Análisis de Señales
Curso 2025

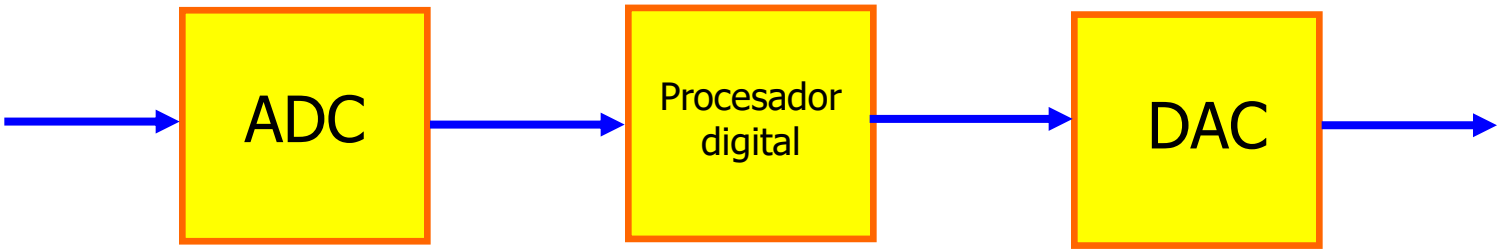
Aplicación Teorema del muesteo y CAD

Instrumentación

Prof. Jorge Runco

Procesamiento digital

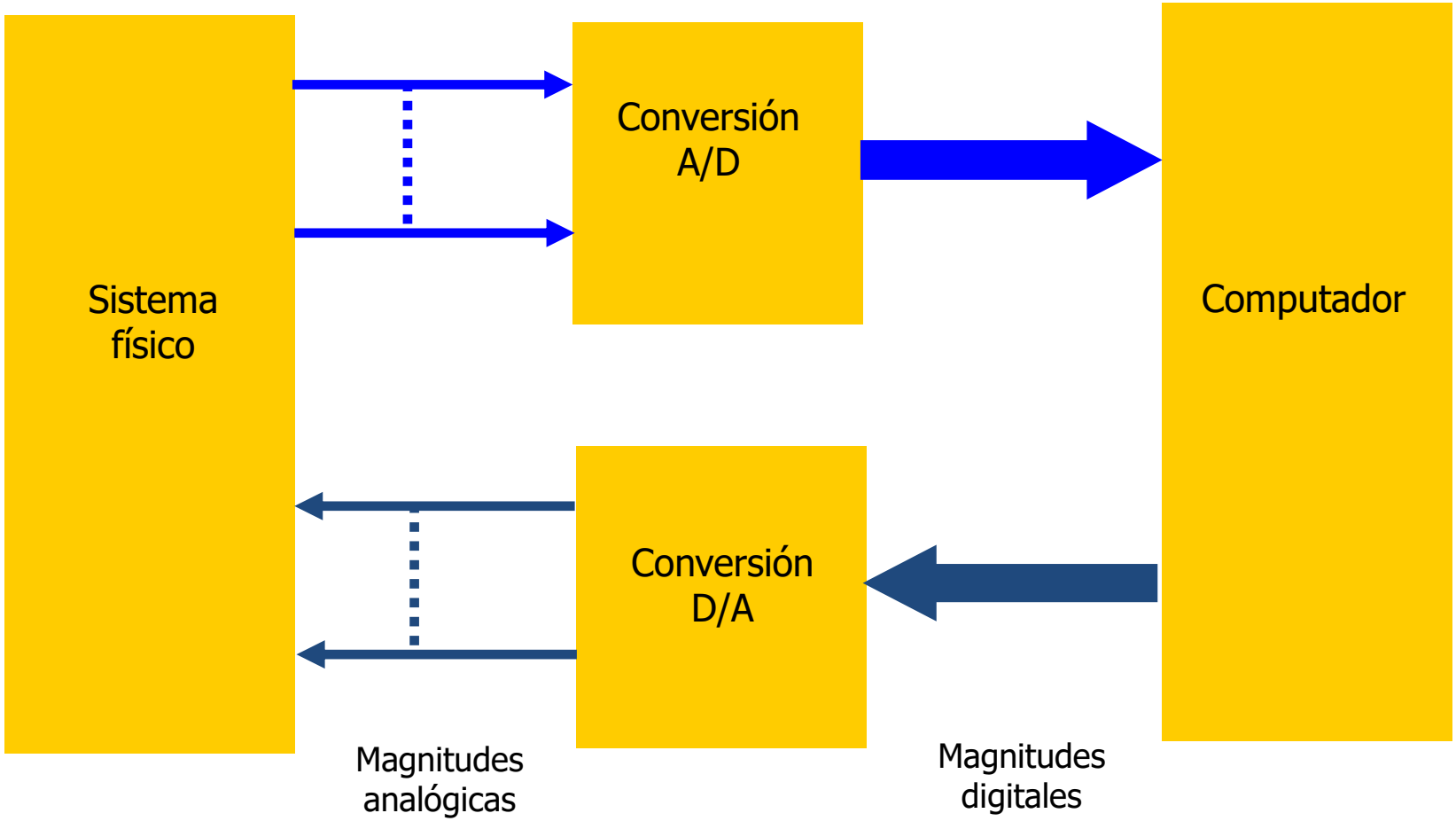
Las técnicas de señales digitales proporciona un método alternativo para procesar una señal analógica de interés práctico tales como la voz, señales biológicas, sísmicas, del sonar y de los distintos tipos de comunicaciones. Para realizar esto, es necesario antes que nada de una interfaz entre la señal analógica y el procesador digital y viceversa. Estas interfaces son el convertidor Analógico-Digital (ADC) y el convertidor Digital-Analógico (DAC) como se muestra en la figura



