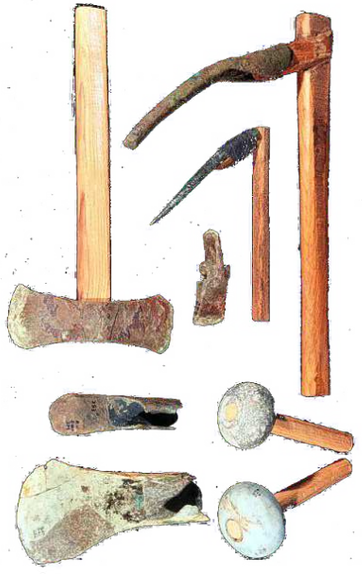
Electrónica programable para óptica

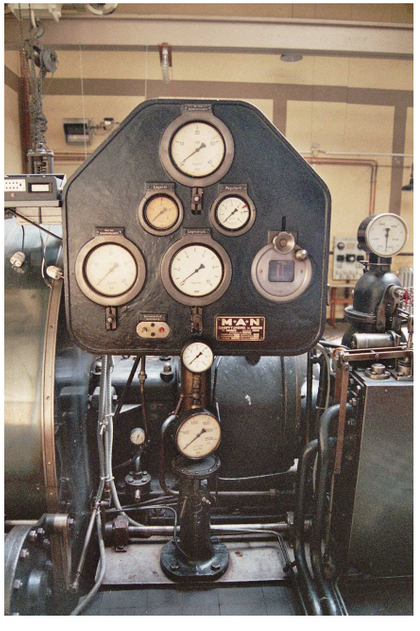
Marcelo Luda*, Matías Risaro, Jorge Codnia

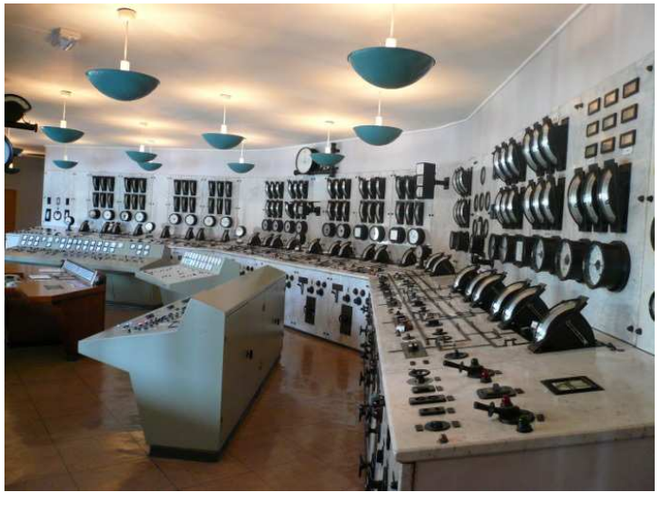
Láseres moleculares - Citedef

* mluda@citedef.gob.ar

Tecnología e intrumentación





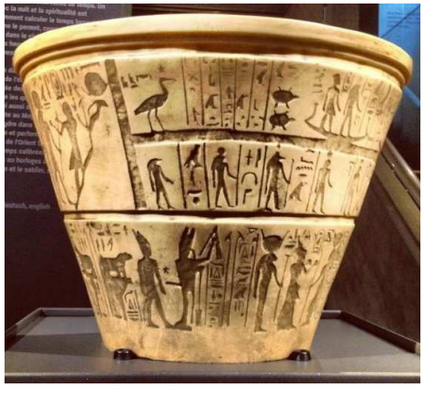


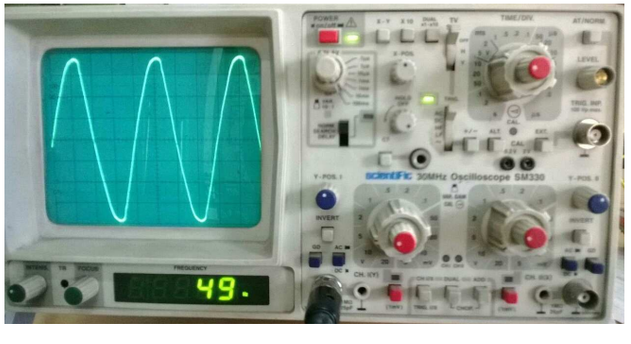
Tecnología

Herramientas

Desarrollo productivo

Instrumentación





Embebidos - ¿Industria 4.0?

"Era de la información"

Embebidos:
computadora
+
electrónica específica

Arduino

Raspberry Pi

Red Pitaya

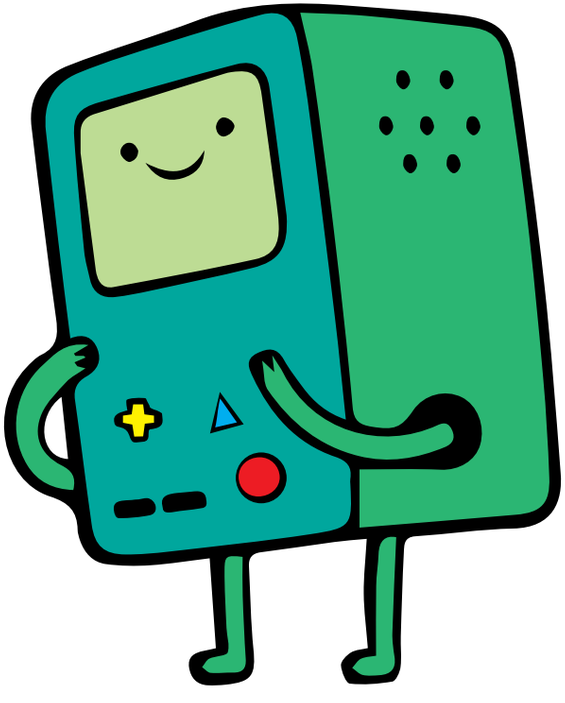
Resumen del taller de electrónica programable

Charla:
¿Que es SoC y FPGA?
Aplicaciones en óptica e instrumentación

mini-clase
Introducción a la programación en Verilog
Framework para FPGA en Red Pitaya

Actividad práctica:
desarrollamos una herramienta
Programar la capa FPGA de una Red Pitaya para procesamiento de señales

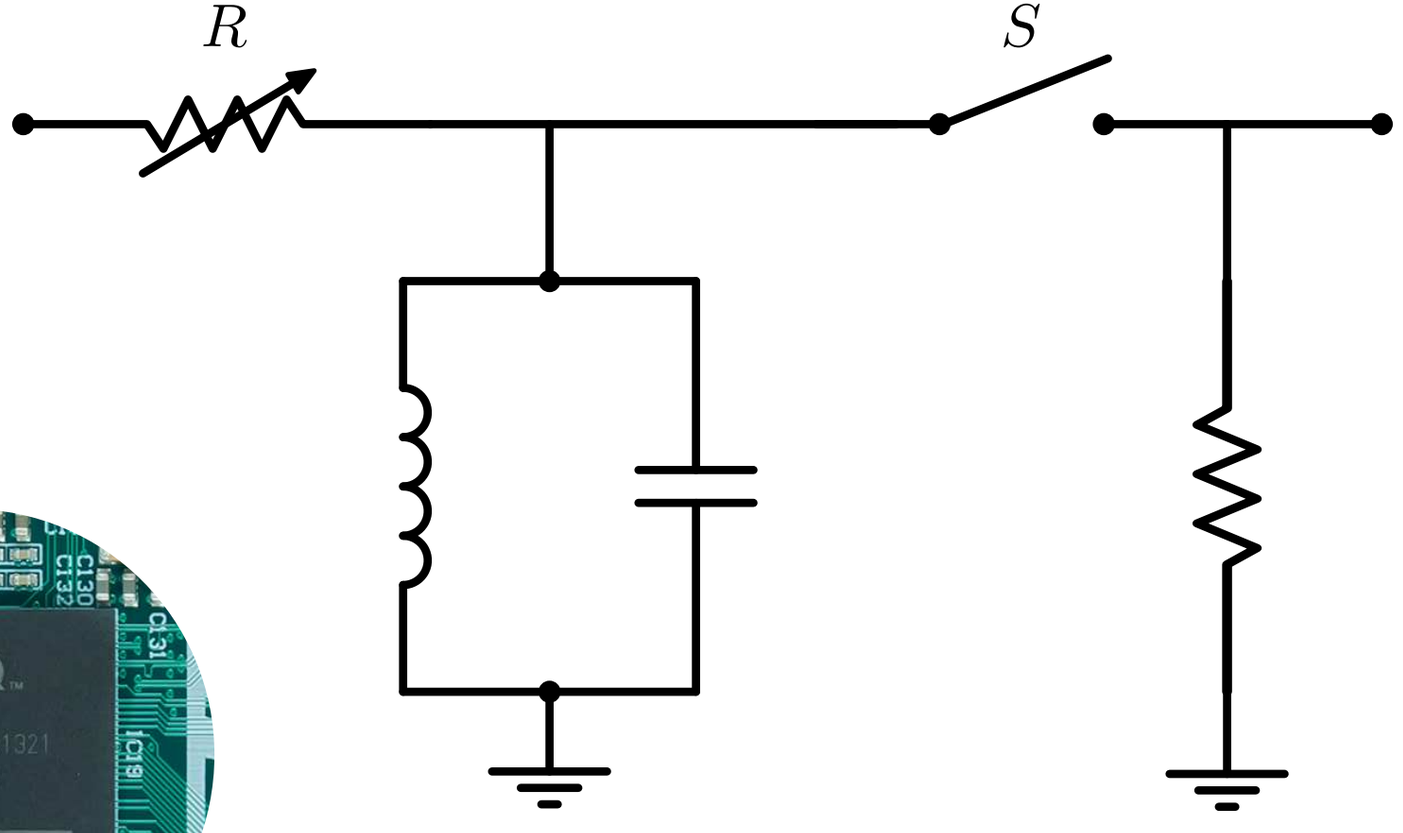
¿Que es Electrónica Programable?



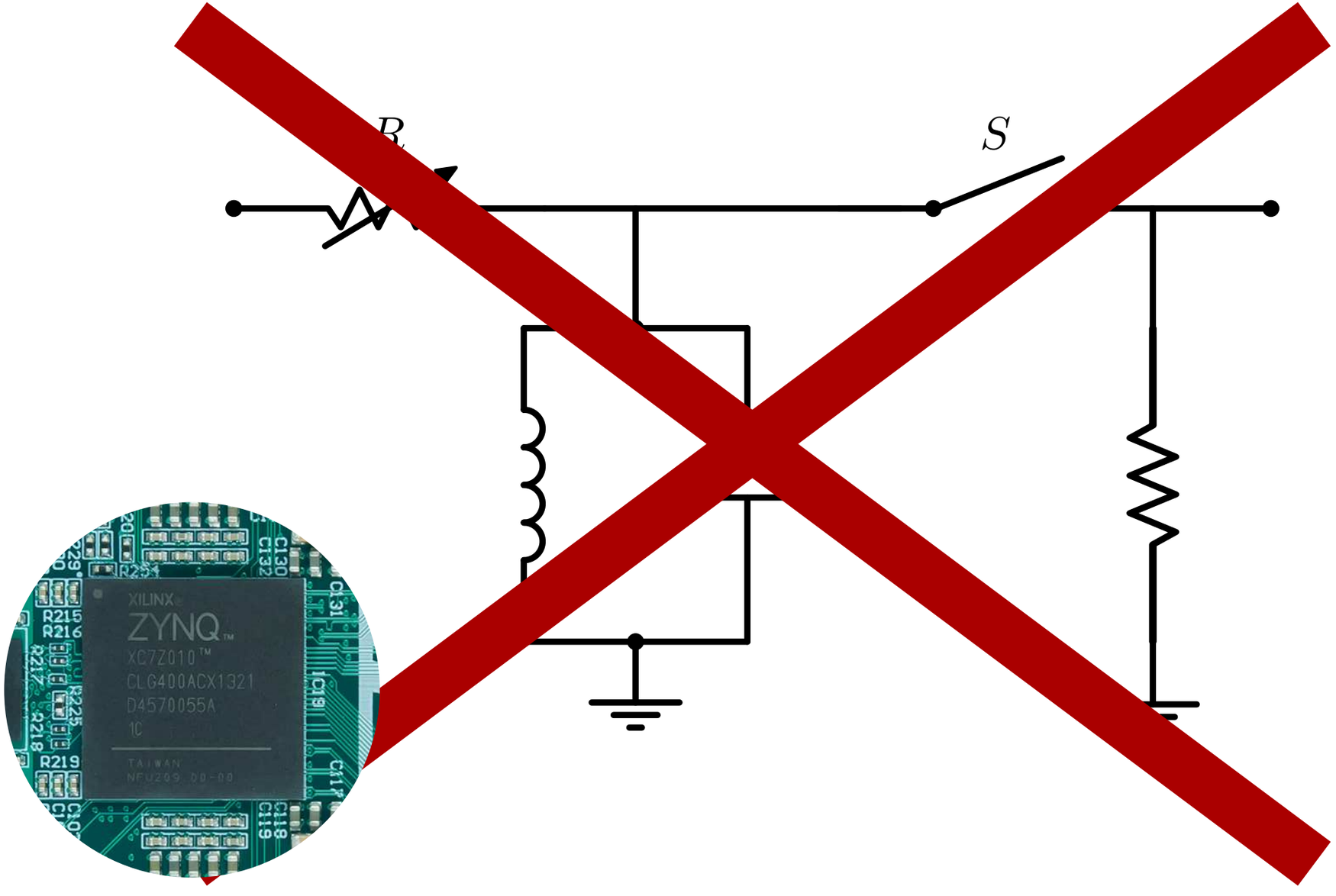
¿Que es FPGA?

"**F**ield-**P**rogrammable **G**ate **A**rray"

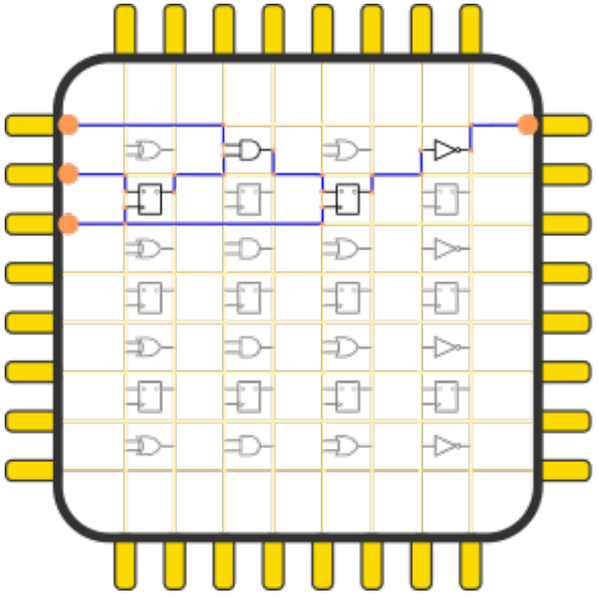
¿Que es Electrónica Programable?

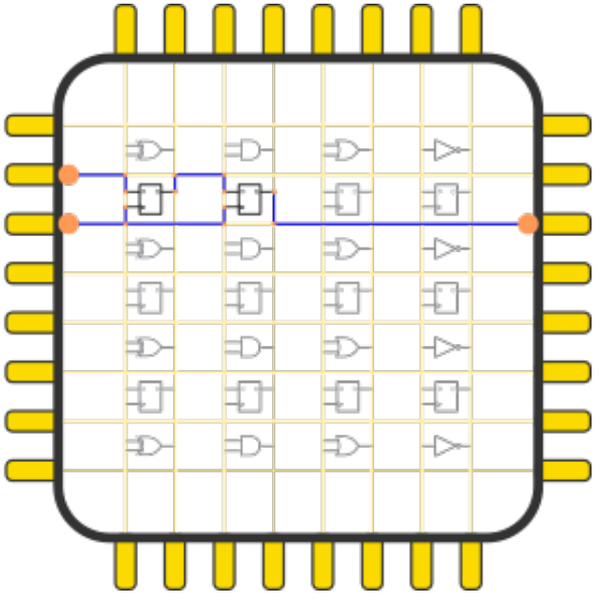


¿Que es Electrónica Programable?



