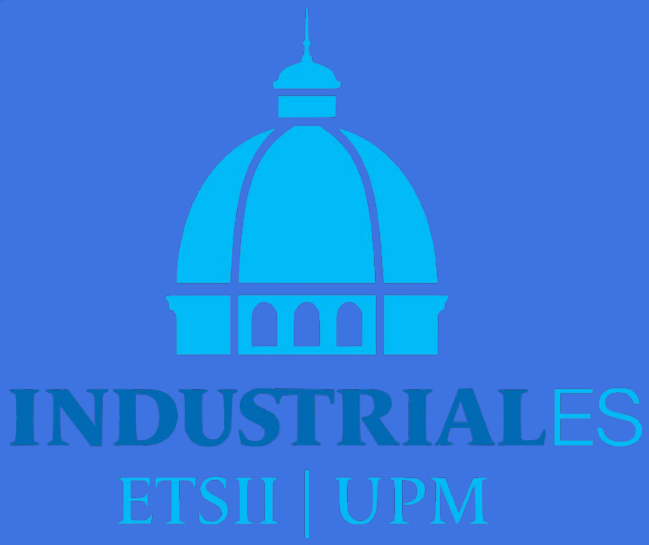
CYBERTECH 2015

TALLER 5: Lógica de control.



Departamento de Automática, Ingeniería Electrónica e Informática Industrial

CYBERTECH
2015

ÍNDICE

- Intro
- Control todo o nada (Bang-Bang Control)
- Cálculo del error.
- Regulador P
- Regulador PI
- Regulador PD
- Regulador PID
- Ajuste PID
- Otros

CYBERTECH 2015

INTRO

(¿Por qué?)

- ¿Cómo le digo a mi robotillo que no se choque contra una pared en el laberinto?

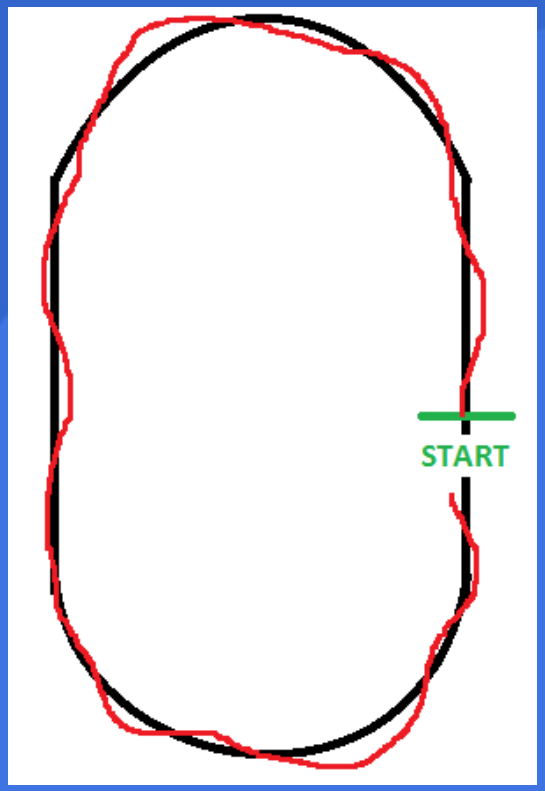
- ¿Cómo convencerle de que la línea negra es lo que quiere?

La lógica de control es lo que nos ayuda a relacionar las lecturas de los sensores con los motores.

CYBERTECH 2015

INTRO

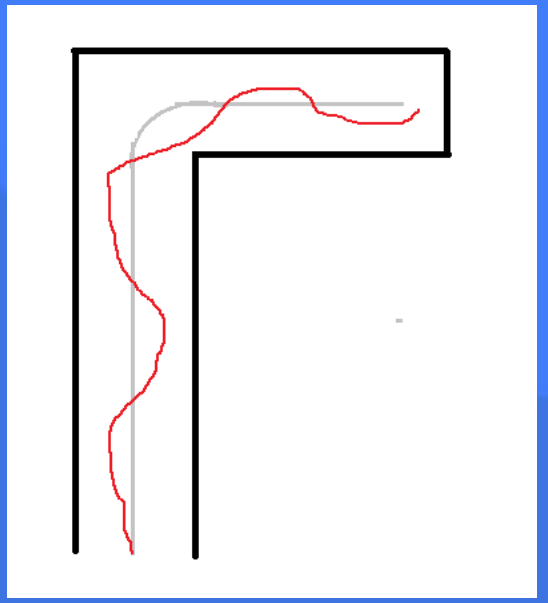
(¿Por qué?)



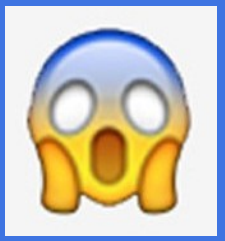
- Negro y gris: IDEAL
- Rojo: REAL

Nuestro objetivo es que la diferencia entre los dos sea mínima.

(Error → 0)



¿Y cómo ******* hago yo eso?



CYBERTECH 2015

ÍNDICE

- Intro
- Control todo o nada (Bang-Bang Control)
- Cálculo del error
- Regulador P
- Regulador PI
- Regulador PD
- Regulador PID
- Ajuste PID
- Otros

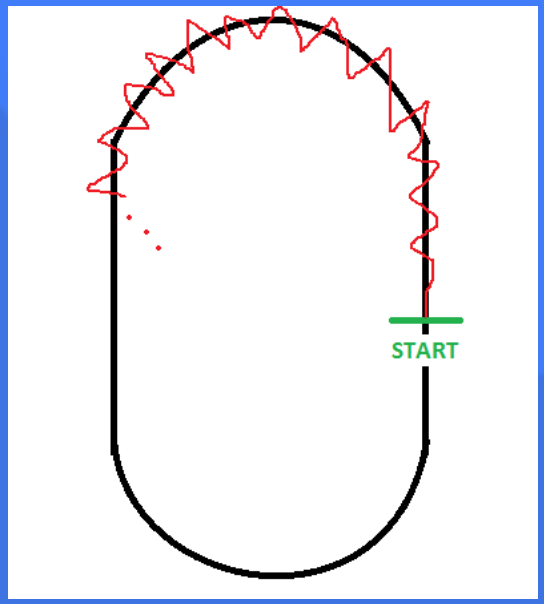
CYBERTECH
2015

Bang-Bang Control

- Línea a la derecha → giro brusco a la derecha.

