

E.T.S. de Ingeniería Industrial,  
Informática y de Telecomunicación

# Diseño e implementación de un sistema de medida de humedad relativa basado en un sensor capacitivo P14 SMD



Grado en Ingeniería  
en Tecnologías Industriales

Trabajo Fin de Grado

David Rodríguez González

Angel María Andueza Unanua

Pamplona, 8 de Septiembre de 2022

upna

Universidad Pública de Navarra  
Nafarroako Unibertsitate Publikoa



# Índice.

ÍNDICE.....	1
ÍNDICE DE FIGURAS.....	2
1. RESUMEN.....	1
2. OBJETIVOS.....	3
3. INTRODUCCIÓN.....	4
3.1. LA HUMEDAD RELATIVA.....	4
3.2. MECANISMOS DE MEDIDA.....	4
3.3. ESTADO DEL ARTE.....	5
4. DISEÑO DEL CIRCUITO DE MEDIDA.....	7
4.1. COMPONENTES.....	7
4.1.1. SENSOR P14 FEMTOCAP.....	7
4.1.2. MKRWAN 1300.....	10
4.1.3. SAMD21G18A.....	12
4.2. MÉTODOS DE MEDIDA DE CAPACIDAD.....	14
4.2.1. CIRCUITO RC.....	14
4.2.2. CARGA DE UN C CONOCIDO.....	16
4.3. MÉTODO ELEGIDO Y CIRCUITO.....	18
4.4. CONFIGURACIÓN DEL ADC.....	20
4.5. CÁLCULO Y SIMULACIÓN DE LA IMPEDANCIA DE ENTRADA DEL ADC.....	23
5. PROGRAMACIÓN.....	26
5.1. PROGRAMA DE MEDIDA DE CAPACIDAD.....	27
5.1.1. INICIALIZACION DE VARIABLES:.....	28
5.1.2. CONFIGURACIÓN (VOID SETUP).....	29
5.1.3. BUCLE (VOID LOOP).....	30
6. PROCESO DE CALIBRACIÓN.....	32
6.1. ANÁLISIS DE ERRORES.....	32
6.2. MEDIDAS EXPERIMENTALES.....	32
6.2.1. PRIMERA SECUENCIA DE MEDIDAS:.....	34
6.2.2. SEGUNDA SECUENCIA DE MEDIDAS.....	35
6.3. ANÁLISIS DE LOS RESULTADOS.....	38
6.4. PROPUESTA DE DISEÑO DE UNA PCB PARA EL CIRCUITO.....	40
7. CONCLUSIONES.....	42
8. BIBLIOGRAFÍA.....	43
9. ANEXOS.....	45

# Índice de figuras

Figura 1 Higrómetros de RS electronics [1].....	5
Figura 2 módulos de humedad de IST [2].....	6
Figura 3 sensores capacitivos de humedad de IST [2].....	6
Figura 4 (Izda) Dimensiones del sensor [3]. (Dcha) Sensor soldado a una PCB..	7
Figura 5 Curva de calibración. [3] .....	8
Figura 6 Ecuación de la influencia de la T en la medida de %HR [3] .....	8
Figura 7 Errores del HR calculados para diferentes temperaturas. ....	8
Figura 8 Margen de error de la curva característica del sensor.....	9
Figura 9 Distribución de pines en el MKRWAN 1300. [4] .....	10
Figura 10 Empaquetado y pines del SAMD21. [5].....	12
Figura 11 características del SAMD21 comparadas con un mega328P [6].....	12
Figura 12 Diagrama de funcionamiento del sistema. ....	14
Figura 13 Circuito RC. ....	14
Figura 14 Curva RC: representación gráfica y matemática. ....	15
Figura 15 Circuito CRC.....	16
Figura 16 Pines A Y B a GND.....	16
Figura 17 Pin a Vcc y Pin B desconectado. ....	16
Figura 18 Pin A desconectando y midiendo, Pin B a GND.....	17
Figura 19 Circuito final RC.....	18
Figura 20 Esquema de los registros del ADC. [5].....	20
Figura 21 Tabla de registros del ADC [5] .....	21
Figura 22 fórmulas para el tiempo de conversión del ADC [5] [7].....	22
Figura 23 Ejemplo de cálculo online del tiempo de conversión [7] .....	22
Figura 24 Circuito para el cálculo de la impedancia interna del ADC. ....	23
Figura 26 Circuito para el cálculo de la Rx del multímetro .....	24
Figura 26 Simulación de los efectos de la Rx del multímetro en la curva RC.....	25

Figura 27 Sección a modificar del archivo "wiring.c" .....	26
Figura 28 Diagrama del programa de medida de capacidad.....	27
Figura 29 Código de inicialización de variables .....	28
Figura 31 Setup del programa .....	29
Figura 32 Loop del programa.....	30
Figura 32 muestras del sensor cada 5 minutos.....	31
Figura 33 dimensiones del DHT11.....	33
Figura 34 Diagrama de conexiones del DHT11 .....	33
Figura 35 Características del DHT11 .....	33
Figura 36 P14 y DHT11 colocados juntos. ....	34
Figura 37 Medidas experimentales .....	34
Figura 39 Configuración final del MKR y P14.....	35
Figura 39 Ortofoto de la ubicación de la estación del GN [8] .....	36
Figura 40 Equipos de la estación meteorológica [9].....	36
Figura 41 Ubicación del equipo de calibrado [8] .....	37
Figura 42 Montaje del equipo de calibrado .....	37
Figura 43 comparación de la medida de humedad entre la estación y el DHT1138	
Figura 45 medidas de DHT11 y P14 para $S=0.3$ y $C30=180$ .....	39
Figura 45 Código de calibración .....	40
Figura 46 footprint del PCB para el circuito de medida de capacidad .....	40
Figura 47 simulación 3D de la PCB .....	40

# 1. RESUMEN

Desde hace años, la creación de dispositivos como las Raspberry Pi, o Arduino, ha acercado el mundo de la electrónica a un mayor número de personas. Actualmente, la electrónica es barata y fácil de programar, lejos de los antiguos sistemas operantes hace más de una década.

La mayoría de dispositivos electrónicos son capaces de medir voltajes y capacidad, si bien lo hacen con limitaciones. Además, muchos sensores basan sus medidas en otro tipo de variables. En el caso de la medida de la humedad relativa de este proyecto, la humedad relativa se suele medir en los cambios de la capacidad de un condensador. Existen diferentes módulos externos para adaptar estas señales, osciladores, circuitos generadores de onda, etc. Pero, la intención de este proyecto es buscar una solución alternativa simple, lineal y más flexible.

Mediante el estudio de diferentes métodos para acondicionar la señal, se diseñará, simulará y montará un circuito de medida, acompañado de la programación requerida para el método elegido. Posteriormente, mediante la realización de varias medidas experimentales se calibrará. El sensor utilizado es el P14 FemtoCap y será procesado mediante un Arduino MKRWAN1300 equipado con un microprocesador SAMD21.

## **Palabras clave:**

- Humedad relativa
- SAMD21
- instrumentación
- MKRWAN1300
- Arduino
- Sensor capacitivo

## SUMMARY

For years, the creation of devices such as Raspberry Pi, or Arduino, has brought the world of electronics to a greater number of people. Nowadays, electronics is cheap and easy to program, far from the old systems that operate one decade ago.

Most electronic devices are capable of measuring voltages and capacitance, but do so with limitations. In addition, many sensors base their measurements on other types of variables. In the case of the relative humidity measurement for this project, it is measured on the capacitance changes of a capacitor. There are different external modules to adapt these signals, oscillators, wave generator circuits, etc. But, the intention of this project is to look for a alternative, simple, linear and more flexible solution.

By studying different methods to condition the signal, a measurement circuit will be designed, simulated and assembled, together with the programming required for the chosen method. Subsequently, by performing several experimental measurements, it will be calibrated. The sensor used is the P14 FemtoCap and will be processed by an Arduino MKRWAN1300 equipped with a SAMD21 microprocessor.

### Key words:

- Relative humidity
- SAMD21
- Instrumentation
- MKRWAN1300
- Arduino
- Capacitive sensor

## 2. OBJETIVOS

El objetivo del presente proyecto es el diseño y desarrollo de un sistema electrónico que sea capaz de medir la Humedad Relativa (%HR) del ambiente. Este sistema está formado por un sensor capacitivo comercial P14 FemtoCap tipo SMD de la marca “Innovative Sensor Technology IST AG” y una placa Arduino MKR equipada con un microcontrolador ( $\mu$ C) SAMD21. De este modo, se pretende conseguir un dispositivo que emplee solo señales DC y evite el uso de señales en alterna, simplificando el circuito y el proceso de medida.

Además de lo anterior, el proyecto persigue los siguientes objetivos secundarios:

- Aplicar los conocimientos adquiridos sobre instrumentación industrial para el ajuste y calibración de un sensor de humedad, analizando características como sistema de instrumentación electrónica.
- Desarrollar un circuito RC sencillo y eficaz para el acondicionamiento de sensores capacitivos sin emplear señales en alterna, solo mediante las propias señales del  $\mu$ C SAMD21.
- Analizar analíticamente y mediante simulación con LTspice el circuito de medida del sensor capacitivo de humedad relativa, teniendo en cuenta las propiedades eléctricas de los componentes que lo forman.
- Diseñar y programar una configuración alternativa a la función *AnalogRead* del IDE de Arduino para controlar el convertidor Analógico-Digital (ADC) del  $\mu$ C y poder trabajar con señales en el orden de los microsegundos ( $\mu$ s).
- Realizar medidas experimentales del sistema diseñado empleando el sensor capacitivo, para evaluar su calidad y nivel de precisión.
- Analizar el nivel de error e imprecisión final del circuito propuesto.
- Compartir los resultados con la comunidad a través de espacios online como GitHub o Instructables.



## 3. INTRODUCCIÓN

### 3.1. LA HUMEDAD RELATIVA

La humedad relativa es un factor climático que expresa el porcentaje de vapor contenido en el ambiente respecto al máximo admisible para una presión y temperatura dadas. Este factor se expresa en porcentaje mediante la nomenclatura %HR.

$$\%HR = \frac{\text{masa de H}_2\text{O en el aire [g]}}{\text{masa max de H}_2\text{O admisible del aire [g]}}$$

La cantidad de vapor de agua que puede absorber el aire depende de su temperatura y presión. El aire caliente admite más vapor de agua que el aire frío.

La humedad puede llegar a dañar materiales sensibles, como es el caso de documentación histórica, o algunos procesos técnicos.,

A su vez, puede llegar a afectar al bienestar y la salud. Una baja HR, un entorno seco, puede tener consecuencias negativas en la piel y conductos nasales. Una elevada HR puede llegar a interferir con los procesos corporales de evacuación del calor, aumentando la sensación térmica de forma considerable.

Por tanto, junto con la temperatura, la HR es uno de los principales factores que son necesarios medir en entornos de climatización controlada.

### 3.2. MECANISMOS DE MEDIDA

Los instrumentos de medida de la humedad por lo general se basan en las mediciones de alguna otra magnitud como la temperatura, la presión, la masa o un cambio mecánico o eléctrico en una sustancia cuando absorbe la humedad.

Los primeros aparatos modernos se basaban en el uso de materiales higroscópico, es decir, capaces a acumular humedad modificando su longitud. Estos cambios se ampliaban mecánicamente para poder ser interpretados por una escala.

Posteriormente se desarrollaron los psicrómetros. Estos aparatos se basan en la influencia de la humedad en la temperatura de un fluido. Se componen de dos termómetros: uno expuesto al exterior y otro permanentemente húmedo. La diferencia de temperaturas permite mediante unas tablas el obtener el valor de forma precisa.

Con la invención de los semiconductores y el desarrollo de la electrónica, han proliferado los sensores basados en la interacción de la HR con las propiedades eléctricas de materiales y ambientes.

Los sensores resistivos se basan en la medida de la variación de la resistencia eléctrica de un material. Suelen requerir circuitos complejos y medidas de T para complementar la medida.

Finalmente, los sensores de humedad electrónicos más extendidos son los capacitivos. Se basan principalmente en la variación de la constante dieléctrica ( $\epsilon$ ) la cual modifica la capacidad del sensor. Son sencillos, asequibles y con una buena calibración pueden llegar a tener una precisión significativa

### 3.3. ESTADO DEL ARTE

Existen diferentes alternativas en el mercado para medir la HR del aire.

En el caso de requerir una medida directa sin necesidad de diseño o montaje existen una gran cantidad de dispositivos de medida, ya sean manuales o de pared que se pueden adquirir completos. Estos dispositivos llamados higrómetros tienden a ser digitales y generalmente vienen con medidas complementarias ambientales como temperatura o % CO<sub>2</sub>. Hay una gran variedad de productos y precios según la precisión, la aplicación (industrial o domestica), el fabricante, etc.




	<b>Higrómetro de mano RS PRO RS1364, calibrado RS, humedad máx. 95%HR, temperatura máx. +140 °F, +60 °C</b> Código RS: 123-2269	286,91 € Unidad	Añadir	- 1 +				
Comparar <input type="checkbox"/>		Rotronic Instruments	-	-	Higrómetro digital			
	<b>Higrómetro Rotronic Instruments, calibrado RS</b> Código RS: 200-5020 Nº ref. fabric.: HL-20D-SET	788,40 € Unidad	Añadir	- 1 +				
Comparar <input type="checkbox"/>		RS PRO	+100 °C, +212°F	100%HR	Higrómetro digital			
	<b>Higrómetro de mano RS PRO, calibrado RS, humedad máx. 100%HR, temperatura máx. +100 °C, +212°F, interfaz USB</b> Código RS: 192-6504	319,50 € Unidad	Añadir	- 1 +				

Figura 1 Higrómetros de RS electronics [1]

En el caso de querer montar el sistema de medida, la mejor opción es la compra de módulos, es decir, pequeñas PCBs equipadas con un sensor, el circuito de acondicionamiento y unos pines adecuados para su incorporación a un sistema electrónico o de procesado. Al venir montado y calibrado de fábrica se asegura la calidad de las medidas y se simplifica el trabajo. Se pueden ver algunos ejemplos de diferentes configuraciones y precios en la Figura 2.