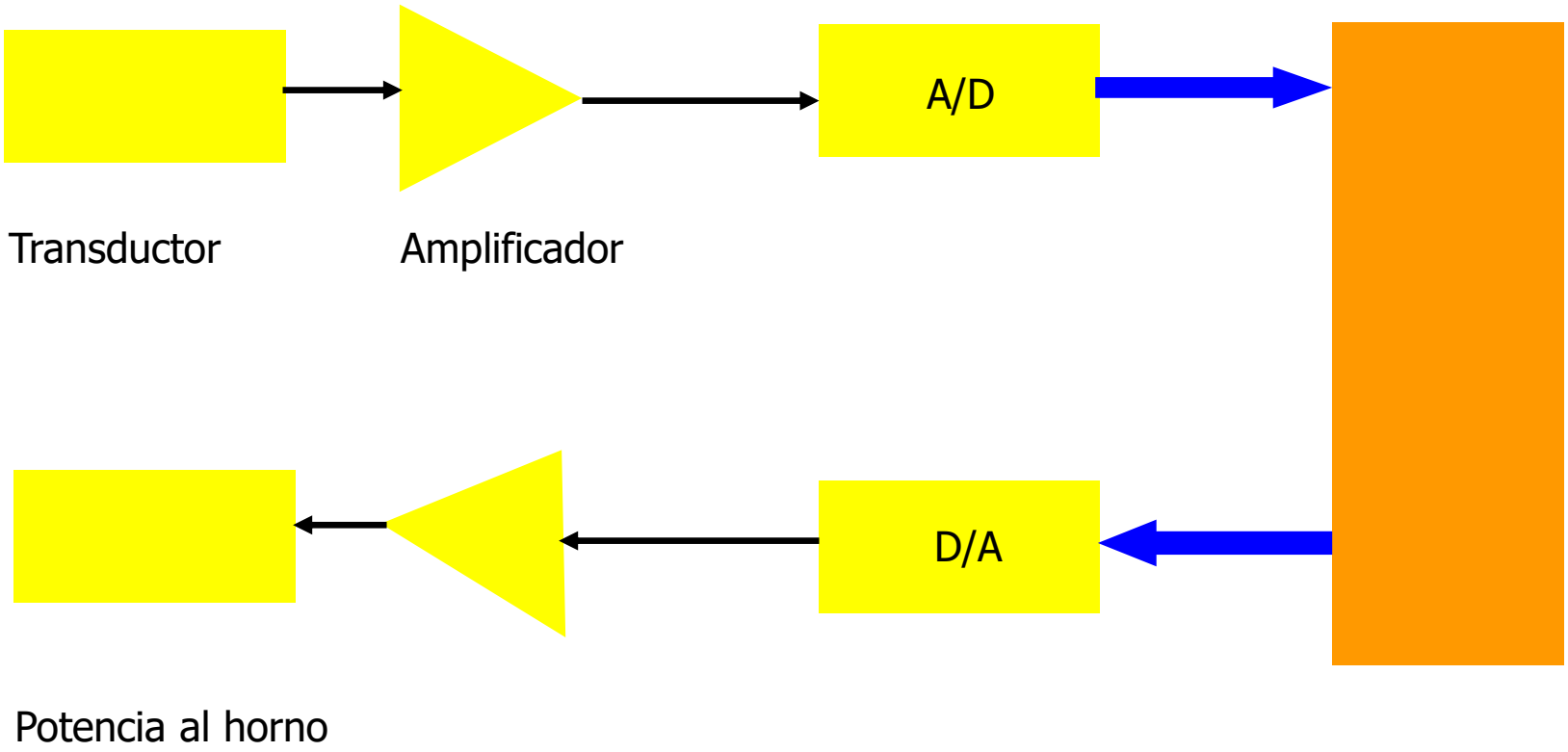
Señal analógica

Señal analógica

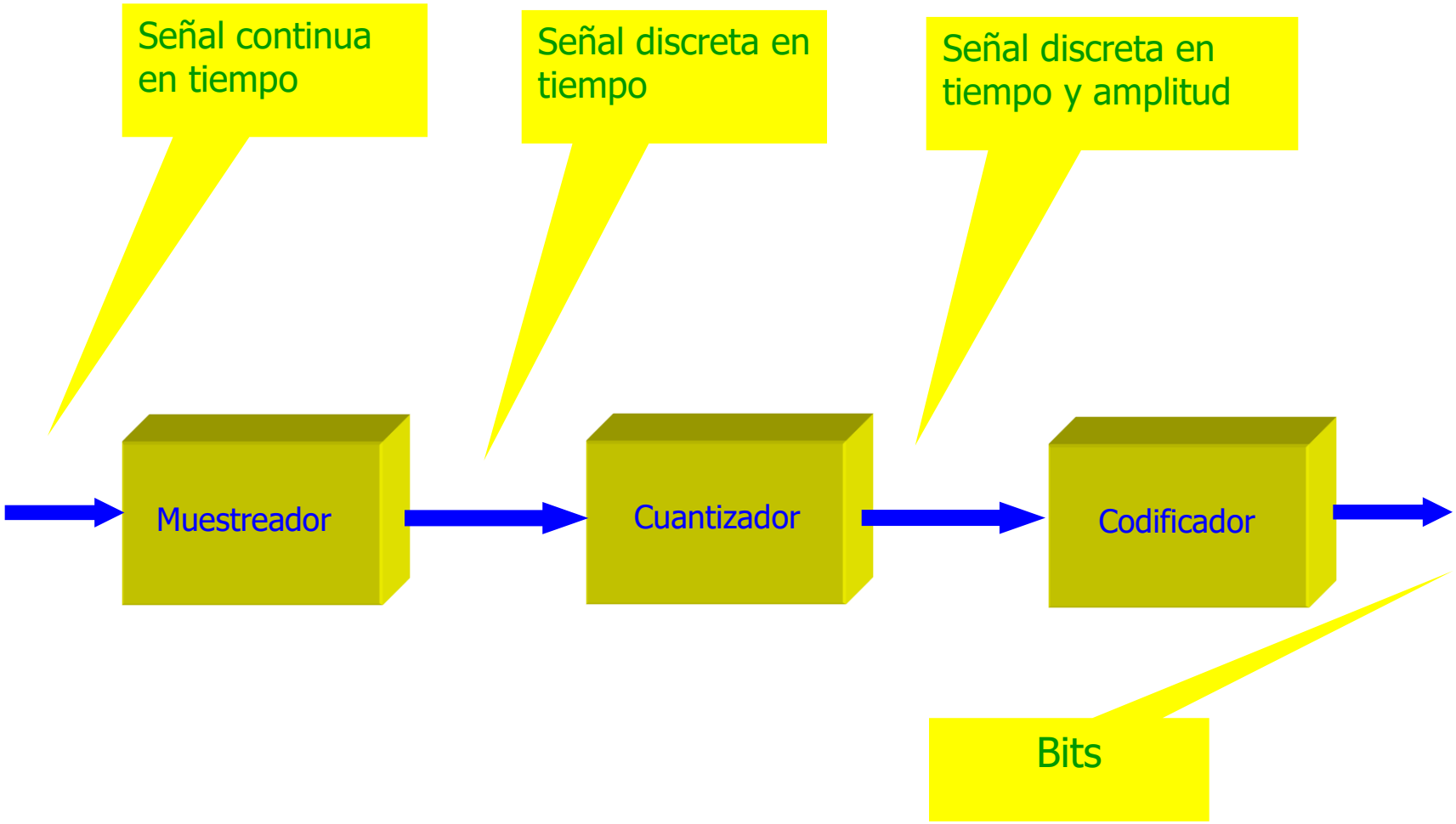
Conversión A/D



Sistema de Adquisición y Accionamiento



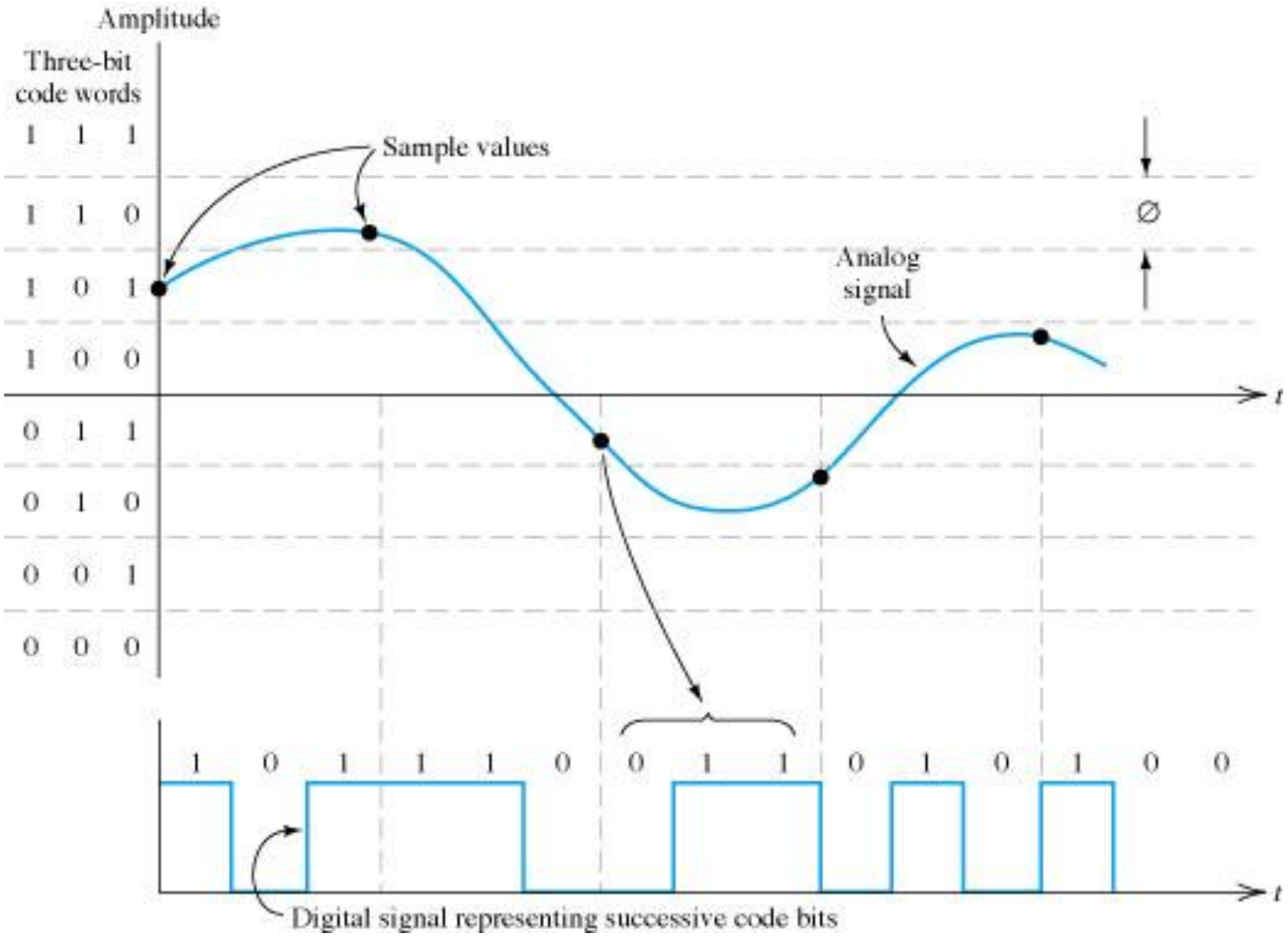
Conversión A/D - Digitalización



Conversión Analógico-Digital y Digital-Analógico

- Para procesar señales analógicas por medios digitales es necesario convertirlas a formato digital, esto es, transformarlas en una secuencia de números de precisión finita. Este procedimiento se denomina conversión analógico-digital (ADC).

Conversión analógica digital



ADC : cuatro procesos

- *Muestreo (sampling)*: tomar muestras periódicas de la amplitud de la onda.
- *Retención (hold)*: las muestras son retenidas hasta “evaluarlas”.
- *Cuantificación*: la amplitud de la señal muestreada toma valores discretos.
