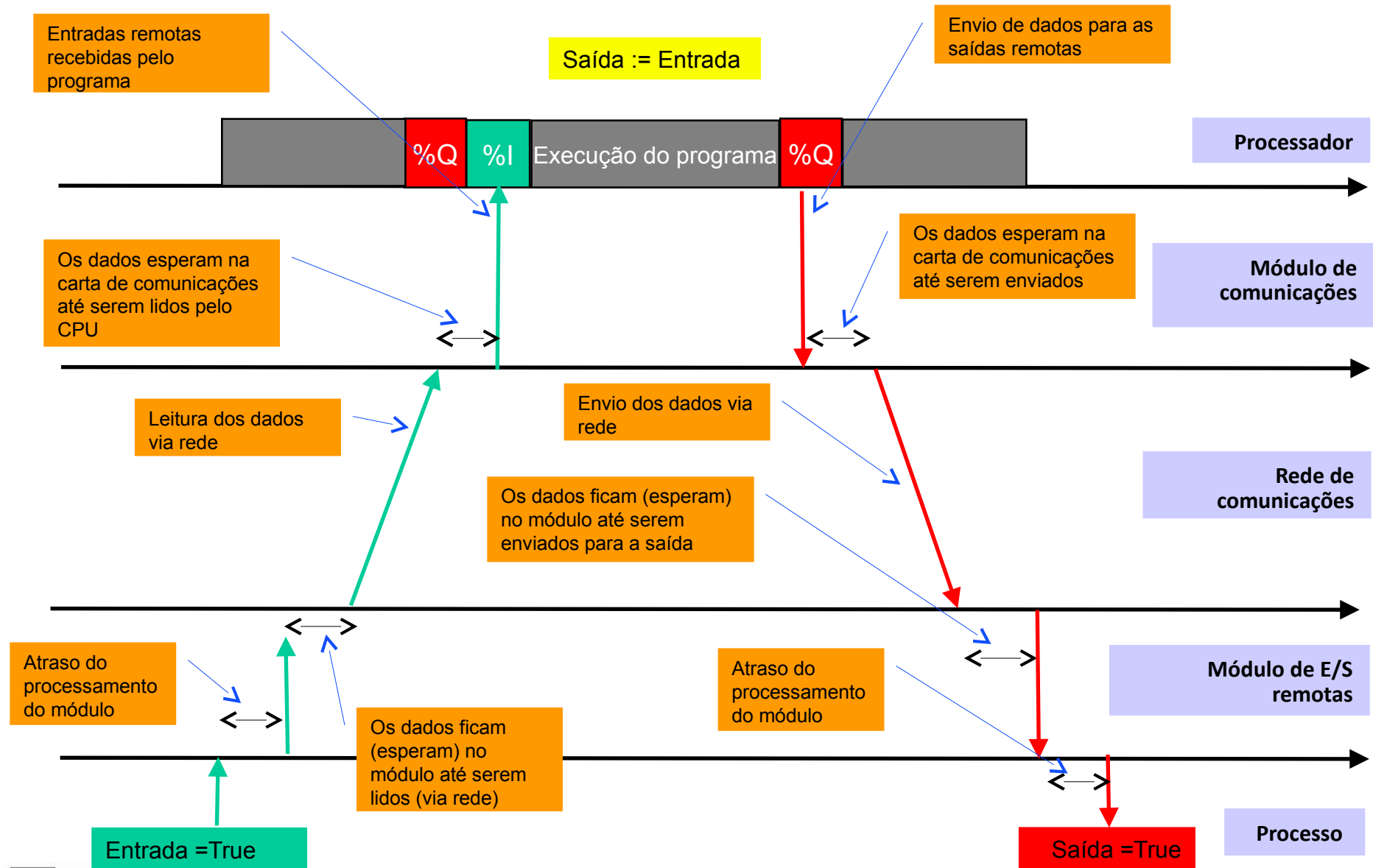


Tempo de resposta: algumas notas

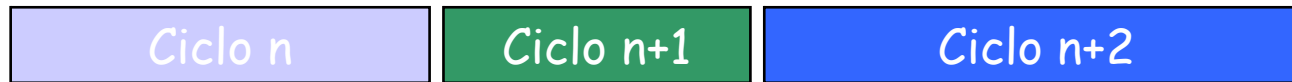
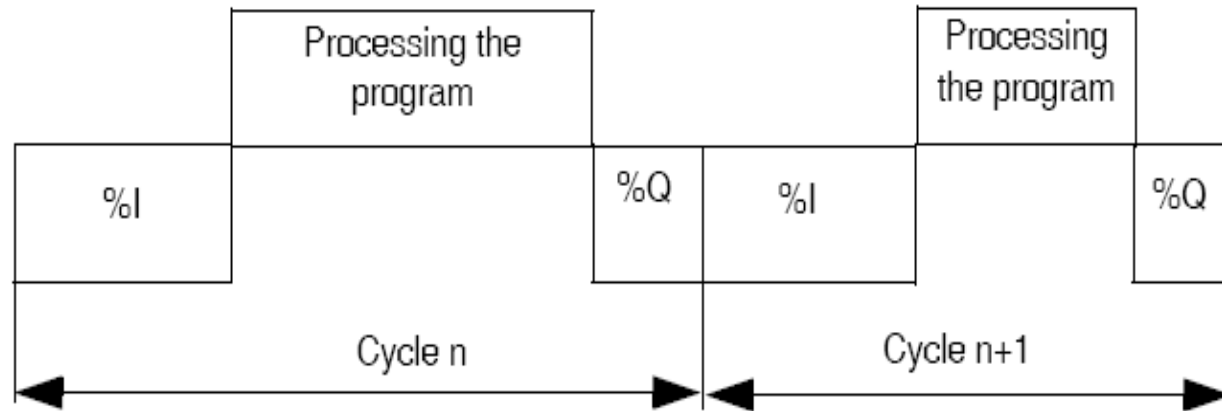
- O tempo de resposta ‘real’ (i.e. medido na interface física com o processo) tem que incluir os tempos de resposta das cartas de E/S envolvidas e também dos sensores e actuadores



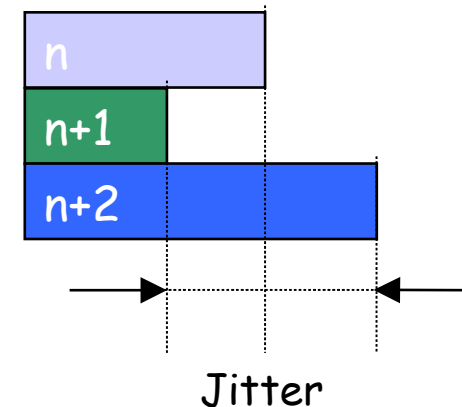
Tempo de resposta: com E/S distribuídas



Funcionamento cíclico



- Se o tempo de ciclo for variável (ex. devido à estrutura do programa) a leitura e a actualização das saídas é afectada por um fenómeno denominado *jitter* (variação do período).
- Este problema torna-se particularmente grave no controlo de processos, quando o *jitter* > 10% ciclo (ver bibliografia de controlo).