	<p><b>Humidity module for dew point monitoring</b></p> <p>Digital RH/T module, in a 16 bar pressure tight TO housing Accuracy humidity: <math>\pm 0.5</math> %RH (0...5 %RH), <math>\pm 1.0</math> %RH (5...10 %RH)   Accuracy temperature: <math>\pm 0.2</math> °C   Product code: 153398</p> <p><input type="radio"/> Compare</p> <p style="text-align: right;"><b>from € 51.33</b></p>	♥
	<p><b>Humidity module for dew point monitoring, assembled</b></p> <p>Digital RH/T module with I<sup>2</sup>C outputs, low humidity calibration, with housing and cable, up to 10 bar Accuracy humidity: <math>\pm 0.5</math> %RH (0...5 %RH), <math>\pm 1.0</math> %RH (5...10 %RH)   Accuracy temperature: <math>\pm 0.2</math> °C   Product code: 153326</p> <p><input type="radio"/> Compare</p> <p style="text-align: right;"><b>from € 92.39</b></p>	♥
	<p><b>Humidity module HYT 271 with pluggable SIL contacts</b></p> <p>Digital RH/T module, for up to 125°C and 100% RH, withstands condensation Accuracy humidity: <math>\pm 1.8</math> % RH at +23 °C   Accuracy temperature: <math>\pm 0.2</math> °C   Product code: 153349</p> <p><input type="radio"/> Compare</p> <p style="text-align: right;"><b>from € 20.02</b></p>	♥

Figura 2 módulos de humedad de IST [2]

Pero en caso de requerir mayor control sobre el acondicionamiento, o para aplicaciones docentes, suele convenir trabajar a nivel de sensor. Si bien se pueden producir daños o generar errores por una mala manipulación de los componentes o errores del montaje, se obtiene un mayor control del diseño y del funcionamiento. Es recomendable el incorporar el sistema a un módulo, para las aplicaciones finales, por tanto los sensores se construyen de manera que sea fáciles de soldar, ya sea en formato SMD o agujero pasante. Se pueden ver algunos ejemplos de diferentes configuraciones y precios en la Figura 3




	<p><b>Capacitive Humidity Sensor P14 Rapid-W (140pF +/-40 pF)</b></p> <p>Capacitive humidity sensor P14-Rapid with wires. Stable and robust humidity sensor with ultra-fast response time, Capacitance: 140pF +/-40pF (at +23 °C and 30 % RH)   Connections: Au/Cu-wire   Product code: 103573</p> <p><input type="radio"/> Compare</p> <p style="text-align: right;"><b>from € 25.66</b></p>	♥
	<p><b>Capacitive Humidity Sensor P14-W</b></p> <p>Humidity allrounder, -50 °C to +150 °C, 5.0 x 3.81 mm, 10 mm wires Capacitance: 150pF +/-50pF (at +23 °C and 30 % RH)   Dimensions: 5.0 x 3.81 x 0.4 mm   Product code: 150466</p> <p><input type="radio"/> Compare</p> <p style="text-align: right;"><b>from € 14.68</b></p>	♥
	<p><b>P14 FemtoCap-G SMD (180 pF +/-50 pF)</b></p> <p>Optimal for automotive and white good applications, -50 °C to +150 °C, 4.0 x 2.0 mm, SMD 1608, automatic assembly compatible Capacitance: 180pF +/-50pF (at +23 °C and 30 % RH)   Connection type: SMD-Pads   Product code: 103563</p> <p><input type="radio"/> Compare</p> <p style="text-align: right;"><b>from € 13.86</b></p>	♥

Figura 3 sensores capacitivos de humedad de IST [2]

## 4. DISEÑO DEL CIRCUITO DE MEDIDA

### 4.1. COMPONENTES

El objetivo principal del proyecto es diseñar un circuito de medida de HR mediante un sensor capacitivo. Dada su naturaleza, es necesario considerar estrategias de medida adaptadas a las propiedades eléctricas de estos componentes. A continuación, introduciremos los componentes principales a utilizar en el sistema:

- Sensor P14 FEMTOCAP tipo SMD
- Arduino MKRWAN 1300
- $\mu\text{C}$  SAMD21

#### 4.1.1. SENSOR P14 FEMTOCAP

Para el desarrollo de este proyecto utilizaremos el sensor capacitivo P14, un sensor SMD cuya capacidad varía entre 170 y 200 pF. Es un sensor pequeño y manejable, asequible y con un elevado rango de temperatura de trabajo.

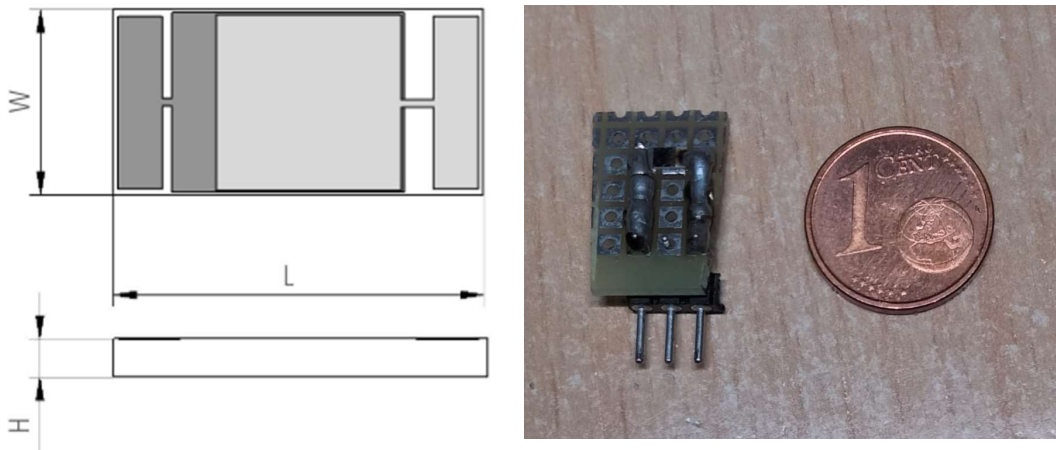


Figura 4 (Izda) Dimensiones del sensor [3]. (Dcha) Sensor soldado a una PCB

Las dimensiones del sensor indicadas en la Figura 4 en mm son (L x W x H): 4 x 2 x 0.4. Éste se ha soldado a una PCB con una serie de pines para su incorporación al circuito de medida, quedando el montaje indicado en la **¡Error! No se encuentra el origen de la referencia..**

Se ha escogido este sensor principalmente debido a que mantiene una relación lineal entre la HR y la capacidad a lo largo de gran parte del rango de medida como se indica en las especificaciones de su datasheet (ver Figura 5). Esta relación simplifica en gran medida las operaciones necesarias para la conversión de los datos.



Como podemos observar en la Figura 7, los efectos son casi nulos entre 20 y 25°C, solo son relativamente considerables en valores de temperatura muy alejados del nominal. Para entornos de temperaturas extremas, sería recomendable la inclusión de un sensor de Tª, para mejorar la precisión de los datos durante el calibrado.

Un problema que sí debe tenerse en cuenta a la hora de tratar el sensor es su elevada variabilidad. Su capacidad típica a %HR de 30 es de 180pF±50pF, por tanto, asumiendo que la sensibilidad de 0,3pF/%HR se mantiene estable, la curva real del sensor estará contenida entre las curvas rojo (valor máximo) y verde (valor mínimo) de la Figura 8.

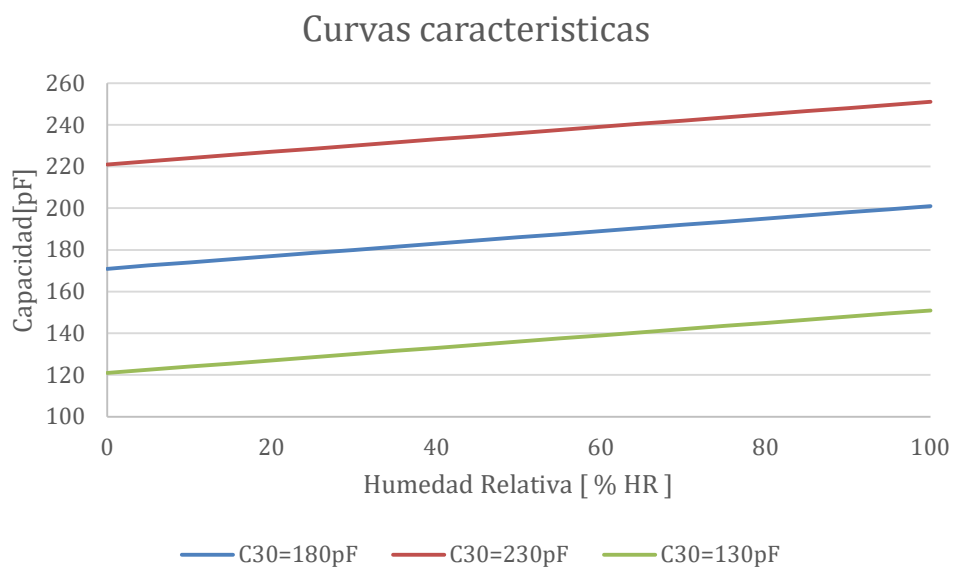


Figura 8 Margen de error de la curva característica del sensor.

### 4.1.2. MKRWAN 1300

Para el proyecto se utilizará una placa Arduino tipo maker (MKR), basada en el  $\mu$ C de **32 bits SAMD21G18A**.

El Arduino MKRWAN1300 es el primero de su clase con compatibilidad con redes LoRa®, lo cual le permite mandar los datos medidos de forma inalámbrica para su almacenaje y procesado en otros equipos.

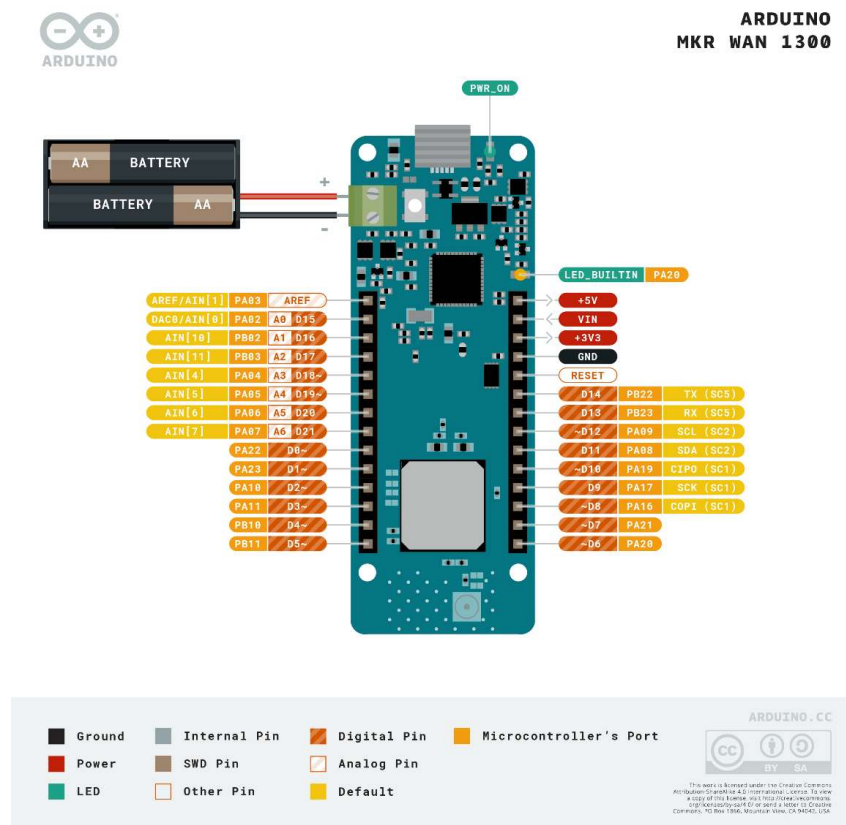


Figura 9 Distribución de pines en el MKRWAN 1300. [4]

En la Figura 9 se puede observar un esquema con la configuración de los 28 pines del Arduino. El MKR está equipado con 4 pines de potencia:

- GND
- 3.3V
- Vin, el cual sirve para introducir una tensión que pueda ser utilizada como referencia por el  $\mu$ C
- 5V

El reinicio del equipo, además de con un pulsador, se puede controlar mediante un pin de reset que se activa al conectarse a GND. Está equipado con 7 entradas analógicas, las cuales también se pueden configurar como pines digitales. En ese caso se pueden llegar a tener hasta 21 pines digitales. Trabaja con tensiones de I/O de 3.3V, importante ya que trabajar con 5V, también presentes en la sección de potencia, puede dañar la placa. Puede llegar a tener hasta 12 pines digitales generando o recibiendo señales de tipo PWM (Pulse Width Modulation). Permite la comunicación serie mediante el protocolo UART/USART haciendo uso de los pines D13 (Rx) para recibir y D14 (Tx) para transmitir datos. Otro protocolo compatible con la placa es el I<sup>2</sup>C, un protocolo síncrono que mediante el pin D12 (SCL) manda la señal de reloj y por el pin D11 (SDA) envía los datos. El último protocolo compatible es el SPI. El pin D9 (SCK) manda la señal de reloj, el pin D8 (COPI "Controller Out / Peripheral In") manda los datos desde el MKR al módulo externo y el pin D10 (CIPO "Controller In / Peripheral Out") manda los datos desde el módulo al MKR. Este protocolo es muy útil a la hora de almacenar datos de forma externa en tarjetas de memoria, pero requiere de los módulos adecuados y una buena programación.



### 4.1.3. SAMD21G18A

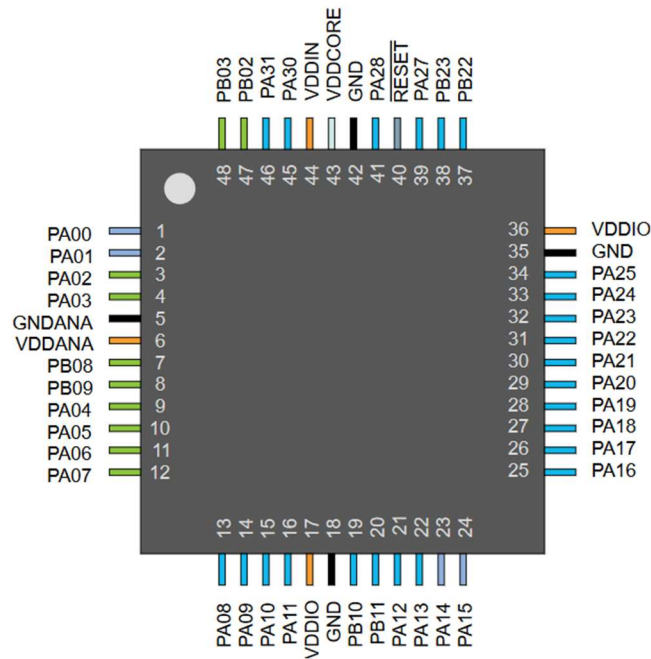


Figura 10 Empaquetado y pines del SAMD21. [5]

Feature	ATSAMD21G18A	ATmega328P
Architecture	ARM Cortex-M0+	AVR
Bus Size	32-bit	8-bit
Max CPU Speed	48MHz	20MHz
Flash	256KB	32KB
SRAM	32KB	2KB
EEPROM	32KB (emulated in Flash)	1KB
Voltage Range	1.62-3.63V	1.8-5.5V
GPIO Count	38	23
ADC Channels	14	8
ADC Resolution	12-bit	10-bit
Digital-to-Analog Converter (DAC)	Yes	No
USB Controller	Yes	No
Direct Memory Access (DMA)	12 Channels	No
Peripheral Touch Controller	Yes	No
Inter-IC Sound I <sup>2</sup> S	Yes	No

Figura 11 características del SAMD21 comparadas con un mega328P [6]

El SAMD21 es un popular  $\mu$ C debido a unas buenas prestaciones y un precio competitivo, que lo hace muy atractivo frente sus competidores. Utiliza un procesador **ARM Cortex-M0+ de 32-bits**, con 256KB de flash, 32KB de SRAM y funcionando a un máximo de **48MHz**.

Está equipado con 6 módulos de comunicación en serie (SERCOM) siendo capaz de utilizar los protocolos UART/USART, SPI o I<sup>2</sup>C, tal y como se han explicado en el apartado anterior.