- *Codificación*: se traducen los valores obtenidos a N bits, asigna un valor entre 2^N posibles.

Hasta aquí ADC

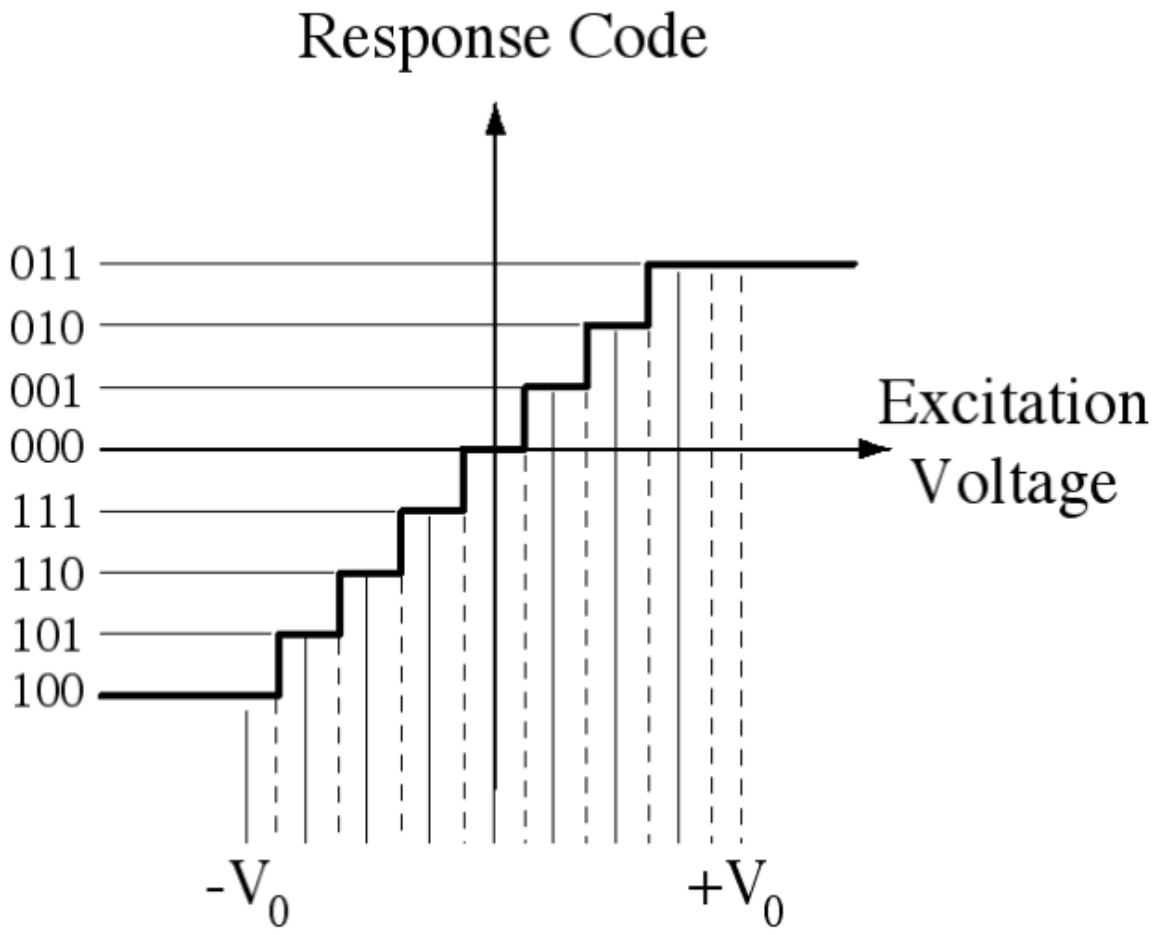
- Luego se podría *comprimir* la información, en bits, para tener en cuenta la capacidad de almacenamiento, tasa de datos, etc. que el sistema de cómputo puede manejar. (MP3, JPEG, etc)

Seguimos……

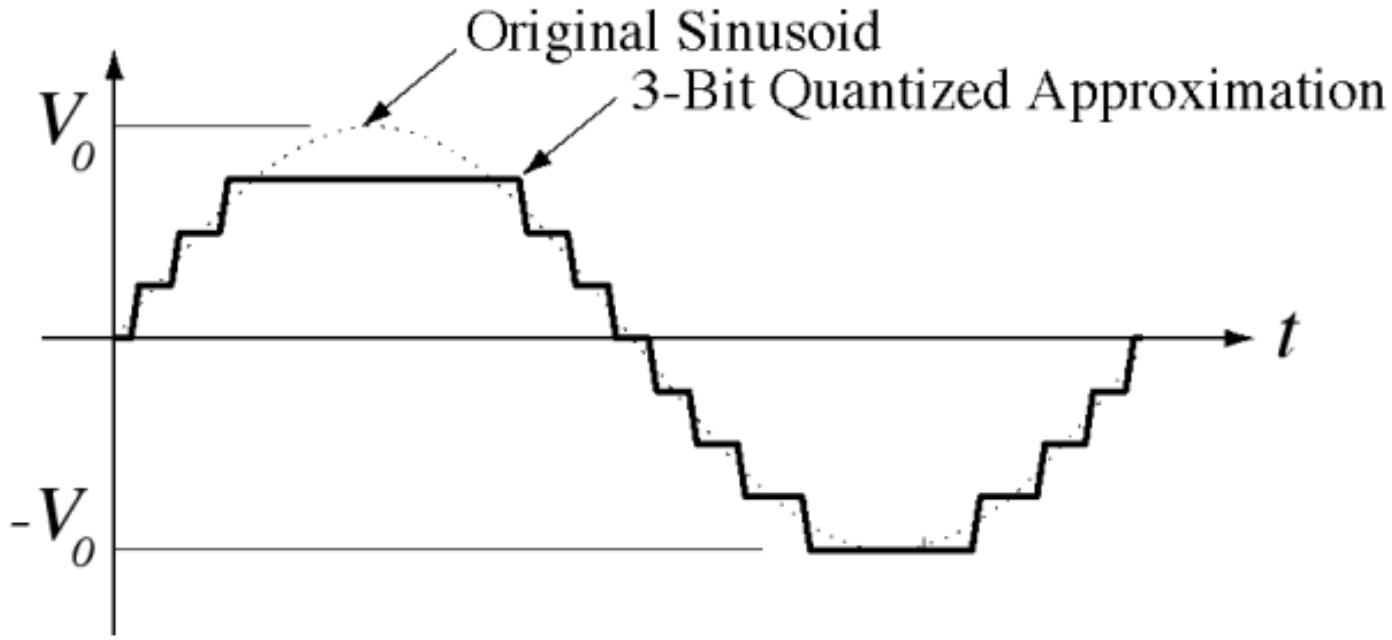
- De acuerdo a lo que es una señal analógica, esta puede tener una cantidad de valores infinitos.
- Como es imposible representarlos a todos, estas señales se dividen en niveles determinados, y así se convertirán en un número finito de niveles (niveles de cuantización).
- Las muestras se convierten en un número binario, cuyo valor es cercano al valor de la muestra.
- Ej : con 8 bits podemos representar 256 niveles