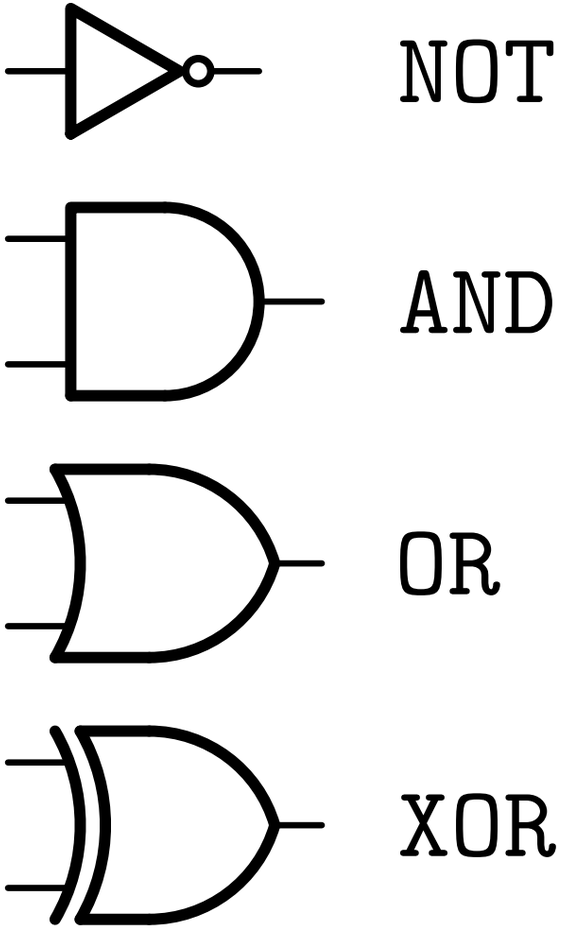
¿Que es Electrónica Programable?

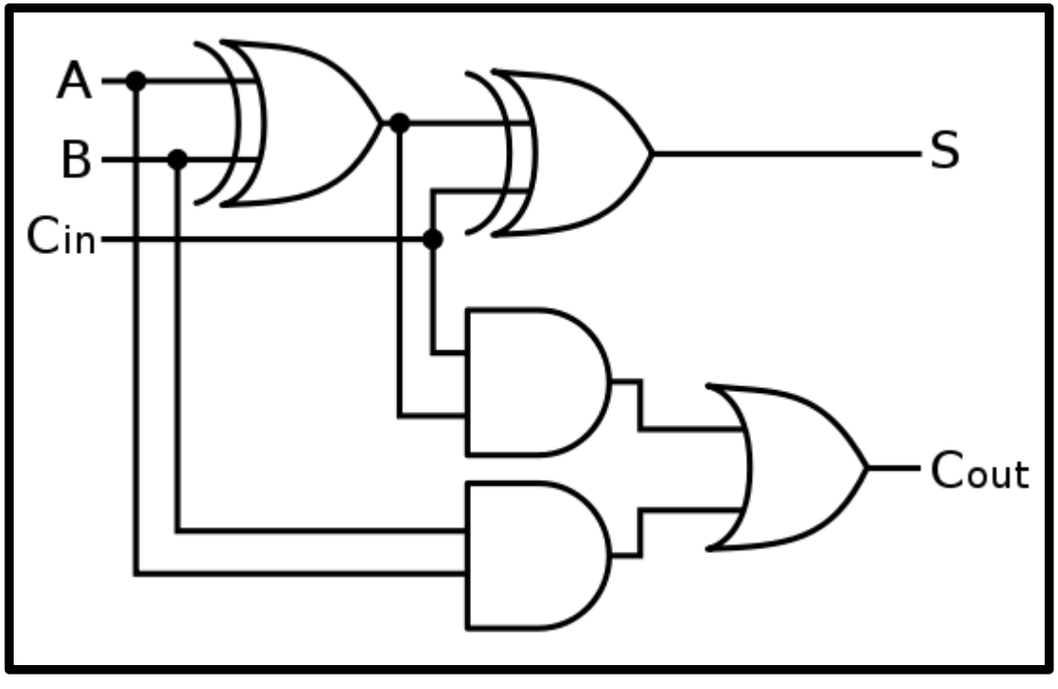




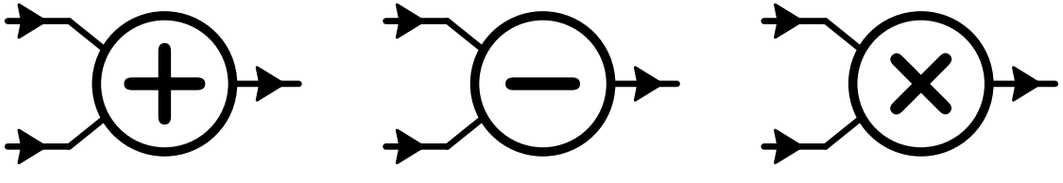
- Cableado físico de componentes
- Circuitos de lógica digital

Lógica digital



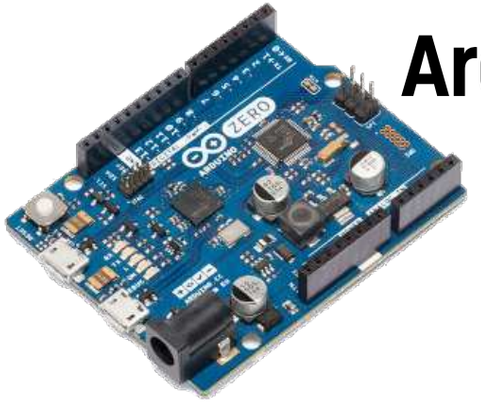


Circuito sumador



¿Para que sirve?

Instrumentación con microelectrónica

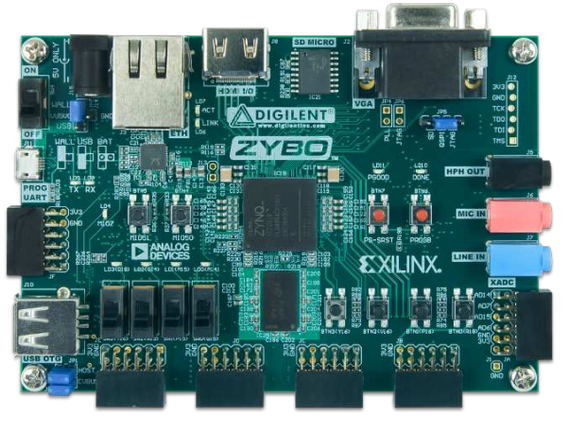


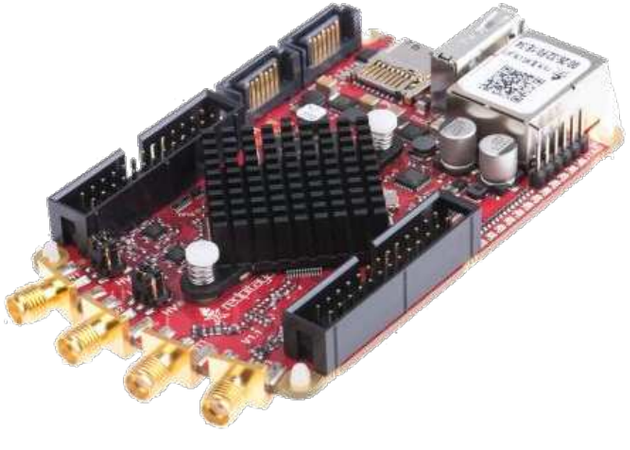
Arduino
Microcontrolador

Raspberry
Microprocesador

Nexys3
FPGA "pura"

Zybo
Micro+FPGA

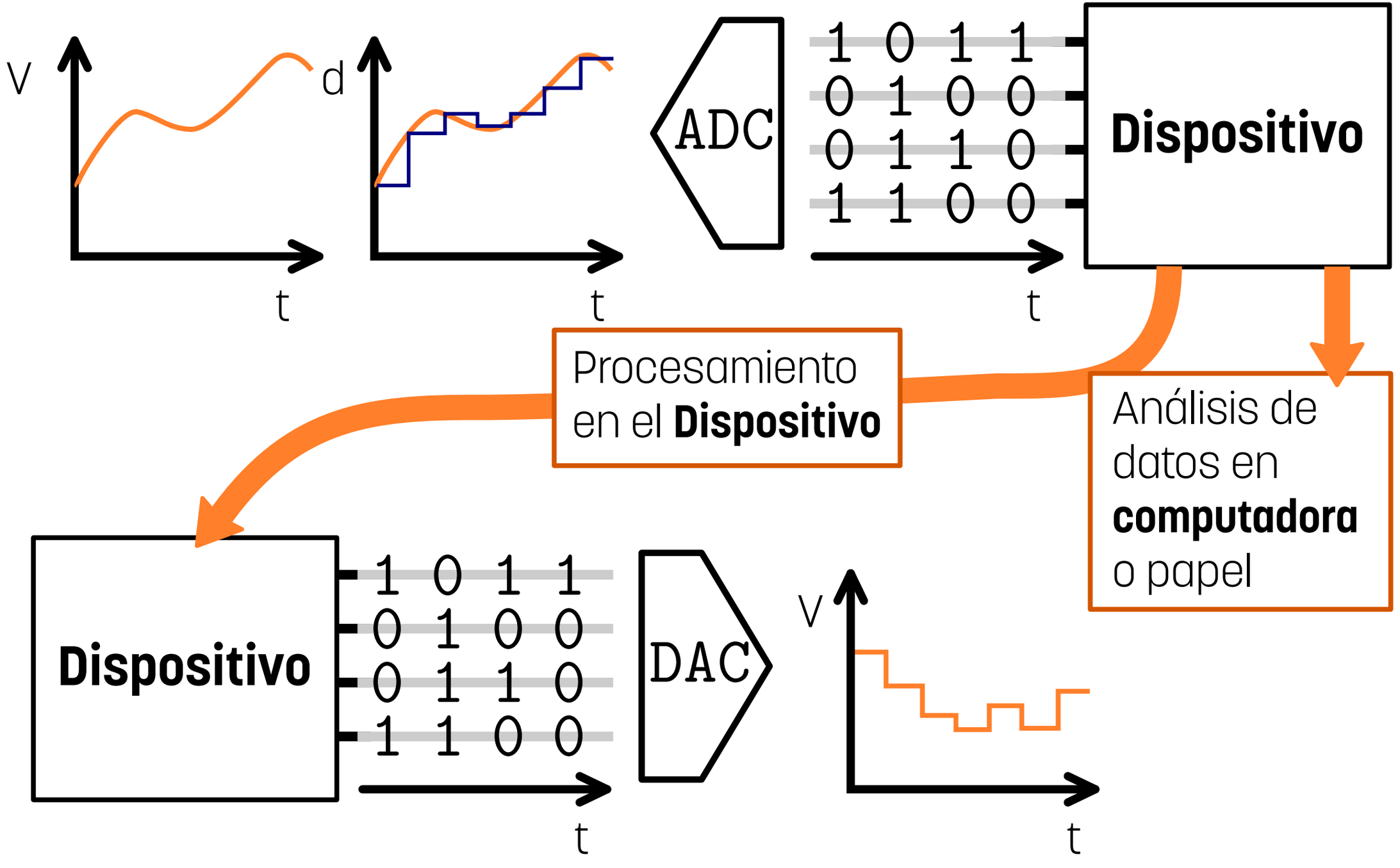




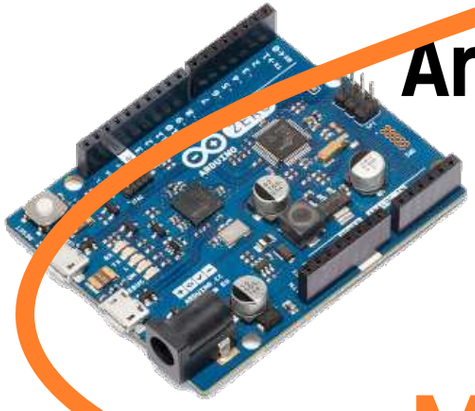
Red Pitaya
Micro+FPGA

Orientada a procesamiento analógico
Parte en software libre → Comunidad

Adquisición, procesamiento y control



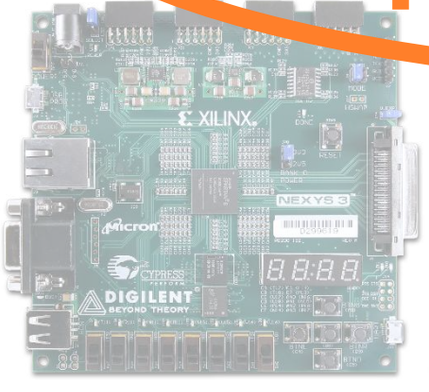
Instrumentación con microelectrónica



Arduino
Microcontrolador

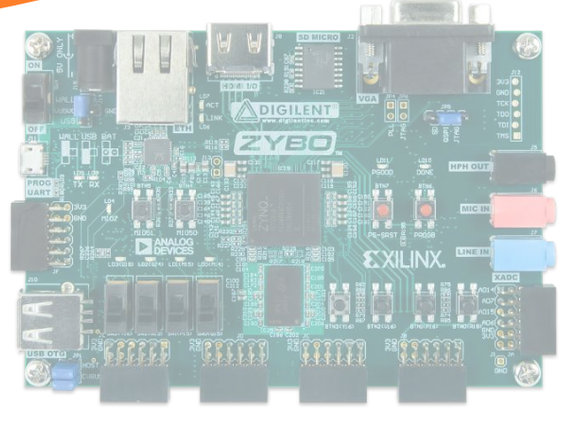
Raspberry
Microprocesador

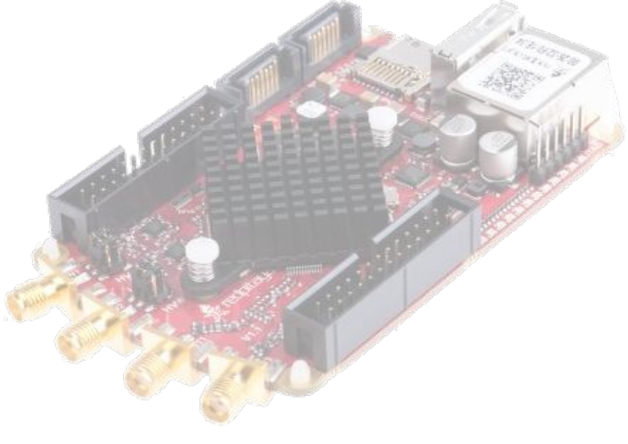
Microprocesadores



Nexys3
FPGA "pura"