Puede utilizar hasta 3 temporizadores/ contadores de 16bit, además de un RTC de 32bits y hasta un calendario interno. Pueden utilizarse para generar interrupciones, pero admite hasta 16 entradas de interrupción.

Una característica muy importante en aplicaciones de largo plazo son los modos de bajo consumo, "Idle" y "Stand-by".

Los modelos de mayor cantidad de pines están equipados con hasta 20 canales PWM.

Respecto a las señales analógicas, incorpora un convertidor analógico-digital (ADC) de hasta 14 canales y resolución de 8, 10 o 12 bits. Además, al contrario que otros  $\mu$ C, tiene un convertidor digital-analógico (DCA) de un canal y resolución de 10 bits.

Para optimizar el espacio de la memoria, las funcionalidades del micro se configuran mediante multiplexación, obteniendo diversas posibilidades de funcionamiento para un mismo pin. La disposición de sus pines facilita la migración entre dispositivos de su misma familia.

Su tensión de funcionamiento es de entre 1.62V to 3.63V, compatible con varios dispositivos, pero con riesgo de sobretensión ante dispositivos de 5V

Al ser un producto de la empresa Microchip y estar muy extendido su uso gracias a Arduino u otros dispositivos, hay una gran cantidad de información y asistencia online. Además es compatible programas como MPLAB X IDE.

## 4.2. MÉTODOS DE MEDIDA DE CAPACIDAD

Los pines de entrada del ADC del MKR solo son capaces de registrar tensión, pero el sensor cambia su capacidad (orden de pF) por lo que es necesario un circuito de acondicionamiento de señal para generar una variación de tensión relacionada con la capacidad del sensor. Clásicamente, el cambio en la tensión puede estar modulado en variación de la amplitud de la señal o en el tiempo necesario para alcanzarlo. En cualquiera de los casos el microcontrolador, debe ser capaz de registrar los cambios de la tensión de entrada para poder determinar la capacidad y con ello la HR.

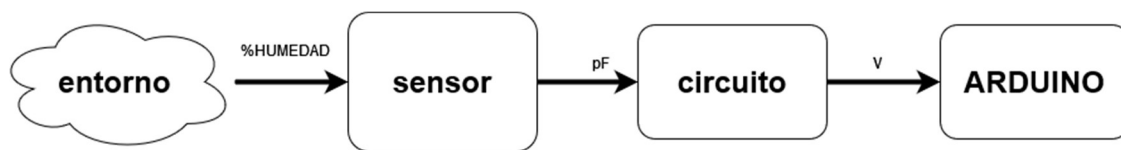


Figura 12 Diagrama de funcionamiento del sistema.

Los valores de capacidad del sensor oscilan entre 100 y 250 pF, y los pines de entrada del ADC de la placa MKR operan con tensiones entre 0 y 3.3V.

Para esta conversión se estudiará el uso de diferentes métodos de medida: un circuito RC y la carga de un condensador conocido.

### 4.2.1. CIRCUITO RC

El circuito está formado, como indica la Figura 13, por una fuente DC, una R y una C colocados en serie.

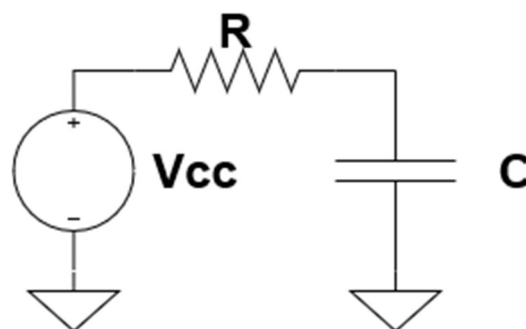


Figura 13 Circuito RC.

Cuando un condensador descargado se ve sometido a una tensión constante, su curva de carga presenta un comportamiento exponencial como se puede observar en la Figura 14. Este proceso de carga puede ser utilizado para determinar su capacidad.

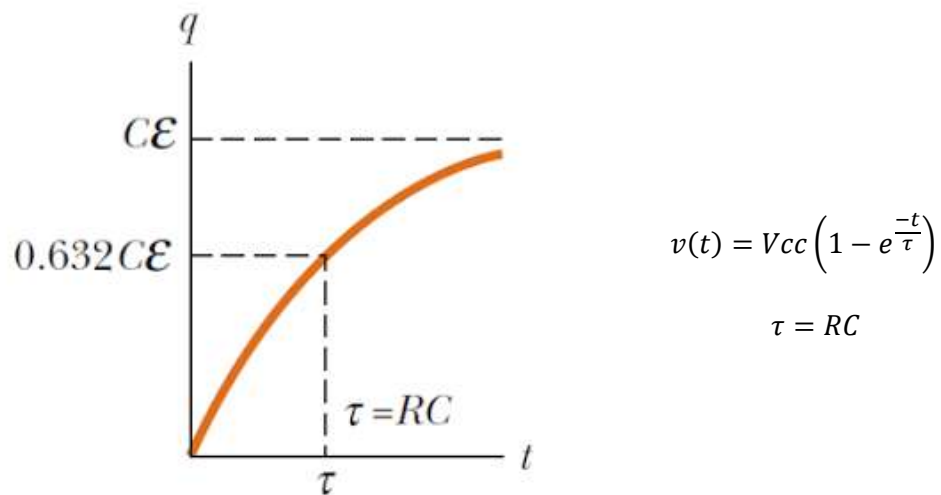


Figura 14 Curva RC: representación gráfica y matemática.

El tiempo que la tensión tarda en alcanzar al 63.2% de la tensión de entrada se denomina constante de tiempo ( $\tau$ ) y su valor corresponde al producto de la resistencia y la capacidad del circuito.

Utilizando esta relación y comparando la tensión medida con el valor umbral de referencia dado por el 63% de la máxima tensión a la que puede cargarse el condensador (en la Figura 13,  $V_{cc}$ ) se puede calcular  $\tau$ , y conociendo el valor de la resistencia R, calcular el valor de capacidad C que presentará el circuito.

#### 4.2.2. CARGA DE UN C CONOCIDO

Este método se utiliza muy comúnmente en los multímetros para la medida de condensadores del orden de 1nF-100pF, el rango de medida del caso a estudiar. El circuito indicado en la Figura 15, consta de dos condensadores, una resistencia y dos pines del microcontrolador. Estos pines se utilizarán para modificar las tensiones del circuito.

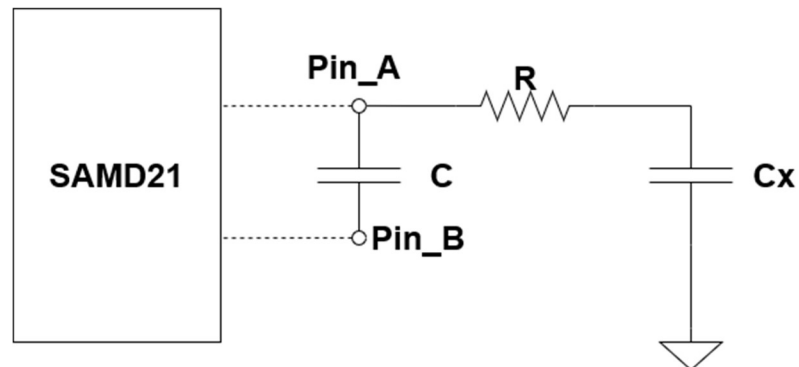


Figura 15 Circuito CRC.

Inicialmente se conectan ambos pines a tierra para descargar el circuito

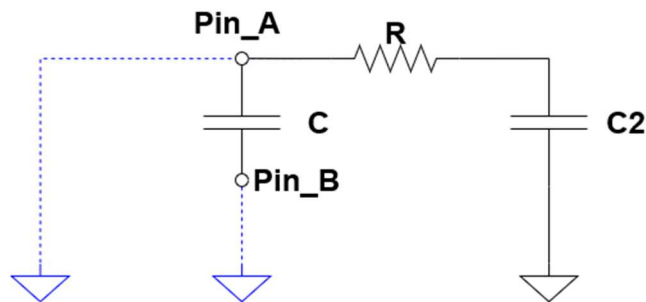


Figura 16 Pines A Y B a GND.

Posteriormente se aísla en PINB poniendo el terminal en estado de alta impedancia, eliminando los efectos de C del circuito. El pin A introduce una tensión para cargar el condensador desconocido quedando un simple circuito RC. Tras pasar un tiempo el condensador acumulara una carga  $q=V_{cc} \cdot C_x$ .

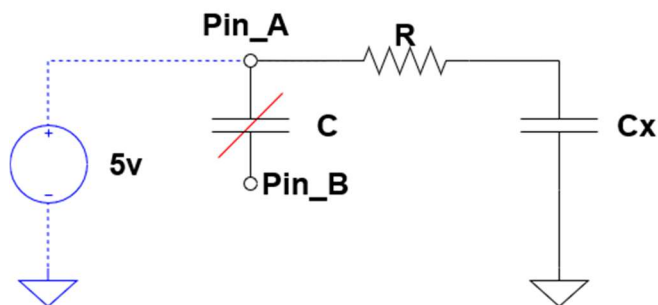


Figura 17 Pin a Vcc y Pin B desconectado.

Finalmente se configura el pin A como alta impedancia y se conecta a tierra el pin B (ver Figura 18).

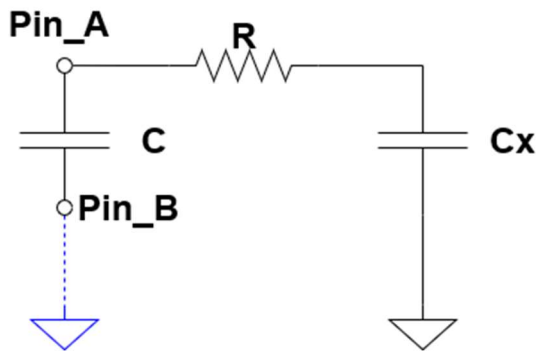


Figura 18 Pin A desconectando y midiendo, Pin B a GND.

Teniendo en cuenta que en los condensadores en serie la carga se conserva (de forma análoga a la corriente para R en serie) midiendo la diferencia de potencial de C y conociendo su valor de capacidad podemos extrapolar el valor de C<sub>x</sub> con la siguiente formula:

$$C_x = C \cdot \frac{V_{pin\_A}}{V_{CC}}$$

Este método tiene como ventaja que el valor de R no condiciona el funcionamiento, y no requiere altas velocidades, pero para que funcione bien se debe tener en cuenta la diferencia de magnitud entre C y C<sub>x</sub>. Si C >> C<sub>x</sub> la precisión del método disminuirá, siendo el error mayor para resoluciones del Convertidor Analógico Digital (ADC) bajas.

Tabla 1 Relación entre C<sub>1</sub> y C<sub>2</sub> para diferentes resoluciones.

C <sub>x</sub> / C	8 bits	10 bits	12 bits
1	255	1024	4096
0,5	128	512	2048
0,1	26	102	410
0,05	13	51	205
0,01	3	10	41

Cuando la diferencia de magnitud aumenta, el ruido y la variabilidad afectaran cada vez más a la medida. Esto se puede compensar con una buena elección de C<sub>1</sub> y con un ajuste de la resolución.

### 4.3. MÉTODO ELEGIDO Y CIRCUITO

Finalmente se ha escogido el método RC al ser el más sencillo de implementar. La clave de este método se sitúa en el correcto ajuste de las configuraciones del convertidor ADC y la elección de una resistencia R adecuada.

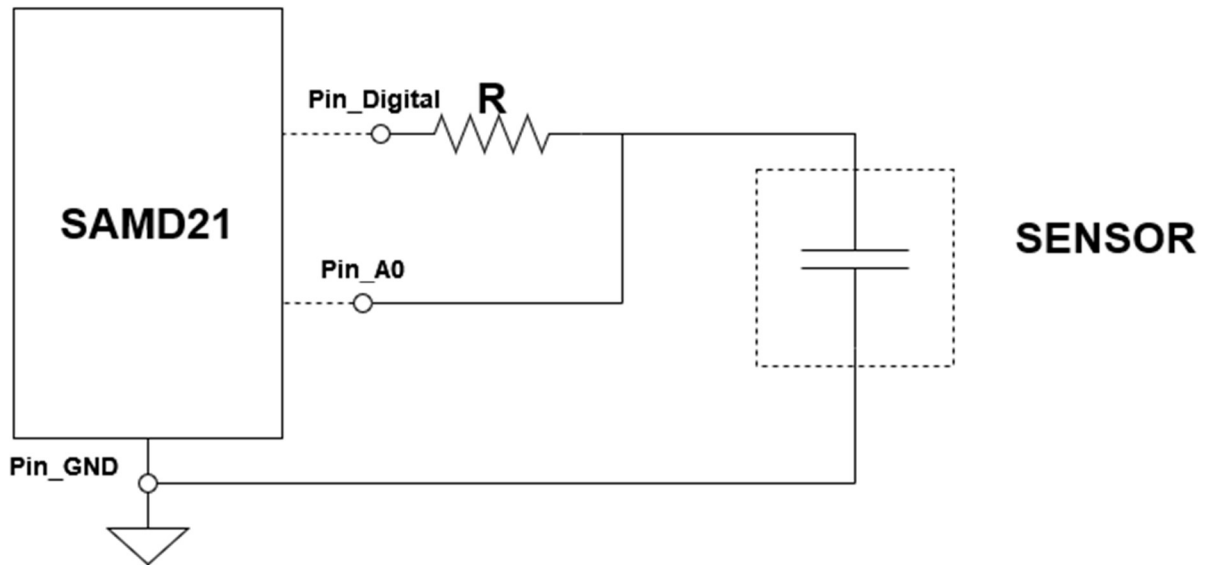


Figura 19 Circuito final RC.

Dado que para poder determinar el tiempo de carga del C debemos emplear el ADC, es necesario que este muestree a suficiente velocidad como para poder seguir la curva de carga de la tensión en el circuito. Por tanto, el factor más limitante para el ADC va a ser el valor de  $\tau$ , y este depende del valor seleccionado para R operando con el valor de variación de C, correspondiente al sensor. En la Tabla 2 se muestra la escala de  $\tau$  para diferentes ordenes de valores de R y C.

Tabla 2 Relación entre  $\tau$  y R para diferentes magnitudes de C

C	R	$\tau$
$\mu\text{F}$	$\text{k}\Omega$	ms
nF	$\text{k}\Omega$	$\mu\text{s}$
pF	$\text{k}\Omega$	ns
pF	$\text{M}\Omega$	$\mu\text{s}$

Inicialmente se debe elegir un valor de R adecuado para el rango de C a medir. En este caso, al tratarse únicamente de valores de 250-100pF, la resistencia a utilizar debe rondar  $1\text{M}\Omega$  para que  $\tau$  sea de unos 250-100  $\mu\text{s}$ .

Como consecuencia de lo todo anterior, es necesario que la tarjeta MKR procese las muestras del convertidor lo suficientemente rápido como para poder estimar el valor

de  $\tau$  sin errores, es decir, que la frecuencia de muestreo sea significativamente mayor que la máxima frecuencia del transitorio de carga del condensador.