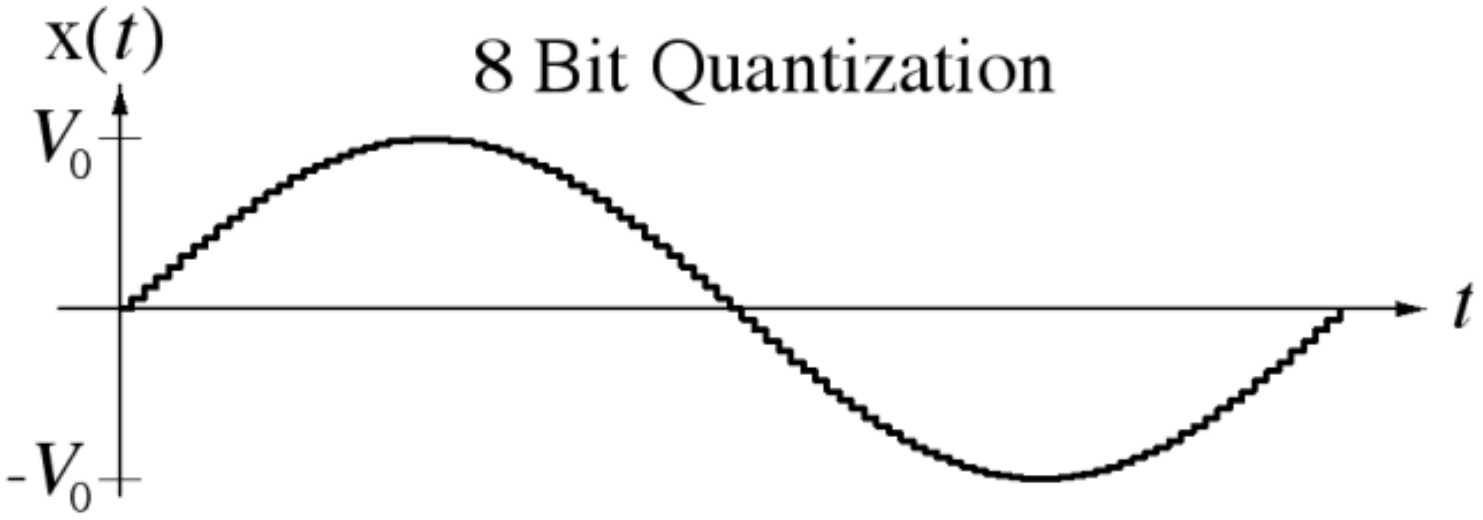
Relación entrada/salida para un ADC

Código digital de salida

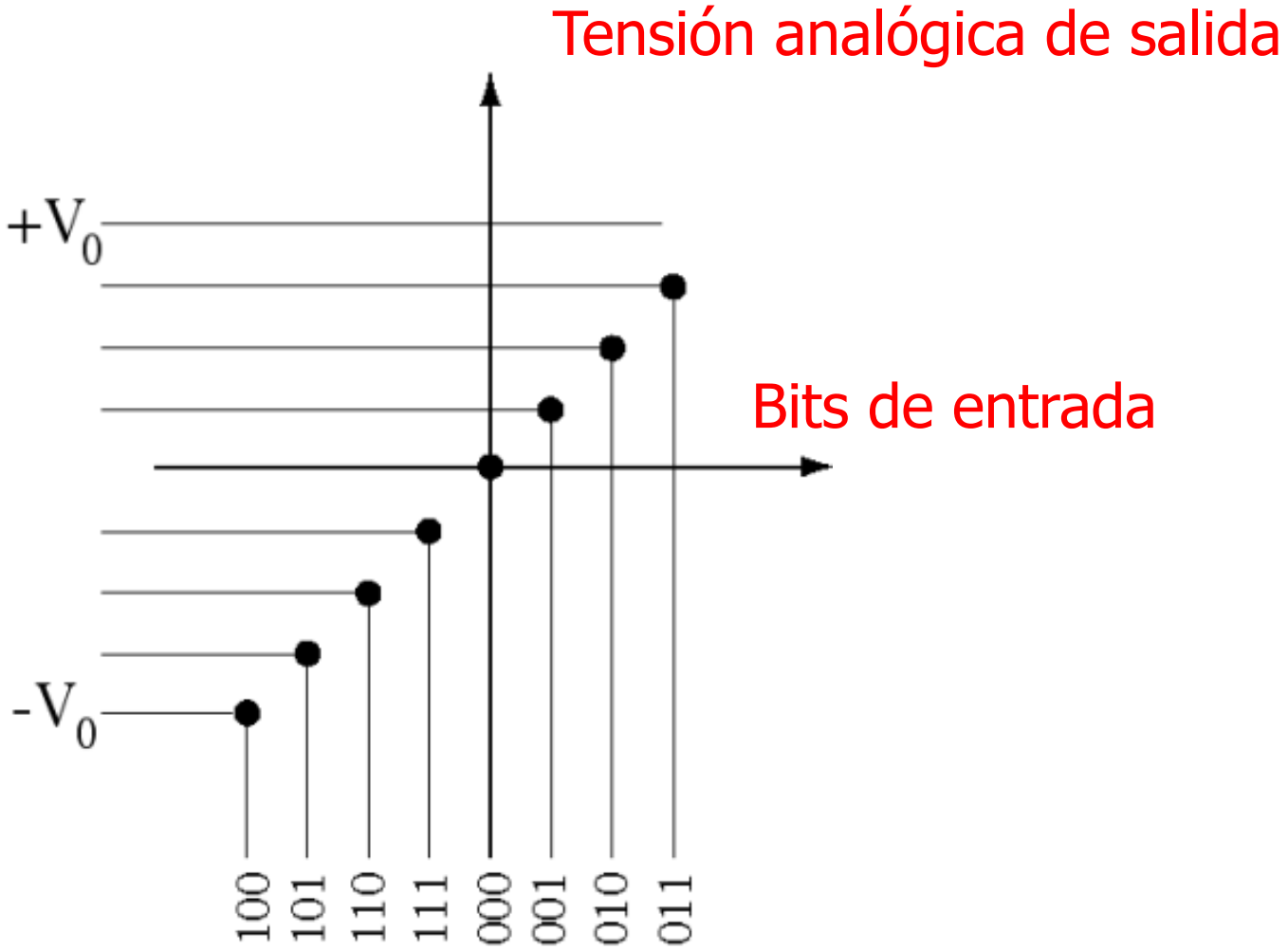


Señal analógica de entrada


Original Sinusoid
3-Bit Quantized Approximation
V_o
$-V_o$
t


x(t)
8 Bit Quantization
V_0
$-V_0$
t

Relación entrada salida DAC



Ejemplo

- Supongamos un convertidor A/D produce 4 bits de salida.
- Con estos 4 bits tenemos 16 niveles de voltaje a representar.
- De acuerdo a la figura, consideremos un intervalo de señal analógica entre 0 y 15 volts.

Ejemplo

- El convertidor A/D divide el intervalo en 16 niveles, los cuales representan 15 incrementos, entre 0 y 15 volts.
- Si medimos el valor 0 Volt, el ADC producirá el valor binario 0000.
- Si la muestra es de 8 volt, obtendremos un 1000.