Zybo
Micro+FPGA

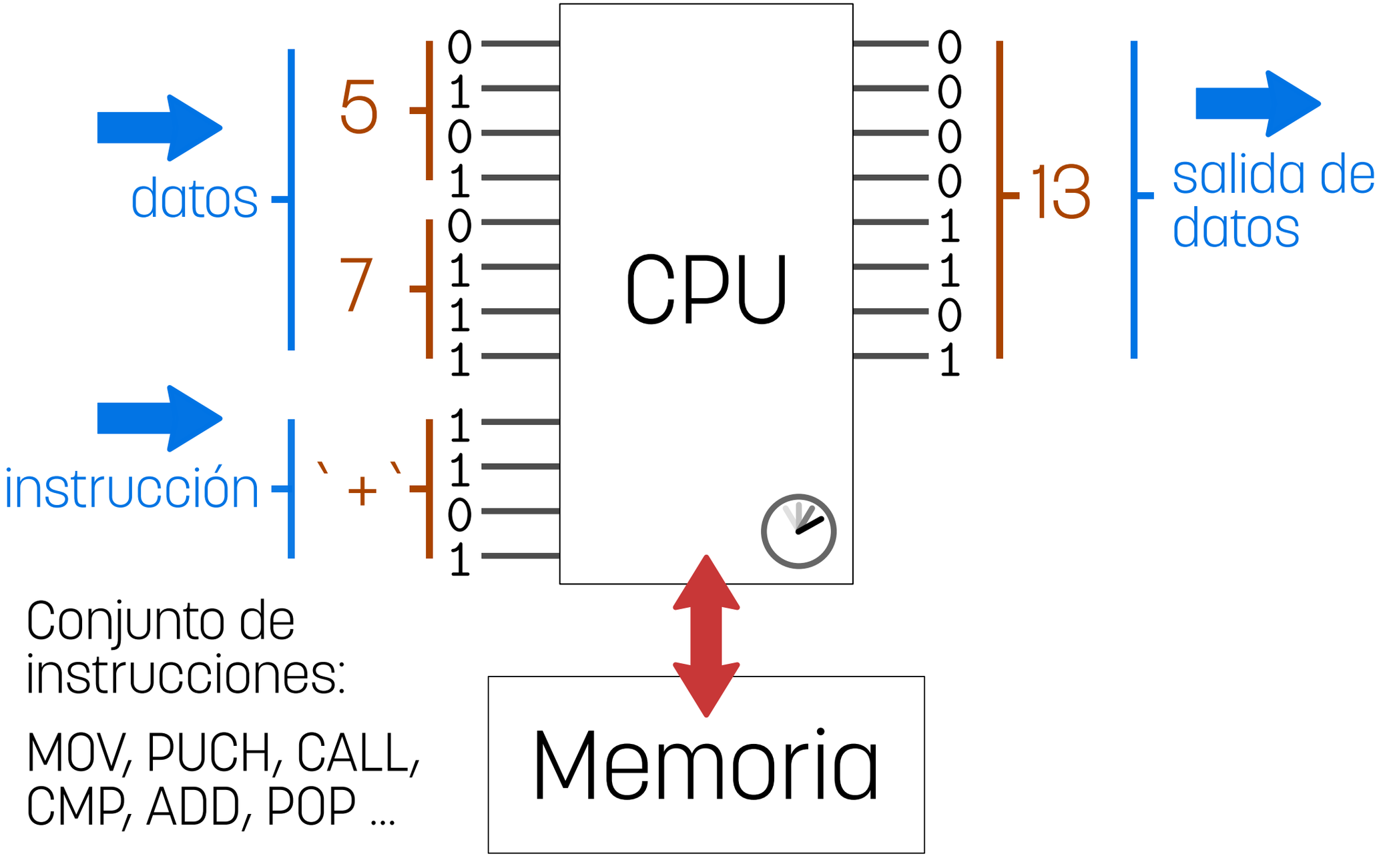




Red Pitaya
Micro+FPGA

Orientada a procesamiento analógico
Parte en software libre → Comunidad

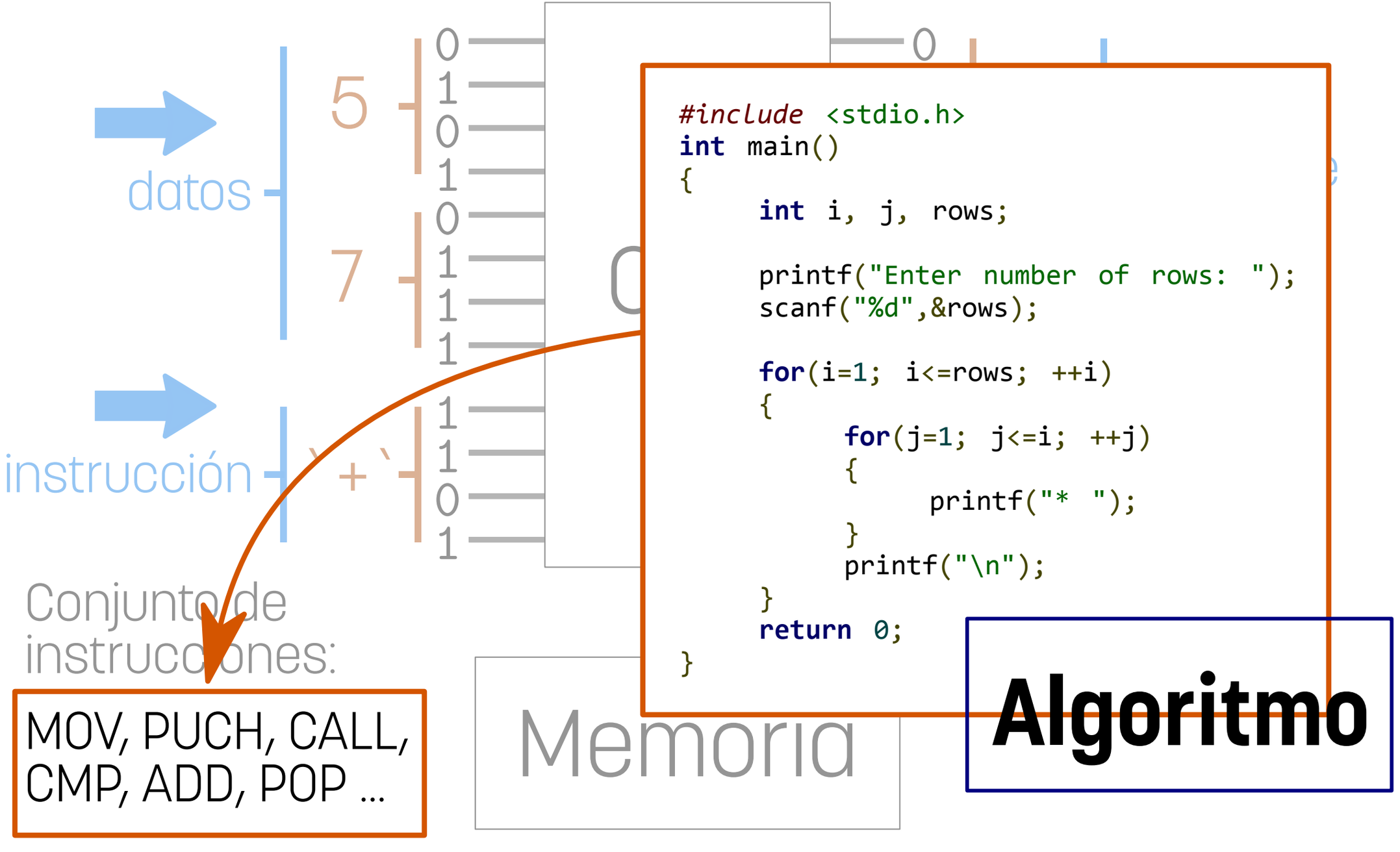
Microprocesador



Conjunto de instrucciones:

MOV, PUCH, CALL, CMP, ADD, POP ...

Microprocesador




```c
#include <stdio.h>
int main()
{
      int i, j, rows;

      printf("Enter number of rows: ");
      scanf("%d",&rows);

      for(i=1; i<=rows; ++i)
      {
            for(j=1; j<=i; ++j)
            {
                  printf("* ");
            }
            printf("\n");
      }
      return 0;
}
```

Microprocesador // FPGA

MICROPROCESADORES

Programanción procedimental

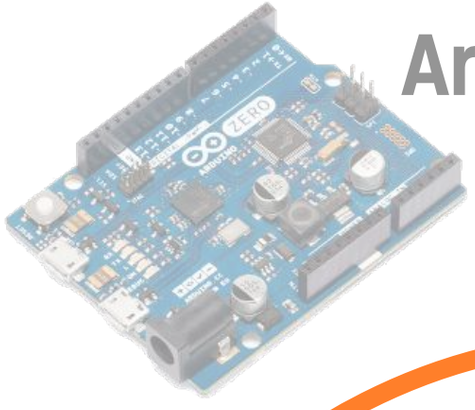
Una instrucción a la vez

Programación muy versátil y flexible
- float, int, array, complex, etc
- Se puede auto-modificar

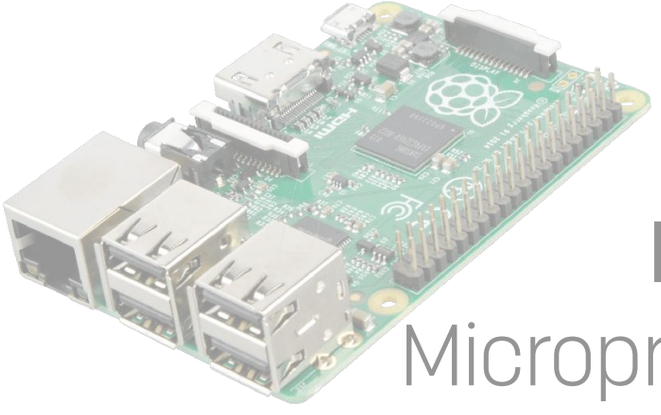
Limitado por reloj

"No-determinista": interrupciones, SO, etc

Instrumentación con microelectrónica



Arduino
Microcontrolador

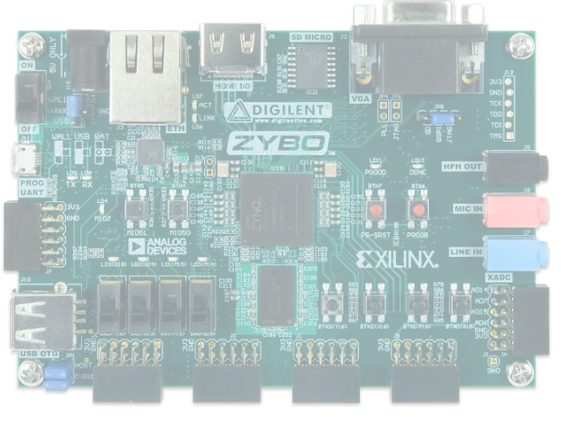


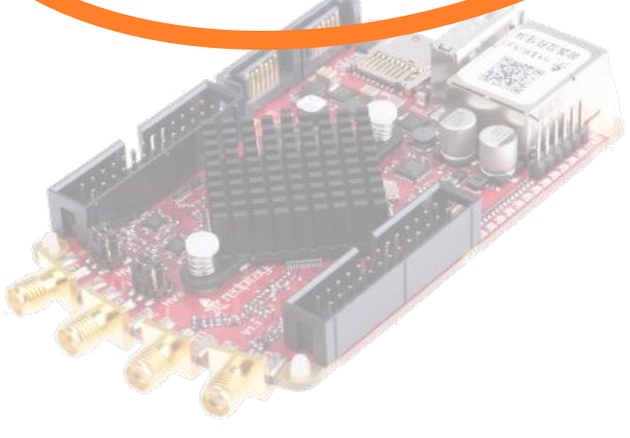
Raspberry
Microprocesador

Nexys3
FPGA "pura"

FPGA "pura"

Zybo
Micro+FPGA

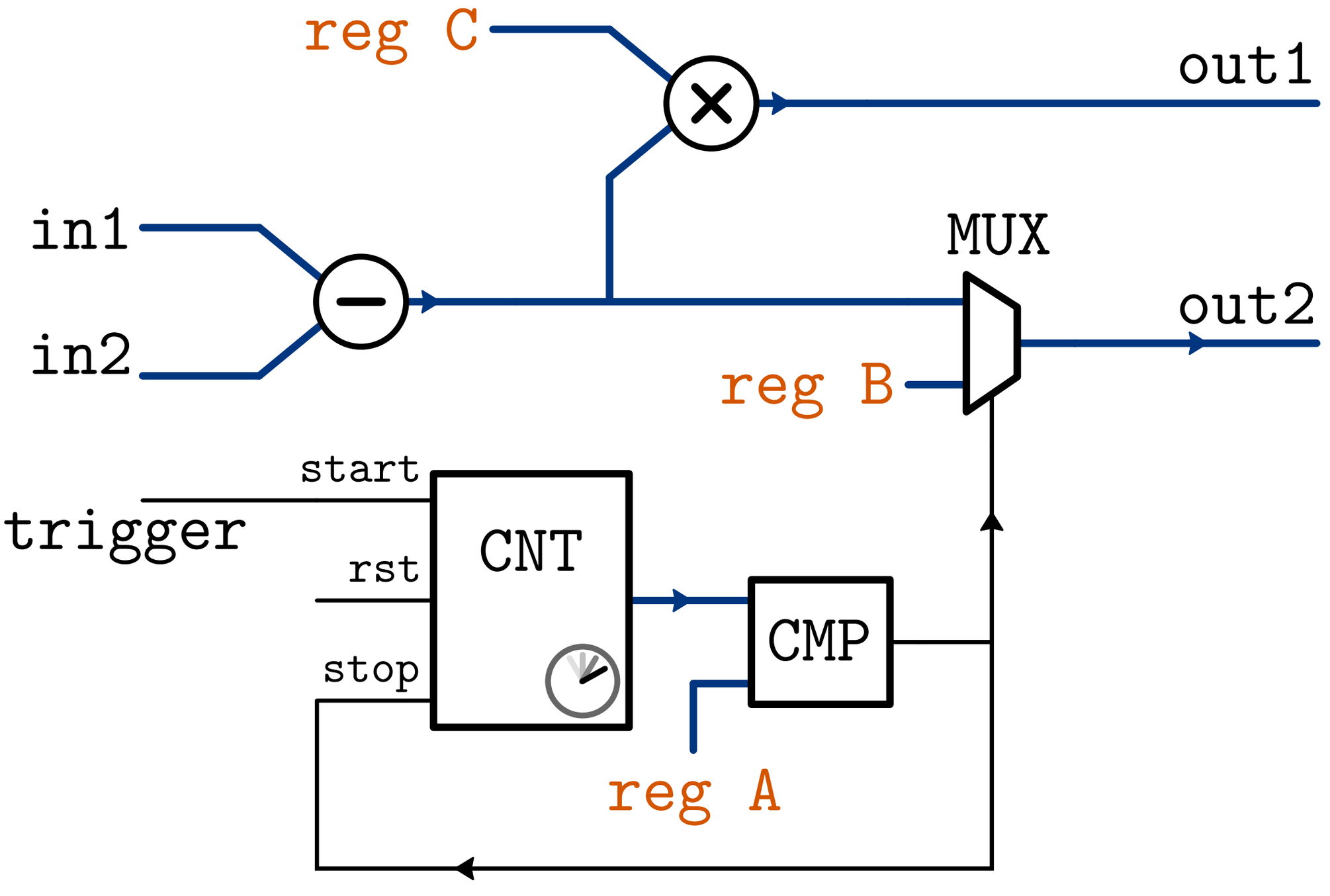


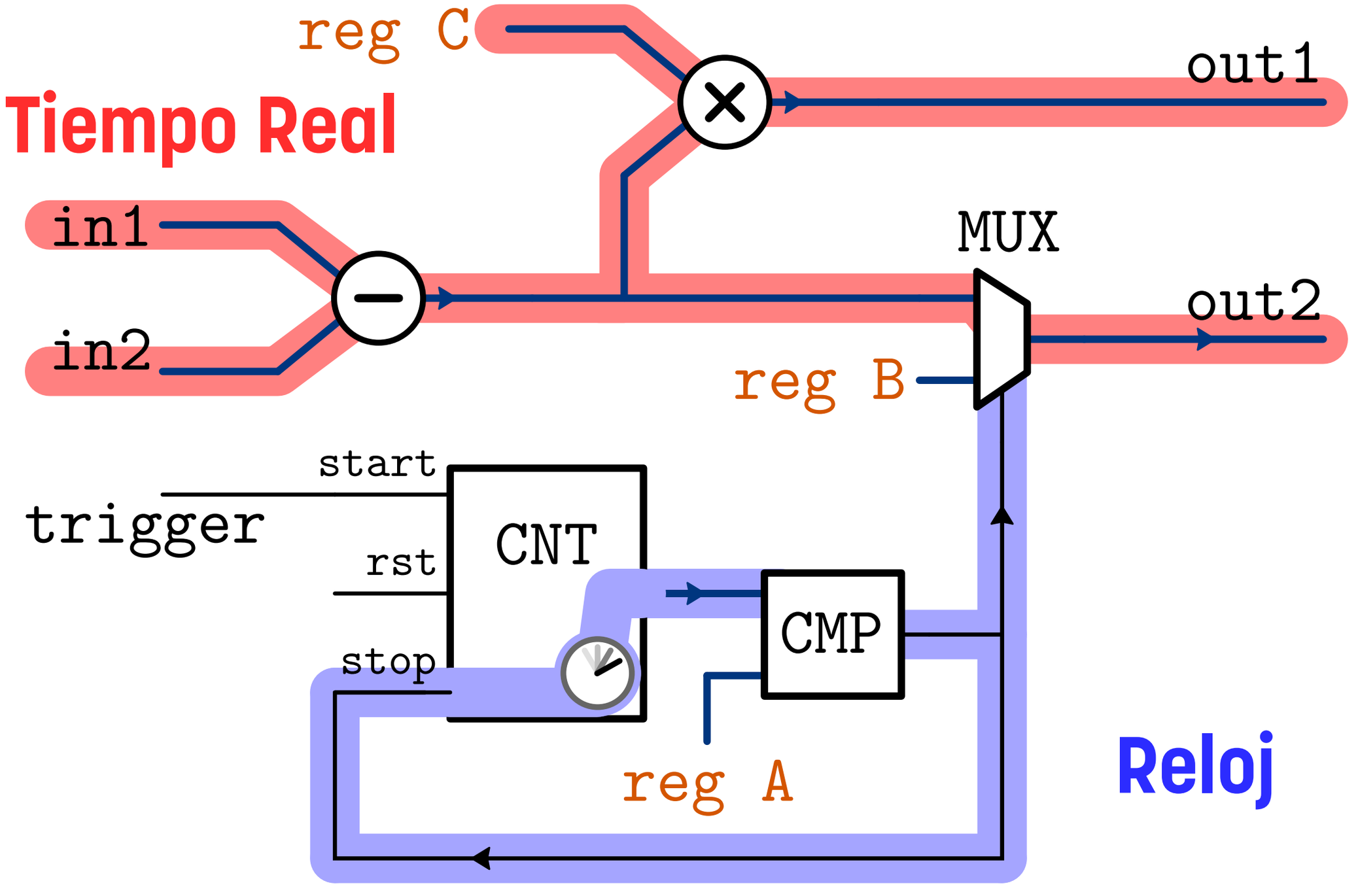


Red Pitaya
Micro+FPGA

Orientada a procesamiento analógico
Parte en software libre → Comunidad

FPGA

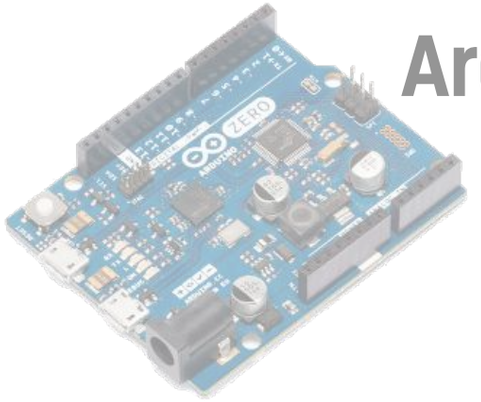

reg C
out1
in1
in2
MUX
out2
reg B
start
trigger
rst
CNT
stop
CMP
reg A


reg C
out1
Tiempo Real
in1
MUX
out2
in2
reg B
start
trigger
CNT
rst
CMP
stop
reg A
Reloj