La medida de la tensión se deberá hacer mediante el uso de los pines analógicos del ADC. En el convertidor analógico-digital hay que tener en cuenta dos factores temporales:

1. Hay que aplicar una frecuencia de muestreo lo suficiente grande como para no perder información significativa, de más del doble que la frecuencia máxima del transitorio (criterio Nyquist).
2. Hay que tener el tiempo suficiente para la conversión y procesado de los datos antes de la siguiente muestra.

En el caso de un Arduino MKR, una lectura analógica mediante la función `AnalogRead()` puede tardar entre 400 y 1200 $\mu$ s, lo cual es entre 3 y 10 veces superior al valor de  $\tau$  requerido para el sensor capacitivo. Es por ello que para realizar las medidas del circuito RC necesitaremos tomar muestras al menos a decenas de  $\mu$ s.

Como el funcionamiento por defecto de la función `AnalogRead()` no satisface las necesidades del proyecto, es necesario modificar la configuración del ADC en el  $\mu$ C SAMD21, en especial el preescalador que emplea para controlar la frecuencia de muestreo de los pines de entrada analógica.



#### 4.4. CONFIGURACIÓN DEL ADC

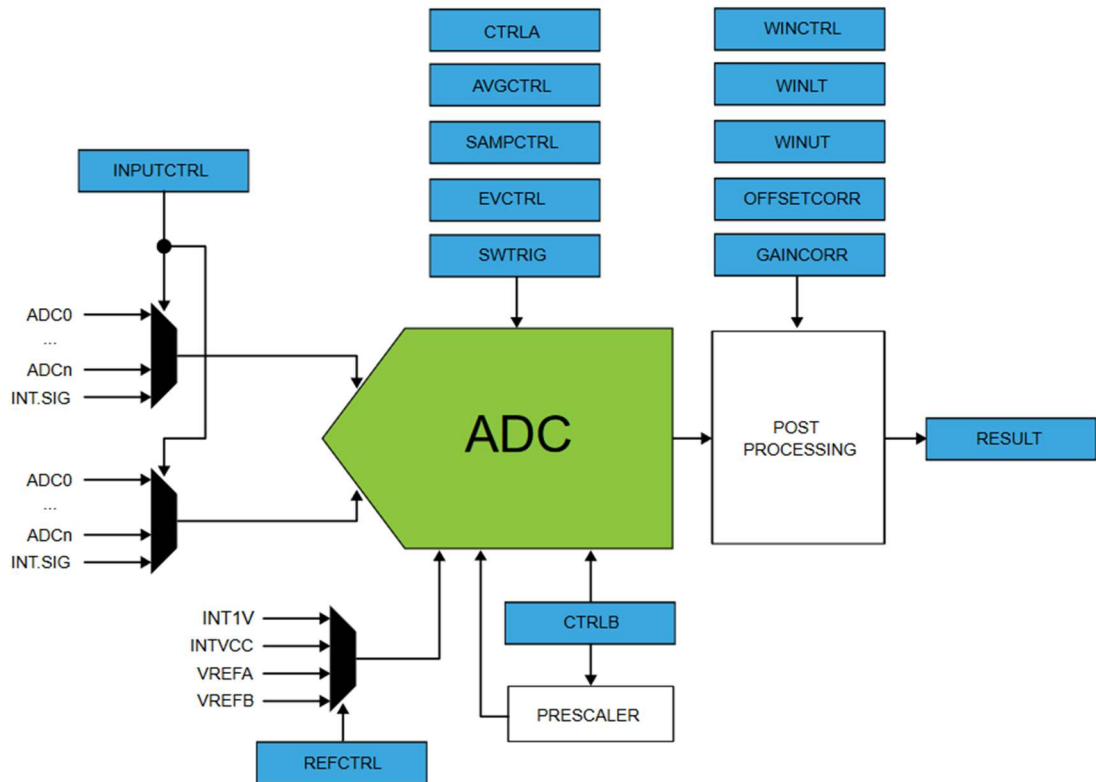


Figura 20 Esquema de los registros del ADC. [5]

El convertidor del SAMD21 se configura mediante la modificación de los registros, los cuales se encuentran explicados en la sección **33. ADC – Analog-to-Digital Converter** del datasheet del  $\mu\text{C}$  SAMD21 [5].

Los registros principales del ADC son:

**REG\_ADC\_REFCTRL:** como se aprecia en la Figura 20, se utiliza para establecer la tensión de referencia del ADC. Puede ser una señal externa, 1V o  $V_{cc}$  (3.3V), siendo esta última la opción por defecto.

**REG\_ADC\_SAMPCTRL:** sirve para aumentar el muestreo basándose en el uso de medio ciclo de reloj. Por defecto está desactivado.

**REG\_ADC\_AVGCTRL:** Se utiliza para realizar una medida promediada, realizando varias muestras a la vez. El número de muestras por defecto es 1.

**REG\_ADC\_INPUTCTRL:** con este registro se puede modificar la ganancia de la medida y seleccionar el pin de muestreo. La ganancia por defecto es de  $\times 1$ , y el pin del convertidor también se puede seleccionar mediante funciones del propio Arduino.

Los registros mencionados no requieren de cambios para nuestro sistema, debido a que se requiere su configuración por defecto, o se pueden modificar de otra manera. Los registros que son imprescindibles modificar para el funcionamiento del sistema son:

ADC->CTRLB.PRESCALER: Se encarga de modificar el preescalador del convertidor, valor que divide la frecuencia de reloj del convertidor. Esta por defecto en 512, pero se puede modificar a 256, 128, 64, 32, 16, 8 y 4.

ADC->CTRLB.->RESSEL: Con este registro se puede modificar la resolución del resultado entre 8, 10 y 12 bits. Cuanto mayor resolución mayor cantidad de información almacenamos, aumentando la precisión pero a costa de la memoria.

ADC->SAMPCTRL.SAMPLEN: Configura el tiempo máximo para tomar una muestra, si es demasiado bajo es posible que no se lea nada. Esto se debe a que el convertidor obtiene el valor de tensión medido mediante la carga de un condensador, y esta carga requiere de cierto tiempo.

Si bien estos son los registros principales, el ADC del SAMD21 tiene una gran diversidad de registros como se aprecia en la Figura 21 para programar diferentes modos de funcionamiento.

**33.7 Register Summary**

Offset	Name	Bit Pos.								
0x00	CTRLA	7:0						RUNSTDBY	ENABLE	SWRST
0x01	REFCTRL	7:0	REFCOMP					REFSEL[3:0]		
0x02	AVGCTRL	7:0		ADJRES[2:0]				SAMPLENUM[3:0]		
0x03	SAMPCTRL	7:0						SAMPLEN[5:0]		
0x04	CTRLB	7:0		RESSEL[1:0]	CORREN	FREERUN	LEFTADJ	DIFFMODE		
		15:8						PRESCALER[2:0]		
0x08	WINCTRL	7:0						WINMODE[2:0]		
0x0C	SWTRIG	7:0						START	FLUSH	
0x10	INPUTCTRL	7:0						MUXPOS[4:0]		
		15:8						MUXNEG[4:0]		
		23:16		INPUTOFFSET[3:0]				INPUTSCAN[3:0]		
		31:24					GAIN[3:0]			
0x14	EVCTRL	7:0		WINMONEO	RESRDYEO		SYNCEI	STARTEI		
0x16	INTENCLR	7:0				SYNCRDY	WINMON	OVERRUN	RESRDY	
0x17	INTENSET	7:0				SYNCRDY	WINMON	OVERRUN	RESRDY	
0x18	INTFLAG	7:0				SYNCRDY	WINMON	OVERRUN	RESRDY	
0x19	STATUS	7:0	SYNCBUSY							
0x1A	RESULT	7:0	RESULT[7:0]							
		15:8	RESULT[15:8]							
0x1C	WINLT	7:0	WINLT[7:0]							
		15:8	WINLT[15:8]							
0x20	WINUT	7:0	WINUT[7:0]							
		15:8	WINUT[15:8]							
0x24	GAINCORR	7:0	GAINCORR[7:0]							
		15:8							GAINCORR[11:8]	
0x26	OFFSETCORR	7:0	OFFSETCORR[7:0]							
		15:8							OFFSETCORR[11:8]	
0x28	CALIB	7:0	LINEARITY_CAL[7:0]							
		15:8							BIAS_CAL[2:0]	
0x2A	DBGCTRL	7:0							DBGRUN	

Figura 21 Tabla de registros del ADC [5]

Con ayuda de una calculadora online [7] basada en las fórmulas de la Figura 22 se puede obtener el tiempo de muestreo para diferentes configuraciones del convertidor y la máxima impedancia de entrada como se muestra en la Figura 23.

- Single-shot mode:

$$\text{PropagationDelay} = \frac{1 + \frac{\text{Resolution}}{2} + \text{DelayGain}}{f_{\text{CLK+ - ADC}}}$$

- Free-running mode:

$$\text{PropagationDelay} = \frac{\frac{\text{Resolution}}{2} + \text{DelayGain}}{f_{\text{CLK+ - ADC}}}$$

Figura 22 fórmulas para el tiempo de conversión del ADC [5] [7]

Parameter	Value	Unit
GCLK frequency (including GCLK division)	48	MHz
ADC clock prescaler	DIV8	
Resolution	8	bits
Gain	1X	
Sample time length	63	
Number of samples	1	
Offset correction	No	
ADC clock	6000	kHz
Sample time	5,33	microseconds
Propagation delay	1	microseconds
Conversion time	6,25	microseconds
Max input impedance	165.437	kΩ

Figura 23 Ejemplo de cálculo online del tiempo de conversión [7]

Utilizando este recurso se calculará el tiempo de conversión de nuestro sistema tras obtener las configuraciones adecuadas:

- preescalador = 8,
- resolución = 12 bits
- Tiempo de muestreo 0x0f=15

Obteniendo un tiempo de conversión de 2,58  $\mu\text{s}$  y una máxima impedancia de 38.734kohm. Esta configuración se ha sacado de forma experimental, ya que con menores valores del preescalador no funcionaba, al no tener tiempo para procesar los datos.

## 4.5. CÁLCULO Y SIMULACIÓN DE LA IMPEDANCIA DE ENTRADA DEL ADC

Debido a que el sistema de este proyecto trabaja a escalas de tiempo y capacidad muy bajas, es bastante sensible. Para poder caracterizar mejor el sistema se debe tener en cuenta la impedancia interna del ADC ( $Z_i$ ), ya que puede afectar a la curva RC introduciendo retrasos y, por lo tanto, errores.

Como no queda claro consultando el datasheet los valores de la  $Z_i$  del ADC, se ha calculado de forma experimental. Inicialmente se ha calculado la componente  $R_i$  de la  $Z_i$ , aplicando una señal continua como muestra el circuito de la Figura 24.

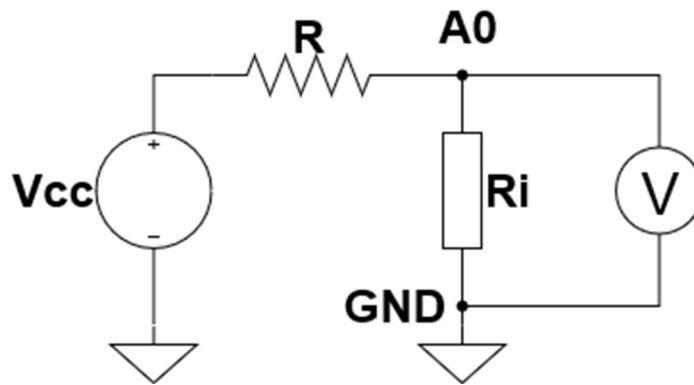


Figura 24 Circuito para el cálculo de la impedancia interna del ADC.

Para obtener el valor de la resistencia interna aplicaremos 2V a diferentes valores de R. Cuanto más similar sea R a  $R_i$ , mayor será la caída de tensión entre los pines medidos, como se aprecia en la siguiente fórmula obtenida por interpretar el circuito como un divisor de tensión:

$$R_i = \frac{V_m}{V_{cc} - V_m} * R;$$

Tabla 3 Cálculo de la  $R_i$  del ADC.

R ( $\Omega$ )	98.5	1002	9900	99000	971000	12000000
$V_i$ (V)	2.03	2.03	2.03	2.03	2.03	2.03
$V_m$ (V)	2.03	2.03	2.03	2.01	1.85	0.95
$R_i$ ( $\Omega$ )	-	-	-	9.95E+06	9.98E+06	1.06E+07

Se puede apreciar en los últimos 3 casos, en los cuales la caída de tensión es apreciable, que la  $R_i$  es de unos  $10\text{ M}\Omega$ .

Se debe tener en cuenta que además de la impedancia del convertidor, el multímetro utilizado para realizar las medidas también tiene impedancia y afecta a la señal. Para poder distinguir el origen de esta caída de tensión se ha repetido el experimento utilizando solo el multímetro.

Para calcular la resistencia interna del multímetro ( $R_x$ ) se ha montado el siguiente circuito:

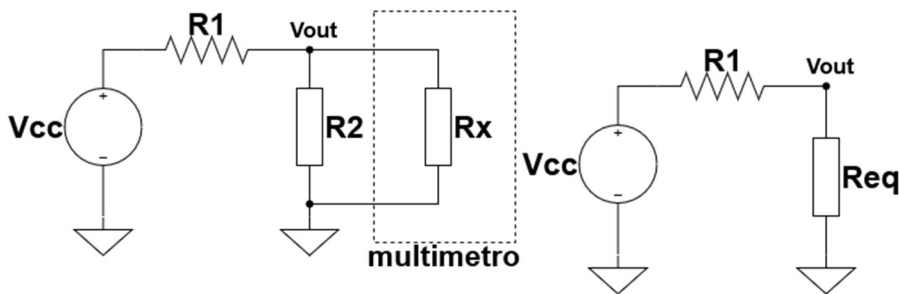


Figura 25 Circuito para el cálculo de la  $R_x$  del multímetro

Aplicando las fórmulas de divisor de tensión y de resistencias en paralelo, podemos definir el circuito de la siguiente manera

$$Req = \frac{Vout}{Vcc - Vout} * R1; \quad Rx = \frac{Req}{R2 - Req} * R2$$

Se han obtenido los siguientes resultados:

Tabla 4 Calculo de la  $R_x$  del multímetro

R1 ( $\Omega$ )	1.00E+06	1.00E+06	1.00E+06	1.00E+06	1.00E+06	1.00E+06
R2 ( $\Omega$ )	100.1	1015	10010	99900	997000	10770000
Vcc (V)	5	5	5	5	5	5
Vout (V)	0	0.0048	0.0495	0.441	2.38	4.18
Req ( $\Omega$ )	-	9.61E+02	1.00E+04	9.67E+04	9.08E+05	5.10E+06
Rx ( $\Omega$ )	-	1.80E+04	9.09E+06	3.05E+06	1.02E+07	9.68E+06

Este circuito es muy sensible a pequeñas variaciones de la tensión, pero con valores elevados de  $R_2$  se observa que la resistencia interna del multímetro configurado para medir tensiones de entre 2V y 20V tiende a los  $10\text{ M}\Omega$ .

