

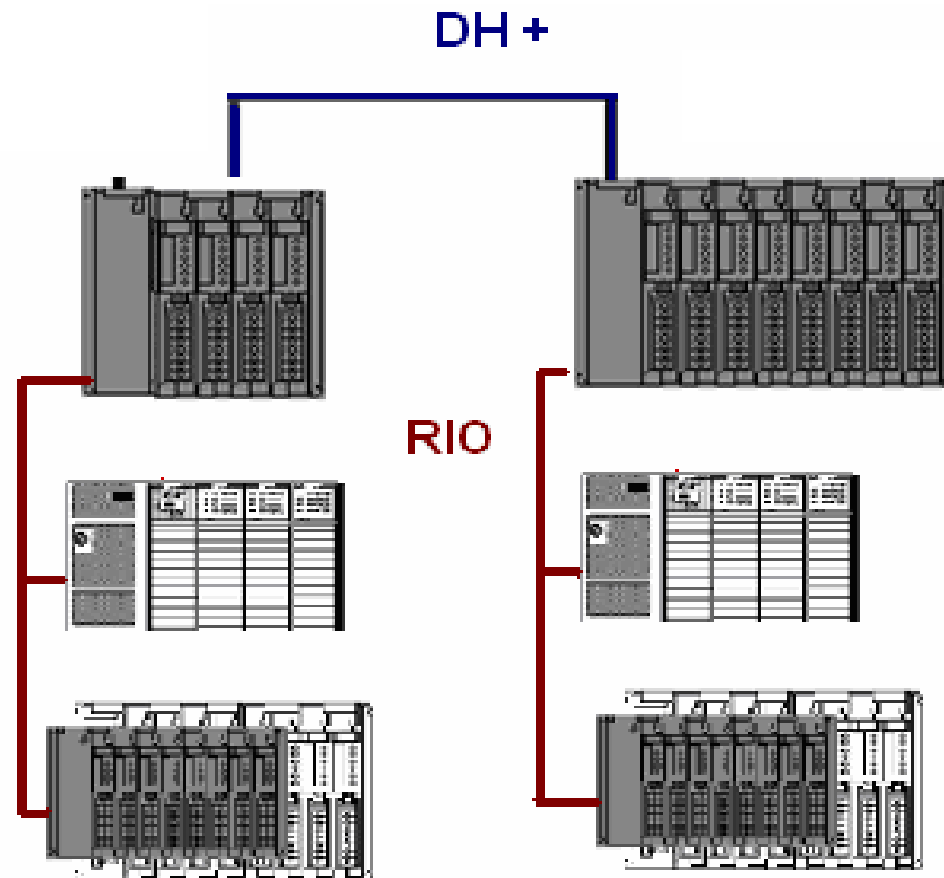
# TENDENCIAS DE LA AUTOMATIZACION

- Arquitecturas
- Programación

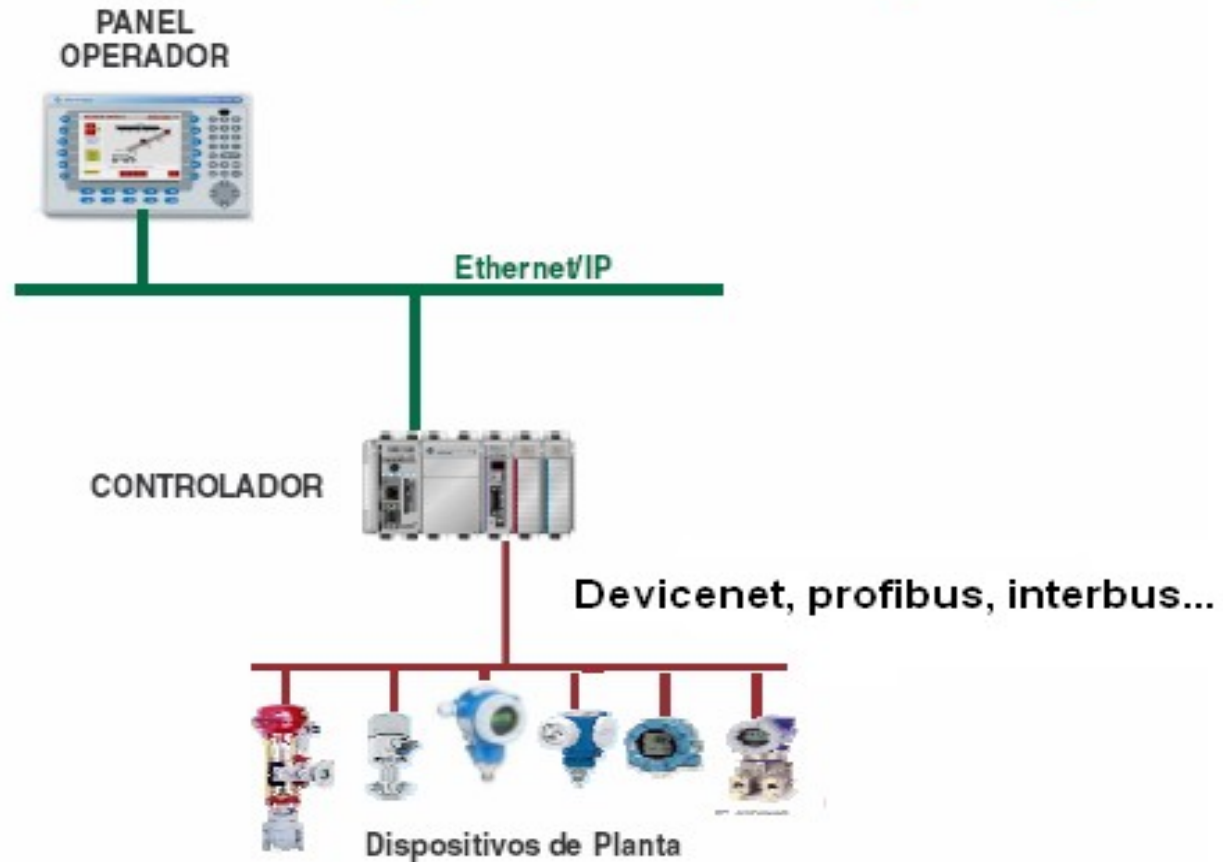
**Autor: Juan Carlos Rubio Calín**



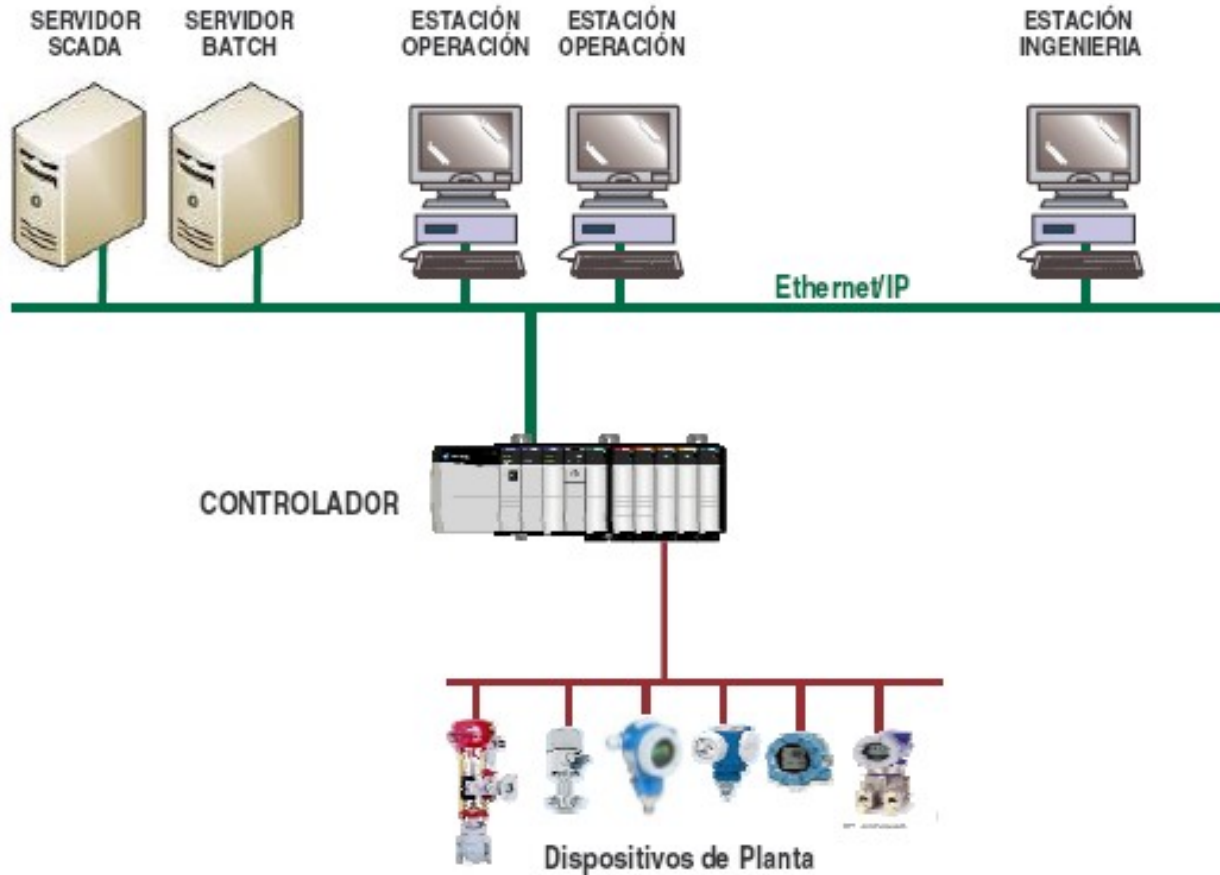
## ARQUITECTURA OBSOLETA



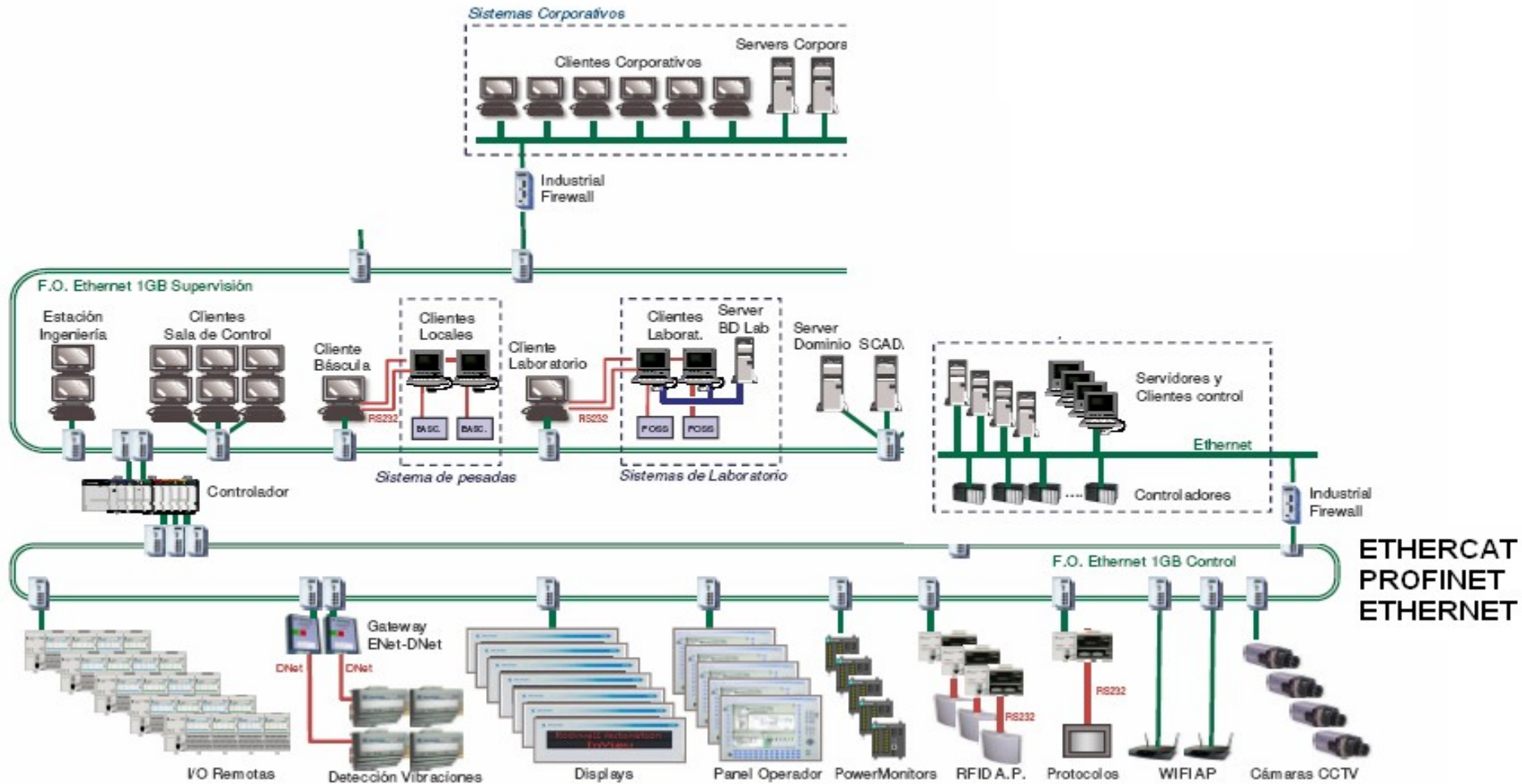
# Arquitectura Independiente: Ejemplo



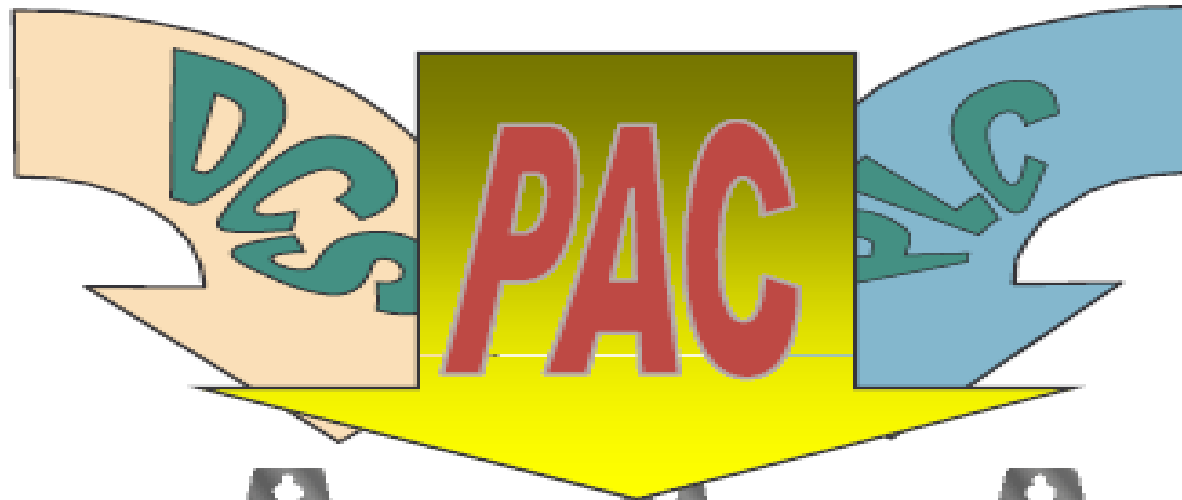
## Arquitectura Centralizada: Ejemplo



# Arquitectura Descentralizada: Ejemplo



**+ MOTION**



PAC: Programmable Automation Controller



## Particularidades del proceso

- Lazos de regulación (PID) de caudales y presiones.
- Coordinación entre los circuitos de captación y osmosis (en los extremos del proceso): Se activan pareados. Si se produce una alarma en un bastidor de osmosis hay que parar de forma controlada la bomba de captación y biceversa.
- Cuando la planta esta en producción se mueven grandes masas de agua a presiones elevadas. Por ejemplo un fallo en la regulación de un lazo podría tener como consecuencia rotura de tuberías, daños en la instrumentación, etc..

## Solucion Tradicional

### Arquitectura de control

El control reside un **PLC de gran potencia y con CPUs redundantes** y con posibilidad de implementar buses de control redundantes y equipos de Entrada/Salidas remotas.

### Las claves

- Para garantizar la disponibilidad del sistema se utilizan PLC potentes con redundancia CPU: Hay 2 procesadores funcionando en paralelo si falla el primario uno el secundario toma el control de la aplicación.