- Línea a la izquierda → giro brusco a la izquierda.

- Centrado → sigue recto.



CYBERTECH
2015

Bang-Bang Control

- VENTAJAS:
  - *Muy Sencillo.
  - *Pocos sensores.

- DESVENTAJAS:
  - *Bajas velocidades.
  - *Alto consumo energético.
  - *No distingue entre errores grandes o pequeños.

CYBERTECH
2015

ÍNDICE

- Intro
- Control todo o nada (Bang-Bang Control)
- Cálculo del error
- Regulador P
- Regulador PI
- Regulador PD
- Regulador PID
- Ajuste PID
- Otros

CYBERTECH
2015

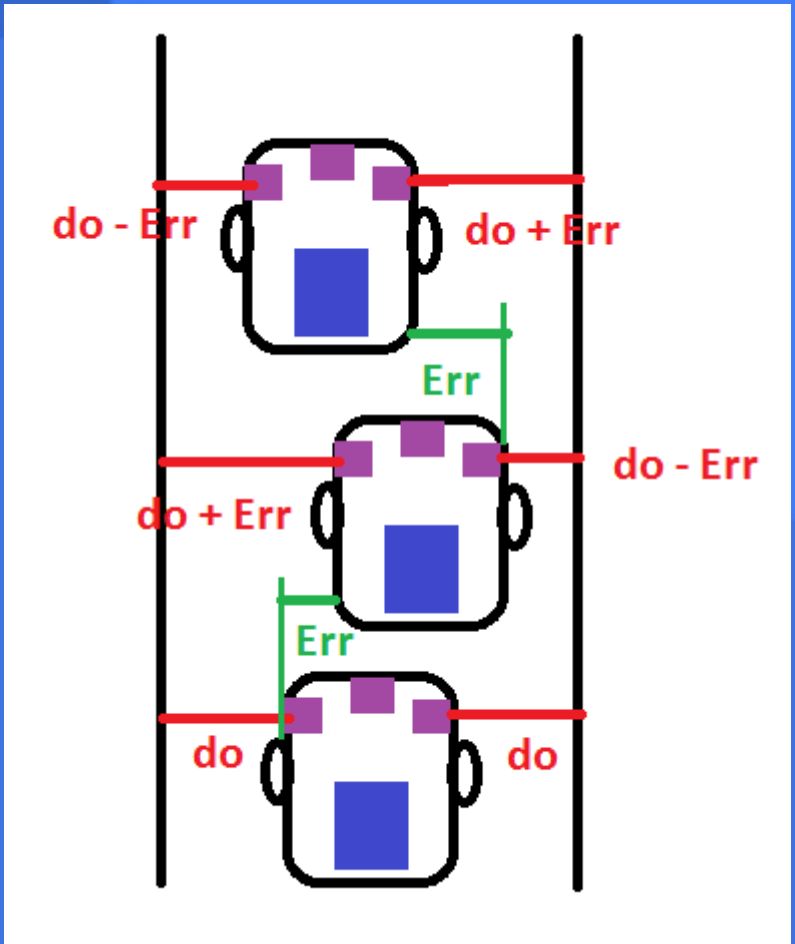
Cálculo del error

Err = (Ddcha – Dizq)/2

→ Err > 0 girar a la dcha
→ Err < 0 girar a la izq
→ Err = 0 centrado

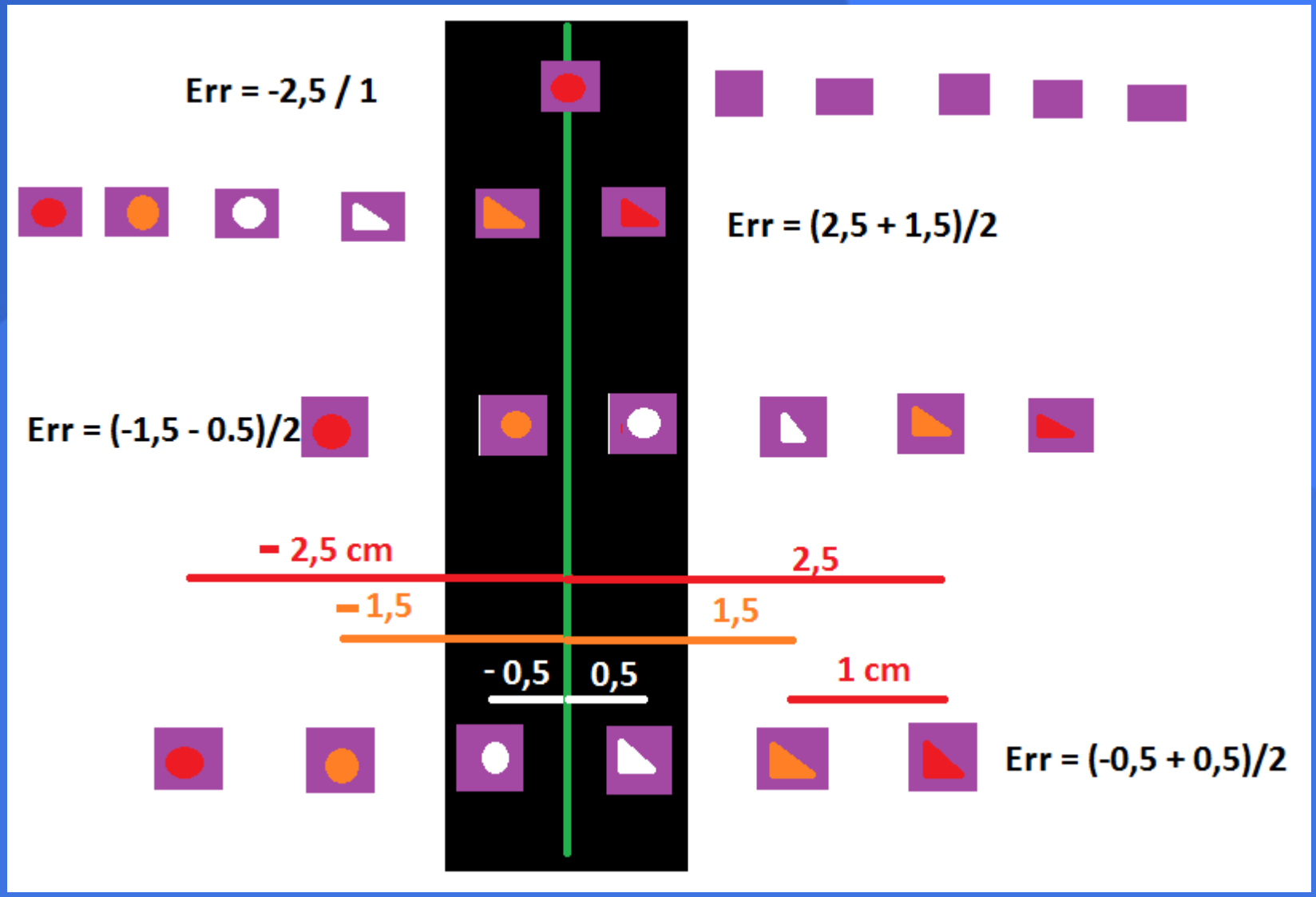
1 sensor

Err = Dsensor - do