Como  $R_x$  y  $R_i$  tienen prácticamente el mismo valor, se puede asegurar que la  $R_i$  medida en la primera prueba era consecuencia de la impedancia del multímetro. Por tanto, se puede considerar que la  $R_i$  del ADC es varios ordenes de magnitud superior y sus efectos nulos.

Para calcular la capacidad interna del convertidor, se ha sustituido la entrada de continua por una señal sinusoidal. Cuando la señal de entrada disminuye su amplitud al 70%, se considera que la frecuencia de la señal se corresponde con la de corte. Sabiendo que la Frecuencia de corte ( $F_c$ ) es de unos 10MHz y que  $R_i \gg 10M\Omega$ :

$$C_i = \frac{1}{2\pi * R_i * F_c} \approx 0pF$$

A partir de estos resultados, se puede asegurar que la impedancia del ADC es despreciable, y sus efectos nulos. Pero mediante estas pruebas se ha comprobado que los aparatos de medida, el multímetro y el osciloscopio, si tienen efecto en el proceso. Aplicando la  $R_i$  calculada del multímetro en el circuito se puede observar que al aumentar la resistencia del circuito, se modifica la curva RC final.

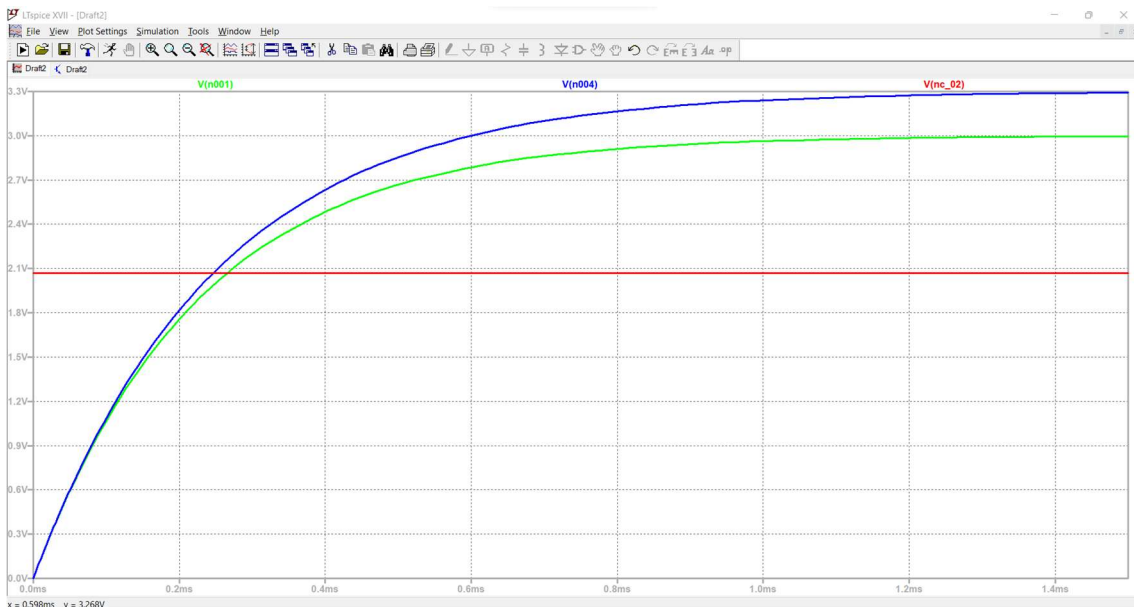


Figura 26 Simulación de los efectos de la  $R_x$  del multímetro en la curva RC

La señal azul es en vacío, la verde con el multímetro y la roja se corresponde con la tensión de referencia.

Por tanto, es importante no utilizar instrumentos externos de medida durante el funcionamiento del circuito.

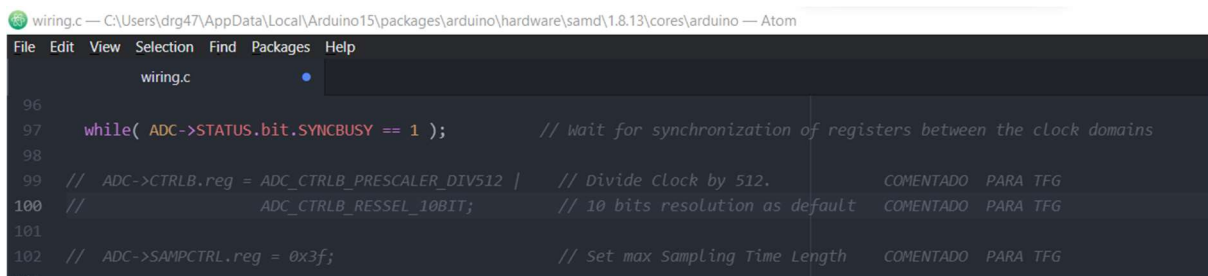
## 5. PROGRAMACIÓN

Si bien desde Arduino se pueden modificar los registros de la configuración del ADC, estos cambios no surgen efecto si no se modifica un archivo del programa. Este archivo almacena la configuración por defecto del convertidor y la protege frente a posibles cambios no deseados.

La carpeta del archivo se puede encontrar en la siguiente dirección, si se sustituye “drg47” por el nombre del usuario del equipo.

`C:\Users\drg47\AppData\Local\Arduino15\packages\arduino\hardware\samd\1.8.13\cores\arduino`

En la carpeta encontramos el archivo ‘wiring.c’, con la configuración por defecto de los registros. Las líneas a modificar se encuentran alrededor de la 100.



```
wiring.c — C:\Users\drg47\AppData\Local\Arduino15\packages\arduino\hardware\samd\1.8.13\cores\arduino — Atom
File Edit View Selection Find Packages Help
wiring.c
96
97 while( ADC->STATUS.bit.SYNCBUSY == 1 ); // wait for synchronization of registers between the clock domains
98
99 // ADC->CTRLB.reg = ADC_CTRLB_PRESCALER_DIV512 | // Divide Clock by 512. COMENTADO PARA TFG
100 // ADC_CTRLB_RESSEL_10BIT; // 10 bits resolution as default COMENTADO PARA TFG
101
102 // ADC->SAMPCTRL.reg = 0x3f; // Set max Sampling Time Length COMENTADO PARA TFG
```

Figura 27 Sección a modificar del archivo "wiring.c"

Para modificar los registros existen dos opciones:

1. Modificar en el archivo los registros por los valores requeridos. De esta manera se configura de manera directa, lo cual deja más limpio el programa pero dificulta la modificaciones y no queda presente este cambio en el programa.
2. Comentarlos en el archivo y modificar desde el programa. Es más cómodo a la hora de experimentar y configurar de forma dinámica.

Hay que tener presente que en ambos casos se está modificando o eliminando parte de la configuración por defecto del ADC del SAMD21, lo cual puede llegar a afectar al funcionamiento de futuros programas con este tipo de  $\mu$ C.

Para un correcto funcionamiento se ha optado por configurar:

- preescalador = 8,
- resolución = 12 bits
- Tiempo de muestreo  $0x0f=15$

Una vez modificado podemos proceder a la programación

## 5.1. PROGRAMA DE MEDIDA DE CAPACIDAD.

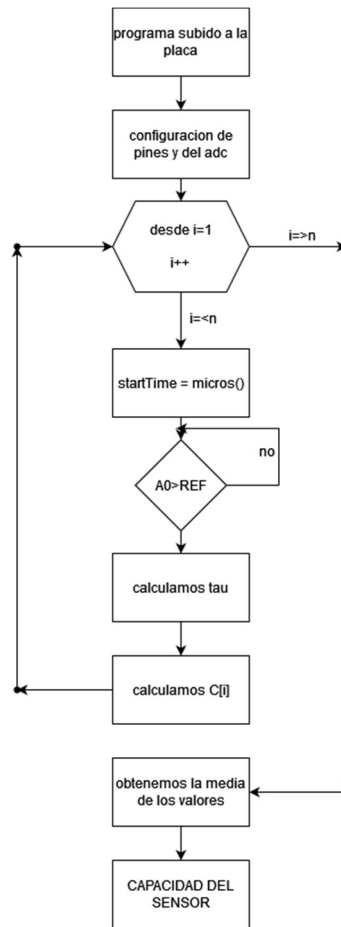


Figura 28 Diagrama del programa de medida de capacidad

El programa se ejecuta nada más subirlo a la placa. Tras realizar las configuraciones necesarias, las cuales en los siguientes subpartados, se inicia el proceso de medida.

El programa se basa en medir tau mediante la diferencia entre el tiempo de inicio de cargado del condensador y el tiempo de llegada a ref. Para determinar estas medidas temporales se hace uso de la función `micros()` de Arduino, la cual devuelve el número de microsegundos desde el inicio del programa. Usando este valor de tiempo y sabiendo la R se calcula la capacidad.

Para un mejor resultado se repite esta operación “n” veces (las que quiera el usuario) para obtener un promedio, siendo este la medida definitiva.

A continuación se va a explicar con capturas del programa de forma detallada cada sección del mismo.



### 5.1.1. INICIALIZACION DE VARIABLES:

En esta sección se definen el tipo correspondiente a las variables , su valor inicial si necesario y los pines a utilizar. Se ejecuta al iniciar el programa .

```
int res = 12; // RESOLUCION DEL ADC
int Vmax = pow(2, res) - 1; //
int ref = (0.63 * Vmax); // CALCULO DEL VALOR DE REFERENCIA CORRESPONDIENTE A LA RESOLUCION ELEGIDA

int analogPin = A1; // PIN ANALOGICO USADO PARA MEDIR
int chargePin = 6; // PIN DE CARGA DE LA RESISTENCIA

float resistorValue = 1.005; // Resistencia en Mohm

unsigned long startTime; // VARIABLE DE TIEMPO INICIAL DE MENDIDA
unsigned long elapsedTime; // TIEMPO ENTRE INICIO DE MUESTREO Y LLEGADA A VREF
float picoFarads; // VALOR DE CAPACIDAD

int n=500; // TAMAÑO DEL BUCLE DE PROMEDIADO
float media[500]; // VECTOR DE PROMEDIO
```

*Figura 29 Código de inicialización de variables*

Como se observa en la Figura 29, se configura aquí la resolución del convertidor, ya que el valor de la tensión de referencia se expresa como el 63% del valor máximo. Este valor máximo para 8 bits es 255, para 10 bits 1023 y para 12 bits es 4095. Con esta disposición se puede modificar fácilmente la resolución sin necesidad de modificar más líneas del programa.

Los pines a utilizar serán el pin analógico A1 como entrada al convertidor y el pin digital 6 para controlar la carga del condensador. Cuando se conecta a 3.3V el condensador se carga y cuando se conecta a GND se descarga.

Para el cálculo de la capacidad del sensor, se necesita conocer el valor de R y  $\tau$ . La resistencia se mide experimentalmente para introducir su valor en una variable. Para el cálculo de  $\tau$  se requiere almacenar el tiempo de inicio de muestreo (startTime) y el tiempo final,  $\tau$  será la diferencia entre ambos( almacenado en elapsedTime).

Por último, para un mejor resultado, se va a realizar un promediado de las medidas para evitar fluctuaciones por ruido. El tamaño del promedio se puede configurar antes de iniciar el programa, modificando el valor de la variable "n".

### 5.1.2. CONFIGURACIÓN (VOID SETUP)

En esta sección se indican las configuraciones necesarias para los pines, el puerto serie y el ADC. Se ejecuta una sola vez.

```
void setup() {
  // put your setup code here, to run once:
  pinMode(chargePin, OUTPUT); // PIN DIGITAL CONECTADO A R COMO SALIDA
  digitalWrite(chargePin, LOW); // CONECTAMOS LA R A GND

  Serial.begin(9600);

  ////CONFIGURACION DEL ADC////
  analogReadResolution(res); // CONFIGURAMOS LA RESOLUCION SELECCIONADA ANTERIORMENTE
  ADC->CTRLB.reg = ADC_CTRLB_PRESCALER_DIV8; // CONFIGURAMOS EL PREESCALADO COMO 8
  ADC->SAMPCTRL.reg = 0x0f; // TIEMPO DE MUESTREO DE 15
  //////////////////////////////////////
}
```

*Figura 30 Setup del programa*

Los pines por defecto se configuran como entradas, por tanto, no hace falta configurar A1. Para empezar con el condensador descargado, se inicia el pin 6 como salida baja, conectando el circuito a GND.

Para la visualización de los datos se va a utilizar el puerto serie. Para un correcto funcionamiento se especificarán los baudios del mismo.

Posteriormente configuraremos el ADC. Tal y como se indicó anteriormente, para que las líneas de código introducidas tengan efecto se deben comentar en el archivo “wiring.c”. Con la variable mencionada anteriormente se configurará la resolución de la medida obtenida por el pin analógico.

Para el resto de las configuraciones se requiere de utilizar una nomenclatura diferente, a nivel de registro. Se utilizará la misma que la comentada en “wiring.c” con los valores modificados para el funcionamiento requerido.

### 5.1.3. BUCLE (VOID LOOP)

En esta sección se ejecuta en bucle las operaciones necesarias. Es la parte principal y más extensa del programa.

```
void loop() {
  // put your main code here, to run repeatedly:
  for (int i = 1; i <= n; i++) {
    digitalWrite(chargePin, HIGH); // CONECTAMOS R A VC, INICIANDO LA CARGA DEL CONDENSADOR
    startTime = micros(); // MEDIMOS EL TIEMPO DE INICIO DE LA CARGA
    while (analogRead(analogPin) < ref) { } // SI LA TENSION DE C ES MENOR A REF, SE QUEDA EN ESPERA
    elapsedTime = micros() - startTime; // MEDIMOS EL TIEMPO ENTRE VCOND=0 Y VCOND=REF QUE ES TAU [us]
    picoFarads = ((float)elapsedTime / resistorValue) ; // CONOCIENDO R Y TAU CALCULAMOS C -> us/Mohm=picoF
    media[0]=0;
    media[i]=media[i-1]+picoFarads; // ALMACENAMOS LA SUMA DE LOS VALORES OBTENIDOS PARA SU POSTERIOR PROMEDIADO

    digitalWrite(chargePin, LOW); // RESISTENCIA A GND PARA DESCARGAR C

    delay(10); // DELAY PARA ASEGURAR QUE EL CONDENSADO SE A DESCARGADO

  }
  float valor=media[n]/n; // PROMEDIO DE LOS VALORES OBTENIDOS
  Serial.print(valor,2); // IMPRIMIMOS EL VALOR FINAL DE CAPACIDAD
  Serial.println("pF") ;

  //delay(295000); // DELAY PARA QUE EL PROGRAMA SE EJECUTE CADA 5 MINS
}
```

*Figura 31 Loop del programa*

Se inicia la carga del condensador configurando el pin 6 a 3,3V mediante la función digitalWrite(). Inmediatamente después se almacena el tiempo en microsegundos utilizando la función de Arduino micros().

Se espera sin realizar acciones a que la tensión leída en A1 sea superior a la referencia.

Cuando se supera ese valor, se almacena el tiempo y mediante la diferencia entre ambas medidas temporales se obtiene  $\tau$ .