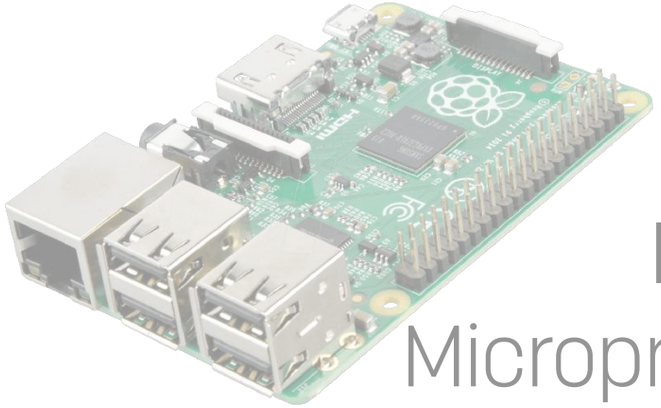
Microprocesador // FPGA

MICROPROCESADORES	FPGA
Programanción procedimental	Diseño de cableado: "flujo"
Una instrucción a la vez	Todo en paralelo
Programación muy versátil y flexible float, int, array, complex, etc Se puede auto-modificar	Diseño rígido y limitado int, uint registros → parametros → multiplexores
Limitado por reloj	Tiempo real (puede usar reloj)
"No-determinista": interrupciones, SO, etc	Determinista

Instrumentación con microelectrónica

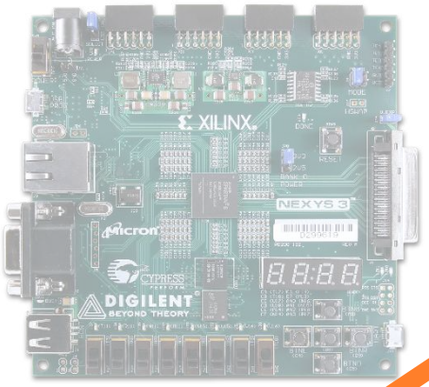


Arduino
Microcontrolador

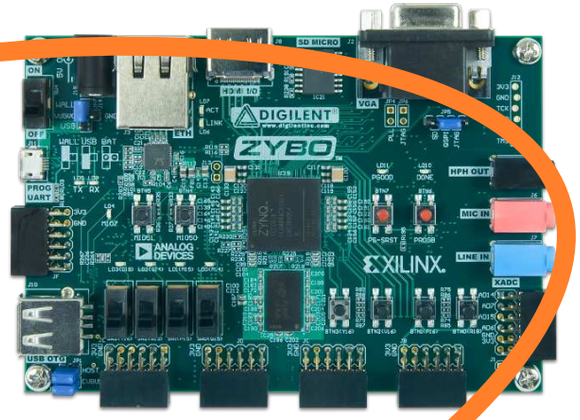


Raspberry
Microprocesador

FPGA + Micro



Nexys3
FPGA "pura"



Zybo
Micro+FPGA

Red Pitaya
Micro+FPGA

Orientada a procesamiento analógico
Parte en software libre → Comunidad

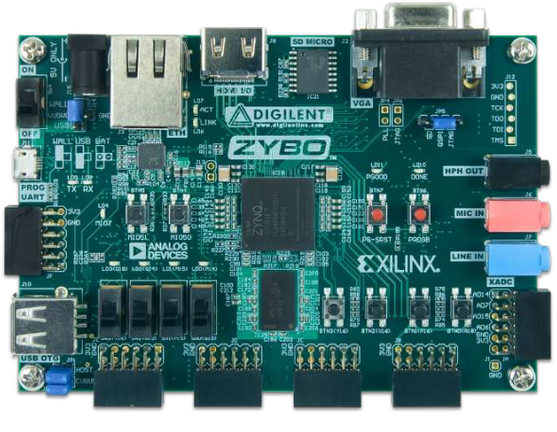
Micro + FPGA

Capa de Microprocesador
Programación versatil
Sistema operativo
Manejo de mermoria