Ejemplo

- Si ahora medimos 11,8 Volts, el ADC, produce el número binario 1100, cuyo equivalente decimal es el 12.
- Por lo tanto aquí apreciamos el error producido y que según vimos es el ERROR DE CUANTIZACION.
- Para reducirlo , no nos queda más que aumentar la cantidad de niveles de voltaje, lo que lleva a aumentar la cantidad de bits a utilizar
- Ej : con 8 bits, tendremos 256 niveles para dividir la señal analógica.
- Así, podemos acercarnos más al valor real analógico

Ejemplo

- La conversión A/D , es un proceso de muestreo o de mediciones de la señal analógica, en intervalos de tiempos regulares.
- En los tiempos mencionados, se mide el valor instantáneo de la señal analógica, y se genera un valor binario proporcional a la muestra.

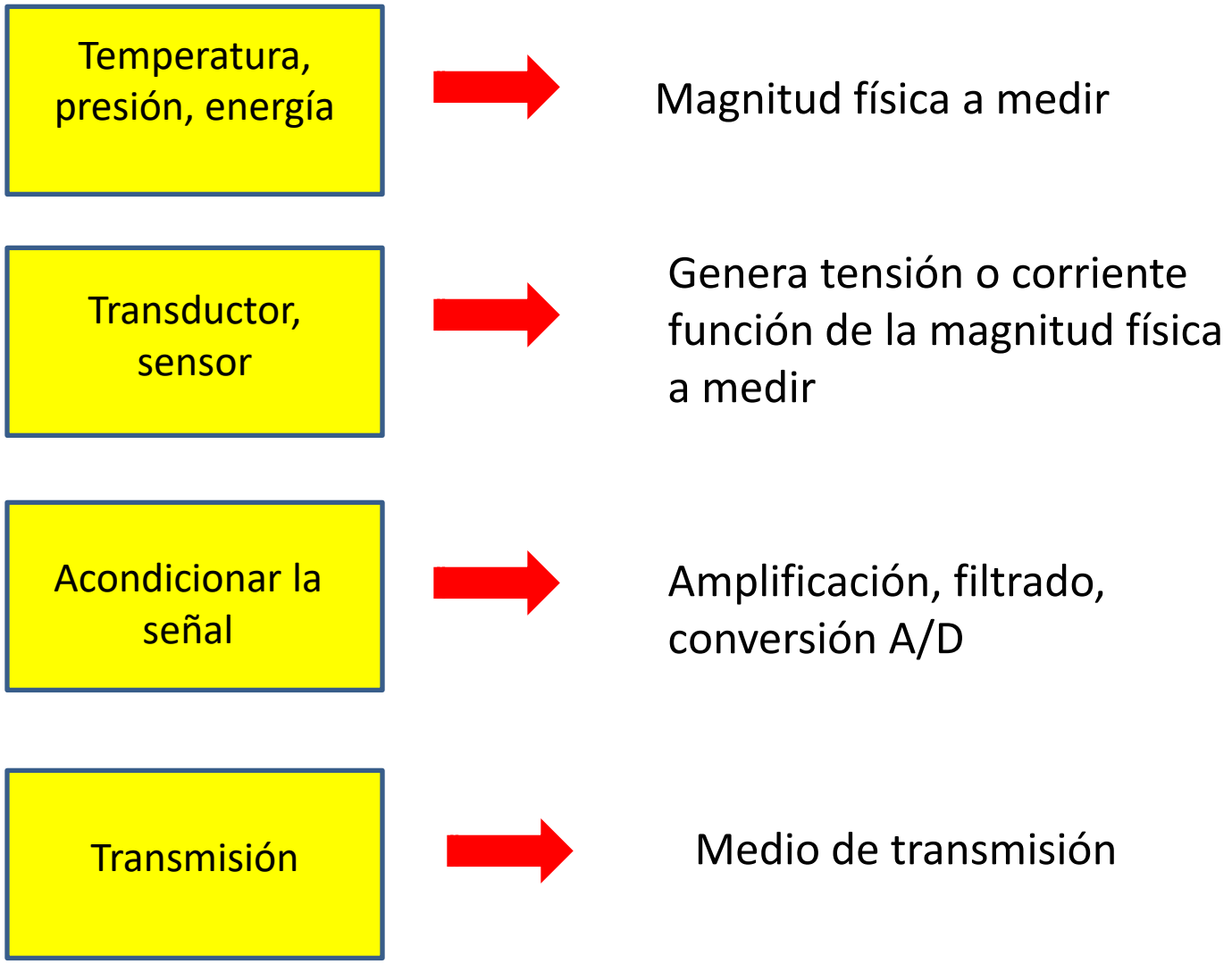
Ejemplo

- Como ya vimos este tiempo está dado por la frecuencia de muestreo expresada por Nyquist, que dará un número de muestras que nos permitirá representar en forma adecuada la señal analógica.
- Por lo tanto “tenemos” que conocer la máxima frecuencia presente en la señal a muestrear.
- El fabricante suele especificar un tiempo llamado de conversión: “de la señal analógica a los bits”.
- Este tiempo nos fija la máxima frecuencia de mustreo con un dado conversor.

Ejemplo

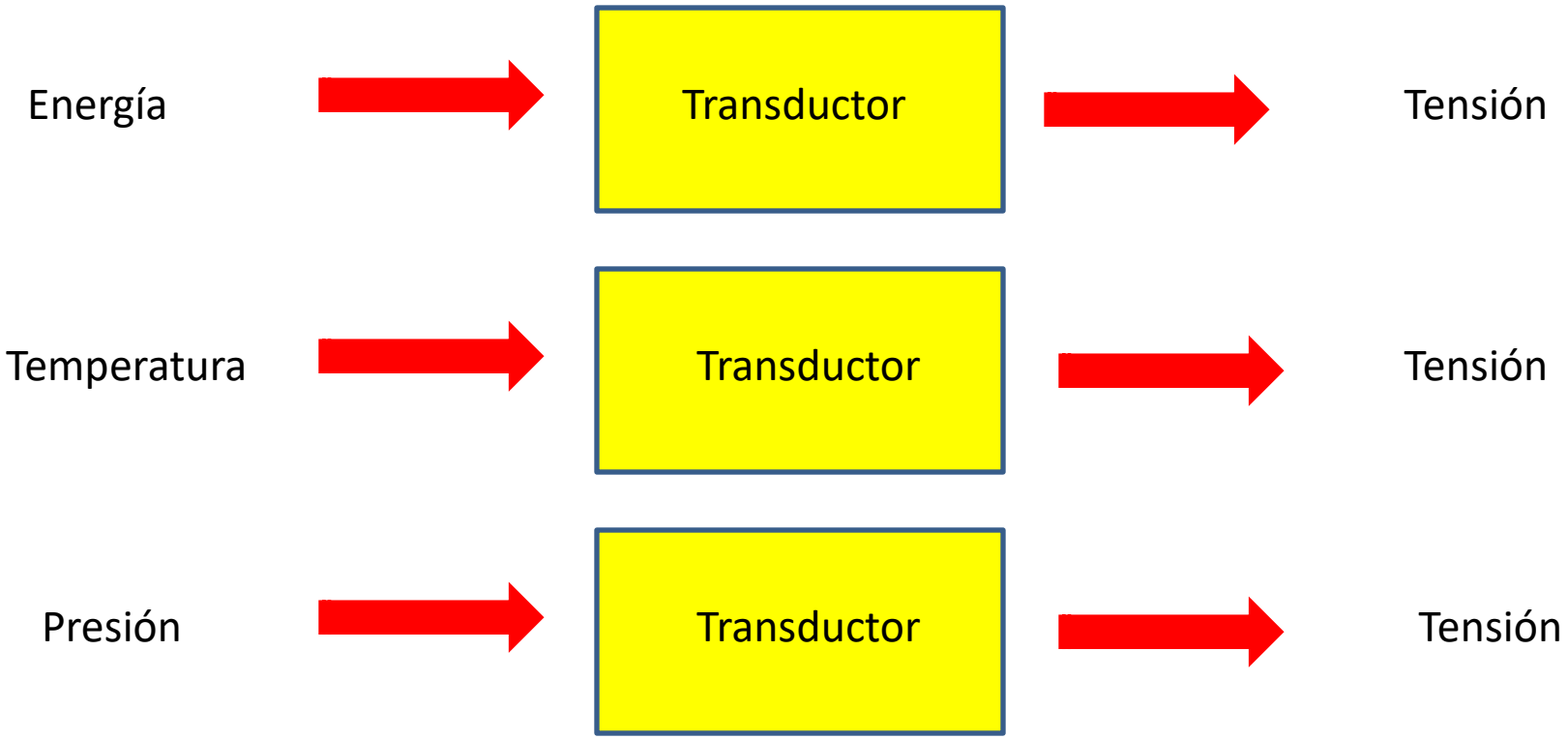
- Analicemos el caso de una emisora de FM, que deseamos digitalizar.
  - La frecuencia más alta de audio es 15 (KHZ)
  - Por lo tanto la frecuencia de muestreo debiera ser de 30(KHZ).
  - Pero la frecuencia real de muestreo se hace entre 3 a 10 veces mayor, es decir entre 45 a 150 (KHZ).
- En el caso de discos compactos CD, que almacenan señales de música hasta 20 (KHZ), usan frecuencias de muestreo de 44,1 a 48 (KHZ)

