

Conversor Analógico Digital  
Laboratorio N.º 3 – SE

### **Actividad 1**

**En base al valor de voltaje entregado por el sensor y al voltaje de referencia utilizado por el ADC, determine la función para convertir cada valor digitalizado en el valor de temperatura correspondiente en C. Incluya los cálculos realizados.**

Función de conversión: Sabemos que el máximo valor digital 1023 corresponde a una tensión de 5v. Con lo que el valor digital obtenido mediante el sensor de temperatura se convertirá con una regla de tres simple, resultando:  $\text{Temperatura [}^{\circ}\text{C]} = \text{valorSensado} * 5000 / 1023 / 10$ . Donde 5000 es el valor de tensión expresado en mV. La división por 10 se efectúa dado que el sensor representa 1°C por cada 10mV de tensión. Si por ejemplo censamos un valor digital de 41, la temperatura asociada será  $41 * 5000 / 1023 / 10 = 20,04^{\circ}\text{C}$

**Estime la frecuencia de muestreo necesaria para obtener los valores utilizados en el cálculo de la temperatura promedio.**

Se deberán obtener aproximadamente 7 muestras por segundo.

**Identifique las tareas del sistema e indique para cada una los dispositivos asociados (display, pulsadores, ADC, etc)**

Obtener muestras de temperatura → Sensor, ADC.

Diferenciar entre modos de operación → Pulsadores, Display, ADC.

Identificar pulsación de botones → ADC, Teclado (driver).

**Determine los eventos que iniciarán la ejecución de cada una de las tareas identificadas.**

El sensor estará activo desde la inicialización del sistema y ningún evento interrumpirá su tarea. (Cuando se esté leyendo el valor arrojado por el teclado, el sensor no habrá interrumpido su labor y viceversa, solo se estará leyendo otro canal).

Desde el instante inicial, el sistema comenzará a medir y mostrar la temperatura actual. El evento que ocasionará el cambio de modos es la pulsación del botón “Select”.

Al pulsar un botón, el ADC realiza la conversión AD, luego el controlador de teclado identificará a que pulsador corresponde.

**Determine las dependencias (sincronización y comunicación) entre tareas así como los datos compartidos entre ellas.**

Las tareas que comparten datos son las asociadas al manejo de temperaturas y justamente el dato que comparten es la temperatura medida en cierto instante de tiempo. Al haber adoptado un enfoque de planificación round robin, las tareas de teclado y sensor son totalmente independientes.

Es necesario sin embargo, sincronizar el acceso a ciertos parámetros del sistema tales como por ejemplo, el valor convertido por el ADC.

## **Actividad 2**

**Describa las modificaciones a realizar en la aplicación desarrollada en el laboratorio anterior para incorporar la capacidad de realizar comunicación bidireccional en serie asincrónica. Añada la descripción de las tareas y qué cambios, si existen, deben producirse en la estructura de control del sensor como consecuencia de su incorporación (ej. manejo de mensajes entrantes por polling o interrupciones, manejo de mensajes salientes, etc).**

A la solución de la actividad 1 se agregan dos funciones al programa principal:

```
void sendTemps()
```

```
void receive()
```

Ambas hacen uso de la librería Serial de Arduino. La primera escribe el buffer con un string conteniendo cada una de las temperaturas obtenidas, (actual, máxima, mínima, promedio). Por otro lado, receive() obtiene un dato provisto por la aplicación Processing que indica que hay que cambiar el modo de operación.

**Describa brevemente las modificaciones a realizar sobre el ejemplo para adaptarlo a los requerimientos del laboratorio.**

Se cambiaron las funciones a dibujar, de manera que en vez de la función coseno y la dependiente de la posición del puntero, ahora se dibujan funciones asociadas a las temperaturas recibidas por la aplicación Arduino.

Se añaden las instrucciones necesarias para recibir y enviar información a través de la librería serial de Processing.

Se eliminan botones considerados sobrantes. Solo se hace uso de un único botón, responsable de efectuar el cambio de modo.