Capa FPGA
Adquisición y procesamiento en tiepo real
Control mediante registros

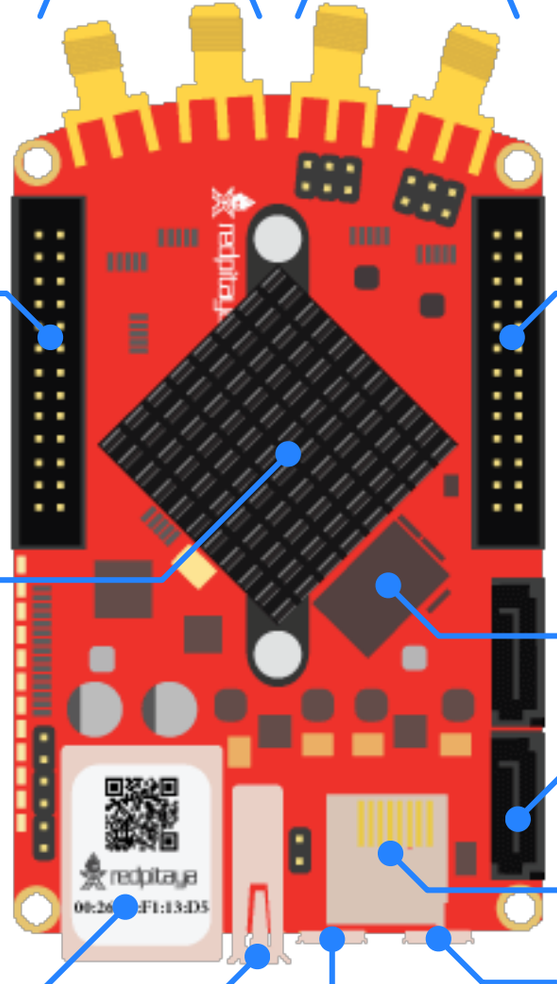
Red Pitaya
Micro+FPGA

Zybo
Micro+FPGA



Hardware Red Pitaya

Hardware cerrado
Software Libre



Fast analog outputs
125 MS/s, 14 bits, ±1 V

Fast analog inputs
125 MS/s, 14 bits, ±1 V

Digital extension
16 I/O ch., 125 MS/s, 3.3 V

Analog extension
~100 kS/s, ~12 bits, 0-1.8 V
4 SAR ADC ch., 4 PWM DAC ch.,

Microprocessor + FPGA
Dual core ARM-Cortex A9 Zync SoC

DDR3 RAM
512 MB

Daisy chain

microSD
OS drive & FPGA design

Ethernet
1 Gb/s

USB
OTG

micro USB
SO console

micro USB
Power in
5 V, 2 A

Osciloscopio
Generador de funciones
Filtros
Microcontrolador

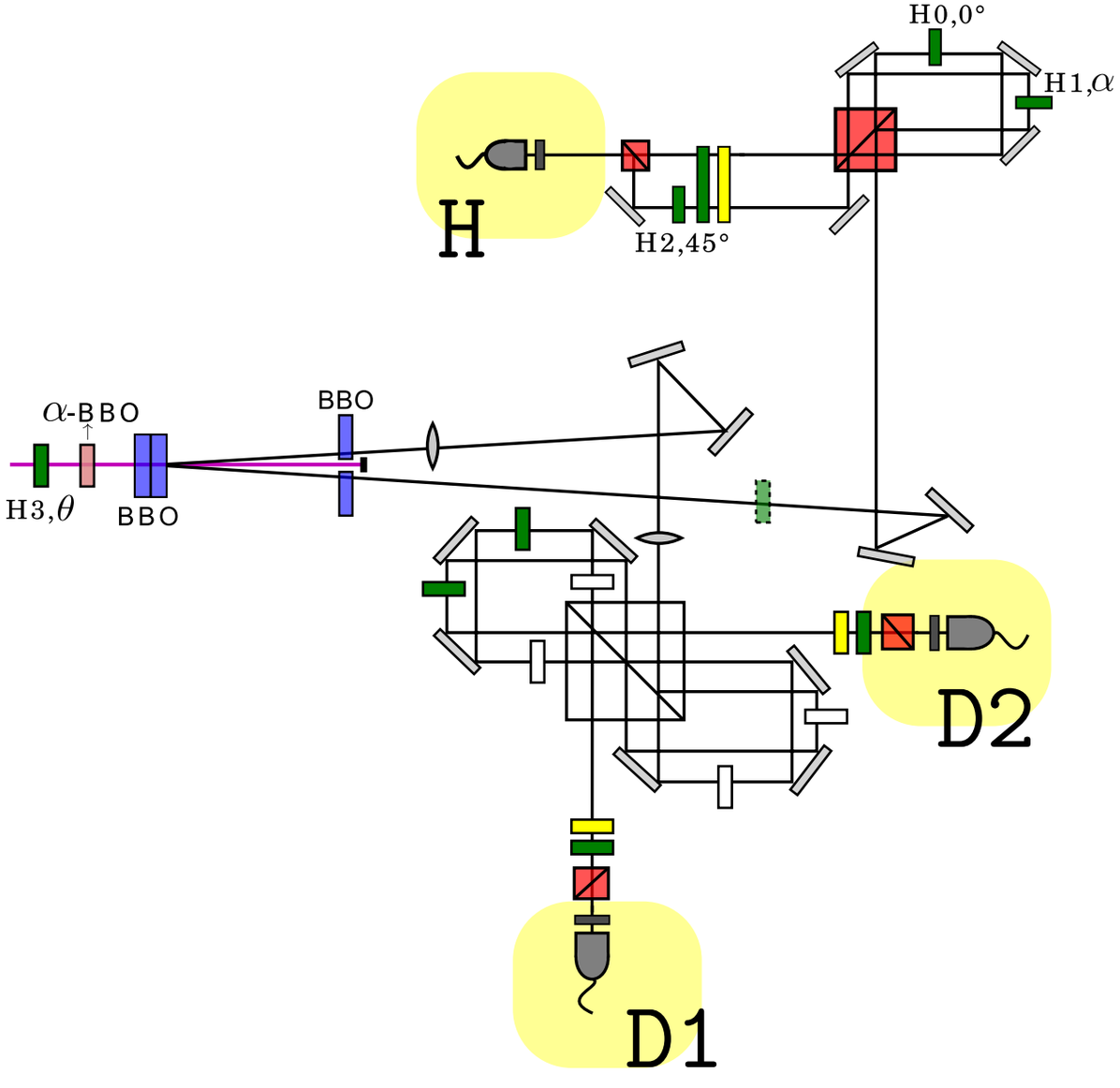
FPGA para óptica

Ejemplo de FPGA para óptica

Adquisición y pre-procesamiento

Experimento de teleportación cuántica*

Detector de coincidencias y conteo

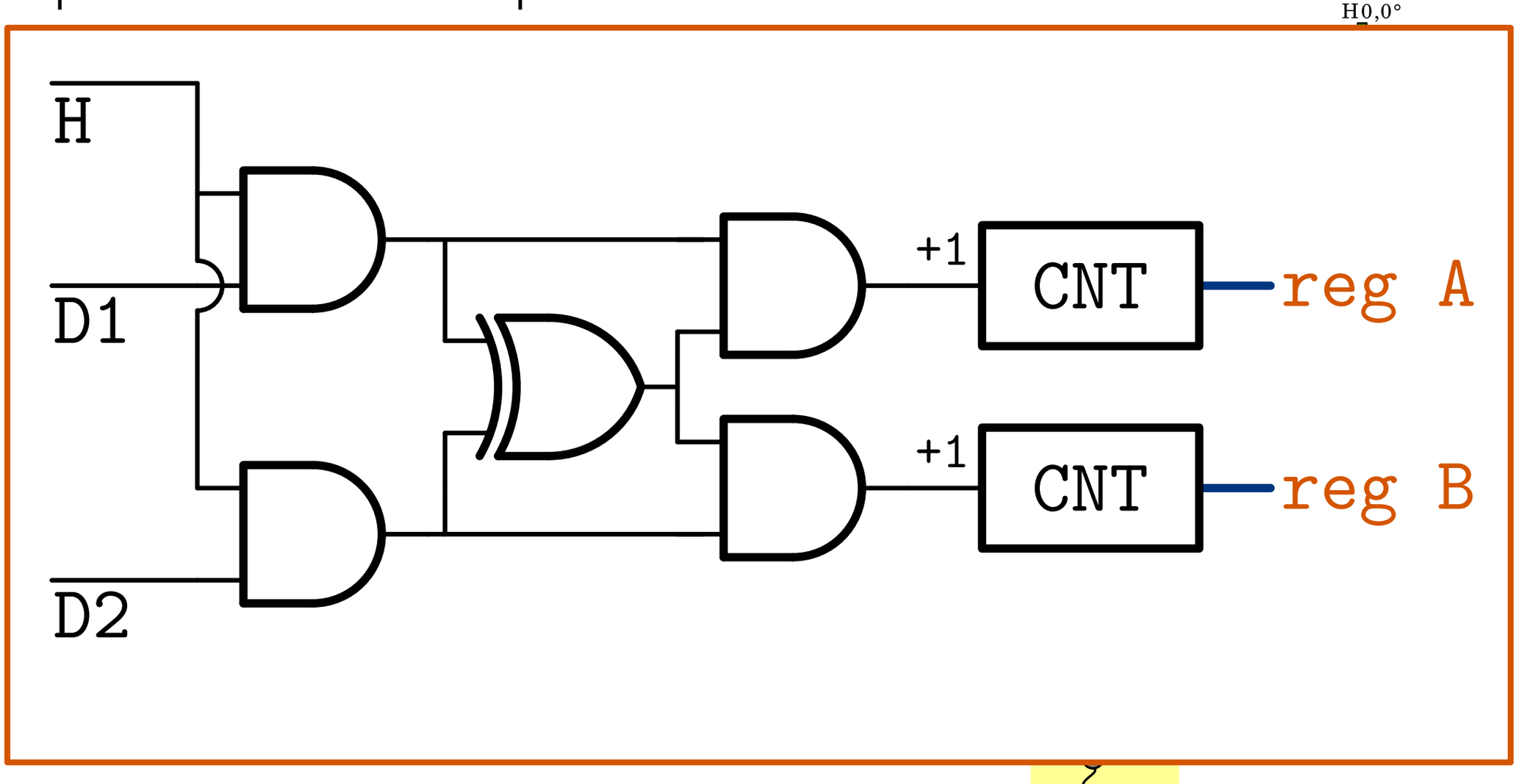


*Knoll et al. , arxiv:1602.07196

Ejemplo de FPGA para óptica

Adquisición y pre-procesamiento

Experimento de teleportación cuántica*



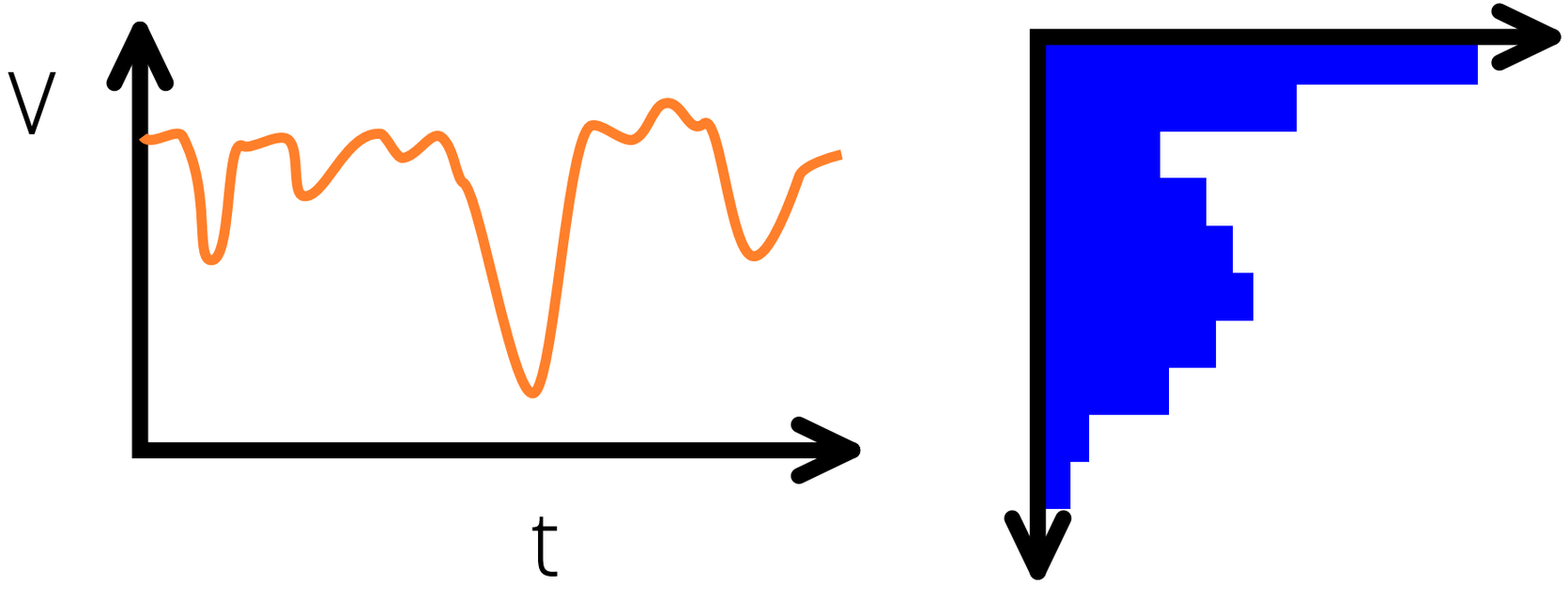
*Knoll et al. , arxiv:1602.07196

Ejemplo de FPGA para óptica

Adquisición y pre-procesamiento

Caracterización de fotomultiplicador

Para conteo de fotones

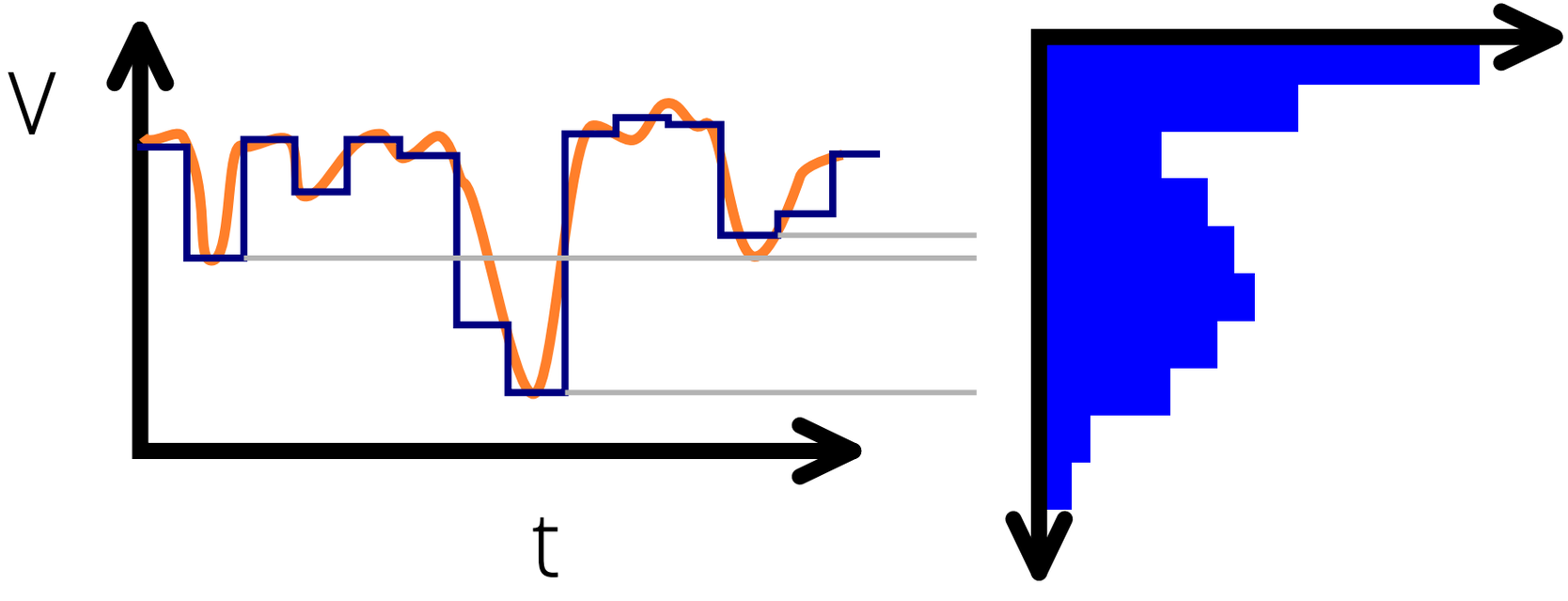


Ejemplo de FPGA para óptica

Adquisición y pre-procesamiento

Caracterización de fotomultiplicador

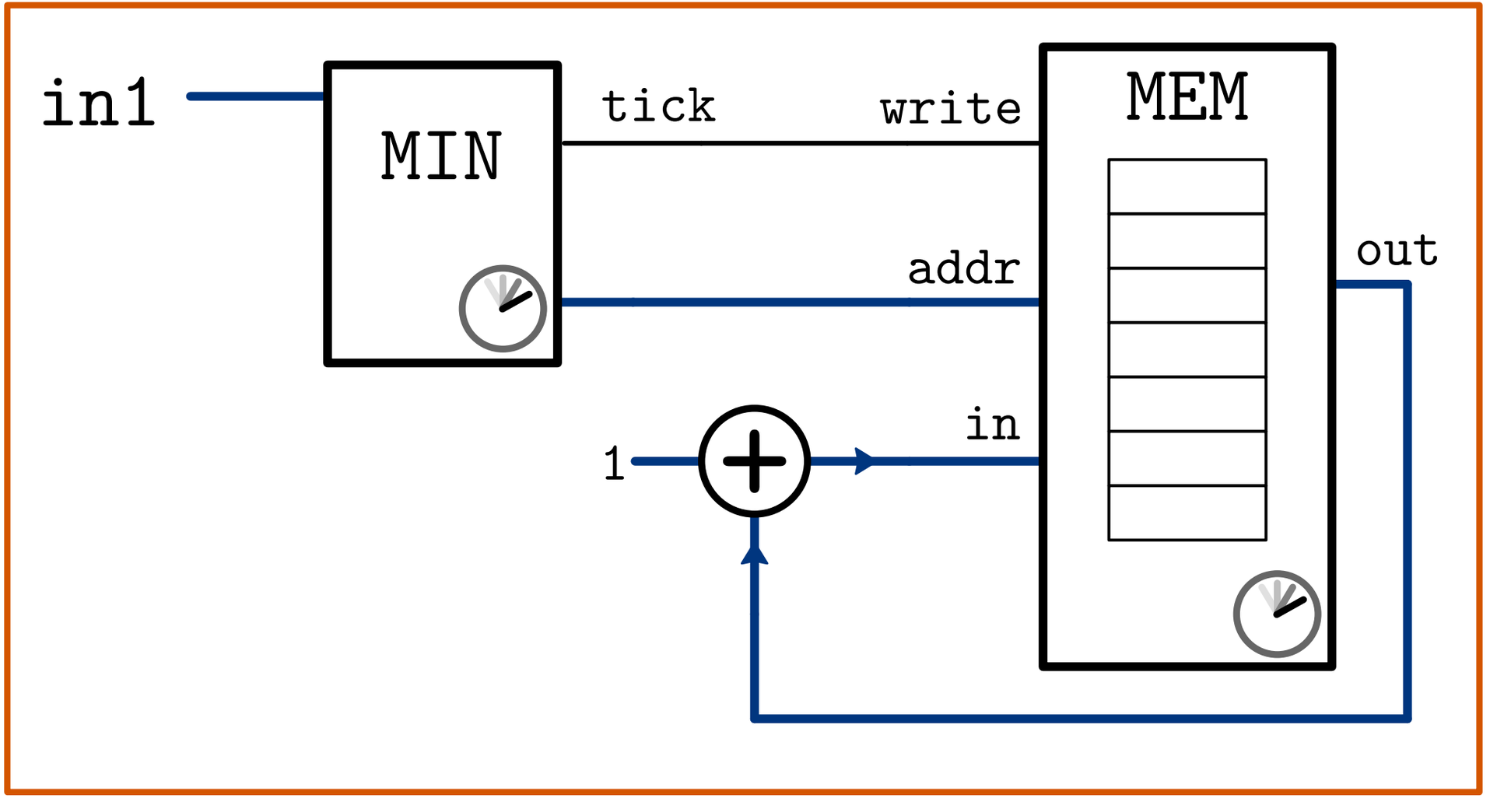
Para conteo de fotones



Ejemplo de FPGA para óptica

Adquisición y pre-procesamiento

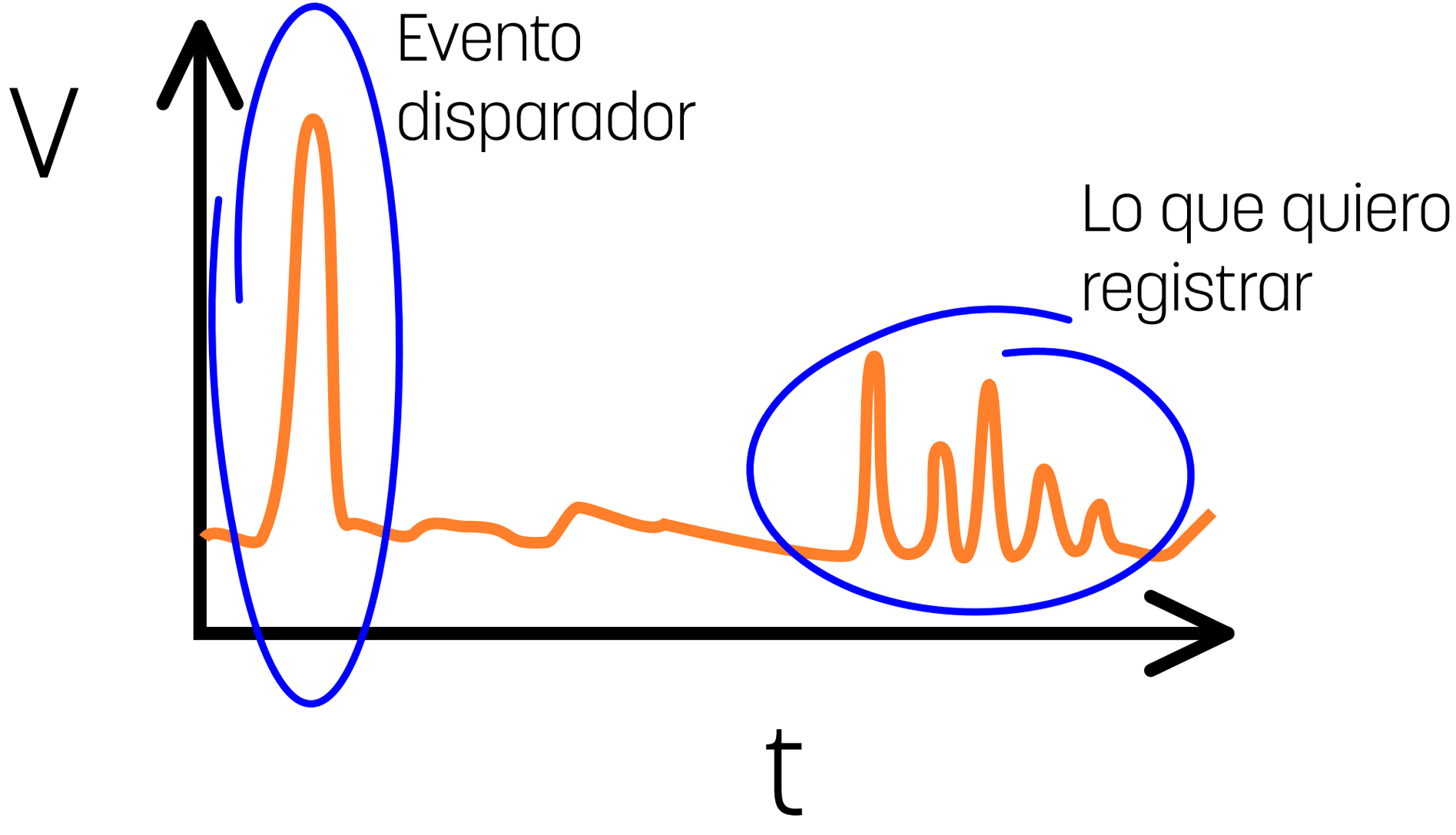
Caracterización de fotomultiplicador



Ejemplo de FPGA para óptica

Adquisición demorada

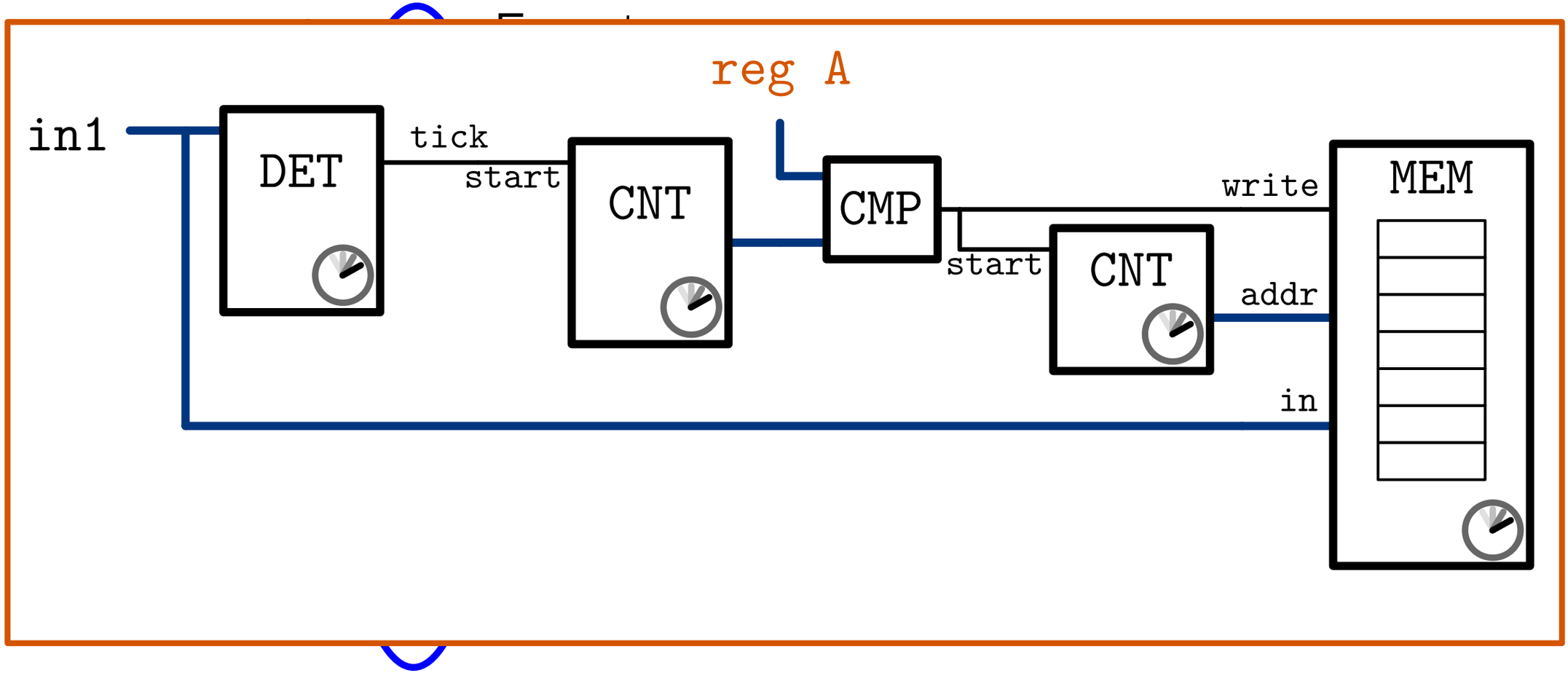
Medición de eventos muy separados en un TOF