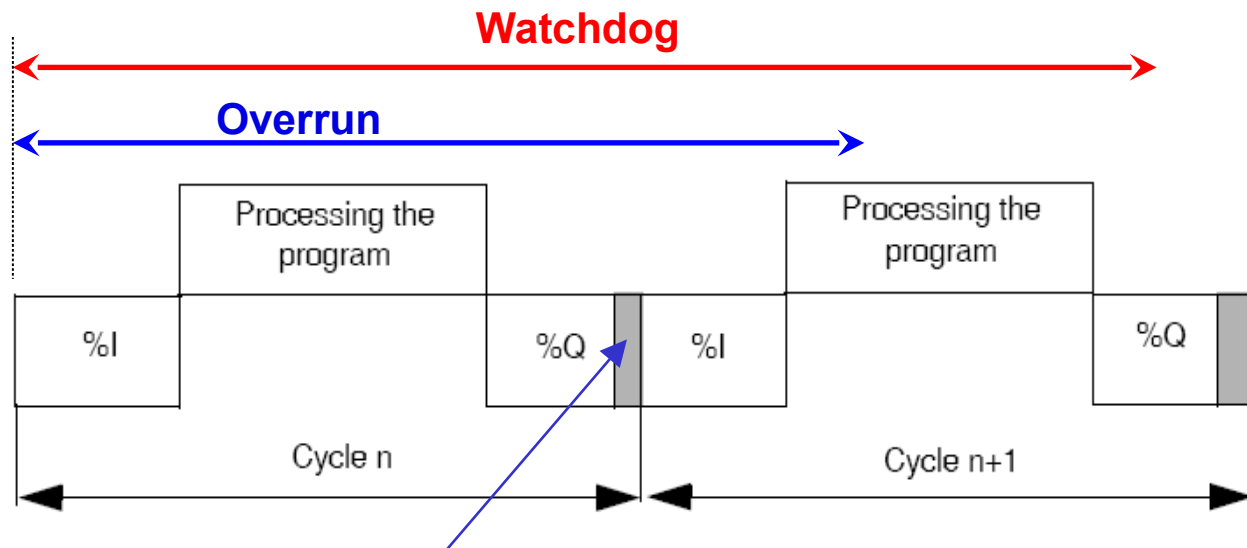


Funcionamento periódico

Utilizado em situações em que o controlo do processo exige que a leitura das entradas e actualização das saídas se realize com um período constante (ex. controlo de velocidade).

Controlo do tempo de ciclo através de 2 temporizadores:

- **Overrun** - permite sinalizar (na memória do autómato) que o tempo de ciclo definido foi excedido. O programa pode depois tomar uma decisão a este respeito.
- **Watchdog** – se o tempo de ciclo exceder o valor do *watchdog* o autómato é colocado me modo de paragem (Stop). Garante um funcionamento seguro.

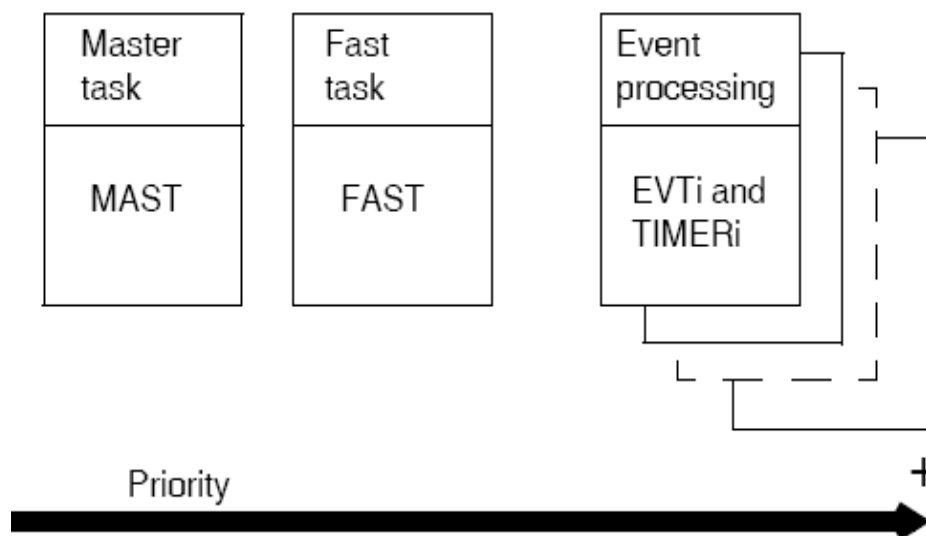


Tempo extra (idle) introduzido para garantir um período constante

Funcionamento multi-tarefa

- De acordo com o IEC 61131, os autómatos Premium suportam a execução de vários tipos de tarefas (Tasks) com características e prioridades diferentes.
 - **Mast Task**: tarefa principal, executada em modo cíclico ou periódico
 - **Fast Task**: utilizada para o processamento de curta duração. Executada de forma periódica. Permite processar eventos com períodos inferiores ao tempo de ciclo.
 - **Event Tasks**: executadas por eventos desencadeados nos canais de E/S e em temporizadores. O tempo de execução deve ser curto. Funciona de forma análoga a uma interrupção.

- As tarefas de maior prioridade interrompem as de menor prioridade
- Este funcionamento representa um compromisso entre a execução orientada por eventos e por tempo