CYBERTECH 2015

Cálculo del error



Cálculo del error

Err = Media de los sensores sobre la línea

→ Err > 0 girar a la dcha
→ Err < 0 girar a la izq
→ Err = 0 centrado

Cálculo del error

```
PesosSensores = {-2.5 , -1.5, -0.5, 0.5, 1.5 , 2.5}
For (i = 0 to 5) {
    if (sensor == negro) {
        Err += PesoSensores[i]
        Cont++;
    }
}
Err = Err/cont
```

CYBERTECH 2015

ÍNDICE

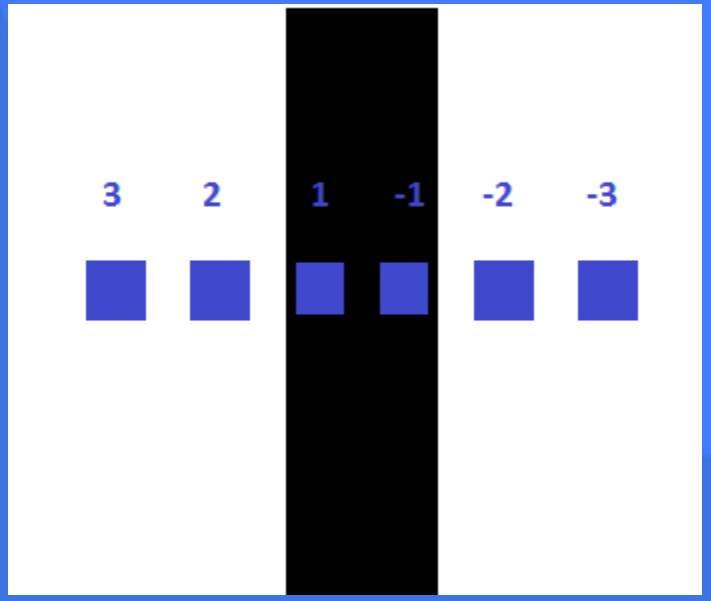
- Intro
- Control todo o nada (Bang-Bang Control)
- Regulador P
- Regulador PI
- Regulador PD
- Regulador PID
- Ajuste PID
- Otros

CYBERTECH 2015

Regulador P

- Línea a la derecha/izquierda → giro proporcional (Kp) al error a la derecha/izquierda.

- Centrado → sigue recto.