- Comunicaciones Redundantes: Los PLC y los equipos de periferia tiene módulos con dos canales de comunicación. Si fallan las comunicaciones por un canal automáticamente se utiliza en canal secundario.



# Solucion Alternativa

## Arquitectura de control

Se utilizan **Control** (realmente) **Distribuido** y comunicaciones redundantes. No hay PLC especiales. Los equipos de Entrada/salida remoto se sustituyen por dispositivos programables, que denominaremos unidades funcionales, que contienen el hardware (tarjetas I/O) y el programa necesario para controlar una pequeña parte del proceso.

## Las claves

Se la modularidad del diseño del programa de control se dispersa por toda la planta aprovechando el carácter modular inherente al diseño de la misma. Aparece el concepto de "Unidad Funcional". Una unidad funcional esta formada por el hardware (CPU y tarjetas) y el software (programa PLC) necesario para controlar una unidad funcional de ingeniería por ejemplo un Filtro de Arena

Para las comunicaciones se utiliza una red Ethernet en anillo (redundante) sobre fibra óptica. En dicha red coexisten dos tipos de comunicaciones

- Comunicaciones con SCADA con protocolo estándar (Modbus/TCP)
- Comunicaciones entre unidades funcionales mediante "variables de red" basadas en la técnica consumidor productor.

## PC-based Control

1986



PC Control  
PC-compatible  
machine  
control

1988



S1000  