Aplicando la relación

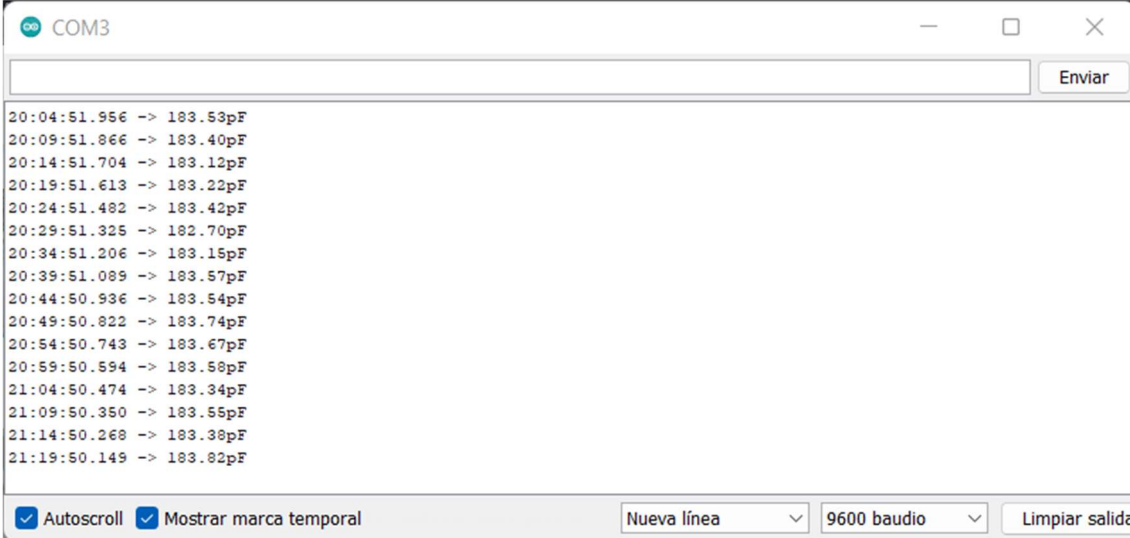
$$C = \frac{\tau}{R}$$

se obtiene la capacidad del condensador C.

Antes de poder repetir el proceso, se conecta el pin 6 a GND para descargar el condensador. Para garantizar la descarga se introduce un pequeño delay.

Para obtener un resultado menos afectado por el ruido o retrasos se realiza esta acción n veces, almacenando los valores en un array. El valor final utilizado es el promedio de los valores leídos. Este valor se imprime por el puerto serie acompañado de las unidades "pF".

Con esta programación se estará imprimiendo indefinidamente los valores leídos por el sensor por el puerto serie en cuanto se sube el programa a la placa. El tiempo entre medidas es dependiente del número de veces que se ejecuta el bucle for ( n). Para las medidas de calibración se ha añadido un delay de tal manera que el tiempo entre las medidas es aproximadamente 5 minutos.



The screenshot shows a serial terminal window titled 'COM3'. The window contains a list of sensor readings, each consisting of a timestamp, a raw value, and a converted value in pF. The readings are as follows:

Timestamp	Raw Value	Converted Value (pF)
20:04:51.956	->	183.53pF
20:09:51.866	->	183.40pF
20:14:51.704	->	183.12pF
20:19:51.613	->	183.22pF
20:24:51.482	->	183.42pF
20:29:51.325	->	182.70pF
20:34:51.206	->	183.15pF
20:39:51.089	->	183.57pF
20:44:50.936	->	183.54pF
20:49:50.822	->	183.74pF
20:54:50.743	->	183.67pF
20:59:50.594	->	183.58pF
21:04:50.474	->	183.34pF
21:09:50.350	->	183.55pF
21:14:50.268	->	183.38pF
21:19:50.149	->	183.82pF

The terminal window also features a control bar at the bottom with the following options:  Autoscroll,  Mostrar marca temporal, a dropdown menu set to 'Nueva línea', a dropdown menu set to '9600 baudio', and a 'Limpiar salida' button.

Figura 32 muestras del sensor cada 5 minutos

## 6. PROCESO DE CALIBRACIÓN

Tras haber comprobado el correcto funcionamiento del programa mediante la medida de condensadores de valor fijo de entre 65 y 250pF, se va a proceder a realizar la calibración para obtener el %HR a partir de la capacidad leída.

Pero antes de ello se va a realizar un simple análisis de los errores arrastrados por el propio funcionamiento del circuito y la programación, para tener en cuenta en las medidas reales.

### 6.1. ANÁLISIS DE ERRORES

Si configuramos la calculadora online comentada en el apartado del ADC con los parámetros escogido obtenemos que el tiempo de conversión es de 2,58  $\mu$ s. Con esta resolución se considera que  $\tau_m = \tau_{real} + \Delta\tau$ , siendo  $\Delta\tau = 2,58\mu$ s.

Si tenemos en cuenta este valor y asumimos que el valor de  $R = 1.005M\Omega$  es cte., aplicando la formula  $C = \tau/R$  y despejando que  $\Delta C = \Delta\tau/R$ , obtenemos que  $\Delta C = 2,57pF$ .

Como  $\%HR = (C - C_0)/S$  siendo  $S = 0.3pF/\%HR$  se despeja que  $\Delta\%HR = \Delta C/S = 8,56\%HR$ . Los valores de HR se van a expresar con una resolución del 1%, por tanto  $\Delta\%HR \approx 9\%$

Teniendo en cuenta que el sensor DHT11 usado de referencia tiene un error de un 5% y considerando despreciables los errores provocados por temperatura, el valor máximo de variación esperado es  $\Delta\%HR = 14\%$ .

### 6.2. MEDIDAS EXPERIMENTALES

Para poder calibrar el sistema, primero se realizarán diferentes medidas experimentales. Teniendo como referencia las medidas y aplicando un criterio sobre los posibles errores acumulados, se procederá a introducir en el programa las operaciones necesarias, obteniendo los valores de RH con cierto nivel de incertidumbre a determinar.

Para el calibrado del sensor se va a utilizar como referencia un sensor de temperatura y humedad DHT11, el cual se puede encontrar en diversos pack de introducción a Arduino.

4. Dimensions (unit: mm)

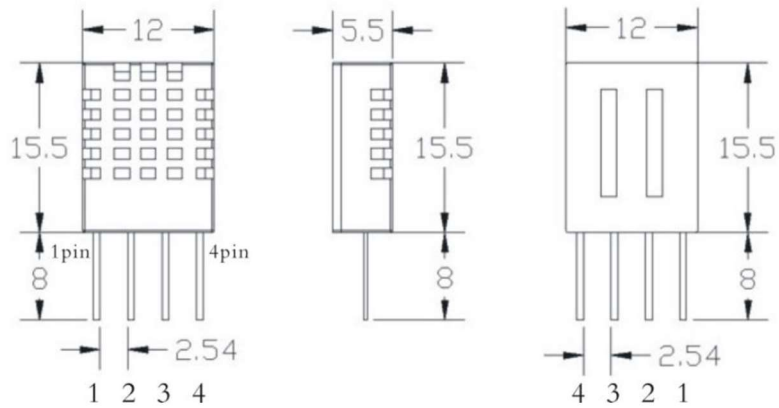


Figura 33 dimensiones del DHT11

El sensor está controlado mediante un Elegoo Uno (similar al Arduino 1) , utilizando el programa que viene con el sensor, además de una librería propia.

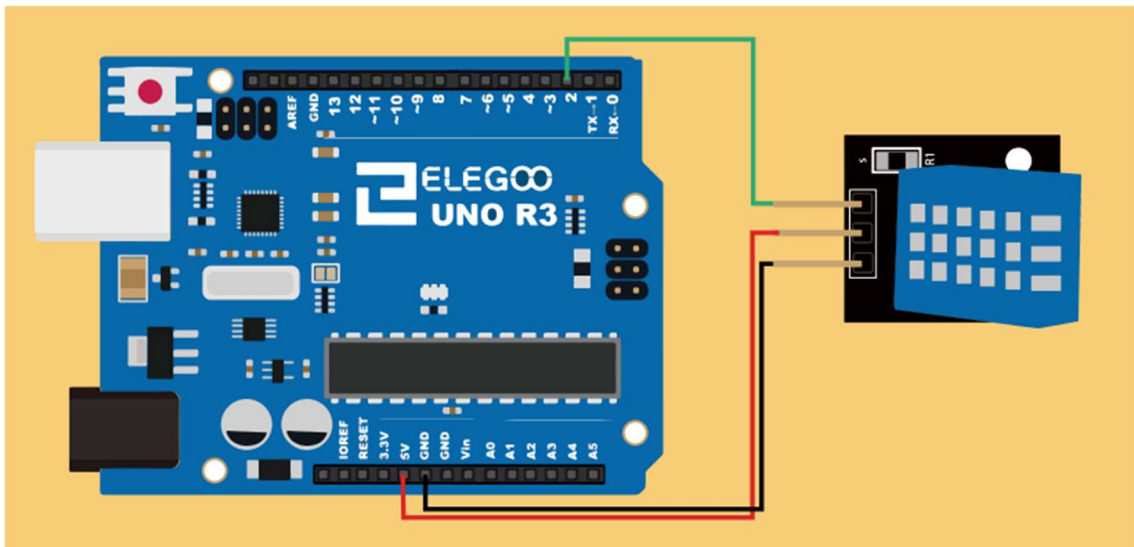


Figura 34 Diagrama de conexiones del DHT11

Si se revisa el datasheet del DHT11 se observa una precisión del 5%HR, como se ve en la Figura 35

- Relative humidity
- Resolution: 16Bit
- Repeatability:  $\pm 1\%$  RH
- Accuracy: At 25°C  $\pm 5\%$  RH
- Interchangeability: fully interchangeable
- Response time: 1 / e (63%) of 25°C 6s
- 1m / s air 6s
- Hysteresis:  $< \pm 0.3\%$  RH
- Long-term stability:  $< \pm 0.5\%$  RH / yr in

Figura 35 Características del DHT11

### 6.2.1. PRIMERA SECUENCIA DE MEDIDAS:

Inicialmente se ha optado por conseguir una configuración del montaje que facilite las medidas experimentales. Por tanto, para poder realizar pruebas en pequeños recipientes cerrados a los cuales se les modifica la humedad, se ha optado por montar los sensores de forma próxima en un soporte. Se han fijado en el extremo de un tubo de cartón y conectado a las placas mediante una serie de cables macho-hembra (ver Figura 36).

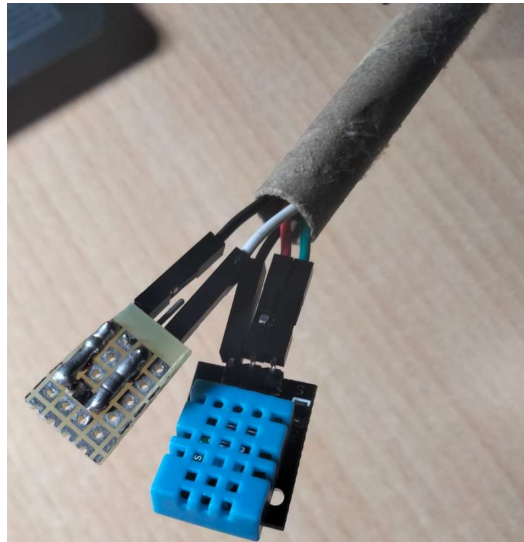


Figura 36 P14 y DHT11 colocados juntos.

Con esta configuración se han realizado una serie de medidas de %HR. Se ha instalado el equipo en habitaciones pequeñas de la casa y expuesto a variaciones de la humedad, mediante el uso de humidificadores, calentadores eléctricos de agua, etc.

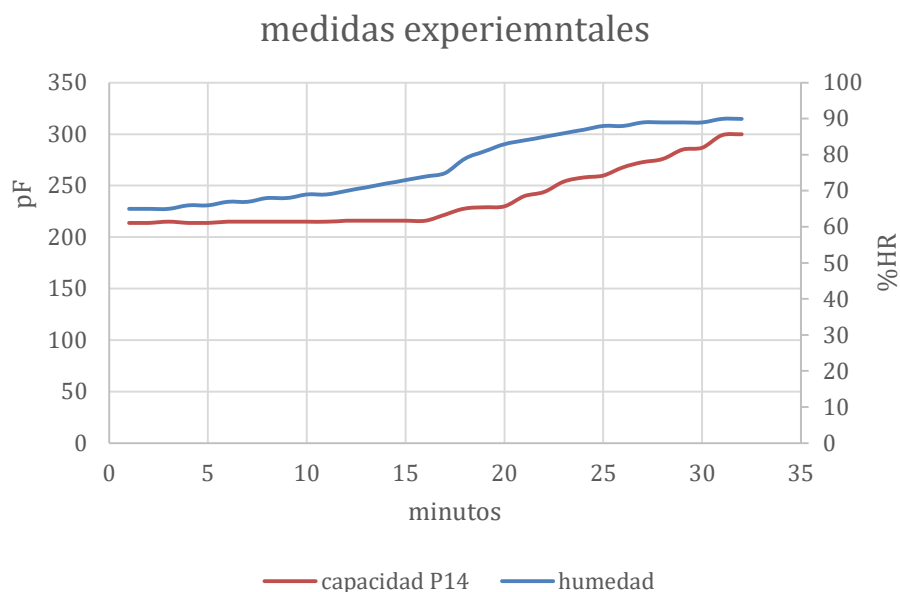


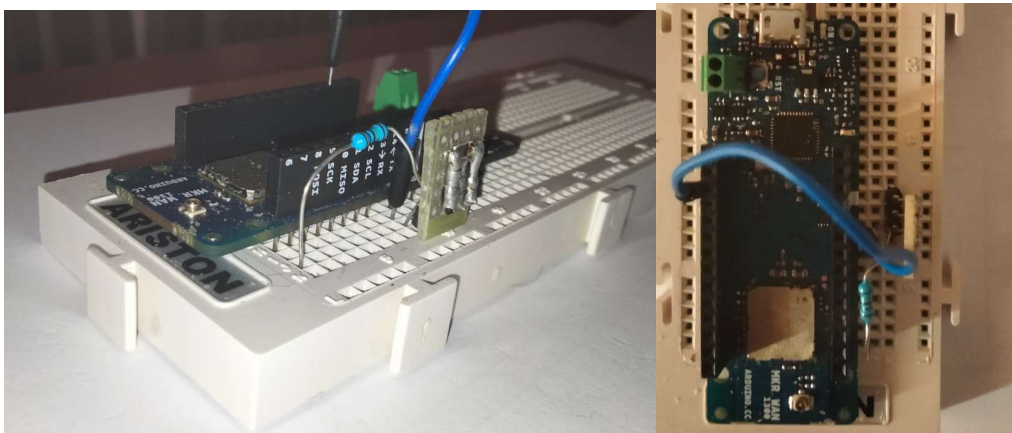
Figura 37 Medidas experimentales

El problema con los datos obtenidos es su incoherencia. Se obtienen valores de capacidad por encima de 300 pF, superiores a lo admisible considerando el máximo error de la Figura 8. También se observa que en muchos casos el valor de capacidad leída se mantiene constante y para valores muy diferentes del %HR medido con el DHT11.