CYBERTECH
2015

Regulador P

Centrado → Err = 0

Vdcha = Vnom + Kp*Err

Vizq = Vnom

Línea a la derecha → Err < 0

Vdcha = Vnom + kp*Err

Vizq = Vnom

Línea a la izquierda → Err > 0

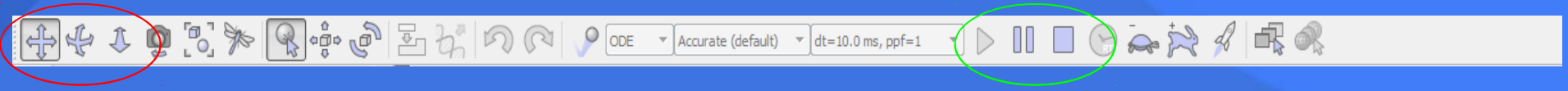
Vdcha = Vnom

Vizq = Vnom - Err

CYBERTECH 2015

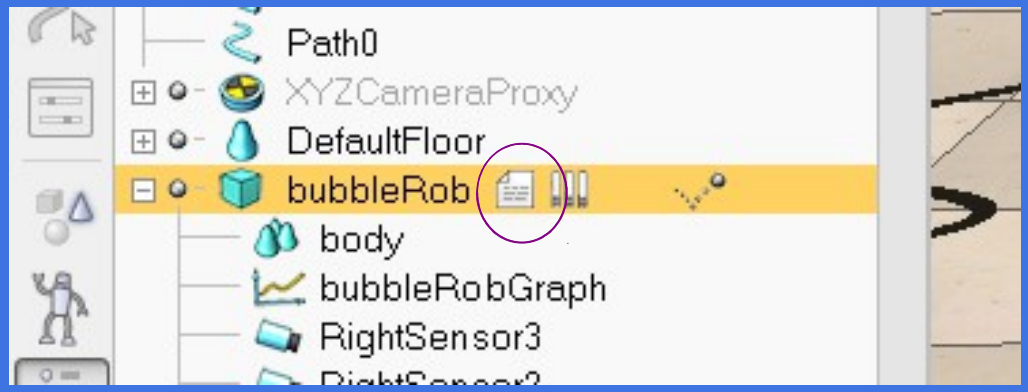
Regulador P

Ajustar la Kp (ki y kd = 0) para una velocidad ¾ Vmax (debajo de la d de speed) probando números enteros entre 1 y 10



Movernos por el escenario

empezar y parar simulación



CYBERTECH
2015

Regulador P

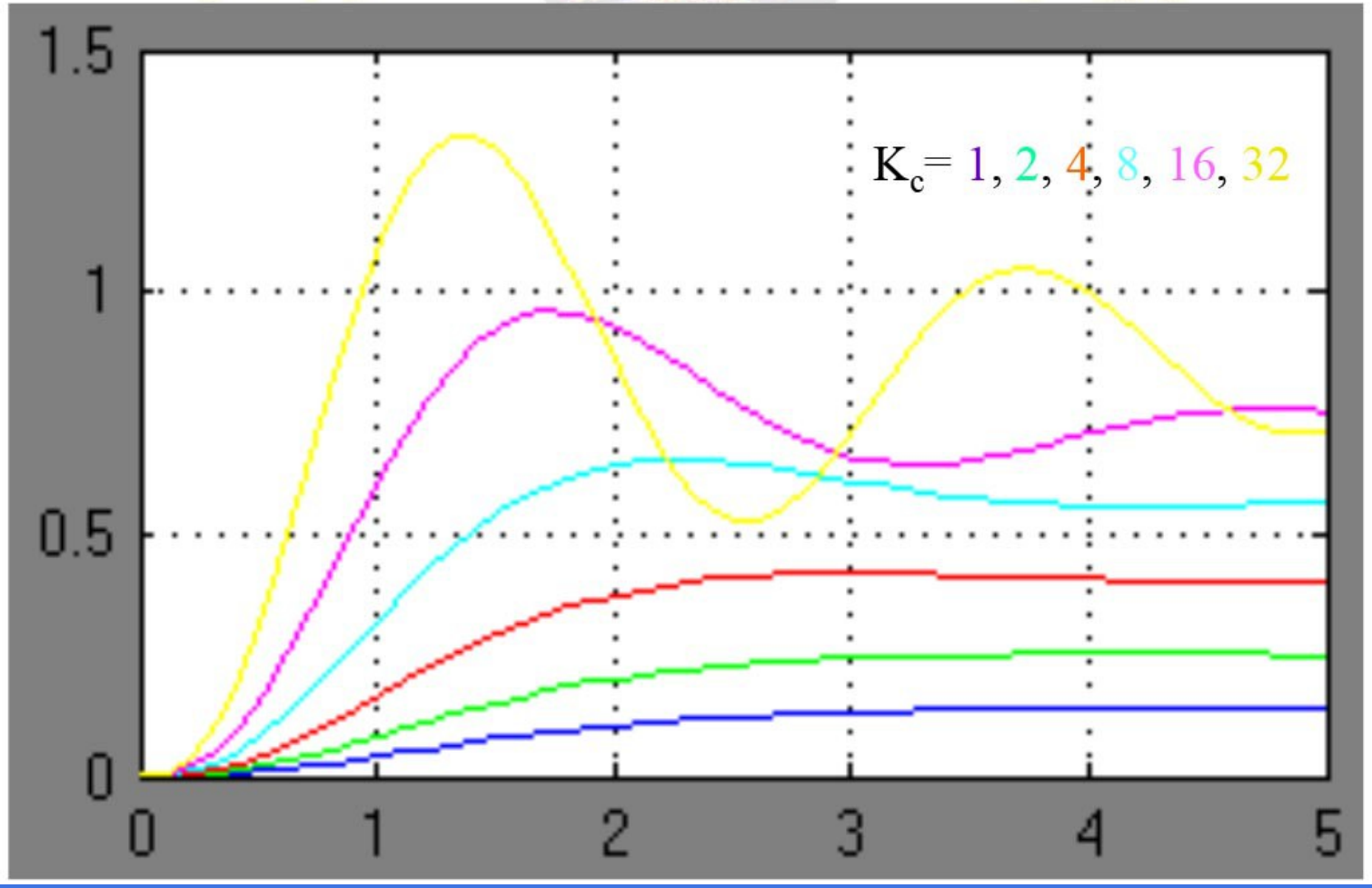
- VENTAJAS:
  *Sencillo.
  *Distingue entre errores grandes y pequeños.

- DESVENTAJAS:
  *Velocidades Medias-Bajas.
  *Mejor cuantos más sensores ($$).