PLC/NC on  
PC

1989



Lightbus

1993



S2000  
PLC/NC/CNC  
on PC

1995



Bus  
Terminal  
universal field-  
bus module

1996



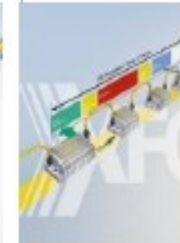
TwinCAT  
IEC 61131  
PLC real-time  
under  
Windows NT

2003



EtherCAT  
real-time  
Ethernet  
Fieldbus

2008



XFC  
eXtreme Fast  
Control  
Technology

2010



TwinCAT3

<b>Control basado en PC</b>	
<b>Características del sistema</b>	
<b>El control basado en PC consta de 4 componentes</b>	<b>El control basado en PC es un sistema de control abierto</b>
<ul style="list-style-type: none"><li>▪ PCs Industriales</li></ul>	<ul style="list-style-type: none"><li>▪ Utiliza componentes de hardware estándar</li></ul>
<ul style="list-style-type: none"><li>▪ E/S en bus de campo</li></ul>	<ul style="list-style-type: none"><li>▪ Utiliza software estándar</li></ul>
<ul style="list-style-type: none"><li>▪ Sistemas operativos estándar (Windows Vista, XP, XP Embedded, CE, 2000, NT, DOS)</li></ul>	<ul style="list-style-type: none"><li>▪ Fácilmente personalizable para aplicaciones e industrias específicas</li></ul>
<ul style="list-style-type: none"><li>▪ Software de control en tiempo real TwinCAT / VXWORKS</li></ul>	

# ¿ Como programar un automatismo?

- Por intuición
- Por experiencia
- Métodos : Karnaugh, grafcet, gemma
- Por normativas: VW, PSA, ...
- Por simulaciones y analisis (Matlab, simulink, labview..)
- Programación orientada a objetos **OOP**  
( Domótica, robótica... )

