

# **Práctica 1**

Percepción y Control para Sistemas Empotrados

Curso 2020-21

---

Izar Castorina

Si sabemos que el sensor tarda 100 ms para realizar el muestreo, el tiempo efectivo que necesitamos tomar como referencia para cada una de las muestras del gráfico será  $T+0.1$ .

Según esto, podemos ver que los valores de voltaje correspondientes a cada muestra son:

Tiempo (s)	Voltaje (mV)
T1 (efectivo: 0.3)	1000
T2 (efectivo: 0.7)	2000
T3 (efectivo: 0.9)	2200

El Arduino interpreta y convierte los valores de voltaje que lee desde cada sensor gracias a la función **getDistance** presente en el código de la placa. En particular, la función es:

```
1 // distance interpolation, 250mV intervals
2 const int TABLE_ENTRIES = 12;
3 const int INTERVAL = 250;
4 // look-up table
5 static int distance[TABLE_ENTRIES] = {150,140,130,100,60,50,40,35,30,25,20,15};
6 int getDistance(int mV) {
7 // this function transforms a voltage in mV from the ADC into a distance in cm
8     if (mV > INTERVAL * TABLE_ENTRIES - 1) return distance[TABLE_ENTRIES - 1];
9     else {
10         int index = mV / INTERVAL;
11         float frac = (mV % 250) / (float)INTERVAL;
12         return distance[index] - ((distance[index] - distance[index + 1]) * frac);
13     }
14 }
```

Si separamos esta función y la incorporamos en un programa de prueba – la manera más rápida de calcular las distancias – obtenemos los siguientes resultados para las tres muestras:

Voltaje (mV)	Distancia aproximada (cm)
1000	60
2000	30
2200	26

Analizando la función, podemos entender mejor cómo se hace la correspondencia entre los valores de voltaje y las distancias.

En la línea 5 de la función se describe un vector de 12 valores, cada uno correspondiente a una distancia, ordenados de manera decreciente de 150 a 15 cm. La constante **INTERVAL** (línea 3) nos indica que entre cada uno de estos valores hay una diferencia de voltaje de 250 mV. Efectivamente, el rango de voltajes para esta tabla iría de 250 mV para 150 cm hasta 2750 mV para 15 cm.

Una vez pasado por parámetro el valor de voltaje que se quiere convertir, la función verifica si es superior al valor máximo de 2750,  $(250 * (12 - 1))$ , y si así es, simplemente se devuelve el último valor de la tabla, indicando que la distancia es igual o inferior a aproximadamente 15 cm.

En el caso de que el voltaje sea menor que 2750, se calcula en qué posición de la tabla caería (línea 10). En la línea sucesiva, se calcula un valor decimal entre 0 y 1 que representa la resta de la división efectuada en la línea 10. Esto nos indica cuánto se acerca el voltaje al siguiente valor de distancia en la tabla.

Finalmente, en la línea 12 devolvemos el valor de distancia, igual al valor de distancia correspondiente al índice calculado en la línea 10 y, si **frac** es diferente de 0, realiza una interpolación para devolver un valor intermedio entre el actual y el siguiente de la tabla, empleando este valor decimal para saber de cuánto se ha acercado a este último.

Desde los resultados que hemos obtenido, es posible ver como el código haya realizado esta interpolación para el valor de voltaje 2200, porque cae entre 30 y 25 en la tabla, y **frac** resulta ser 0.8. Por tanto, el valor final, 26, viene dado por  $30 - (30 - 25) * 0.8$ .