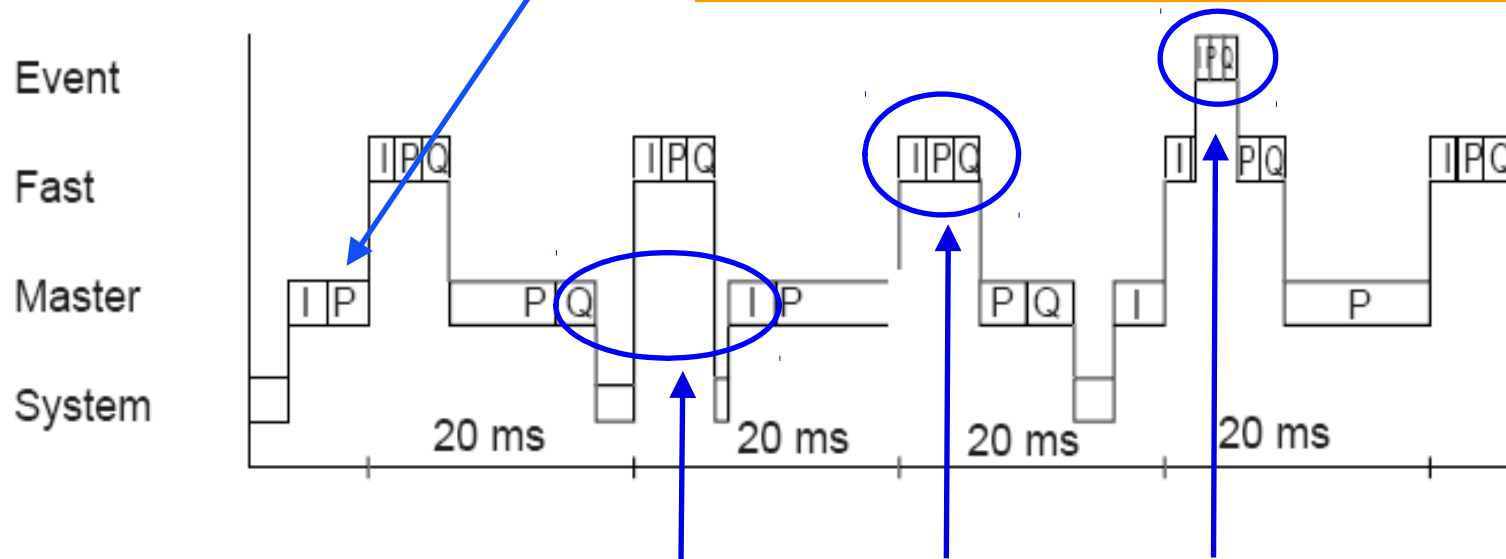
Exemplo de execução (1)

Example of multitask processing

- cyclic master task
- fast task with 20 ms period
- event-triggered task.

A tarefa MAST é interrompida (periodicamente) pela FAST

O ciclo da tarefa Fast permite, se necessário, processar periodicamente eventos num período inferior ao tempo de ciclo (da MAST).



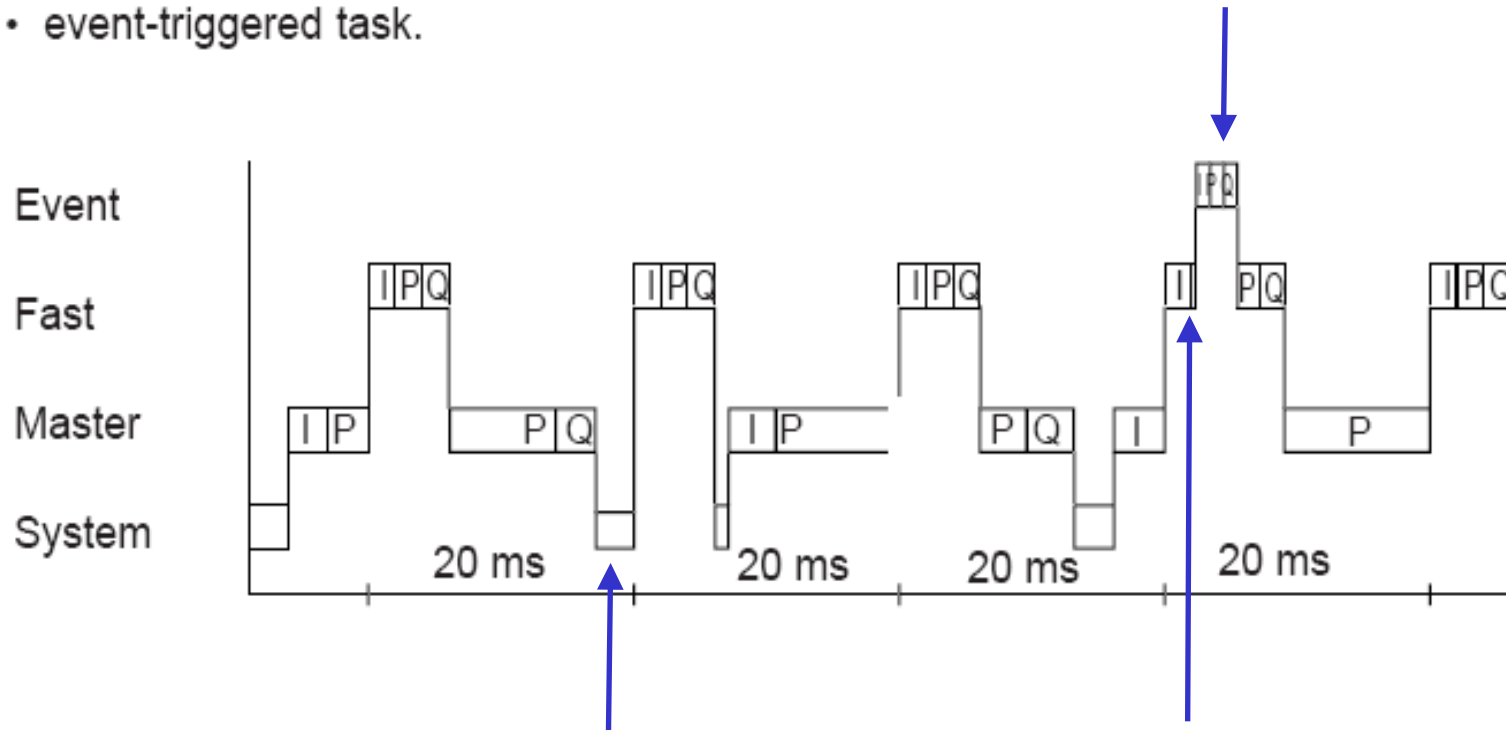
- São apenas adquiridas (I) e actualizadas (Q) as E/S que a tarefa utiliza (o utilizador pode definir quais).
- Minimiza-se assim o tempo gasto nestas operações.

Exemplo de execução (2)

Example of multitask processing

- cyclic master task
- fast task with 20 ms period
- event-triggered task.

As Event Task permitem processar eventos assíncronos (i.e que não são periódicos e que precisam de resposta urgente)



Processamento interno (sistema operativo do autónato)

A tarefa FAST é interrompida pela tarefa de Evento

Exemplo de execução (3)

- As ferramentas de programação do autómato permitem configurar e analisar os tempos de execução das várias tarefas

The screenshot shows a software interface for configuring and monitoring a task. At the top, there are two tabs: "Task" (selected) and "Real-time clock". Below the tabs is a table with the following data:

	Programmed Period	Adjusted Period	Minimum Duration	Current Duration	Maximum Duration	Watchdog	Operating Mode	Cmd	State	Activation Task	Init Duration
MAST		CYCLIC	1	2	7	250	RUN	STOP	E	Disable	Init

Below the table, there are three main sections:

- Events:** Contains a button "Enable or Disable all", "State: D", and "Number: 0".
- Start/Restart:** Contains two buttons: "Warm restart" and "Cold start".
- Output fallback:** Contains two buttons: "Applied Outputs" and "Output Fallback".