Ejemplo de FPGA para óptica

Adquisición demorada

Medición de eventos muy separados en un TOF

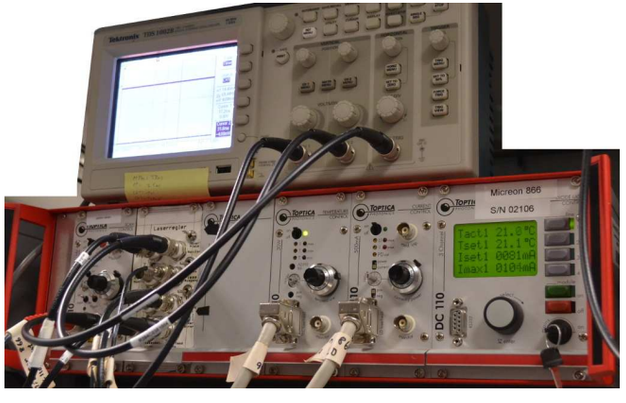


t

Ejemplos más sofisticados

Múltimples instrumentos en uno

Modulación + demodulación lock-in + PID



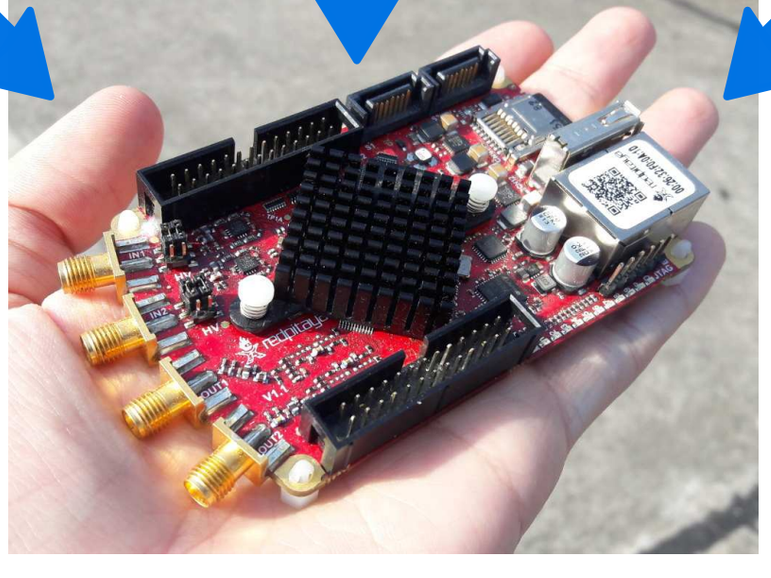
Osciloscopio

+

Filtros PID

Generador de funciones

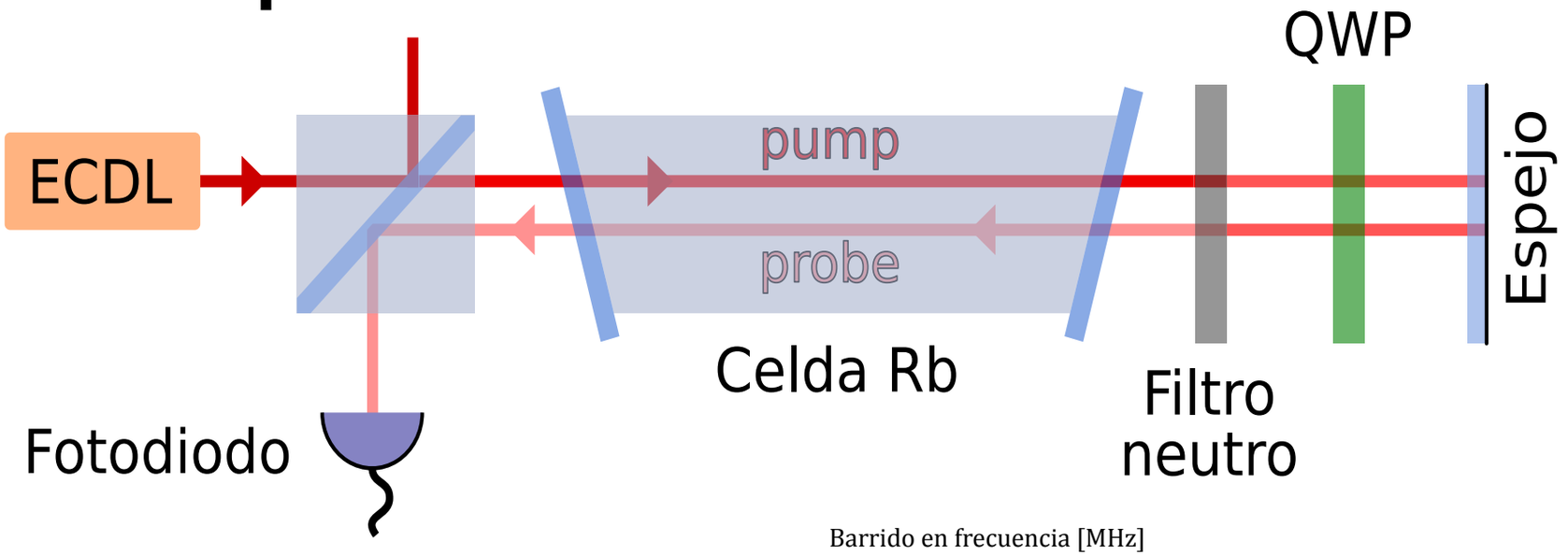
Amplificador Lock-in

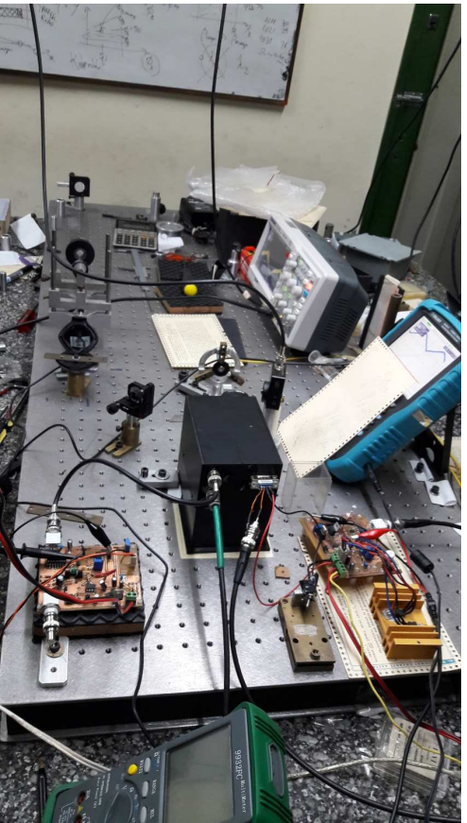


Estabilización de longitud de onda de láser

Sistemas de lockeo para estabilizar λ en láseres

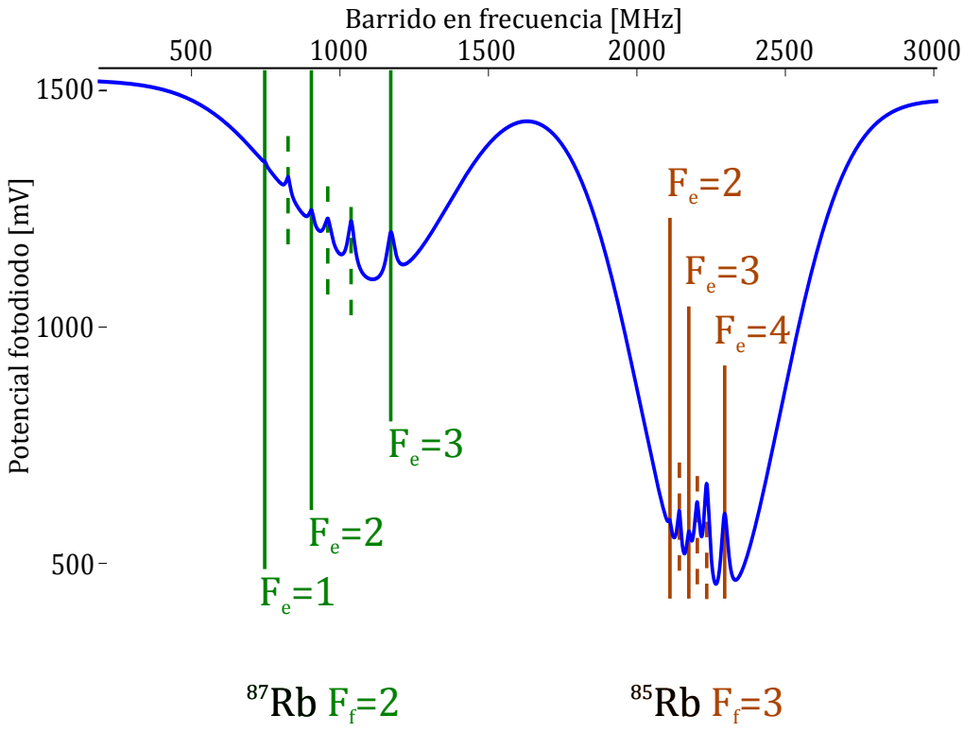
Patrón metrológico





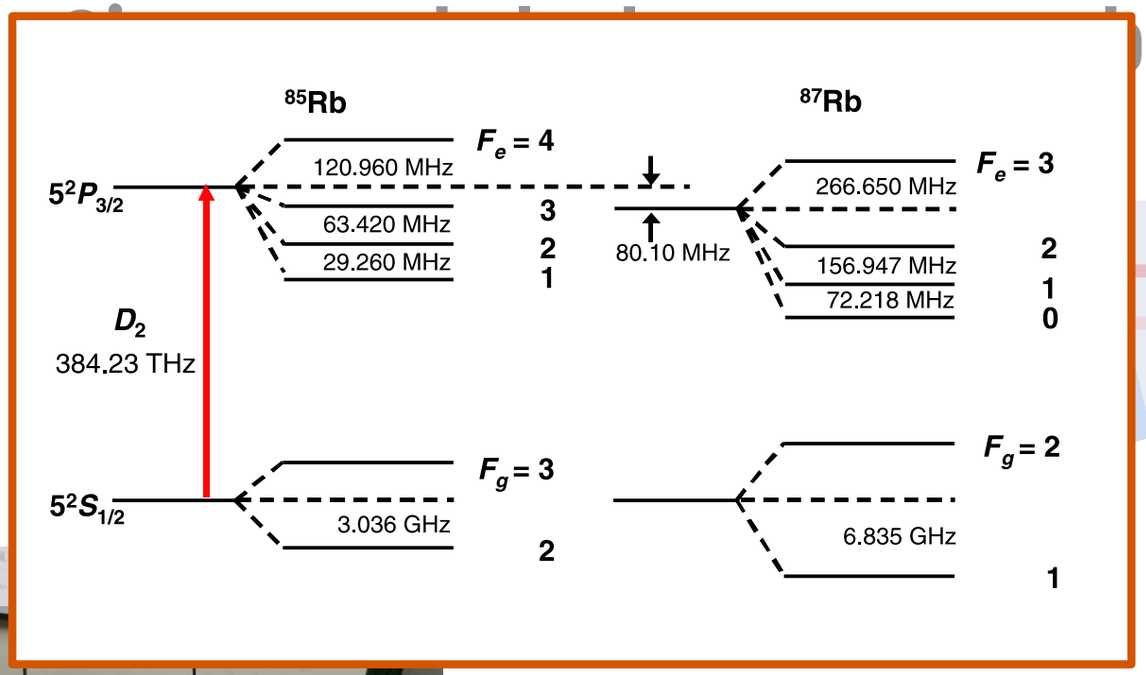
Barrido en frecuencia

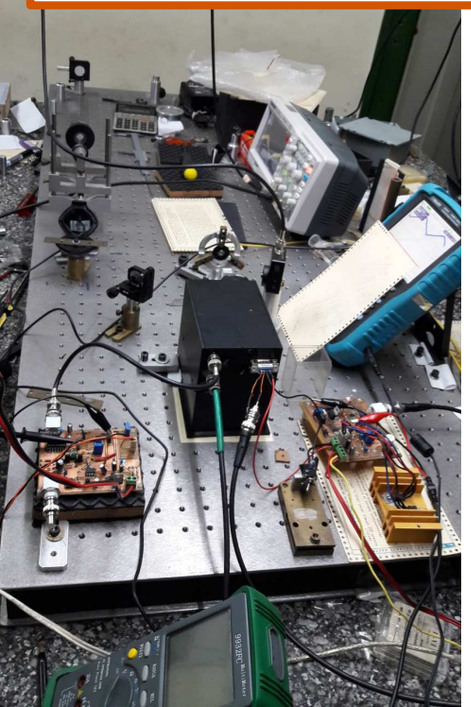