Ante tales errores se ha decidido revisar el montaje del circuito. Lo primero que se ha observado con esta configuración, utilizando condensadores de valor fijo, es un offset de capacidad provocado por los cables. Cabe destacar que este error de capacidad parasita es diferente según la distancia entre los cables, siendo los cables que vienen pegados entre si los mayores generadores de error. Por tanto se va a reestructurar el montaje para reducir el número y longitud de los cables. De esta manera también se pretende evitar los posibles errores de capacidad parasita consecuencia del acoplo de la señal del DHT11.

### 6.2.2. SEGUNDA SECUENCIA DE MEDIDAS

Para evitar los errores ya mencionados se ha modificado la configuración física del sistema. Se han reducido el número de cables al mínimo y se ha cambiado el pin analógico de muestreo, ya que el A0 introducía unos 7-10pF de error.



*Figura 38 Configuración final del MKR y P14*

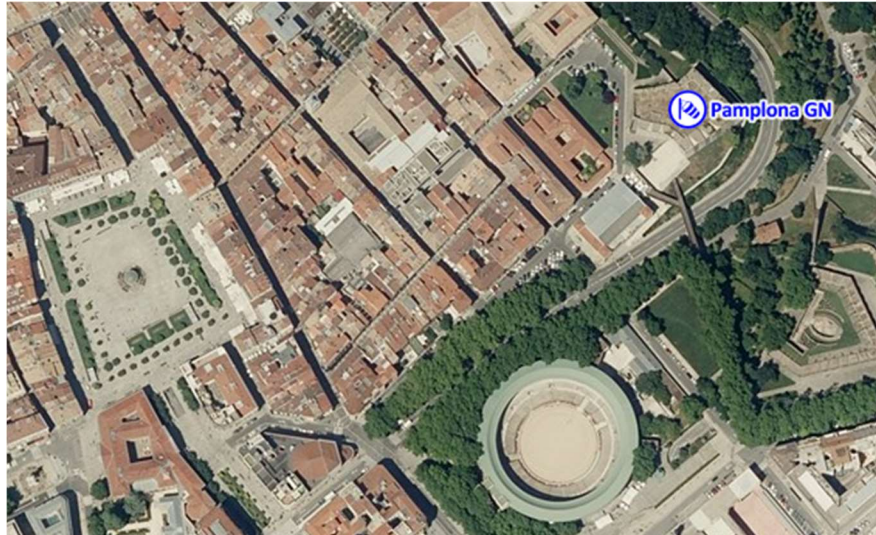
En esta secuencia de medidas diferenciaremos dos equipos que serán empleados para tener valores de referencia comparables a los del sistema diseñado:

- La estación meteorológica automática del Gobierno de Navarra (GN) [8].
- El conjunto formado por el DHT11, el P14 y el ordenador portátil que ejecuta el programa y registra los datos. Se denominará equipo de calibrado.



### 6.2.2.1. Estación meteorológica del GN

La estación meteorológica está ubicada en el baluarte del Labrit, como se puede observar en la ortofoto de la proporcionada por IDENA de la Figura 39.



*Figura 39 Ortofoto de la ubicación de la estación del GN [8]*

Esta estación está equipada con una gran diversidad de sensores los cuales registran diversos datos climatológicos de forma diaria o cada 10 minutos según se requiera. Los valores de humedad, viento, temperatura, radiación, presión, precipitación, se almacenan de forma regular en la página web de Meteo navarra, gestionada por la administración de la comunidad foral. Además, como se observa en la Figura 40, la estación está completamente al aire libre, sin edificios cerca que puedan afectar a las medidas por sus sombras o el bloqueo de corrientes de aire.



*Figura 40 Equipos de la estación meteorológica [9]*

#### 6.2.2.2. Equipo de calibrado

El equipo de calibrado está situado en el barrio pamplonés de la Rochapea, concretamente en un balcón de un primer piso cuya pared está orientada al noroeste, en la calle Joaquín Beunza. Como se aprecia en la Figura 41, enfrente a unos 20-25m se encuentra otro edificio de 7 pisos. La configuración de la calle tiene como consecuencia que durante las primeras horas de la mañana no incide el sol (hasta las 10 aproximadamente), y por la tarde la luz solar se ve reflejada por el edificio de enfrente. Además, estos edificios también bloquean parte de las corrientes de aire.



Figura 41 Ubicación del equipo de calibrado [8]

Debido a esta diferencia en el entorno se esperan variaciones entre los datos recogidos por ambos equipos, pero con tendencias similares, debido a su cercanía.

En el balcón, se ha utilizado un armario bajo como soporte y como espacio para situar el ordenador conectado a los sensores, como se ve en la Figura 42, además de añadir un plástico para evitar la entrada de agua o suciedad al ordenador.

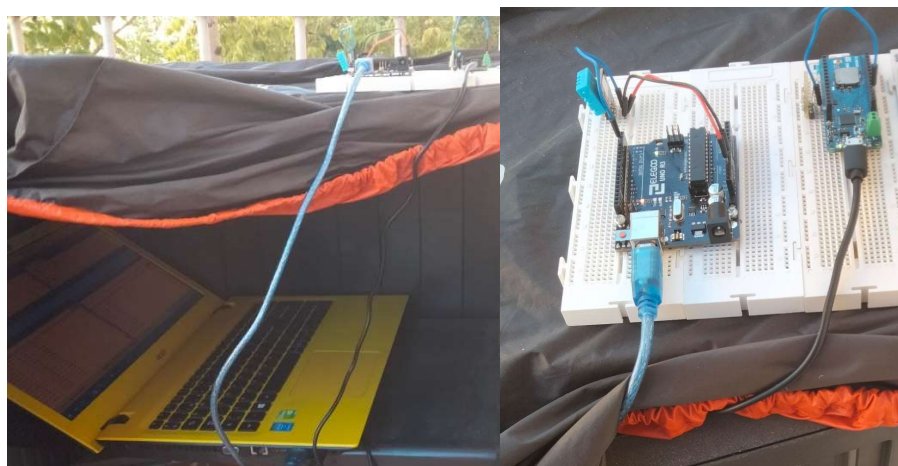


Figura 42 Montaje del equipo de calibrado

### 6.3. ANÁLISIS DE LOS RESULTADOS.

Tras realizar las medidas entre las noche del 26 de agosto de 2022 y la mañana del 29 de agosto de 2022 se han recopilado los siguientes datos del sensor DHT11 y la estación meteorológica:

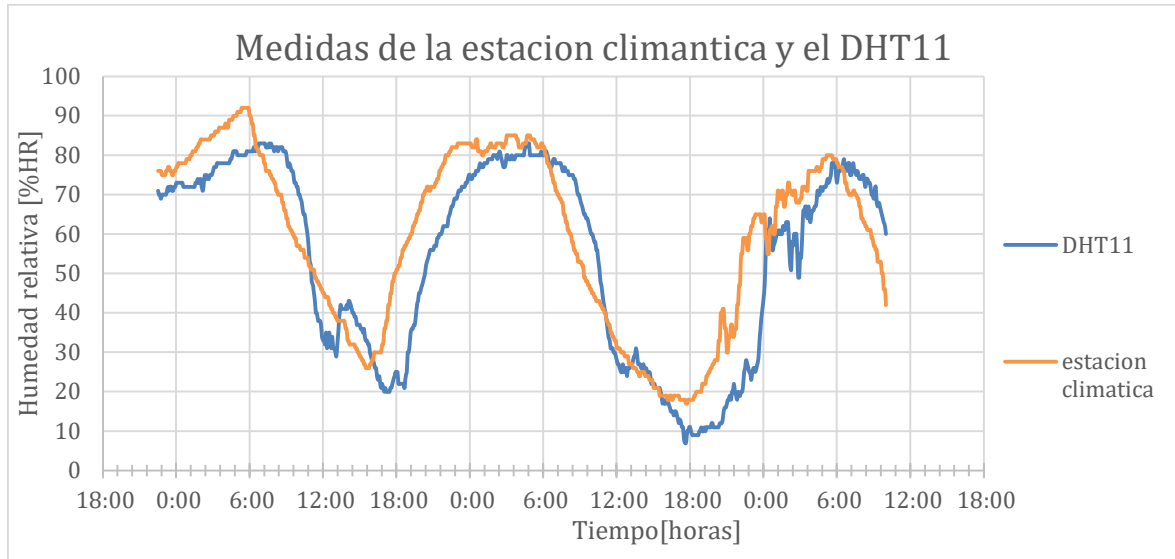


Figura 43 comparación de la medida de humedad entre la estación y el DHT11

En la Figura 43 se observa claramente que, durante la mañana, al salir el sol y aumentar la temperatura la humedad relativa disminuye, ya que aumenta la capacidad de agua admisible por el aire o porque parte de vapor sube en la atmósfera. El pico de mínima humedad se corresponde con la 6 de la tarde, tras varias horas de exposición solar. Conforme el sol baja, y la temperatura también, la HR aumenta considerablemente y se mantiene relativamente constante a lo largo de la noche. Cabe destacar que en la noche del 28 al 29 la humedad se dispara debido a la presencia de precipitaciones.

Tras analizar los datos se aprecia claramente que el comportamiento de la humedad es similar en ambas medidas, con cierto retraso en el DHT11. Ese retraso se debe a las diferencias de la ubicación, ya que se observa claramente que en el equipo de calibrado la humedad baja hacia las 9-10, lo cual se corresponde con la hora a partir de la cual el sol incide en la calle. En la estación, los efectos del sol se producen desde el amanecer, ya que está expuesta.

Por tanto, se considera que los datos obtenidos son fiables y se procederá a analizar los datos del P14. Mediante el programa descrito en el proyecto, se han tomado medidas de capacidad durante todo el proceso. Como se ha eliminado gran parte del ruido y las

cargas parasitas del circuito se supone que el comportamiento del sensor se corresponde con el de la curva de la Figura 5.

De esta manera, aplicando una sensibilidad de  $S=0.3\text{pF}/\%HR$  y sabiendo que  $C_{30}=180\text{pF}$ , se tiene:

$$C = \%HR * S + C_0 ; S = 0.3 \text{ pF}/\%HR; C_{30} = 30 * S + C_0 = 180 \text{ pF} \rightarrow C_0 = 171 \text{ pF}$$

Para calcular la HR hace falta invertir la formula de tal manera que se obtiene que:

$$HR (\%) = \frac{(C - C_0)}{S}$$

Con esta formula y los datos de capacidad se obtiene:

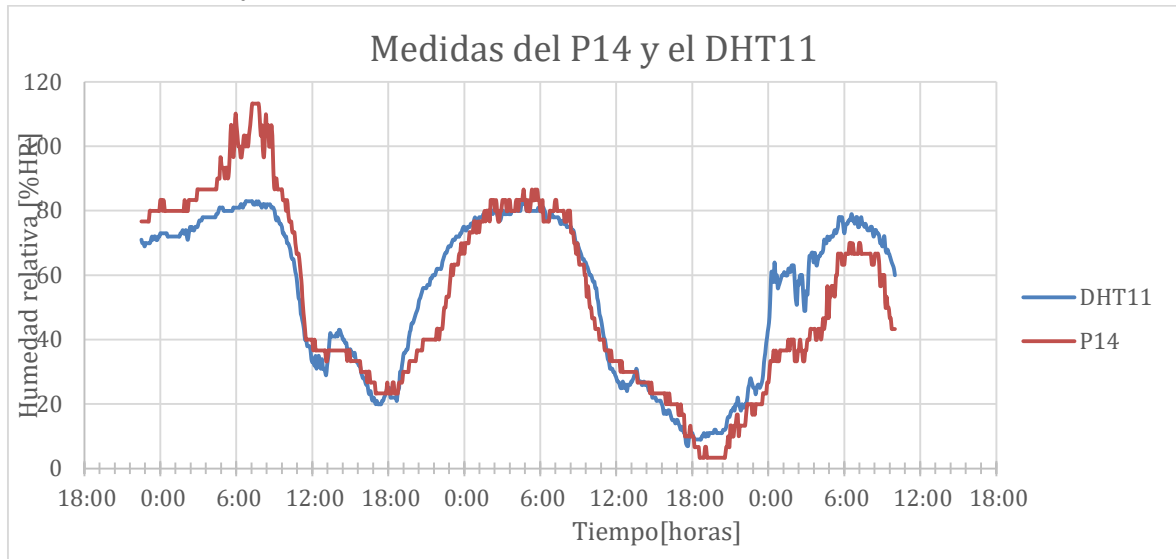


Figura 44 medidas de DHT11 y P14 para  $S=0.3$  y  $C_{30}=180$

Si bien es cierto que se pueden observar ciertos momentos de elevado error, como es el caso de los valores del día 27 entre las 6 y 12 de la mañana, no se observa ese comportamiento de nuevo, por tanto, se supone que esta causado por algún factor externo inusual o no controlado. Otro momento a destacar es la madrugada del 29, donde la diferencia entre las medidas podría ser debido al efecto de la lluvia, que puede haber influido de manera diferente a ambos sensores.

Pero a pesar de momentos puntuales con bastante error, el promedio de la diferencia entre ambas medidas es de menos del 3% HR. A la luz de estos resultados y con falta de tiempo para realizar más medidas, se ha decidido que la calibración se realizará utilizando una relación lineal aplicando los factores mencionados

anteriormente, de manera que el programa final realizara estas operaciones de forma autónoma. Esta calibración del programa se indica en la Figura 45.

```
float valor=media[n]/n;// PROMEDIO DE LOS VALORES OBTENIDOS
//Serial.print(valor,2);//IMPRIMIMOS EL VALOR FINAL DE CAPACIDAD
//Serial.println("pF") ;
float S=0.3;//Sensibilidad en pF/%HR
int Co=171;// valor en el cual la %HR es 0 si se aplica que la S=0.3 y que C30=180pF
float HR= (valor-Co)/S;
Serial.print(HR,0);//IMPRIMIMOS EL VALOR FINAL DE %HR
Serial.println("%") ;

//delay(295000);//DELAY PARA QUE EL PROGRAMA SE EJECUTE CADA 5 MINS
```

Figura 45 Código de calibración

#### 6.4. PROPUESTA DE DISEÑO DE UNA PCB PARA EL CIRCUITO.

Para eliminar definitivamente cualquier ruido de los cables se ha diseñado una tarjeta PCB para conectarse directamente a los pines del MKR tal y como se visualiza en las siguientes figuras:

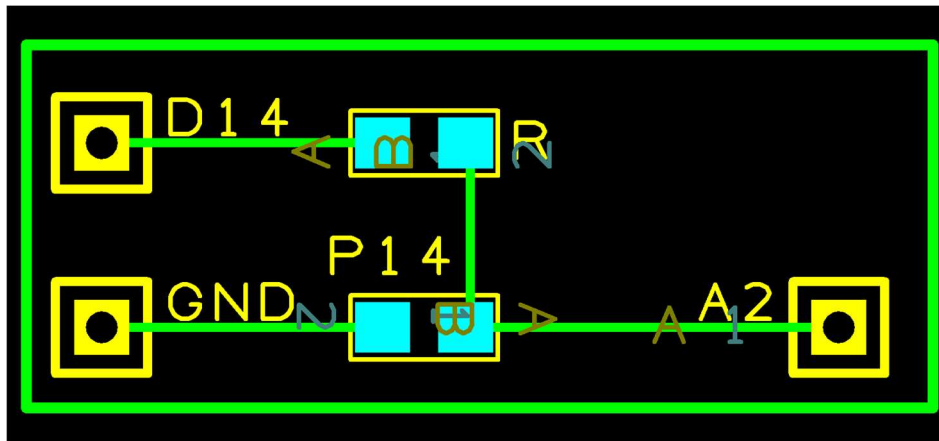


Figura 46 footprint del PCB para el circuito de medida de capacidad

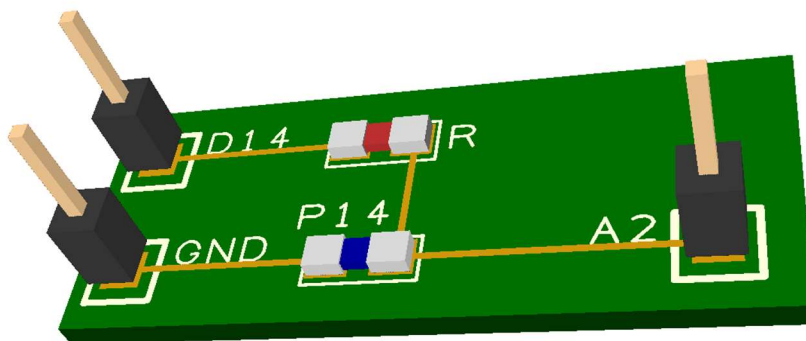


Figura 47 simulación 3D de la PCB

El incorporar una PCB justo encima del MKR, se crea el inconveniente de tapar algunos pines clave como  $V_{cc}$  y GND. Este problema se puede solucionar modificando la PCB para que se pueda acceder a estos pines a través de ella, o simplemente accediendo mediante las patillas inferiores que incorpora el propio Arduino.