At the bottom, there is a "Last Stop" section with a yellow warning icon, the text "quarta-feira, 04 Setembro 1991 22:51:00", and the cause: "Cause: Changeover from RUN to STOP by the terminal or dedicated input".

Tempo de ciclo : algumas notas (1)

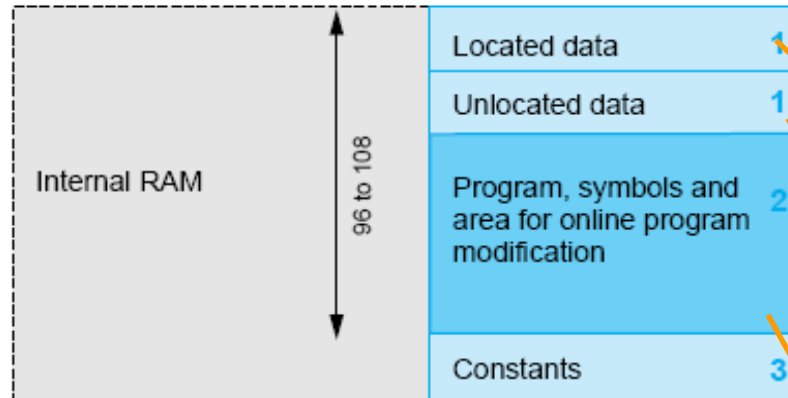
- Considerando apenas a tarefa Mast, não é possível:
 - Ter tempos de resposta inferiores ao tempo de ciclo;
 - Detectar eventos cuja duração é inferior ao tempo de ciclo (ex. impulsos);
- Os tempos de actualização da imagem do processo (I/Q) são função do:
 - número de E/S (aumento c/ o número de E/S)
 - tipo (aumento c/ o volume de dados)
 - Os fabricantes disponibilizam dados sobre estes elementos
- Os valores dos temporizadores são actualizados apenas durante a fase de actualização da imagem do processo, ou seja:
 - Durante a execução de um ciclo de programa o valor do temporizador está “congelado” (tal como os restantes E/S);
 - Não é possível (com temporizadores) medir com rigor tempos da ordem do tempo de ciclo.

Tempo de ciclo : algumas notas (2)

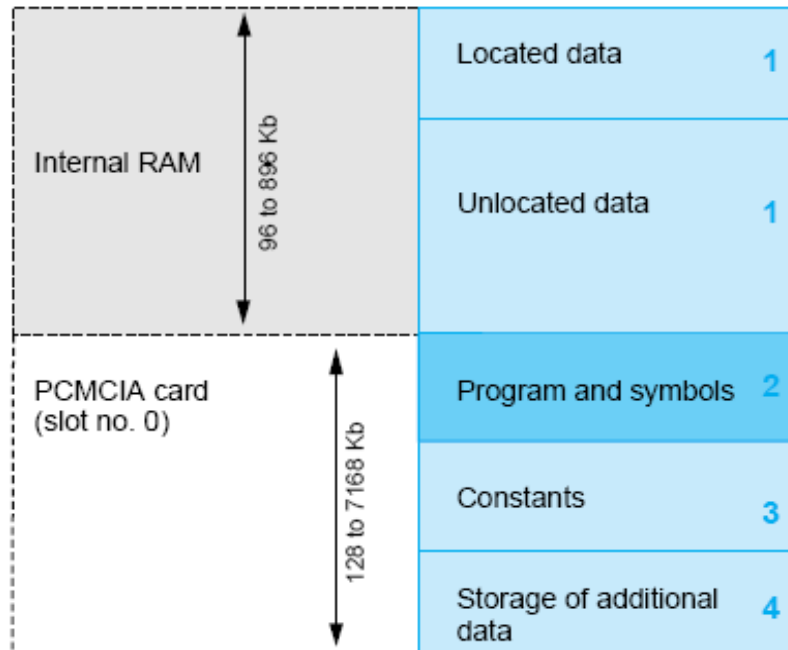
- É necessário ter algum cuidado com as variáveis que são partilhadas entre as várias tarefas.
- Podem existir comportamentos imprevisíveis quando uma tarefa de maior prioridade modifica o valor de uma variável utilizada por uma tarefa de menor prioridade (esta última tarefa pode não se aperceber que a modificação foi realizada, ex. interrupção da tarefa)
- Regra: partilhar um mínimo de variáveis entre tarefas (ideal: nenhum), assegurar-se da forma como o acesso é realizado (só leitura/escrita)

Organização da memória

Processor without PCMCIA memory card



Processor with PCMCIA memory card in slot no. 0



A memória do autômato está localizada no módulo do processador

Variáveis endereçadas em posições de memória específicas (%MW, %I, %Q, etc.)