CYBERTECH 2015

Regulador P

$$U(s) = K_c \epsilon(s)$$



ÍNDICE

- Intro
- Control todo o nada (Bang-Bang Control)
- Regulador P
- Regulador PI
- Regulador PD
- Regulador PID
- Ajuste PID
- Otros

CYBERTECH
2015

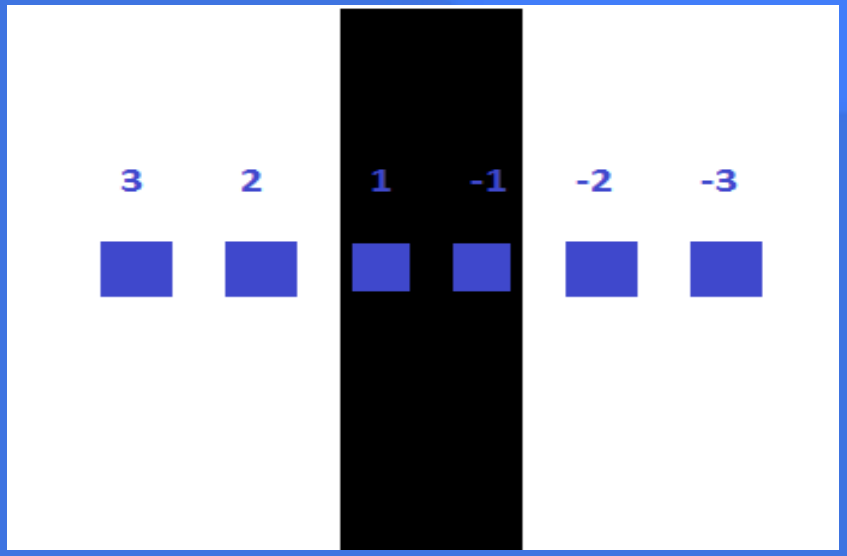
Regulador PI

- La integral acumula el error.
- Más error acumulado → respuesta más brusca.
  * Se centra antes. :)
  * Oscila más. :(