D_2 Rb: ~384 THz

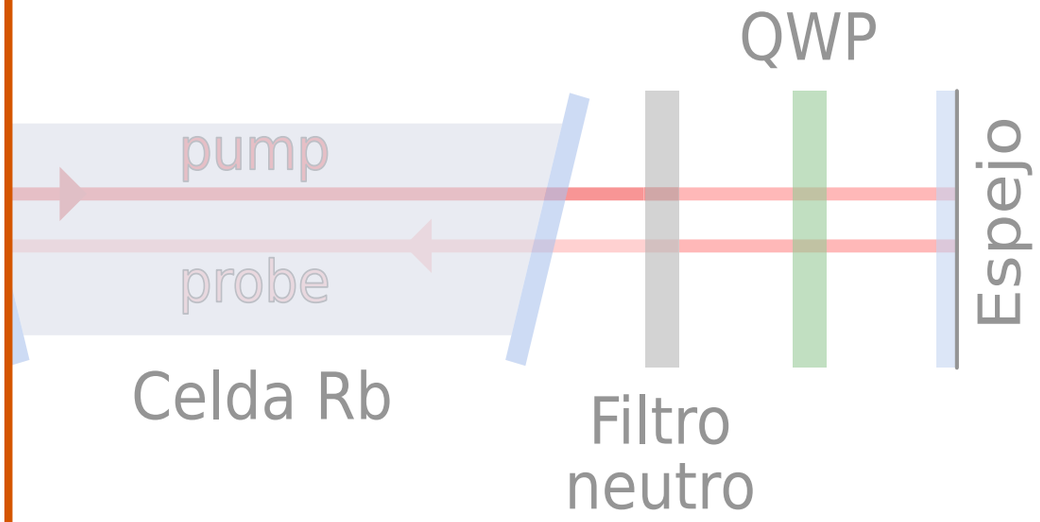


Estabilización de longitud de onda de láser

ilizar λ en láseres

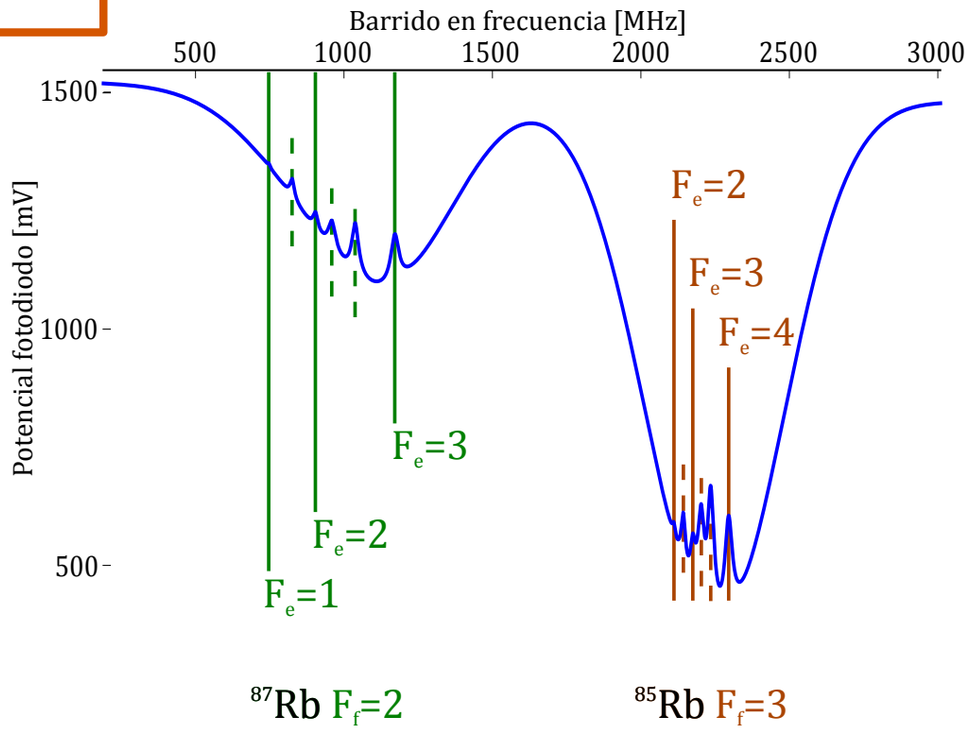




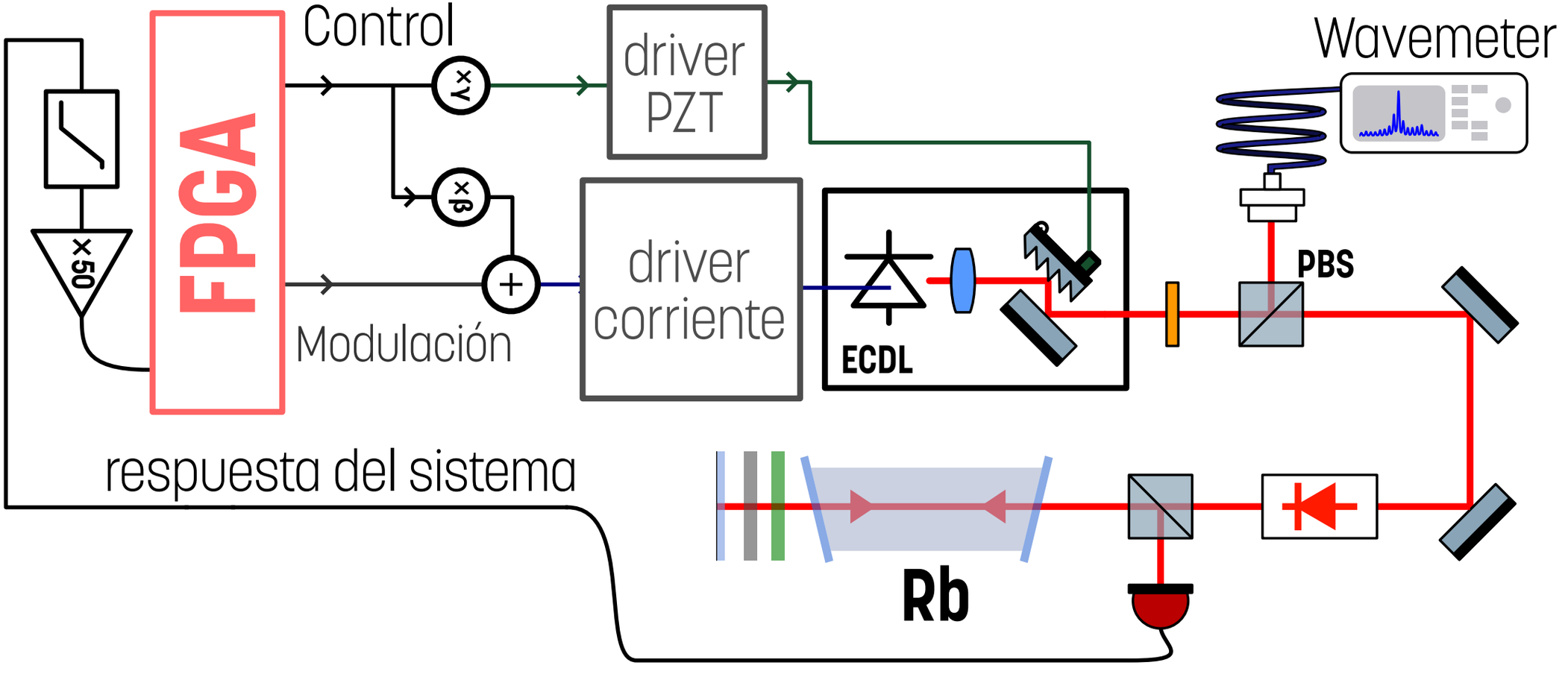


Barrido en frecuencia

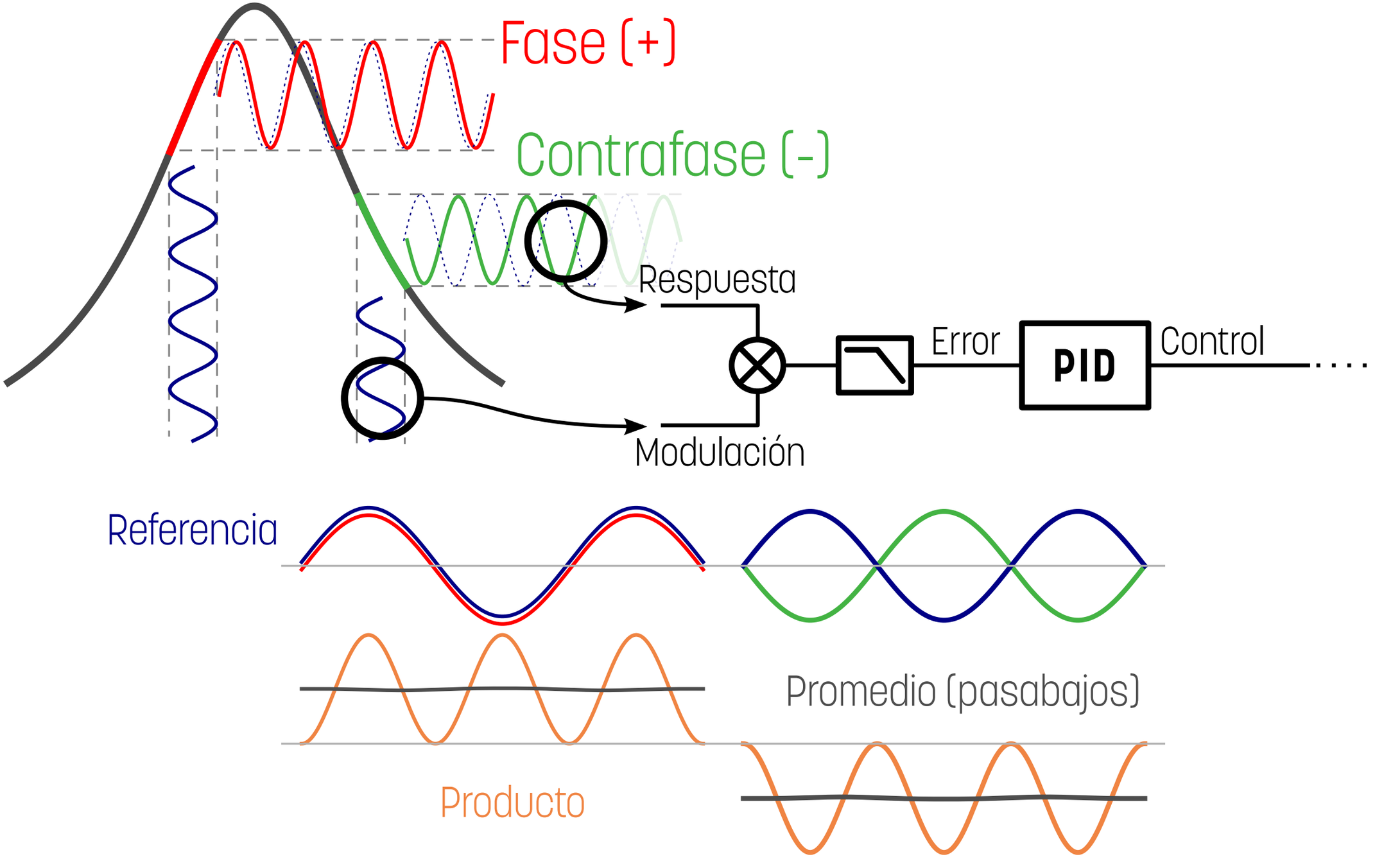
D_2 Rb: ~384 THz



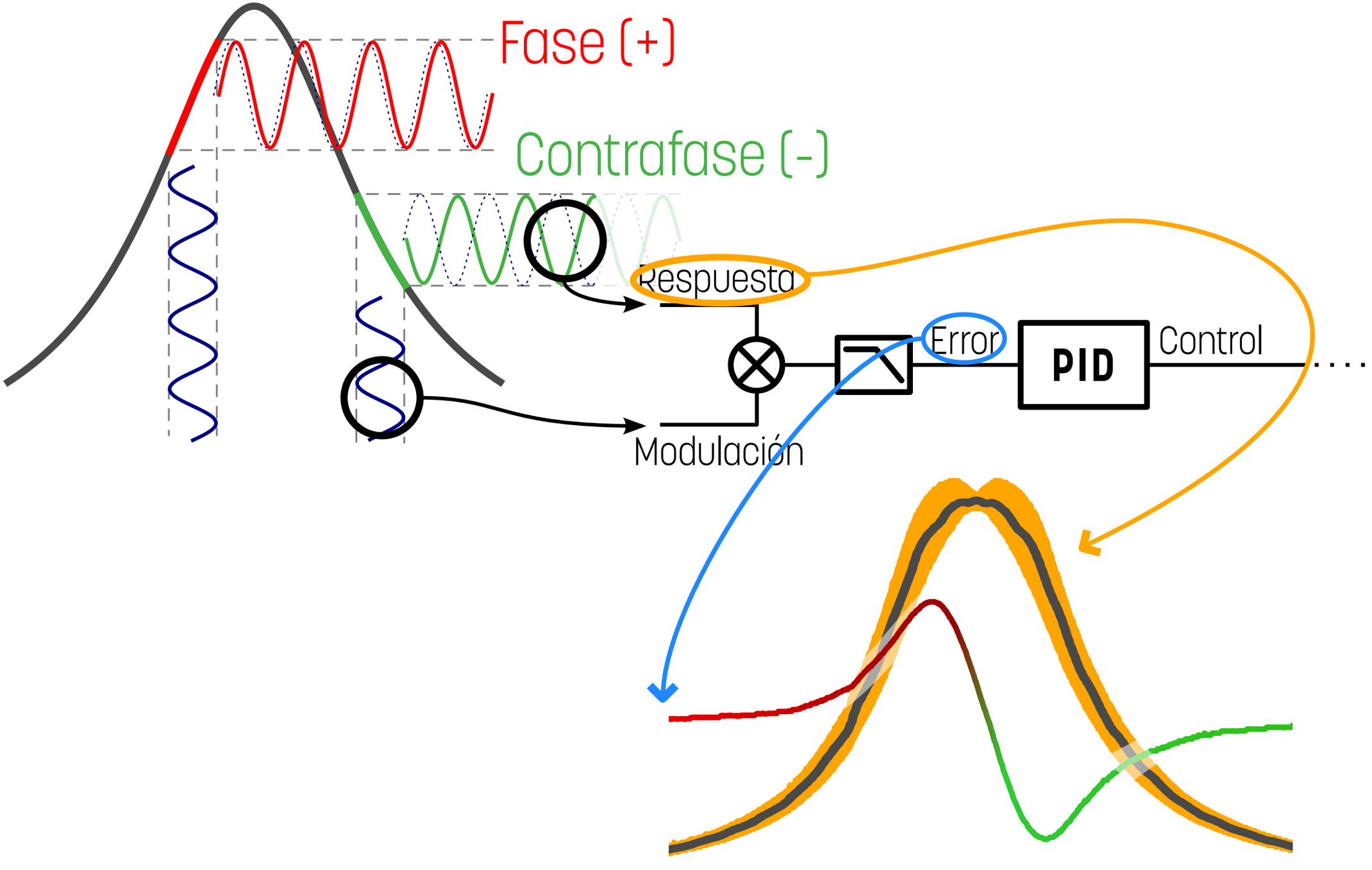
El experimento



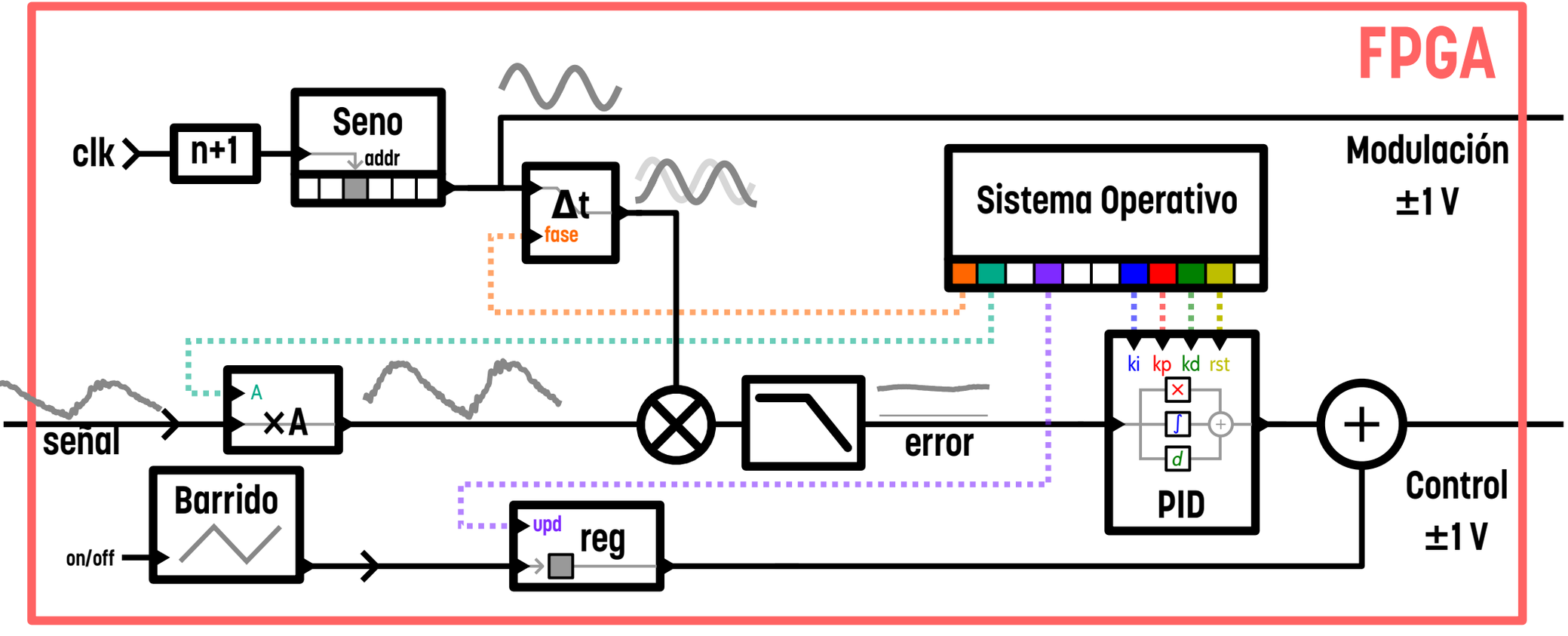
Técnica de amplificación lock-in

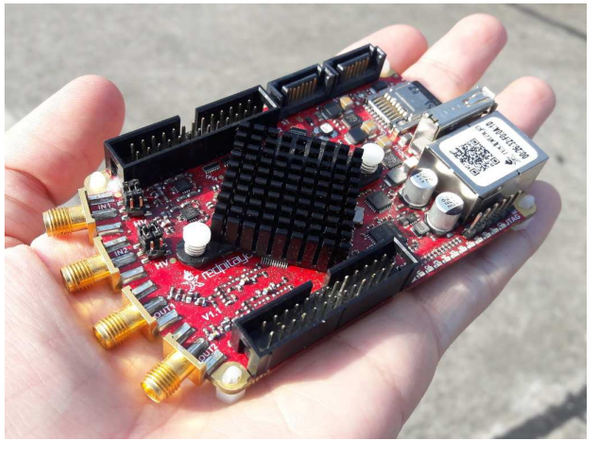


Técnica de amplificación lock-in



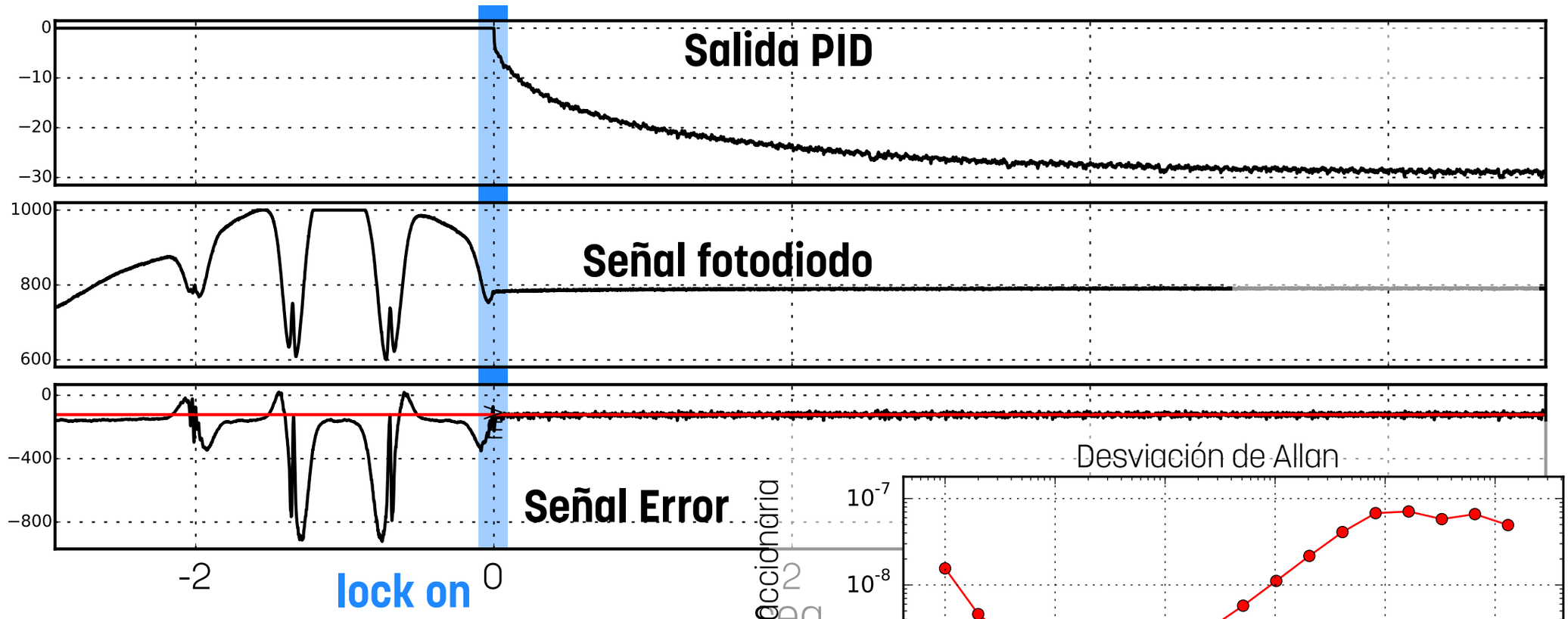
Implementeación Lock-in FPGA

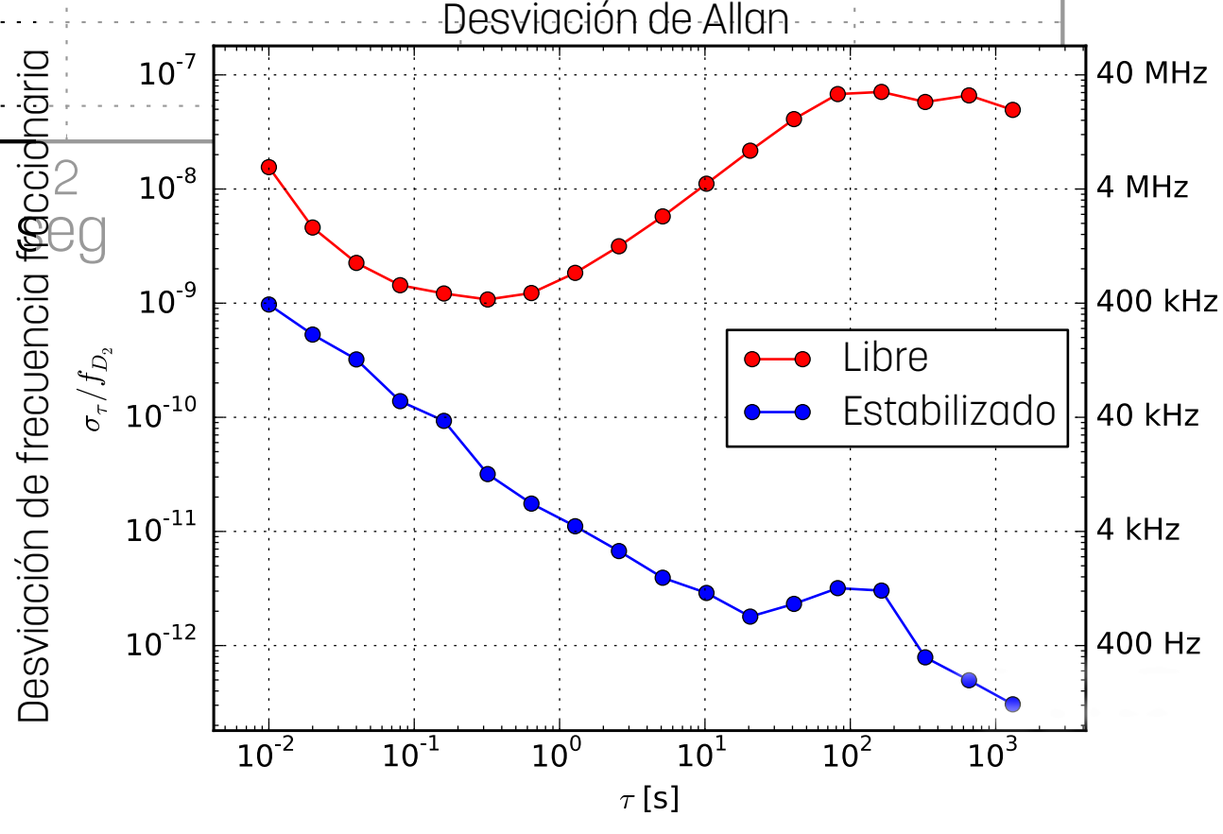




Implementeación Lock-in FPGA

Resultados



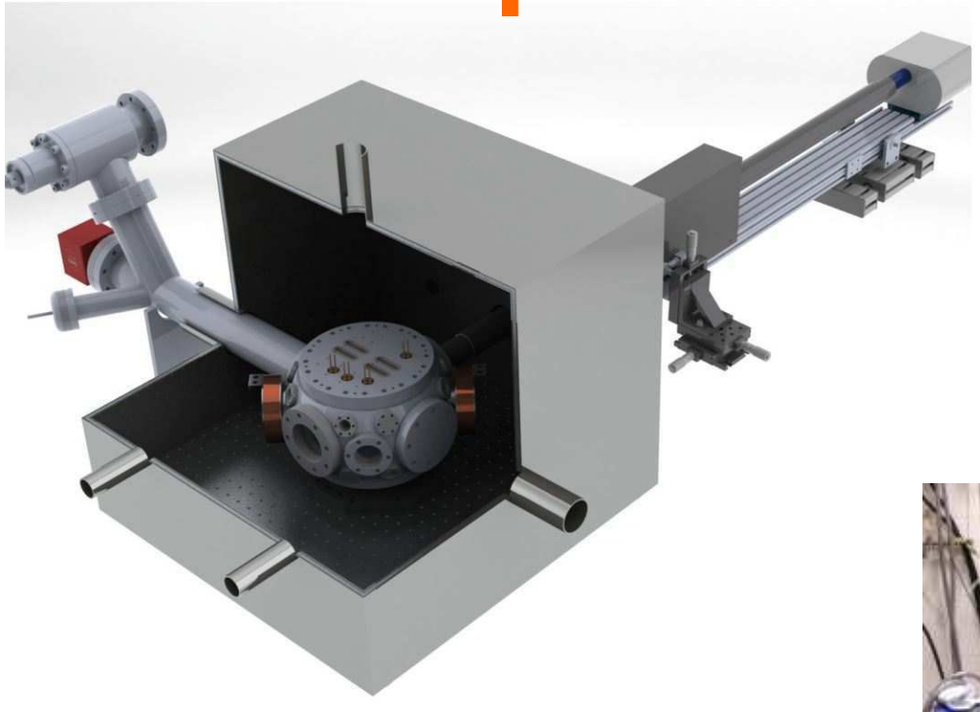