# The IEC 61131-3 Programming Languages

IEC 61131-3:  
Sistema de  
programación  
con los 5  
lenguajes  
Standard

LD

Diagrama de Contactos

IL

Lista de  
Instrucciones

FBD

Bloque de Funciones

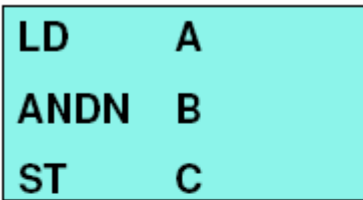
ST

Estructurado

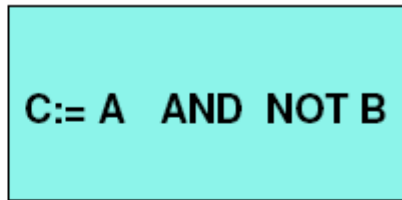
SFC

Funciones Secuenciales

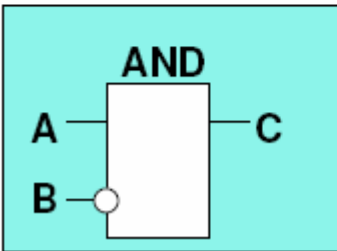
Instruction List



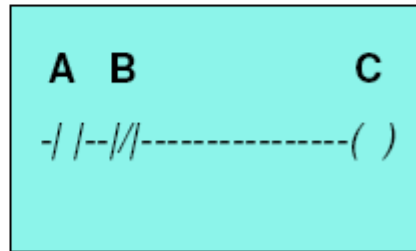
Structured Text



Function Block Diagram

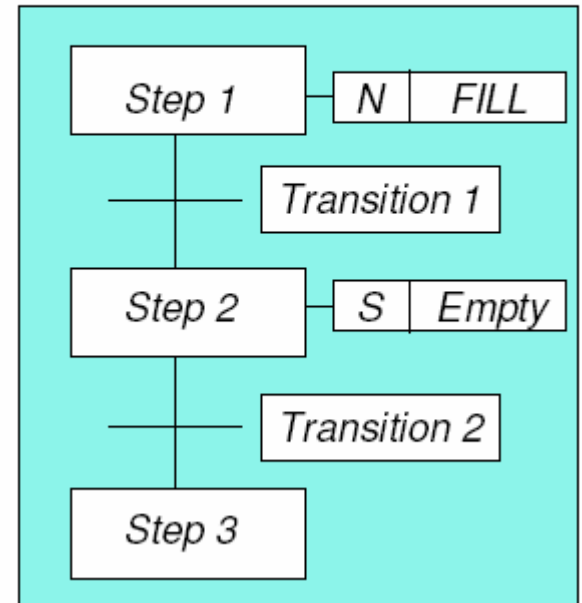


Ladder Diagram



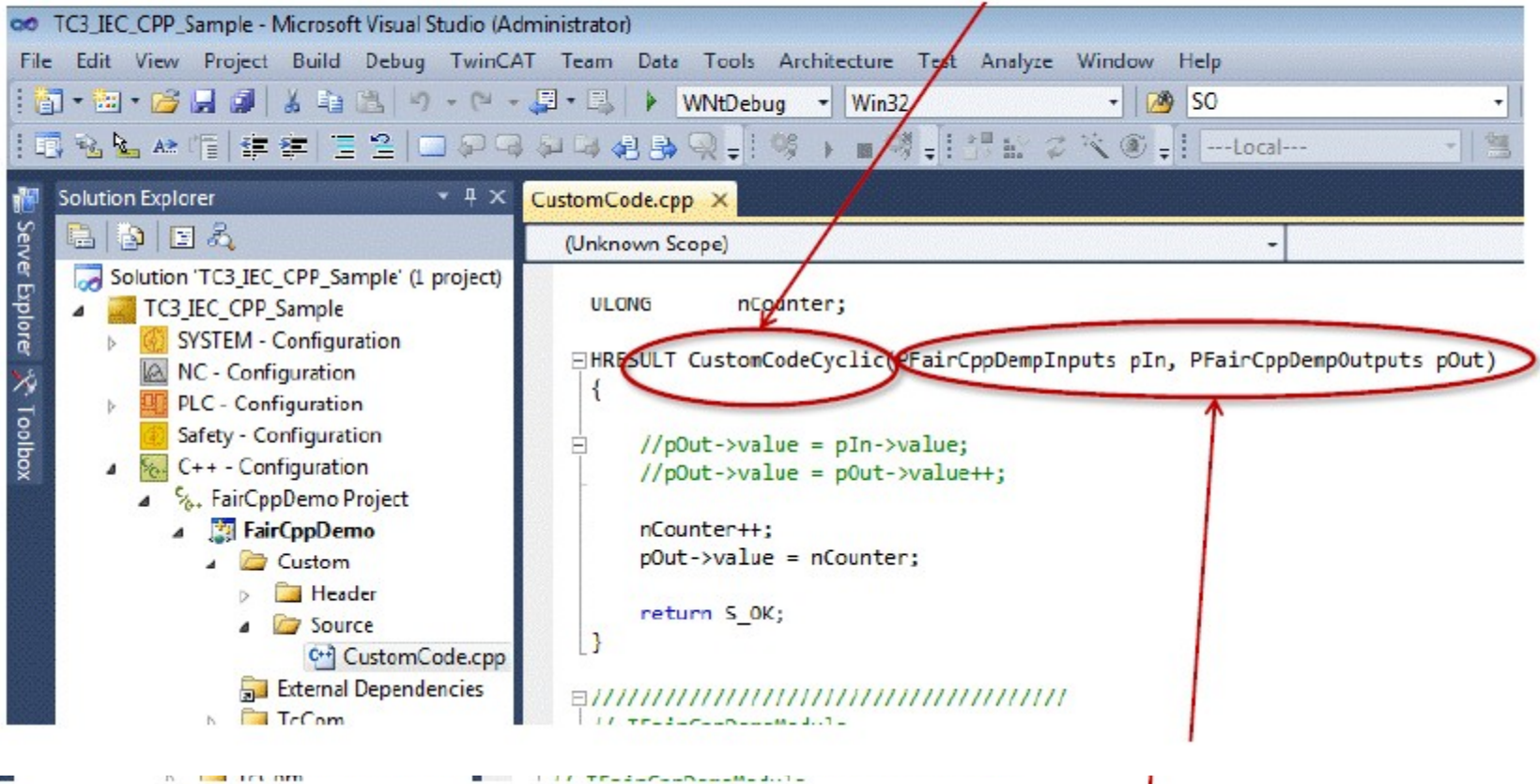
## SFC

Funciones Secuenciales



lenguajes de alto nivel

# C/C++ Programming languages

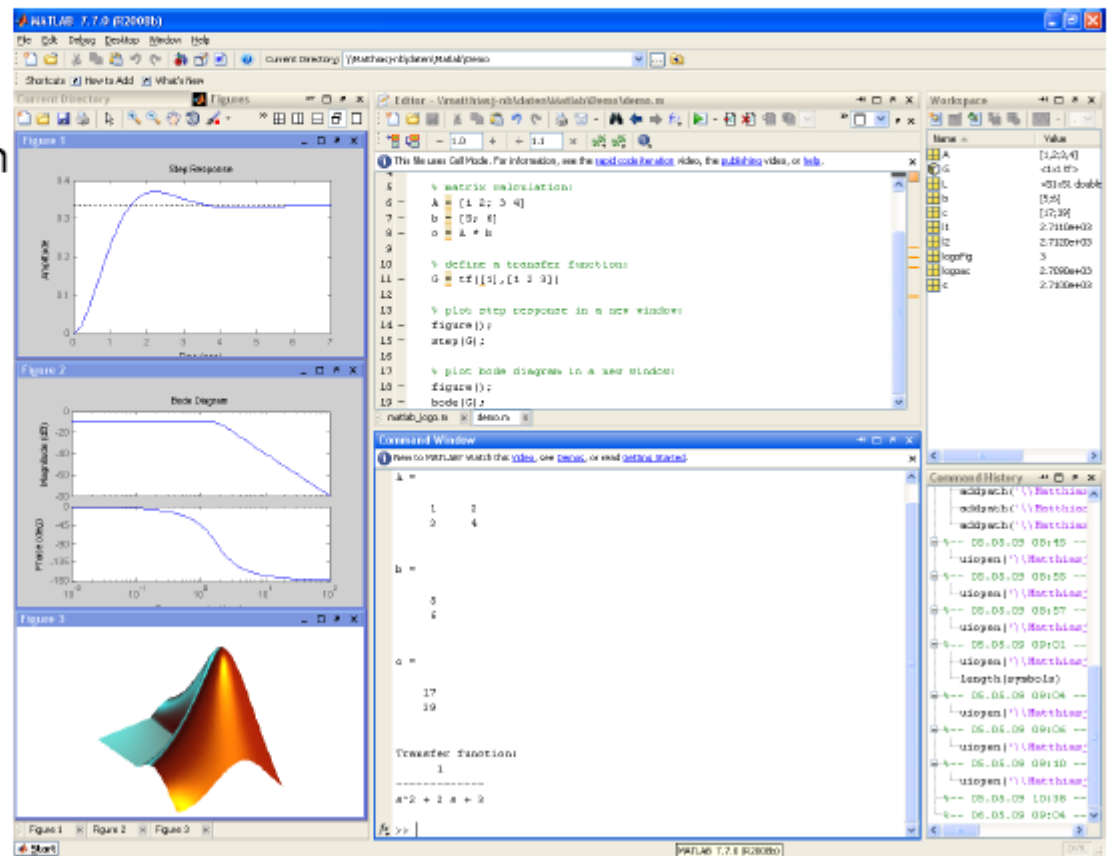




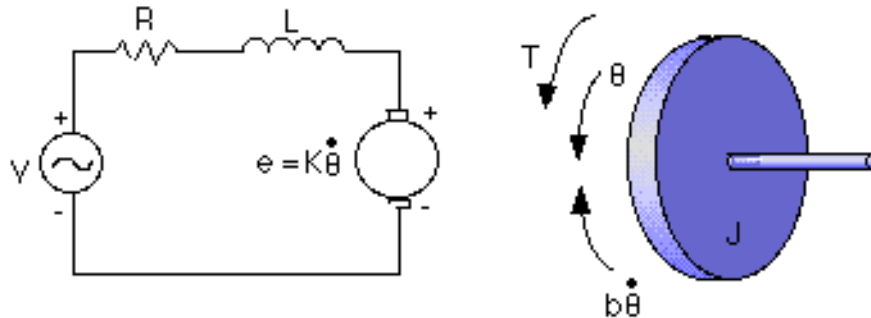
# What is Matlab ,Simulink, Labview ?

## Matlab

- Matrix-Operations
- Easy programmability
- Graphical data preparation
- Many special functions for a wide field of application
- Very common in the scientific/ university environment



## Comportamiento de un sistema motor eléctrico frente a variaciones de su par de carga



- $v(t)$  tensión aplicada (variable de entrada)
  - $i(t)$  intensidad en el devanado del motor
  - $f_{cem}(t)$  fuerza contraelectromotriz generada
  - $w(t)$  velocidad angular de giro del motor
  - $p(t)$  par proporcionado por el motor.
- $R$  y  $L$  representan la resistencia e inductancia de los devanados del motor.