Adquisición de señales

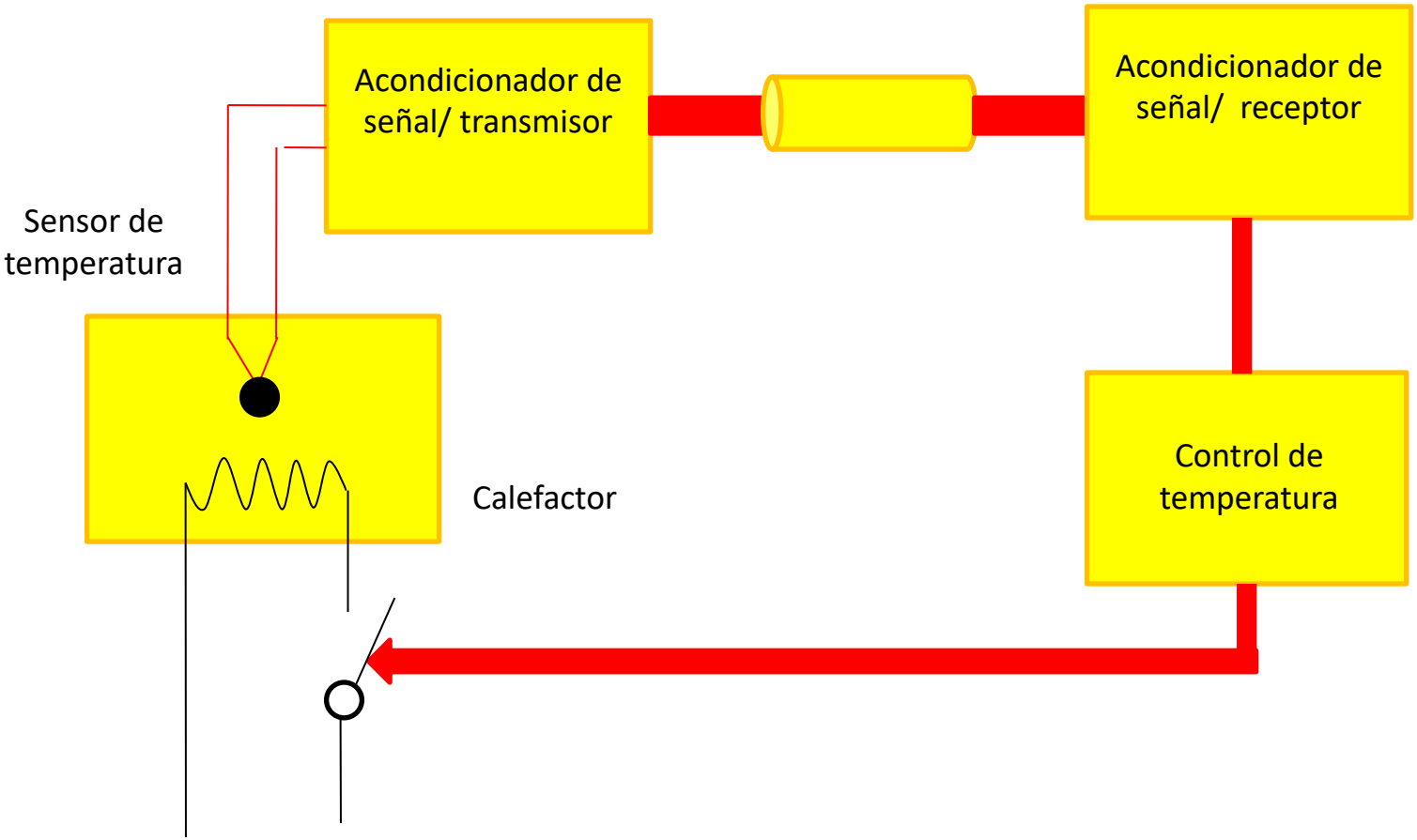


Transductor

- Dispositivo que convierte una señal física de entrada en una señal de salida de tipo: eléctrico (tensión, corriente).



Esquema en bloques de un control de temperatura



- La primera tarea a realizar por el acondicionador de señal, será amplificar las débiles señales entregadas por el transductor, hasta niveles utilizables por el resto de la cadena de medida.
- El amplificador de señales débiles que realice esta tarea será, por lo tanto, un componente crítico (y fundamental) del sistema de instrumentación.
- Hoy en día la mayoría de los amplificadores para sistemas de instrumentación se realizan utilizando amplificadores operacionales, por su versatilidad y sencillez de utilización.

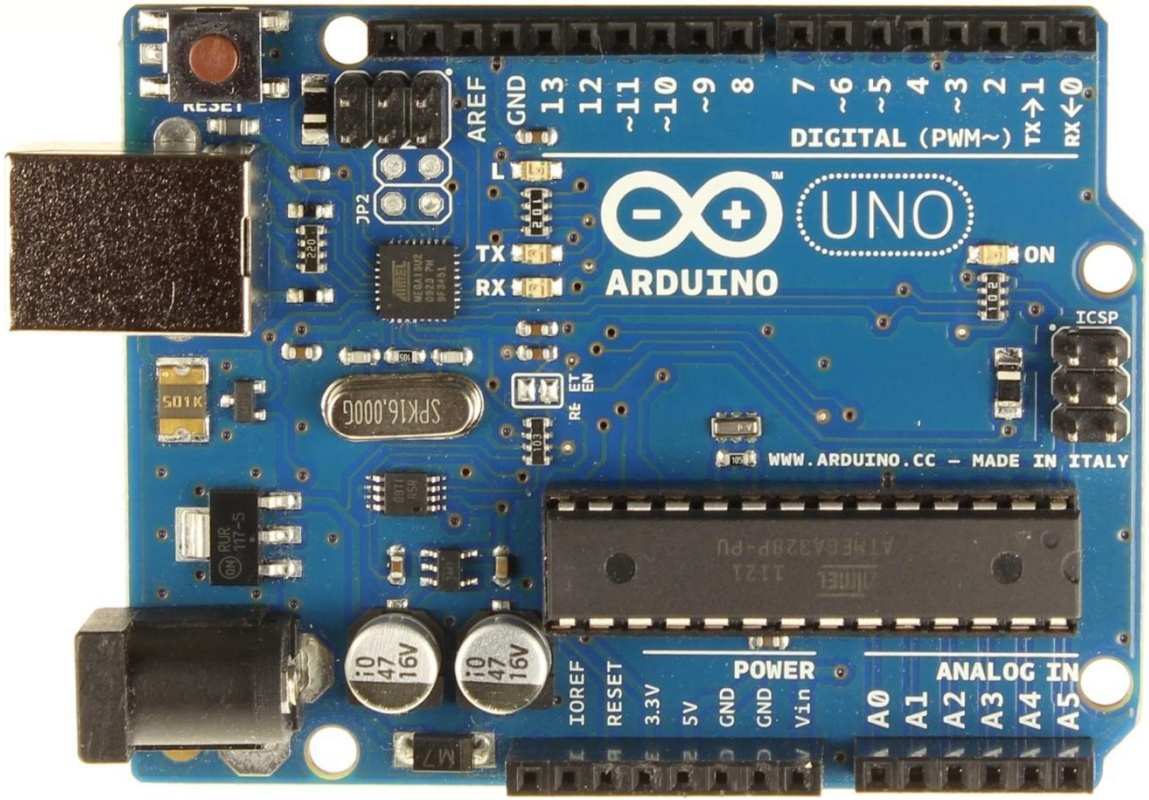
Transductor

- Es un dispositivo que convierte la magnitud física a medir en una señal eléctrica como una tensión o una corriente.