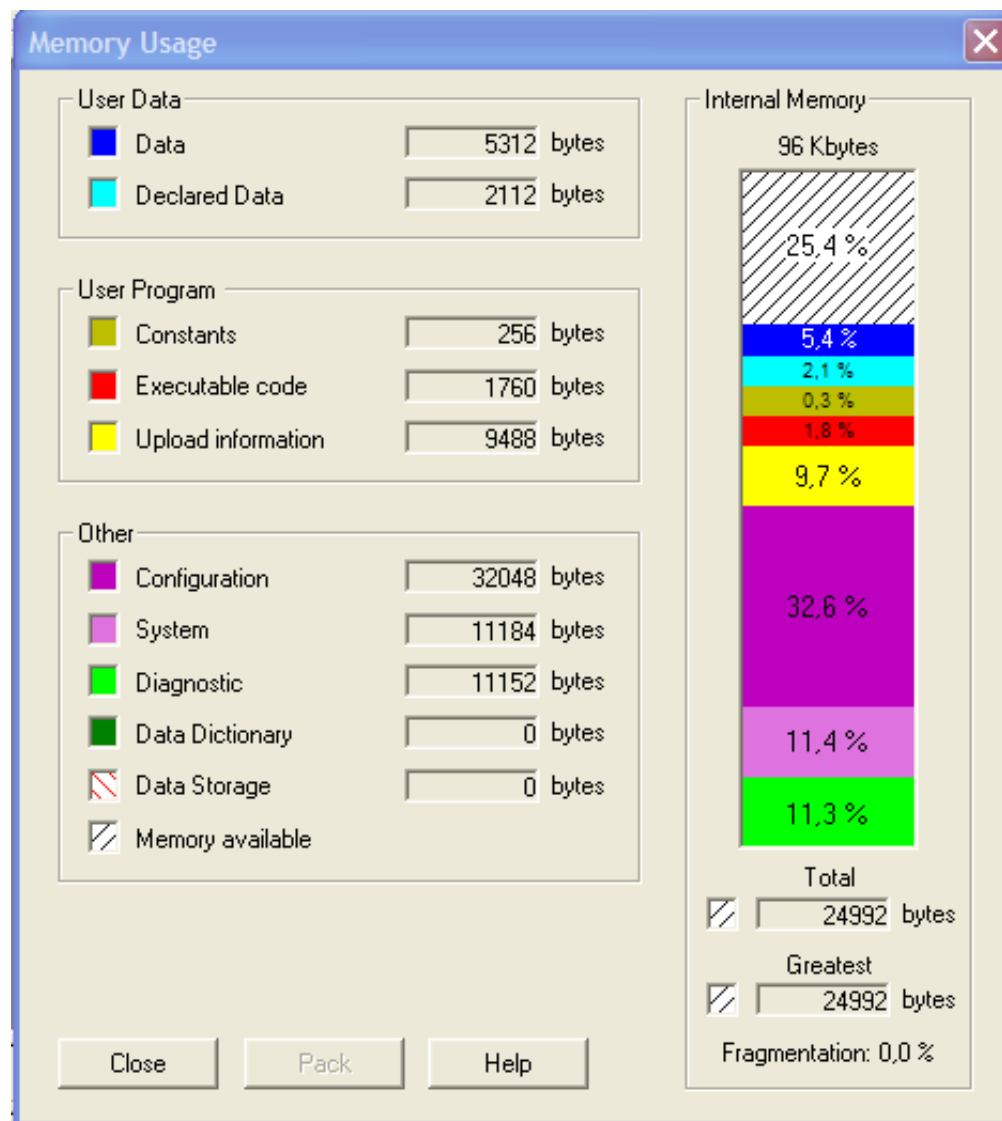
Variáveis definidas por símbolos e FB sem localização específica

Código do programa e símbolos

Podem também armazenar a configuração da aplicação e dados de diagnóstico

Os dados armazenados em RAM são mantidos pela bateria quando não existe alimentação eléctrica.

Exemplo da utilização da memória



Livre

Programa

Configuração

Utilização de memória e tempo de ciclo (1)

- É possível ter uma aproximação do número de instruções de um programa que a memória do autómato pode suportar
- O tempo de ciclo depende do tipo e numero de instruções executadas, bem como das características do processador utilizado

Type of processor			TSX P57 0244M (1)	TSX P57 104M	TSX P57 1634M	TSX P57 154M
Execution time for one instruction	Without PCMCIA card	Boolean	µs	0.19	0.19	
		On word or fixed-point arithmetic	µs	0.25	0.25	
		On floating points	µs	1.75...2.60 (7)	1.75...2.60 (7)	
	With PCMCIA card	Boolean	µs	0.25	0.25	
		On word or fixed-point arithmetic	µs	0.50	0.50	
		On floating points	µs	1.75...2.60 (7)	1.75...2.60 (7)	
No. of Kinstructions executed every ms	Without PCMCIA card	100% Boolean	Kinst/ ms	4.76	4.76	
		65% Boolean and 35% fixed arithmetic	Kinst/ ms	3.71	3.71	
	With PCMCIA card	100% Boolean	Kinst/ ms	3.10	3.10	
		65% Boolean and 35% fixed arithmetic	Kinst/ ms	2.10	2.10	
System overhead	Master task		ms	1.00	1.00	
	Fast task		ms	0.30	0.30	

Sistema operativo do autómato

Utilização de memória e tempo de ciclo (2)

- A linguagem em que é escrito o programa ocupa +/- instruções
- As linguagens mais complexas (ex. ST e SFC), em princípio utilizam mais memória e demoram mais tempo a executar

Ki = K instruções

Program 100% Boolean	
• LD language	1.6 Ki
• IL language	2 Ki
• ST language	1.3 Ki

Program 90% Boolean	
• LD language	1.1 Ki
• IL language	1.4 Ki
• ST language	1.1 Ki

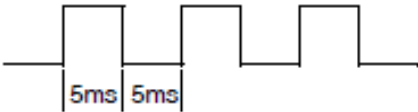
Program 65% Boolean	
• LD language	0.9 Ki
• IL language	1.0 Ki
• ST language	1.0 Ki

IL optimiza o uso da memória para programas booleanos

Mas quando é exigida complexidade (ex. cálculos) ST é melhor

Variáveis do sistema

- Para além das variáveis normais, o autómato disponibiliza uma número (elevado) de variáveis do sistema (%Si) que são utilizadas para sinalizar o estado de funcionamento do equipamento
- Alguns exemplos:

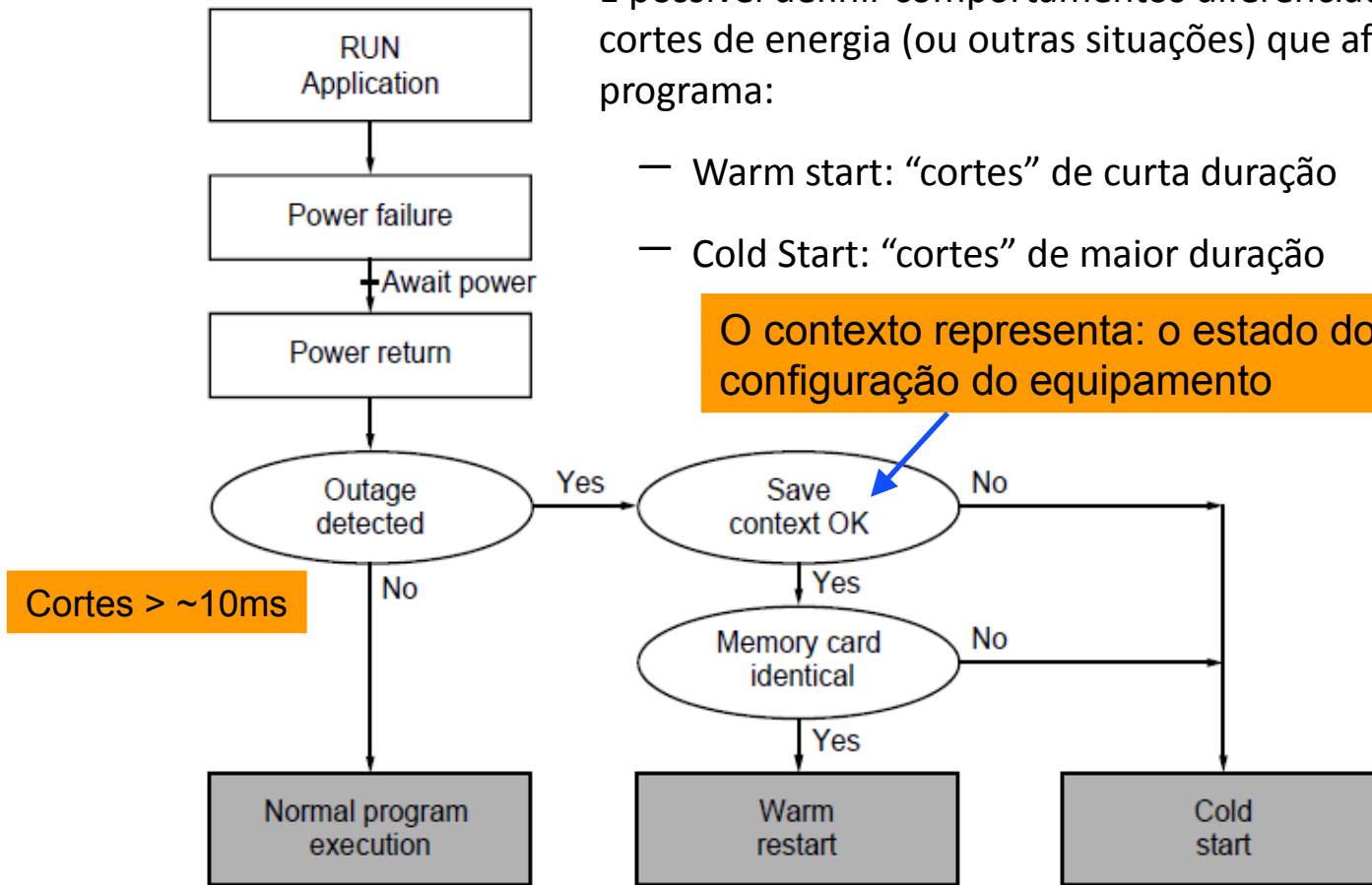
%S13 1RSTSCANRUN	First cycle after switching to RUN	Switching the PLC from STOP mode to RUN mode is indicated by setting system bit %S13 to 1. This bit is reset to 0 at the end of the cycle of the MAST task in RUN mode.
%S4 TB10MS	Timebase 10 ms	An internal timer regulates the change in status of this bit. It is asynchronous in relation to the PLC cycle. Graph: 
%S11 WDG	Watchdog overflow	Normally at 0, this is set to 1 by the system as soon as the task execution time becomes greater than the maximum execution time (i.e. the watchdog) declared in the task properties.

Modos de operação especiais

— É possível definir comportamentos diferenciados quando ocorrem cortes de energia (ou outras situações) que afectam a execução do programa:

- Warm start: “cortes” de curta duração
- Cold Start: “cortes” de maior duração

O contexto representa: o estado do programa, dados e configuração do equipamento



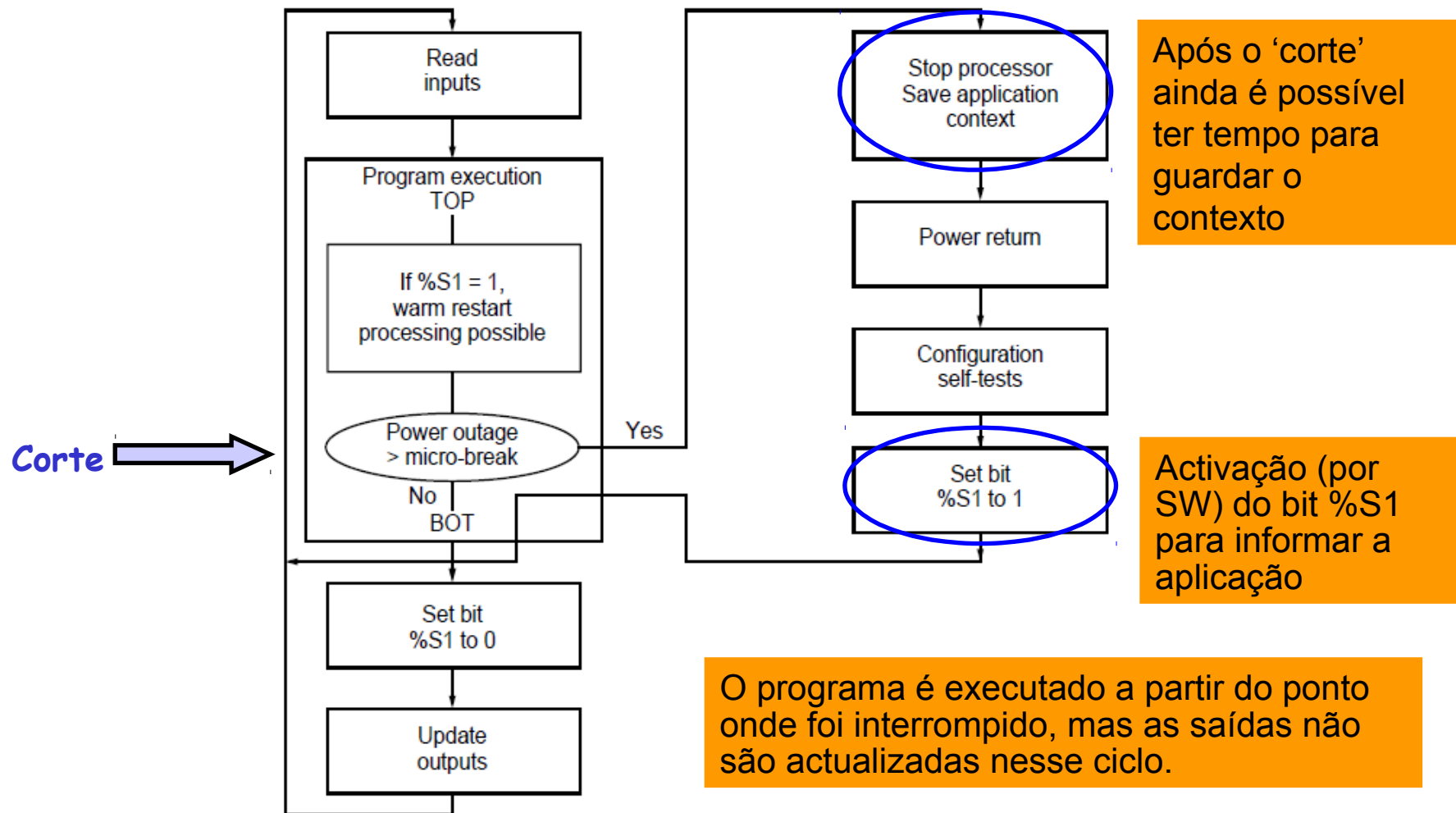
Cortes > ~10ms

Quando é detectado um corte de energia eléctrica, as saídas podem ser colocadas em:

- Fallback: valor pré-definido pelo utilizador durante a configuração (ex. 0 ou 1), ou
- Manter o último valor processado (executado no último ciclo).