- Error de posición = 0

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t)\partial t$$

Kp/Ti → Ki



CYBERTECH
2015

Regulador PI

Centrado →

Vdcha = Vnom + Plout

Vizq = Vnom

Línea a la derecha →

Vdcha = Vnom + Plout

Vizq = Vnom

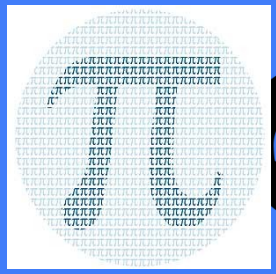
Línea a la izquierda →

Vdcha = Vnom

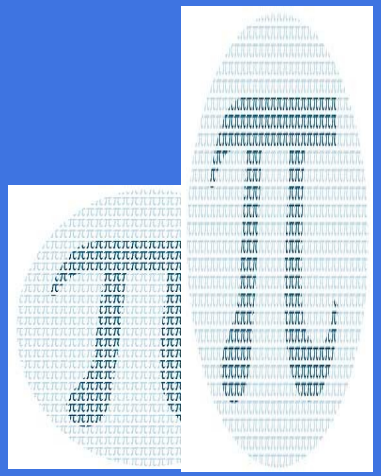
Vizq = Vnom - Plout

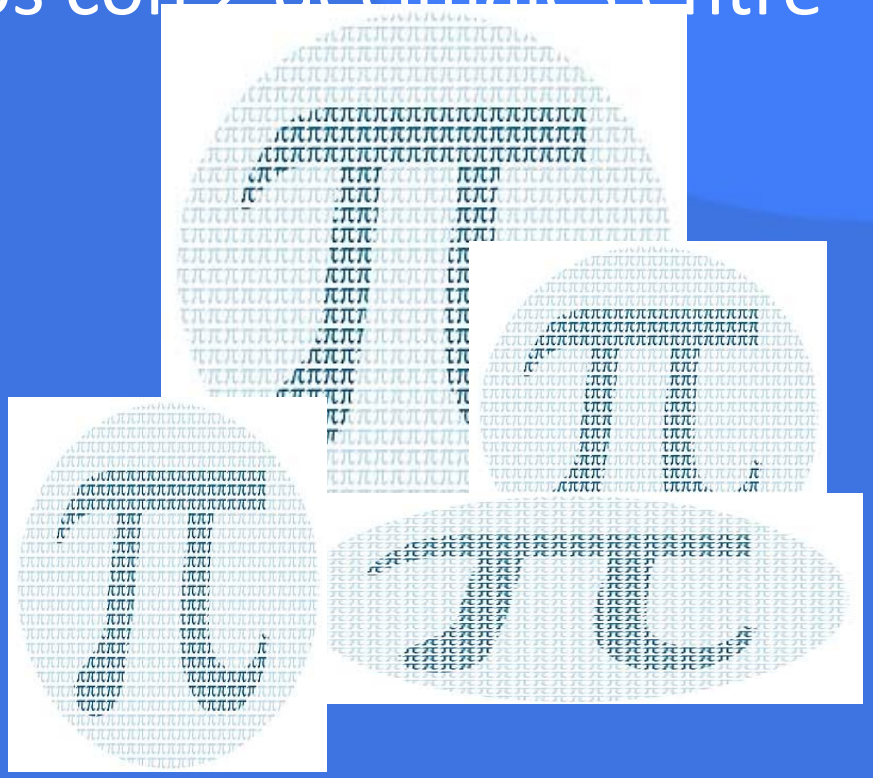
CYBERTECH 2015

Regulador PI



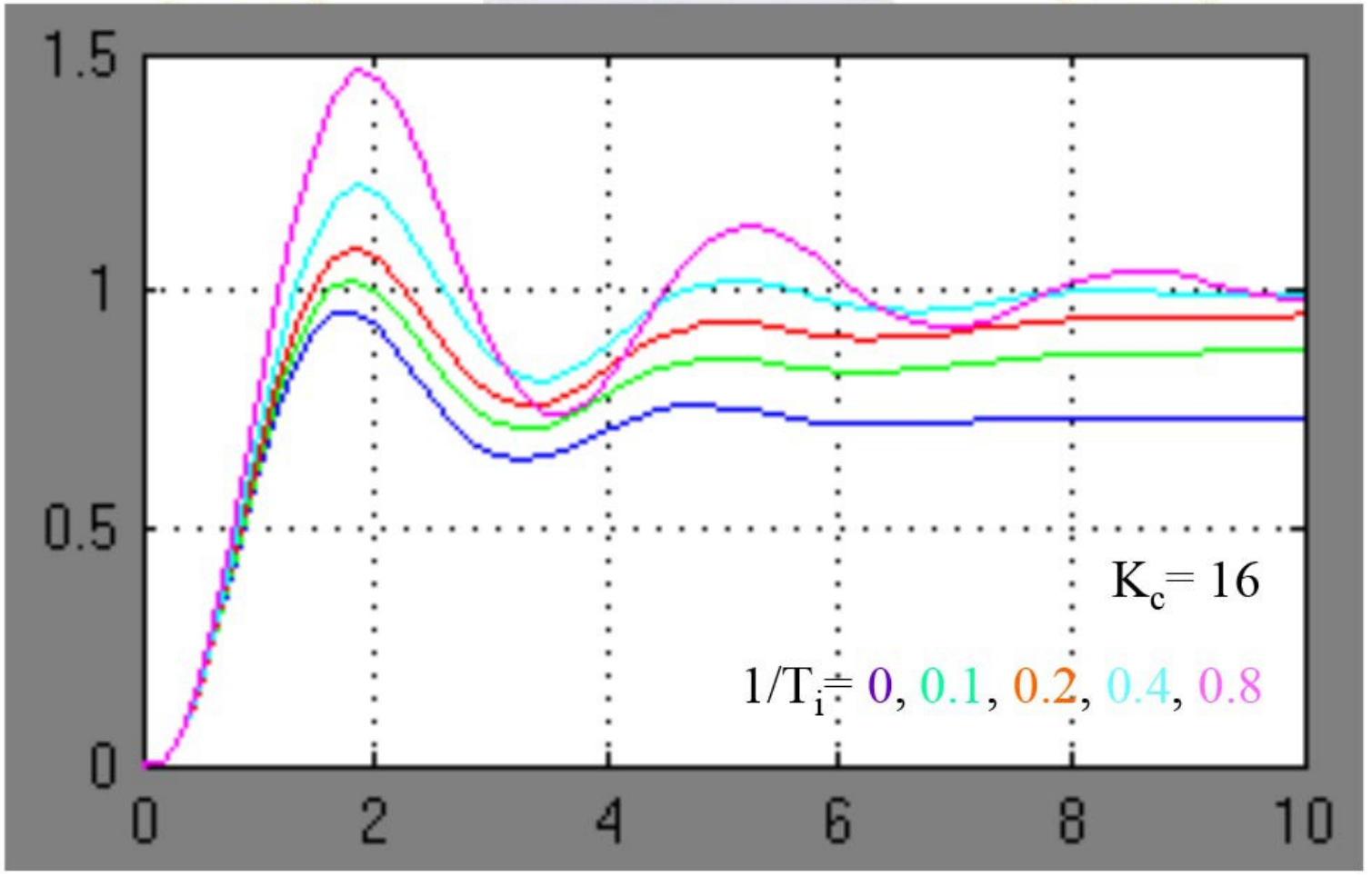
Ajustar la Ki (kp= la guay del ej. anterior ; kd = 0) para una velocidad ¾ Vmax (debajo de la d de speed) probando números con 2 decimales entre 0.01 y 0.10.





Regulador PI

$$U(s) = K_c(1 + \frac{1}{T_i s})\epsilon(s)$$



CYBERTECH
2015

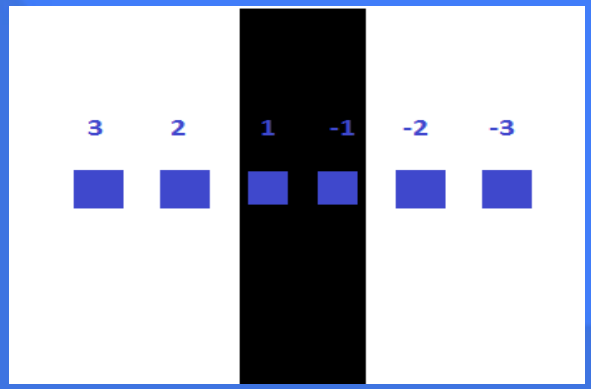
INDICE

- Intro
- Control todo o nada (Bang-Bang Control)
- Regulador P
- Regulador PI
- Regulador PD
- Regulador PID
- Ajuste PID
- Otros

CYBERTECH
2015

Regulador PD

- La derivada se anticipa al error.
- Mayor diferencia de error → respuesta más brusca.
    * Reduce las oscilaciones. :)
    * La respuesta es más lenta. :(



- Más sensible a ruido y/o lecturas erróneas.