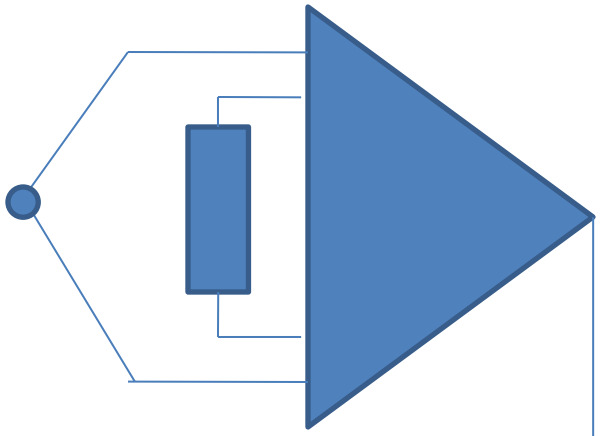
- Ejemplos
  - Termocupla tipo K 40 uV/^{o}C
  - LM35 10 mV/^{o}C

- El transductor entrega una tensión que es función de la magnitud física a medir.
- Para almacenar esta magnitud física en el sistema de computo debemos convertirla en un numero (bits), usamos un conversor analógico digital.
- Luego hacemos una cuenta y de acuerdo a alguna condición, tomamos una decisión.

Para aplicaciones dedicadas:

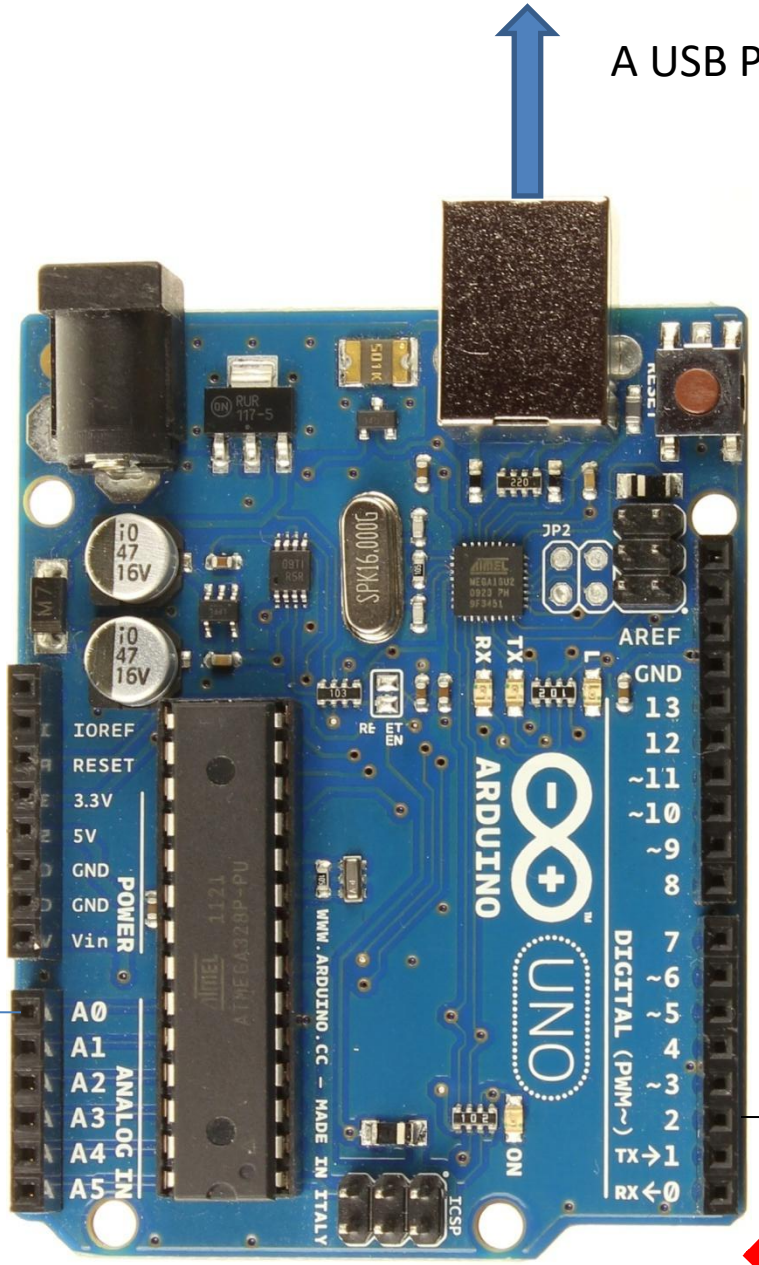


Sensor de temperatura

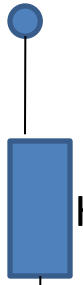


Amplificador de instrumentación

A USB PC

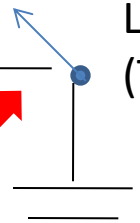


Fuente de alimentación

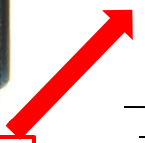


Horno

Entrada analógica



Llave (Tran)



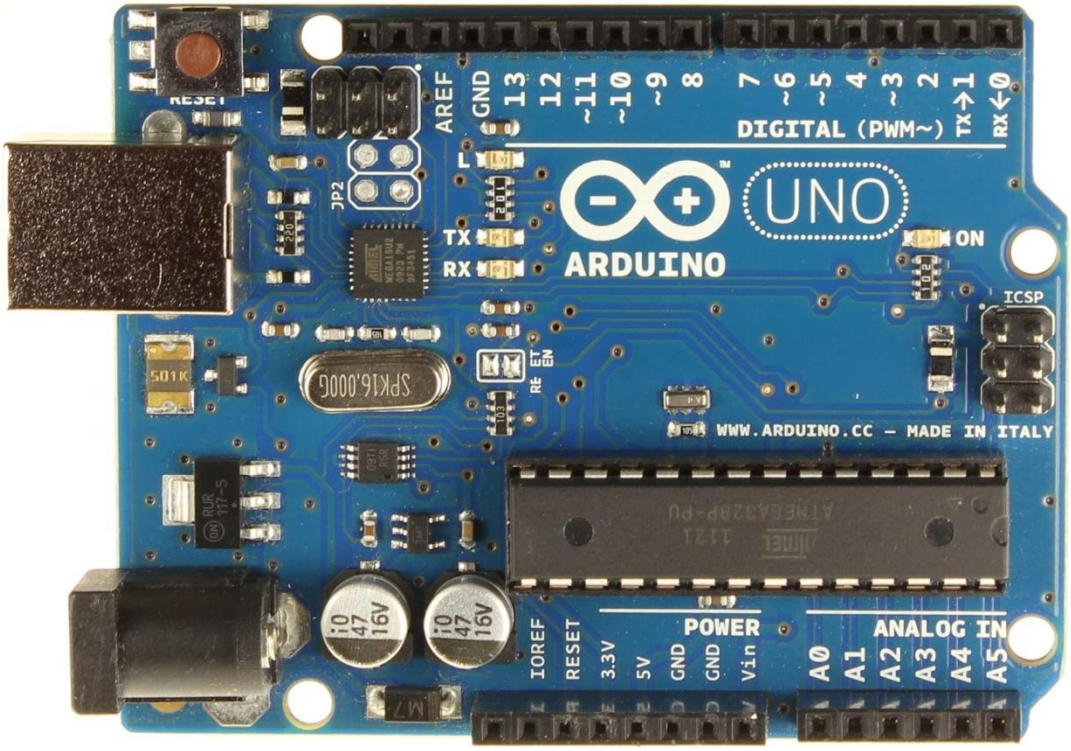
Salida digital

¿Cómo elegir un conversor AD?