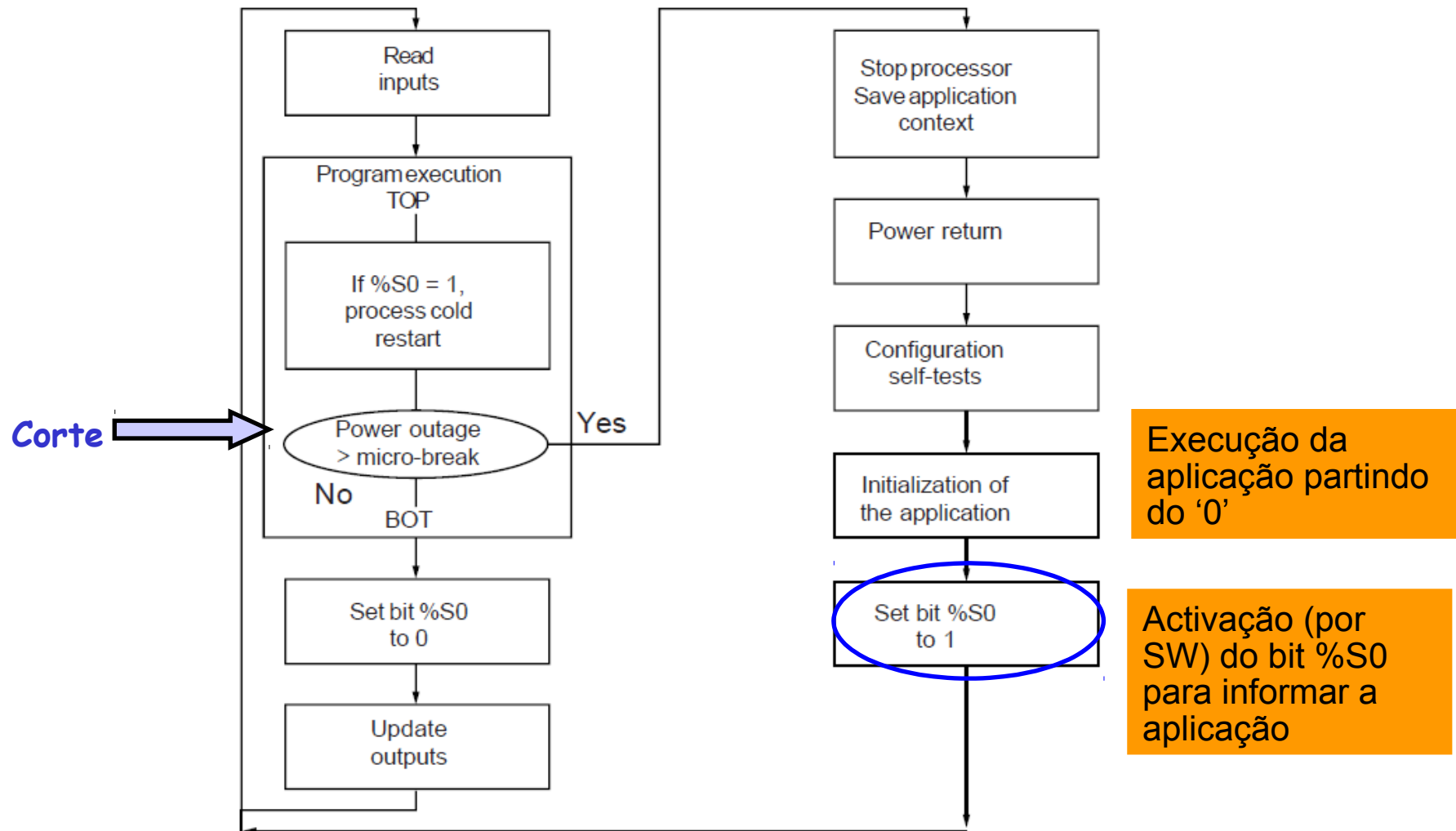
Warm start

- Retorno da alimentação sem perda de contexto



Cold start

- Retorno da alimentação com perda de contexto, reset do processador, transferência de um programa, sem bateria no momento do corte.



Exemplo de exercício relacionado com tempo de ciclo:

- Um autómato semelhante ao que utilizou nas aulas práticas executa duas tarefas: MAST e FAST. A tarefa MAST executa de forma periódica a cada 50 ms e possui um tempo de execução variável entre 12 e 30 ms (que já inclui o tempo de leitura das entradas utilizadas e a atuação nas saídas afetadas por esta tarefa). A tarefa FAST é periódica com período de 10 ms e tem um tempo de execução de 2ms (que já inclui o tempo de leitura das entradas utilizadas e a atuação nas saídas afetadas por esta tarefa). Considere que ambas as tarefas ficam prontas para executar no instante em que o autómato passa para RUN ($t=0$).
- a) O tempo máximo que pode ocorrer entre a primeira ativação da tarefa MAST e o fim da sua execução é de _____ ms.
- b) O tempo máximo que pode ocorrer entre a primeira ativação da tarefa FAST e o fim da sua execução é de _____ ms.
- c) A tarefa FAST executa um algoritmo de proteção que desativa todas as saídas caso o botão de paragem de emergência seja pressionado. A tarefa MAST executa um algoritmo que controla diversas saídas em função do estado das entradas, ignorando por completo o estado de um botão de paragem de emergência. Com esta configuração pode-se garantir que o tempo máximo entre a ativação do botão de emergência e a desativação de todas as saídas de forma permanente é sempre inferior a 20 ms.