Las ecuaciones de comportamiento son las siguientes:

•  $v(t) = R \cdot i(t) + L \cdot di(t)/dt + f_{cem}(t)$  [segunda Ley de Kirchoff] [1]

•  $f_{cem}(t) = K_v \cdot w(t)$  [ $f_{cem}$  proporcional a velocidad de giro] [2]

•  $p(t) = K_e \cdot i(t)$  [par ofrecido proporcional a intensidad] [3]

$p(t) = J \cdot dw(t)/dt + B \cdot w(t)$  [4] El par producido por el motor

**Ecuación del Comportamiento Eléctrico:**

$$L_a \frac{d}{dt} i_a = v_a - r_a i_a - k_v w_r$$

Donde:

$L_a =$	120 mH	Inductancia eléctrica	
$i_a =$	variable de estado.	Corriente del circuito del rotor	
$v_a =$	variable de control.	Tensión en el circuito del rotor	$v_{nom} = 6 \text{ V}$
$r_a =$	7 ohm	Resistencia del rotor	
$k_v =$	$1.41 \cdot 10^{-2} \text{ Vs/rad}$	Constante de la armadura	
$w_r =$	variable de estado.	Velocidad angular del rotor	$w_r^{nom} = 104 \text{ rad/s}$

**Ecuación del Comportamiento Mecánico:**

$$T_l = k_e i_a = J \frac{d}{dt} w_r + B_m$$

Donde:

$J =$	$1.06 \cdot 10^{-6} \text{ Kg m}^2$	Momento de Inercia del eje del rotor	
$k_e =$	$k_v (*)$	Constante del motor	
$B_m =$	$6.04 \cdot 10^{-6} \text{ Nms}$	Coefficiente de rozamiento viscoso del rotor	
$T_l =$	perturbación	Par de giro	$T_l^{nom} = 0.0085 \text{ Nm}$

Usando Transformadas de Laplace, las ecuaciones ([1],[2],[3],[4]) del modelo de arriba pueden expresarse en términos de s.