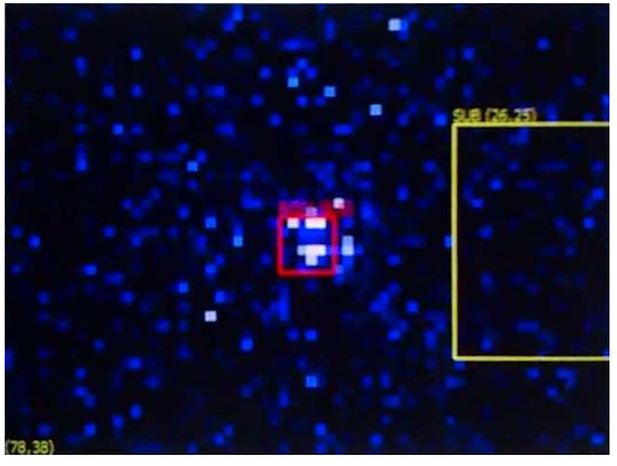
Lockeo de láser
Referencia de
frecuencia óptica

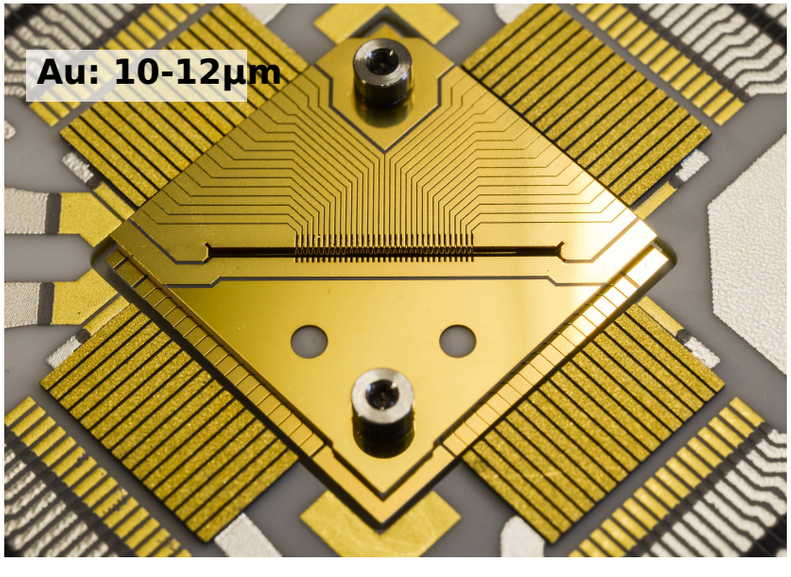
Aplicaciones a gran escala

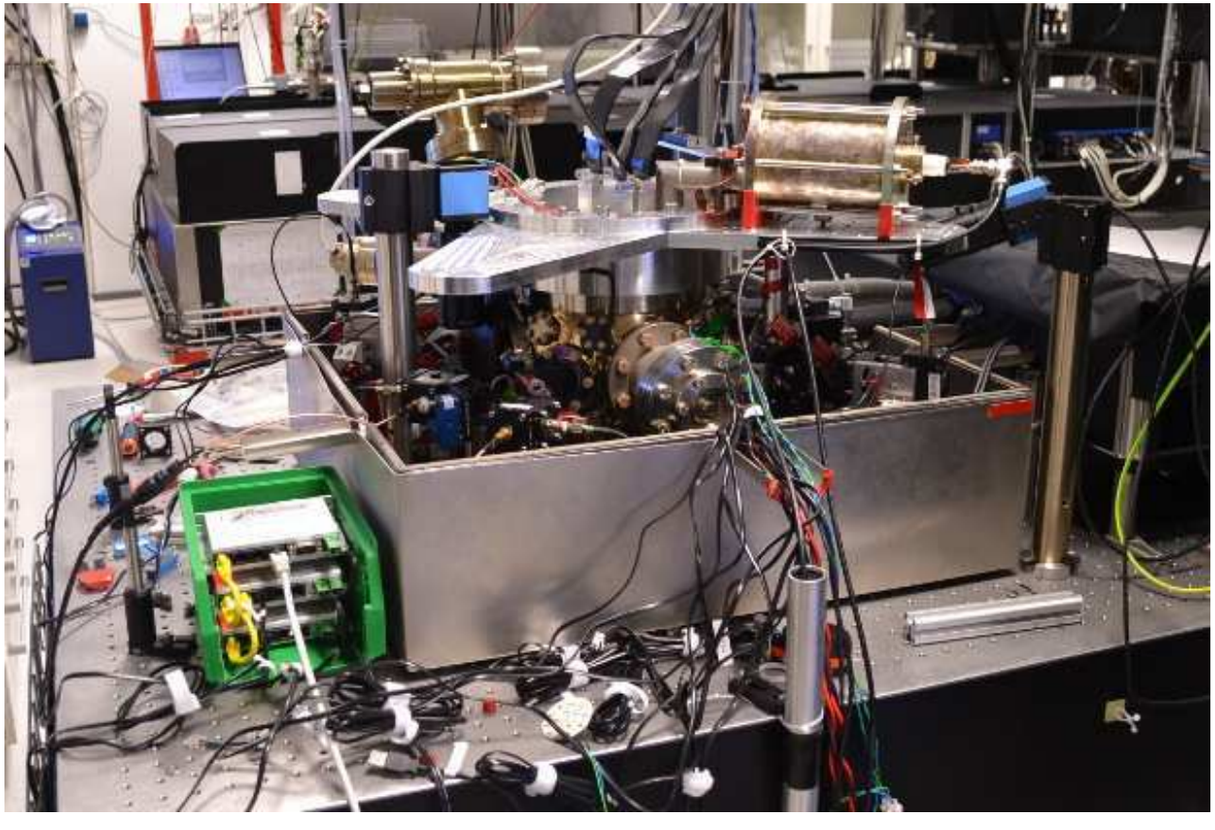
Comentarios finales

Trampa de iones - Mainz







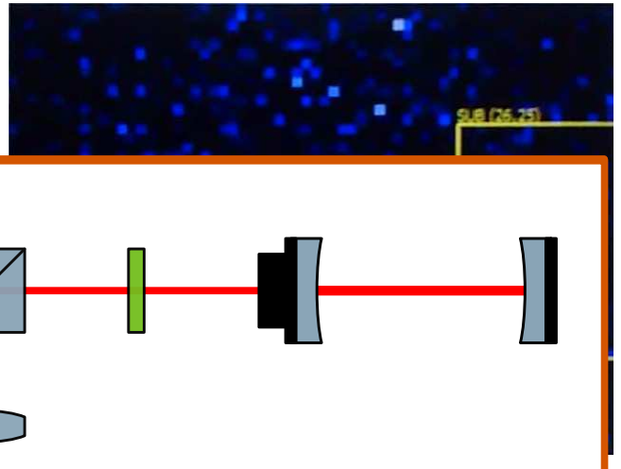


Comentarios finales

Trampa de iones - Mainz

Comentarios finales

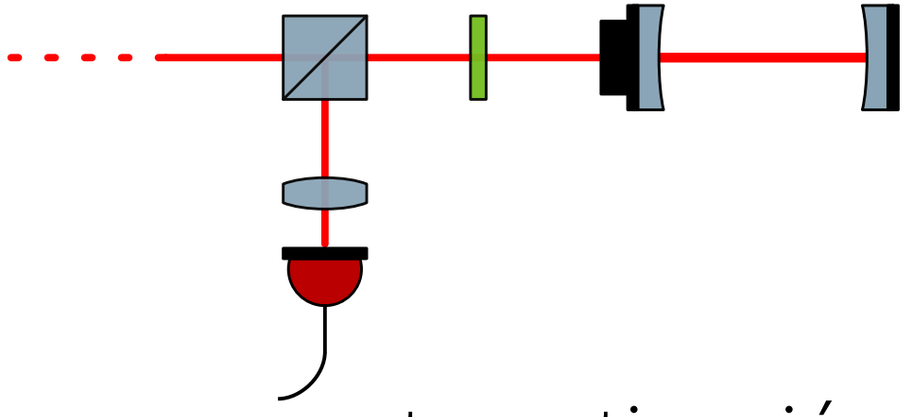
Trampa de iones - Mainz



Pound-Drever-Hall

modulación: ~30 MHz

colaboración con LIAF - FCEN-UBA



5 lásers para el experimento

Control de λ por PDH

Módulos RP para:

- Generación de RF
- Demodulación Lock-in
- Control PID

automatización

monitoreo

control remoto

integración


Muchas Gracias
¿Preguntas?