$$u(t) = K_p e(t) + K_p T_d \frac{\partial e(t)}{\partial t}$$

Kp*Td → Kd

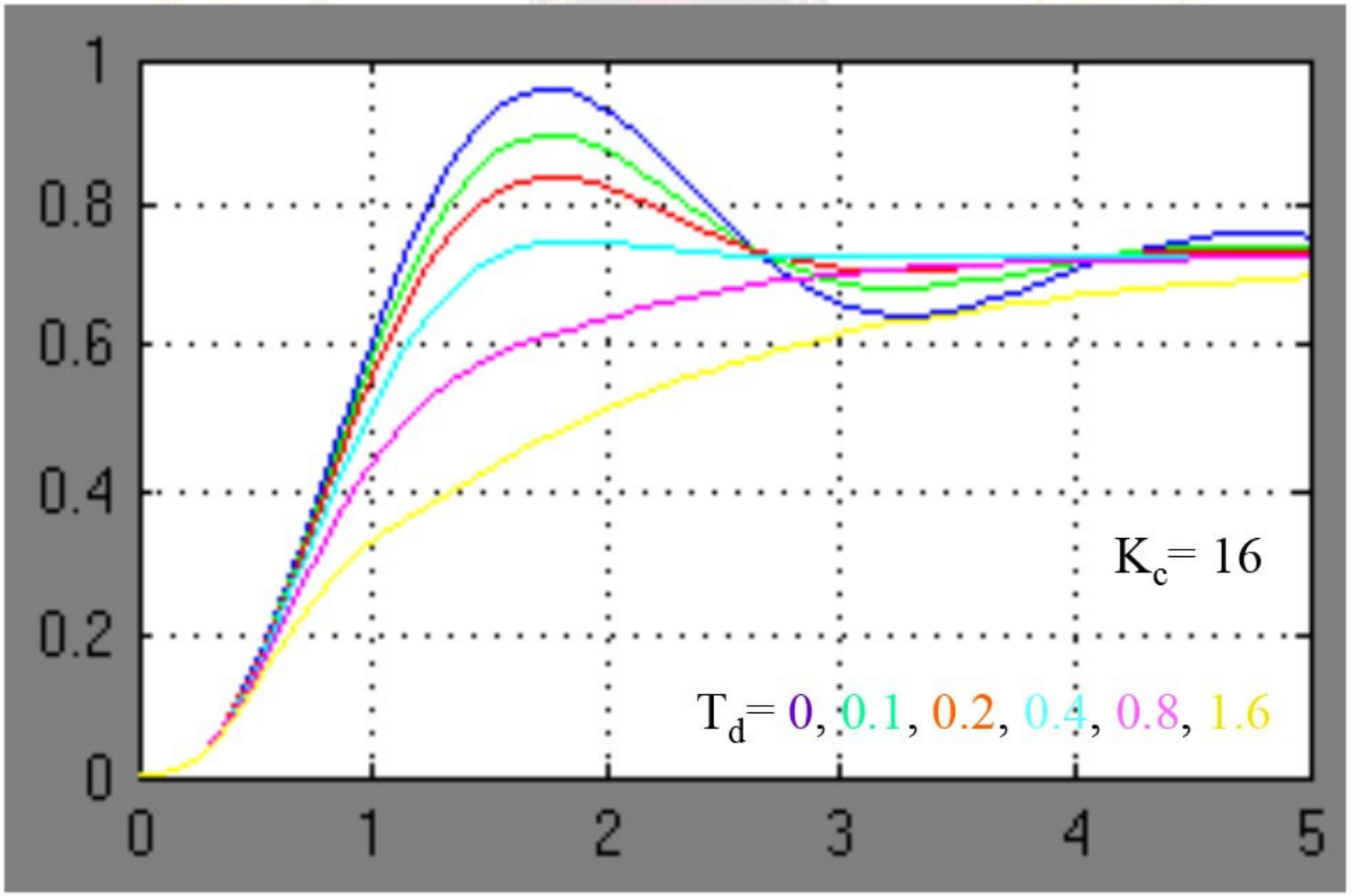
CYBERTECH
2015

Regulador PD

Ajustar la Kd (kp= la guay del ej. anterior2 ; ki = 0) para una velocidad ¾ Vmax (debajo de la d de speed) probando números con 2 decimales entre 1 y 10.

Regulador PD

$$U(s) = K_c(1 + T_d s)\epsilon(s)$$



CYBERTECH
2015

INDICE

- Intro
- Control todo o nada (Bang-Bang Control)
- Regulador P
- Regulador PI
- Regulador PD
- Regulador PID
- Ajuste PID
- Otros

CYBERTECH 2015

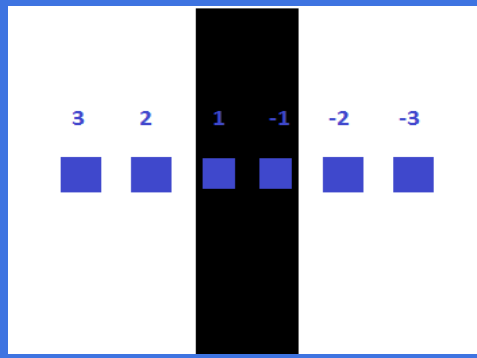
Regulador PID

- El más usado.
- Compuesto de 3 partes: P (proporcional), I (integral), D (derivativa).

Cada una influye en la respuesta de los motores según la siguiente fórmula:

$$u(t) = K_p e(t) + \frac{K_p}{T_i}\int_0^t e(t)\partial t + K_p T_d \frac{\partial e(t)}{\partial t}$$

Kd



CYBERTECH 2015

Regulador PID

```
-- Constantes PID:
   --tiempoViejo=0
   PrevError=0
   ErrorAcumu=0
   Kp=4
   Ki=1
   Kd=1
```

```
diffTime = 10 -- Diferencia de tiempos entre errores calculados.
           -- En la simulacion vale 10.
           -- En robots reales hay que hallarla (función millis() de Arduino).
           -- tiempoNuevo=millis();
           -- diffTime = tiempoNuevo - tiempoViejo

Derivada=(ErrorAhora - PrevError)/diffTime
Integral=Integral + ErrorAhora

RespuestaMotor = Kp*ErrorAhora + Ki* Integral + Kd*Derivada

if RespuestaMotor<0 then
   rightV=Ref_speed+RespuestaMotor
end

if RespuestaMotor>0 then
   leftV=Ref_speed-RespuestaMotor
end
--tiempoViejo = tiempoNuevo
PrevError=ErrorAhora
```

CYBERTECH
2015

Regulador PID

Re-ajustar Kd y Ki (kp= la guay del ej. anterior3) para una velocidad ¾ Vmax (debajo de la d de speed) probando números en los rangos que os dijimos antes.