Mast: periodica, $T = 50\text{ms}$ $C = 12 \text{ a } 30 \text{ ms}$

Fast: periodica, $T = 10\text{ms}$ $C = 2 \text{ ms}$

Exemplo de exercício relacionado com tempo de ciclo:

- Um autómato semelhante ao que utilizou nas aulas práticas executa duas tarefas: MAST e EVT1. A tarefa MAST executa de forma cíclica e possui um tempo de execução variável entre 20 e 25 ms (inclui o tempo de leitura de entradas utilizadas e atuação nas saídas afetadas por esta tarefa). A execução da tarefa EVT1 é despoletada por um evento gerado pela carta de entradas e a sua execução interrompe a execução da MAST. Sempre que é despoletada, a execução de EVT1 demora 5 ms (inclui o tempo de leitura de entradas utilizadas e atuação nas saídas afetadas por esta tarefa). O evento que despoleta a execução de EVT1 nunca ocorre com intervalos inferiores a 100ms (ou seja, intervalo entre dois eventos consecutivos nunca é inferior a 100ms). A tarefa MAST executa um algoritmo no qual copia o estado da entrada E1 para a saída S1. Inicialmente todas as entradas e saídas estão a falso. Em $t=10\text{ms}$ a entrada E1 passa a verdadeiro.
- a) O instante de tempo máximo em que a saída S1 passa a verdadeiro é de _____ ms.
- b) O instante de tempo mínimo em que a saída S1 passa a verdadeiro é de _____ ms.
- c) A tarefa EVT1 executa um algoritmo de proteção, o qual desativa todas as saídas caso um botão de paragem de emergência seja pressionado. O evento que despoleta a execução de EVT1 é a ativação do botão de emergência. A tarefa MAST executa um algoritmo que controla diversas saídas dependente do estado das entradas, ignorando por completo o estado de um botão de paragem de emergência. Com esta configuração pode-se garantir que o tempo máximo entre a ativação do botão de emergência e a descativação de todas as saídas de forma permanente é sempre inferior a 10 ms.

Regras de evolução e interpretação de SFC – Multi-token

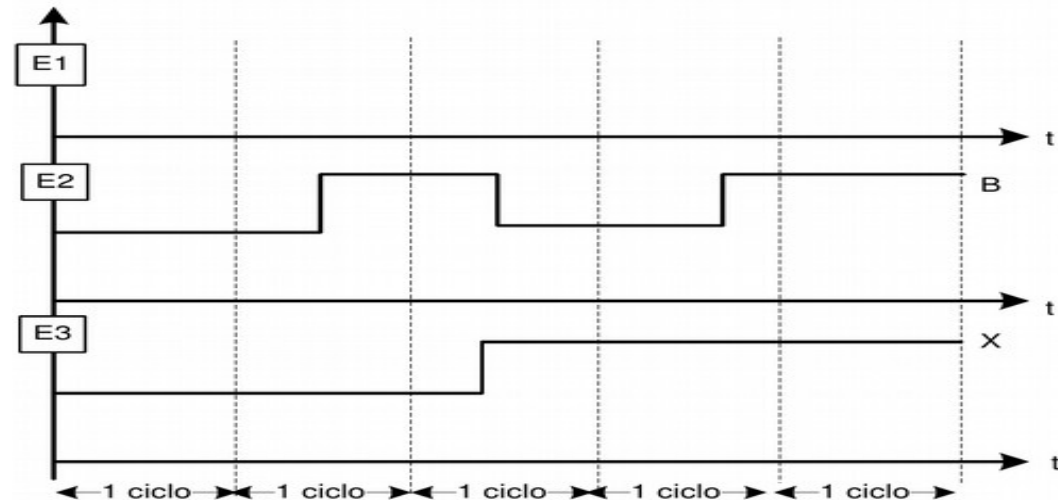
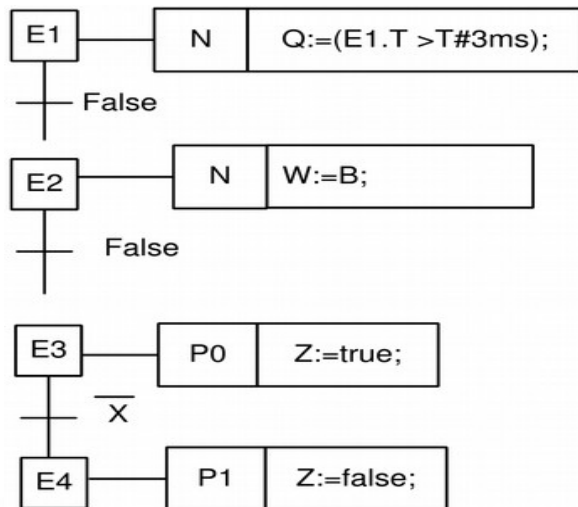
- The original situation is defined in a number of initial steps (0 to 100) which can be defined.
- A transition is enabled if the steps immediately preceding it are active. Transitions whose immediately preceding steps are not active are not analyzed.
- A transition is triggered when the transition is enabled and the associated transition conditions are satisfied.
- Triggering a transition leads to the disabling (resetting) of all immediately preceding steps that are linked to the transition, followed by the activation of all immediately following steps.
- If more than one transition condition in a row of sequential steps has been satisfied then one step is processed per cycle.
- If an active step is activated and deactivated at the same time then the step remains active.
- More than one branch can be active with alternative branches. The branches to be run are determined by the result of the transition conditions of the transitions that follow the alternative branch. Branch transitions are processed in parallel. The branches with satisfied transitions are activated.

If jumps are to be made into a parallel branch or out of a parallel branch then this option can be enabled. All parallel branches do not have to be combined at the end by a parallel joint in this case.

Regras de evolução e interpretação de SFC

- No início de cada ciclo do programa do autómato:
 1. Determinar quais as etapas que estão ativas
 2. Determinar quais as transições que podem disparar
 3. Disparar todas as transições possíveis
 4. Determinar quais as etapas que ficaram ativas
 5. Para cada etapa que ficou inativa executar as respetivas ações finais
 6. Para cada etapa que ficou ativa executar as respetivas ações

Este SFC é executado num autómato (semelhante ao que utilizou nas aulas práticas) dentro de uma tarefa cíclica com um tempo de execução constante (1 ciclo=4ms). Considere que antes do SFC ser executado todas as saídas têm o valor False e que Q, W, Z são saídas do autómato e B é uma entrada do autómato.



Bibliografia

- Principal

- D. Coggan, “Fundamentals of Industrial Control, 2nd edition”, ISA, 2005 (na biblioteca)
 - Capítulo 9 : Programmable Logic Controllers

- Complementar

- Programmable Logic Controllers – An Emphasis on Design and Application, K. Erickson, Dogwood Valley Press, LLC, 2005 (na biblioteca)
 - Capítulos: 1, 2, 3 (secções 3.1 e 3.3), 4 e 20.
 - W. Bolton, “Programmable Logic Controllers (4th Edition)”, Elsevier, 2006
 - Capítulos 1, 2 e 4: Programmable logic controllers, Input-output devices, I/O processing
 - Acessível via e-books da biblioteca: Ebooks->Lista de colecções->Knovel”. Pesquisar pelo título do livro
- Documentação adicional fornecida pelo docente