Conjunto de ecuaciones del sistema:

- $v(t) = R \times i(t) + L \times di(t)/dt + f_{cem}(t)$
- $f_{cem}(t) = K_v \times w(t)$
- $p(t) = K_e \times i(t)$
- $p(t) = J \times dw(t)/dt + B \times w(t)$

Estas ecuaciones son directamente transformables al dominio de Laplace:

- $V(s) = R \times I(s) + L \times s \times I(s) + F_{CEM}(s)$
- $F_{CEM}(s) = K_v \times W(s)$
- $P(s) = K_e \times I(s)$
- $P(s) = J \times s \times W(s) + B \times W(s)$

De la figura de arriba podemos escribir las siguientes ecuaciones basadas en la ley de Newton combinado con la ley de Kirchhoff:

- $V(s) = R \times I(s) + L \times s \times I(s) + K_v \times W(s)$
- $P(s) = K_e \times I(s)$
- $P(s) = s(J \times s + B) \times W(s)$

Las ecuaciones obtenidas del sistema y la función de transferencia son:

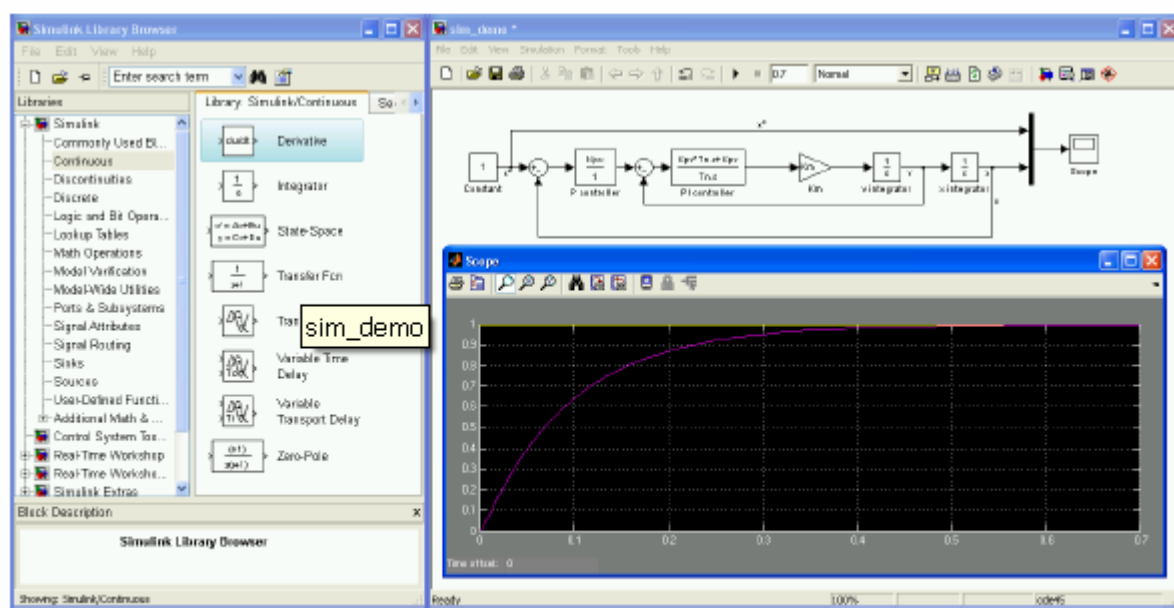
$$s(J \times s + B) \times W(s) = K_e \times I(s)$$

$$(L \times s + R) \times I(s) = V(s) - K_v \times W(s)$$

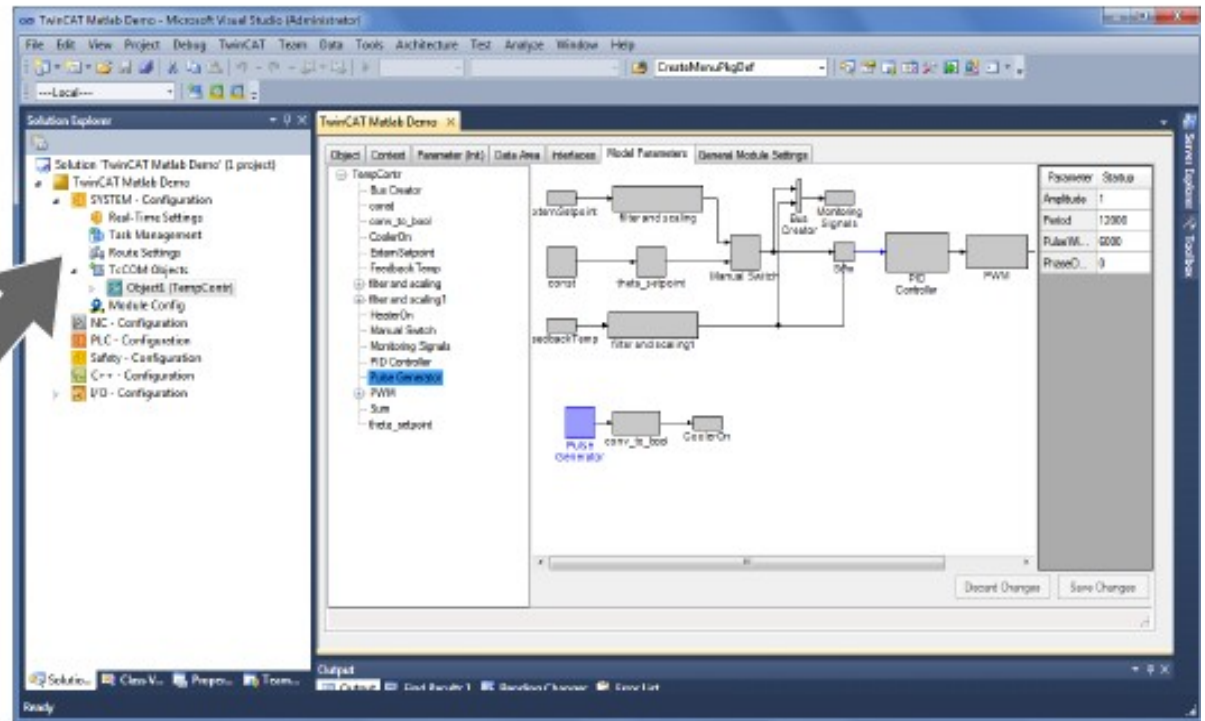
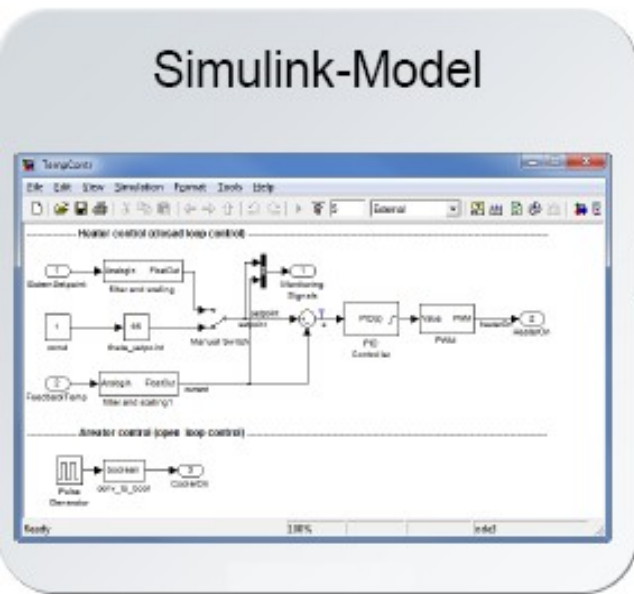
# What is Simulink?