CYBERTECH
2015

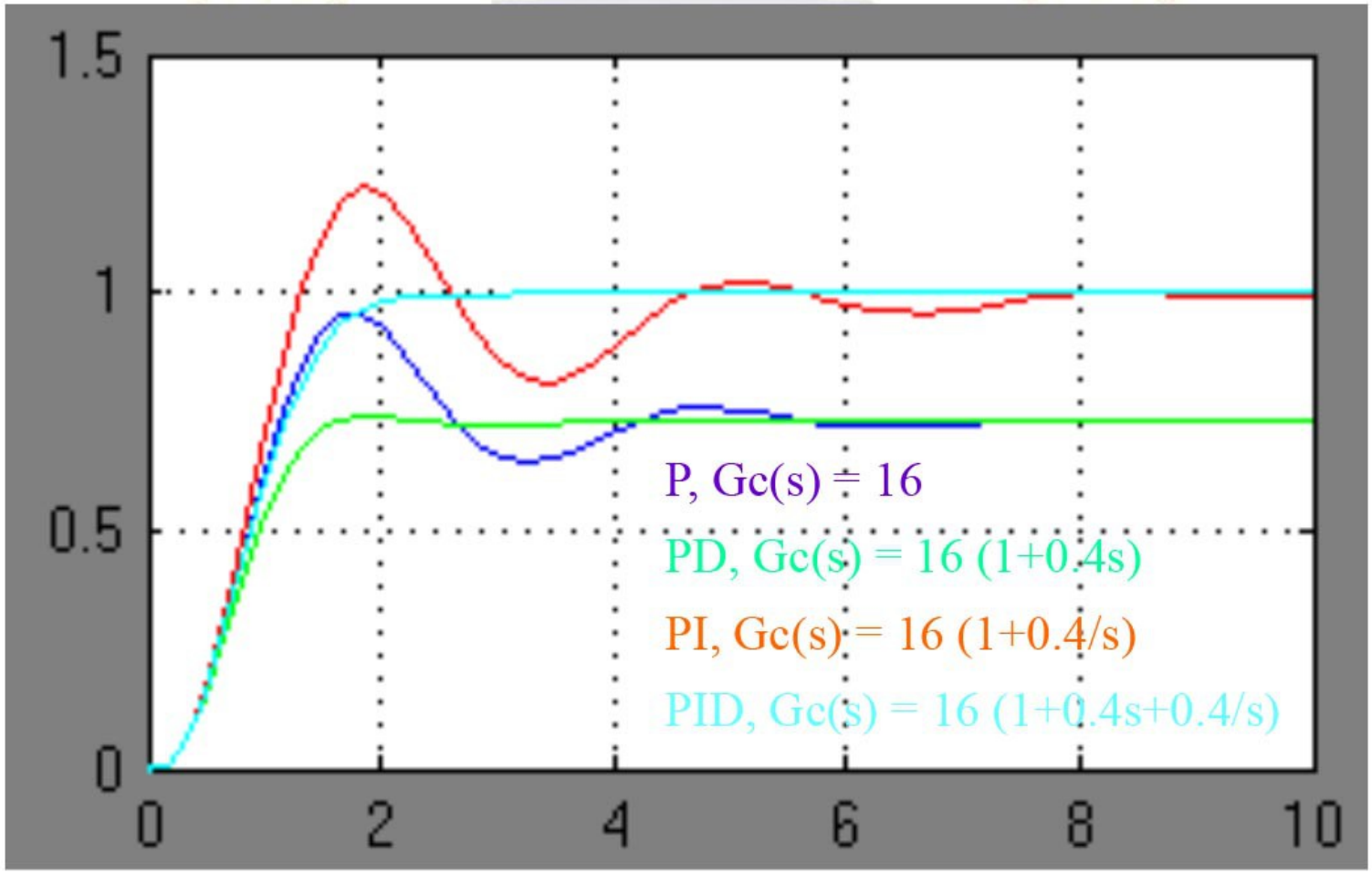
Regulador PID

- VENTAJAS:
  - *Altas velocidades.
  - *Anticipación y acumulación del error.
- DESVENTAJAS:
  - *Alta complejidad de ajuste.
  - *Mejor cuantos más sensores ($$).

CYBERTECH
2015

Regulador PID

$$U(s) = K_c(1 + \frac{1}{T_i s} + T_d s)\epsilon(s)$$



CYBERTECH
2015

ÍNDICE

- Intro
- Control todo o nada (Bang-Bang Control)
- Regulador P
- Regulador PI
- Regulador PD
- Regulador PID
- Ajuste PID
- Otros

CYBERTECH
2015

Ajuste PID

1.- Ajustar la P (Kp).
    Que “siga” la línea con bastante oscilación.
2.- Ajustar ligeramente la D.
    Que mejore respecto a la P sola.
3.- Afinar el ajuste con D e I.
    Modificar D. Introducir I.

Y probar, probar, probar hasta que os sangren los ojos. :D

*Aumentar la D reduce el efecto de la I y viceversa.

CYBERTECH 2015

ÍNDICE

- Intro
- Control todo o nada (Bang-Bang Control)
- Regulador P
- Regulador PI
- Regulador PD
- Regulador PID
- Ajuste PID
- Otros

CYBERTECH 2015

Otros

- Control adaptativo:
  *Se basa en variar las constantes Kp, Ki y Kd según las condiciones del sistema.

- Fuzzy logic:
  *No es lógica binaria. Difícil de explicar.

- Variables de estado

-.....

MUCHO MÁS TIEMPO Y CAPACIDAD DE PROCESAMIENTO

CYBERTECH
2015

FIIIIINNNN!!! :D