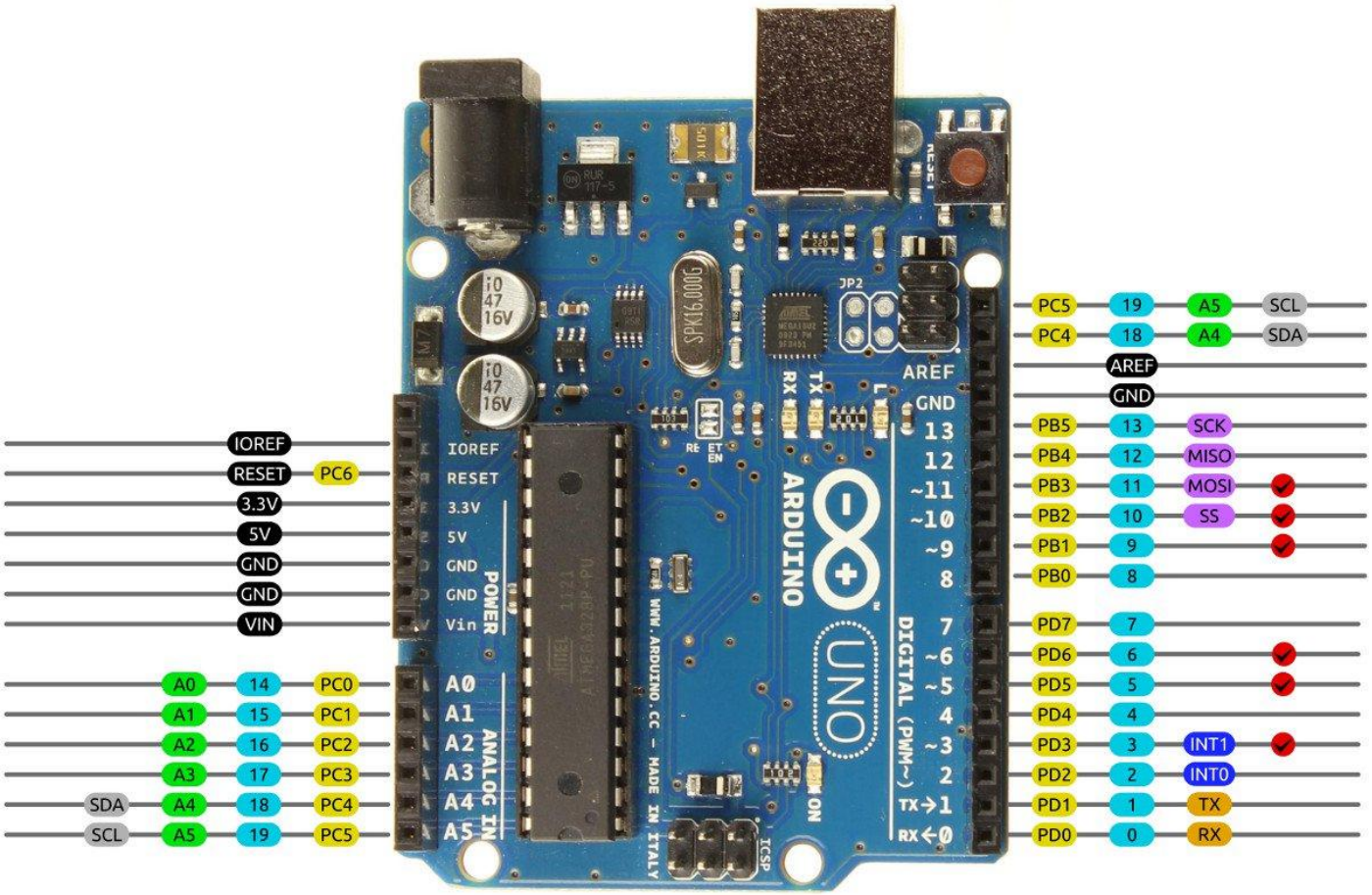
- 0^{o} C → 0 volt
- 500^{o} C → 5 volt (máx tensión en el conversor)
- Error = 0.5^{o} C
- CAD → 8 bits, 9 bits y 10 bits
- 8 bits → $500/2^{8} = 500/256$ aprox 2^{o}C
- 9 bits → $500/2^{9} = 500/512$ aprox 1^{o}C
- 10 bits → $500/2^{10} = 500/1024$ aprox 0.5^{o}C

¿Cómo elegir un conversor AD?

- Tiempo de conversión = Tc
- Tc= 1 mseg → $f_{muestreo}$ = 1/1= 1000Hz
- $f_{muestreo}$ = 2 f_{max} → f_{max} = 500 Hz
- Tc= 1 useg → $f_{muestreo}$ = 1/1= 1MHz
- $f_{muestreo}$ = 2 f_{max} → f_{max} = 500 kHz = 0.5 MHz


RESET
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
JP2
L
TX
RX
UNO
ARDUINO
ON
ICSP
WWW.ARDUINO.CC – MADE IN ITALY
POWER
ANALOG IN
IOREF
RESET
3.3V
5V
GND
GND
Vin
A0
A1
A2
A3
A4
A5

Arduino Uno R3 Pinout



AVR DIGITAL ANALOG POWER SERIAL SPI I2C PWM INTERRUPT

```
int T0=0;
int P1=0;
int opcion=0;
String medidas;
int ValvulaPV1=2;


void setup() {
  //Inicializa puerto serie
  Serial.begin(115200);
  pinMode(ValvulaPV1, OUTPUT);
}
void loop() {
  if (Serial.available()>0)
 {
     opcion=Serial.read();

 switch (opcion)
  {
   case ('1'):
   T0=analogRead(A1);
   medidas+=T0;
   medidas+=":";
   delay(1);
   P1=analogRead(A2);
   medidas+=P1;
   medidas+=":";
   delay(1);
   Serial.println(medidas);
   delay(200);
   medidas="";
   break;
   case('2'):
   digitalWrite(ValvulaPV1,HIGH);
   break;
   case('A'):
   digitalWrite(ValvulaPV1,LOW);
   break;
   }
}
}
```

```
python -m pip install PySerial
```

```
void setup() {
Serial.begin(9600);
}
void loop() {
Serial.println("Hola mundo");
delay(1000);
}
```

```
import serial, time
arduino = serial.Serial('COM4', 9600)
time.sleep(2)
rawString = arduino.readline()
print(rawString)
arduino.close()
```

```
import serial, time
arduino = serial.Serial("COM4", 9600)
time.sleep(2)
arduino.write(b'9')
arduino.close()
```

Type of Output Data	Output Data Format
Non-reading queries	<80 ASCII character string
Single reading (IEEE-488)	SD.DDDDDDDDESDD<nl>
Multiple readings (IEEE-488)	SD.DDDDDDDDESDD,...,... <nl>
Single reading (RS-232)	SD.DDDDDDDDESDD<cr><nl>
Multiple readings (RS-232)	SD.DDDDDDDDESDD,...,...<cr><nl>
	S Negative sign or positive sign
	D Numeric digits
	E Exponent
	<nl> newline character
	<cr> carriage return character

```
lectura1:=(lectura1+(devuelto[r]-$30)*Power(10,E));
r:=r-1;
E:=E+1;

        for k:=1 to 17 do

        devuelto1:=devuelto1+Chr(devuelto[k]);

            t2h:=devuelto[2];
            t2h:=t2h-48;
            t4h:=devuelto[4];
            t4h:=t4h-48;
            t5h:=devuelto[5];
            t5h:=t5h-48;
            t15h:=devuelto[15];
            t15h:=t15h-48;
```

Otro instrumento…….

```
fa:=devueltof[12];
        if fa <= 57 then fa:=fa-48
                else fa:=fa-55;
    fb:=devueltof[13];
        if fb <= 57 then fb:=fb-48
                else fb:=fb-55;
    fc:=devueltof[14];
        if fc <= 57 then fc:=fc-48
                else fc:=fc-55;

    fd:=devueltof[15];
        if fd <= 57 then fd:=fd-48
                else fd:=fd-55;

        lecflujo:=fd+fc*16+fb*256+fa*4096;

        lecflujo1:=(lecflujo/32000)*100;
```