Simulation of dynamic systems

$$\frac{\theta}{V} = \frac{K}{(Js + b)(Ls + R) + K^2}$$



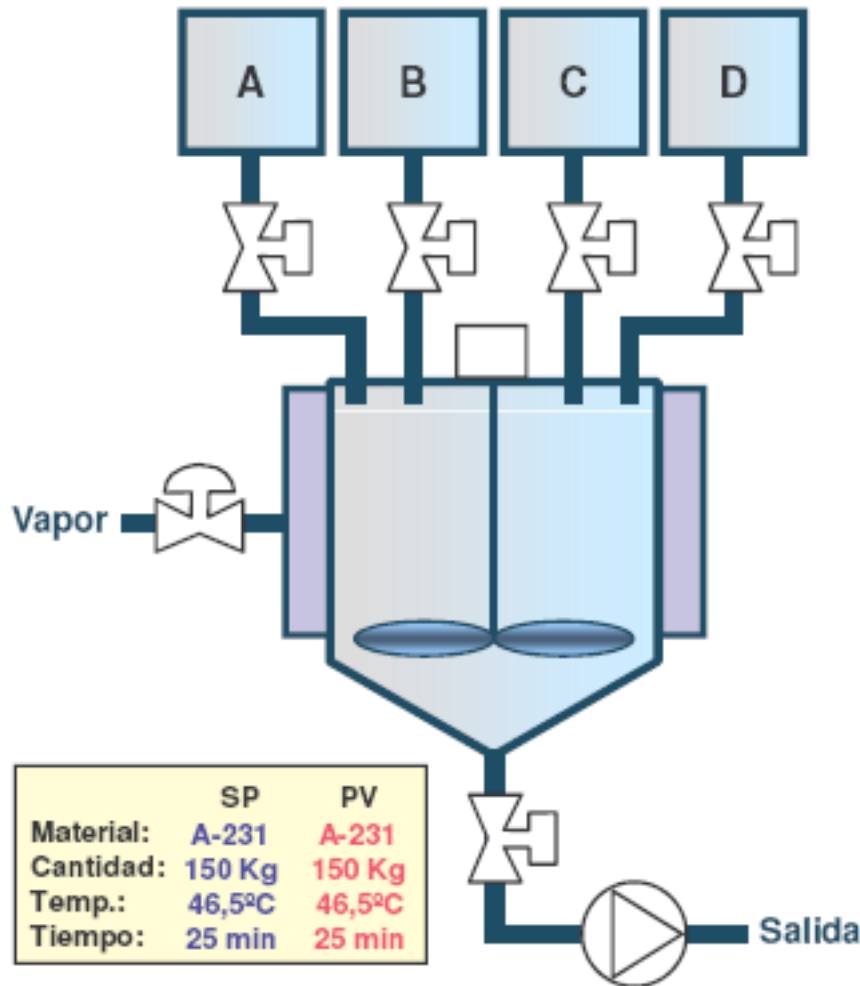
## Simulink-Model



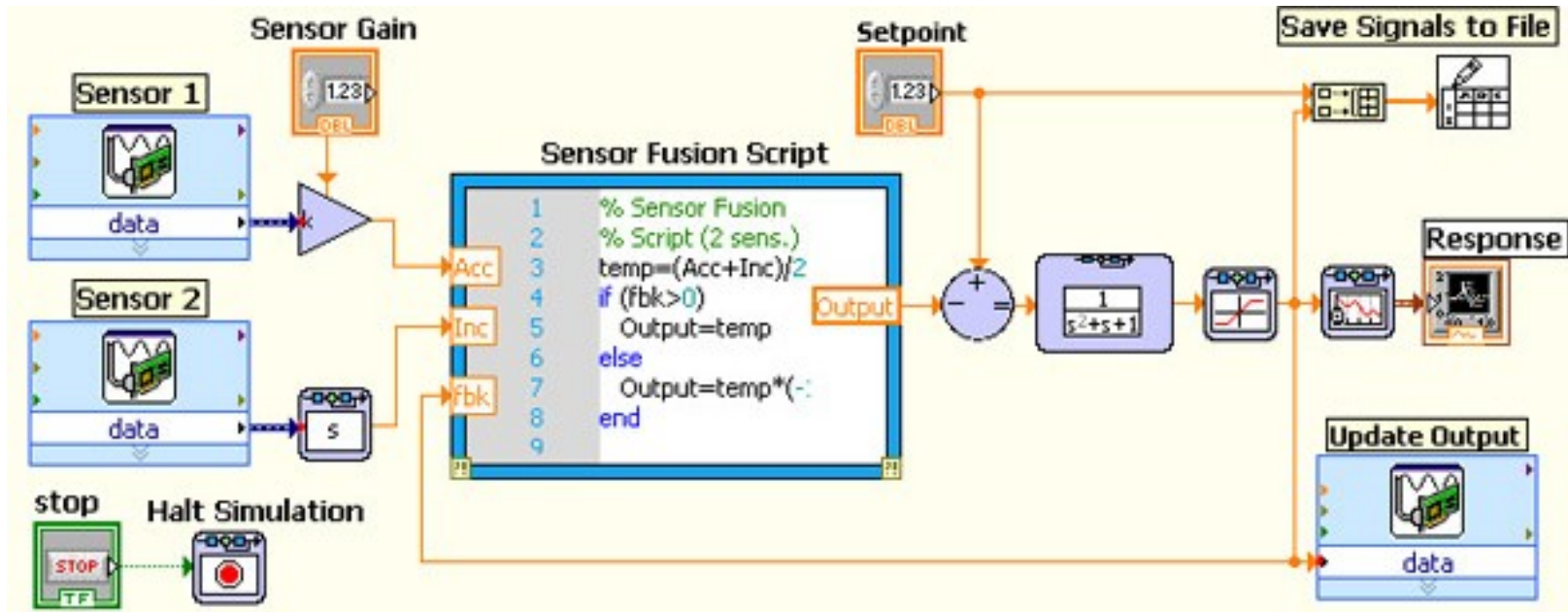
The screenshot shows the TwinCAT software interface, which is a MATLAB-based environment for PLC programming. The main window displays a Simulink model of a motor control system. The model includes a 'Manual Switch' block, a 'PID Controller' block, and a 'Pulse Generator' block. The 'Manual Switch' block is selected, and its parameters are shown in the 'Parameter (Int)' tab. The parameters are:

Parameter	Status
Amplitude	1
Period	12000
PulseW...	6000
PhaseO...	0

The 'Parameter (Int)' tab also shows a list of parameters for the 'Manual Switch' block, including 'Bus Creator', 'Manual Switch', 'Monitoring Signals', 'Pulse Generator', and 'Sum'. The 'Manual Switch' block is highlighted in blue. The 'Parameter (Int)' tab also shows a list of parameters for the 'Manual Switch' block, including 'Bus Creator', 'Manual Switch', 'Monitoring Signals', 'Pulse Generator', and 'Sum'. The 'Manual Switch' block is highlighted in blue.



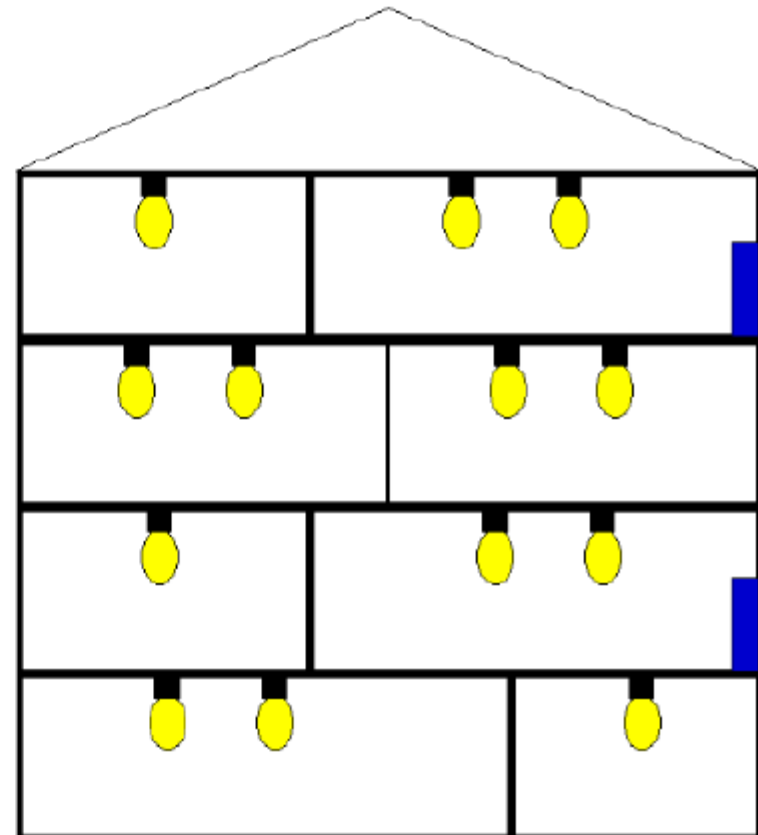
PAC: Programmable Automation Controller

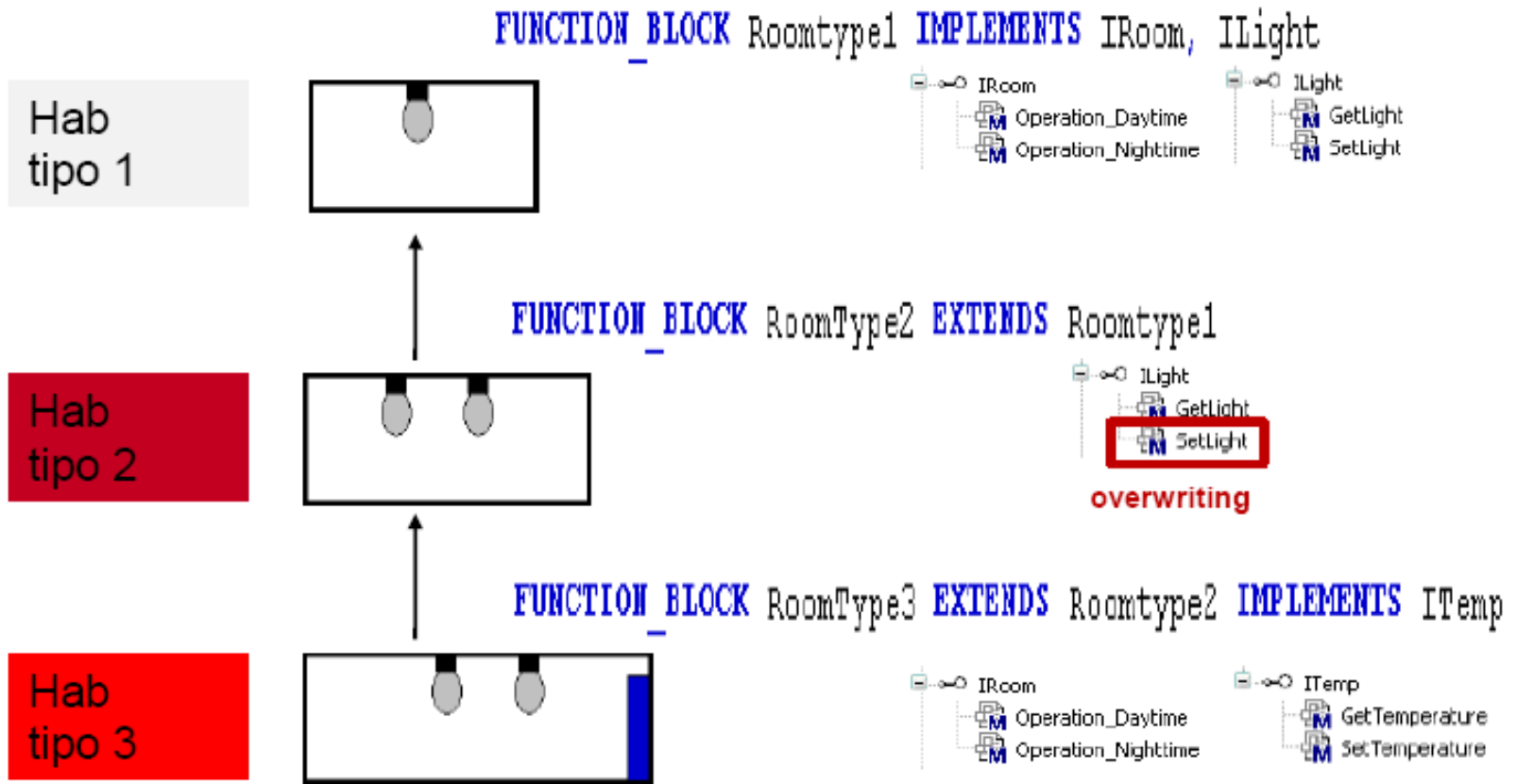




## Un ejemplo sencillo

- Realizaremos un sencillo ejemplo de domótica
- Una casa con 3 tipos de habitaciones diferentes:
  - Hab. tipo1, solo 1 luz
  - Hab. tipo2, 2 luces
  - Hab. Tipo3, 2 luces + control de temperatura
- Funciones:
  - Manejar las luces (dia/noche)
  - Controlar la temperatura





OOP es un Standard en formación para programadores de software para PC

**!!! MUCHAS GRACIAS !!!**