Por desgracia, la falta de tiempo no ha permitido la fabricación de la placa, si bien se espera que pueda hacerse en los próximos días.

## 7. CONCLUSIONES

Gran parte de la dificultad del proyecto ha sido lograr entender y modificar correctamente los registros del convertidor, debido a la necesidad de una nomenclatura diferente a la usual empelada en el IDE de Arduino, y a tener que modificar un archivo de instalación del programa, algo poco habitual. Además, el formato del datasheet del SAMD21 puede llegar a ser un poco confuso, ya que su funcionamiento es mas complejo que otros  $\mu\text{C}$ . Pero sus prestaciones aportan una mayor facilidad y flexibilidad para poder utilizar diversos protocolos de comunicación serie. De tal manera se puede utilizar en aplicación de medidas y control de la humedad en tiempo real.

Otro gran problema ha sido el ruido, los errores, provocados generalmente por capacidades parasitas. Al trabajar en escalas de tiempo y capacidad mas bajas de lo convencional, se han tenido que tomar en cuenta factores en otros casos despreciables. Esto ha dificultado sobre todo la calibración, ya que algunos de estos efectos no eran tan notables en los condensadores constantes, por tanto se detectaron durante las medidas. Sin embargo, la detección de este tipo de influencias es mejor en la fase experimental que durante su incorporación práctica.

Tras haber encontrado la manera adecuada de tomar las medidas, el incorporar mas referencias y eliminar los ruidos, el proceso de calibrado a sido satisfactorio, al funcionar el sensor de forma similar a la especifica en el datasheet. Si bien hay circunstancias ambientales que pueden alterar los resultados y el propio sistema tiene una incertidumbre de un 14%, el funcionamiento es adecuado para entornos controlados, con un error habitualmente bajo.

El programa, aunque sencillo, es mejorable. Se podrían haber incorporado interrupciones, utilizado más configuraciones del ADC o funciones de bajo consumo en el  $\mu\text{C}$  para lograr un menor consumo de memoria y energía. Sin embargo, cumple su objetivo, es fácil de entender y modificar.

Teniendo en cuenta la sencillez del circuito utilizado y que el programa es simple y efectivo, se ha cumplido el objetivo principal: diseñar y desarrollar un sistema electrónico que sea capaz de medir la Humedad Relativa (%HR) del ambiente mediante un circuito DC.

## 8. BIBLIOGRAFÍA

- [1] «Higrómetros,» RS Components, [En línea]. Available: <https://es.rs-online.com/web/c/prueba-y-medida/medicion-ambiental/higrometros/?pn=1&rpp=100&sortBy=Calibrado&sortType=ASC>. [Último acceso: Agosto 2022].
- [2] «Sensor Finder,» IST Innovative Sensor Technology, [En línea]. Available: <https://www.ist-ag.com/en/sensor-finder?f%5B0%5D=category%3A10>. [Último acceso: Agosto 2022].
- [3] «P14 FemtoCap,» IST Innovative Sensor Technology, [En línea]. Available: [https://www.ist-ag.com/sites/default/files/downloads/p14\\_femtocap.pdf](https://www.ist-ag.com/sites/default/files/downloads/p14_femtocap.pdf). [Último acceso: Agosto 2022].
- [4] «MKR WAN 1300,» Arduino, [En línea]. Available: <https://docs.arduino.cc/hardware/mkr-wan-1300?queryID=7cc6740e0dc19a0ae53335850e44af1e>. [Último acceso: Agosto 2022].
- [5] «Datasheet SAMD21,» Arduino, [En línea]. Available: [https://content.arduino.cc/assets/mkr-microchip\\_samd21\\_family\\_full\\_datasheet-ds40001882d.pdf](https://content.arduino.cc/assets/mkr-microchip_samd21_family_full_datasheet-ds40001882d.pdf). [Último acceso: Agosto 2022].
- [6] Jimblom, «SAMD21 Mini/Dev Breakout Hookup Guide,» SparkFun Electronics, [En línea]. Available: <https://learn.sparkfun.com/tutorials/samd21-minidev-breakout-hookup-guide/samd21-overview>. [Último acceso: Agosto 2022].
- [7] S. Flowers, «Getting the most out of the SAM D21's ADC,» thea codes, 18 Julio 2020. [En línea]. Available: <https://blog.thea.codes/getting-the-most-out-of-the-samd21-adc/>. [Último acceso: Agosto 2022].
- [8] «Visor,» Infraestructura de Datos Espaciales de Navarra, [En línea]. Available: <https://idena.navarra.es/navegar/>. [Último acceso: Agosto 2022].



- [9] G. d. Navarra, «Pamplona GN Estacion Automantica,» Meteo Navarra, [En línea]. Available: <http://meteo.navarra.es/estaciones/estacion.cfm?IDestacion=455>. [Último acceso: Agosto 2022].
- [10] «Humedad Relativa,» Wikipedia, [En línea]. Available: [https://es.wikipedia.org/wiki/Humedad\\_relativa](https://es.wikipedia.org/wiki/Humedad_relativa). [Último acceso: Agosto 2022].
- [11] ForceTronics, «Speeding up the ADC on Arduino SAMD21 Boards (Zero, Mkr, etc) P1,» Youtube, [En línea]. Available: [https://www.youtube.com/watch?v=glulleL2lxA&ab\\_channel=ForceTronics](https://www.youtube.com/watch?v=glulleL2lxA&ab_channel=ForceTronics). [Último acceso: Agosto 2022].
- [12] Theran, «Lectura de capacidades muy bajas con Arduino,» Electronica Guru, [En línea]. Available: <https://electronica.guru/questions/9188/lectura-de-capacidades-muy-bajas-con-arduino>. [Último acceso: Agosto 2022].

## 9. ANEXOS

### CÓDIGO

```

int res = 12;// RESOLUCION DEL ADC
int Vmax = pow(2, res) - 1;//
int ref = (0.63 * Vmax);// CALCULO DEL VALOR DE REFERENCIA
CORRESPONDIENTE A LA RESOLUCION ELEGIDA

int analogPin = A1;// PIN ANALOGICO USADO PARA MEDIR
int chargePin = 6 ;// PIN DE CARGA DE LA RESISTENCIA

float resistorValue = 1.005 ; //Resistencia en Mohm

unsigned long startTime;// VARIABLE DE TIEMPO INICIAL DE MENDIDA
unsigned long elapsedTime;// TIEMPO ENTRE INICIO DE MUESTREO Y
LLEGADA A VREF --> TAU
float picoFarads;// VALOR DE CAPACIDAD

int n=500;// TAMAÑO DEL BUCLE DE PROMEDIADO
float media[500];// VECTOR DE PROMEDIO

void setup() {
  // put your setup code here, to run once:
  pinMode(chargePin, OUTPUT);// PIN DIGITAL CONECTADO A R COMO
SALIDA
  digitalWrite(chargePin, LOW);// CONECTAMOS LA R A GND

  Serial.begin(9600);

  ///CONFIGURACION DEL ADC///
  analogReadResolution(res);// CONFIGURAMOS LA RESOLUCION
SELECCIONADA ANTERIORMENTE
  ADC->CTRLB.reg = ADC_CTRLB_PRESCALER_DIV8;// CONFIGURAMOS EL
PREESCALADO COMO 8
  ADC->SAMPCTRL.reg = 0x0f;// TIEMPO DE MUESTREO DE 15
  //////////////////////////////////////
}

void loop() {
  // put your main code here, to run repeatedly:
  for (int i = 1; i <= n; i++) {
    digitalWrite(chargePin, HIGH);// CONECTAMOS R A VC, INICIANDO LA
CARGA DEL CONDENSADOR
    startTime = micros();// MEDIMOS EL TIEMPO DE INICIO DE LA CARGA
    while (analogRead(analogPin) < ref) { }// SI LA TENSION DE C ES MENOR A
REF, SE QUEDA EN ESPERA
    elapsedTime = micros() - startTime;// MEDIMOS EL TIEMPO ENTRE VCOND=0
Y VCOND=REF QUE ES TAU [us]
    picoFarads = ((float)elapsedTime / resistorValue) ;// CONOCIENDO R Y TAU
CALCULAMOS C -> us/Mohm=picoF
    media[0]=0;
    media[i]=media[i-1]+picoFarads;// ALMACENAMOS LA SUMA DE LOS
VALORES OBTENIDOS PARA SU POSTERIOR PROMEDIADO
  }
}

```

```

digitalWrite(chargePin, LOW);// RESISTENCIA A GND PARA DESCARGAR C
delay(10);// DELAY PARA ASEGURAR QUE EL CONDENSADO SE A
DESCARGADO

}
float valor=media[n]/n;// PROMEDIO DE LOS VALORES OBTENIDOS
//Serial.print(valor,2);//IMPRIMIMOS EL VALOR FINAL DE CAPACIDAD
//Serial.println("pF") ;
float S=0.3;//Sensibilidad en pF/%HR
int Co=171;// valor en el cual la %HR es 0 si se aplica que la S=0.3 y que
C30%=180pF
float HR= (valor-Co)/S;
Serial.print(HR,0);//IMPRIMIMOS EL VALOR FINAL DE %HR
Serial.println("%" );

//delay(295000);//DELAY PARA QUE EL PROGRAMA SE EJECUTE CADA 5
MINS

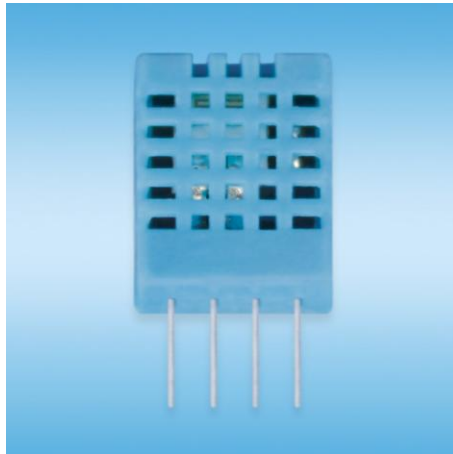
}

```

# AOSONG

## Temperature and humidity module

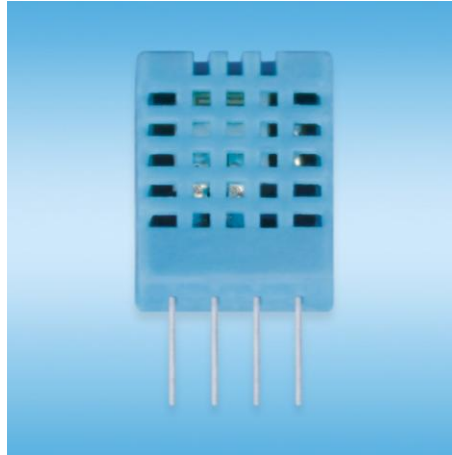
### DHT11 Product Manual



[www.aosong.com](http://www.aosong.com)

### 1、 Product Overview

DHT11 digital temperature and humidity sensor is a composite Sensor contains a calibrated digital signal output of the temperature and humidity. Application of a dedicated digital modules collection technology and the temperature and humidity sensing technology, to ensure that the product has high reliability and excellent long-term stability. The sensor includes a resistive sense of wet components and an NTC temperature measurement devices, and connected with a high-performance 8-bit microcontroller.



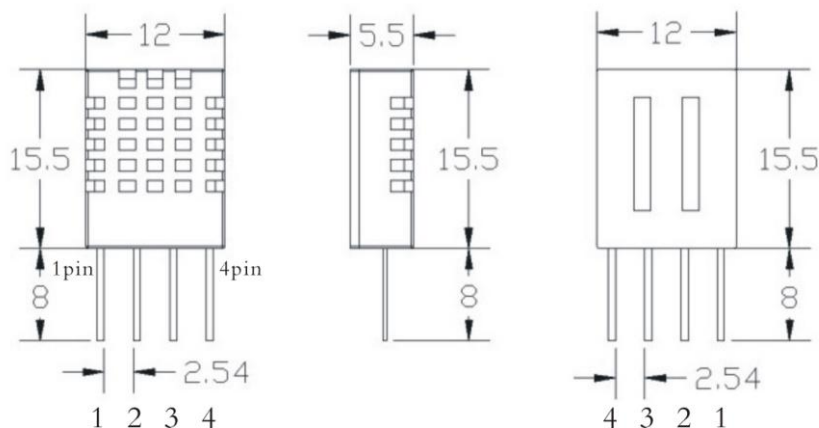
### 2、 Applications

HVAC, dehumidifier, testing and inspection equipment, consumer goods, automotive, automatic control, data loggers, weather stations, home appliances, humidity regulator, medical and other humidity measurement and control.

### 3、 Features

Low cost, long-term stability, relative humidity and temperature measurement, excellent quality, fast response, strong anti-interference ability, long distance signal transmission, digital signal output, and precise calibration.

### 4、 Dimensions (unit: mm)



## 5、Product parameters

Relative humidity

Resolution: 16Bit

Repeatability:  $\pm 1\%$  RH

Accuracy: At 25°C  $\pm 5\%$  RH

Interchangeability: fully interchangeable

Response time: 1 / e (63%) of 25°C 6s

1m / s air 6s

Hysteresis:  $< \pm 0.3\%$  RH

Long-term stability:  $< \pm 0.5\%$  RH / yr in

Temperature

Resolution: 16Bit

Repeatability:  $\pm 0.2^\circ\text{C}$

Range: At 25°C  $\pm 2^\circ\text{C}$

Response time: 1 / e (63%) 10S

Electrical Characteristics

Power supply: DC 3.5 ~ 5.5V

Supply Current: measurement 0.3mA standby 60 $\mu$  A

Sampling period: more than 2 seconds

Pin Description

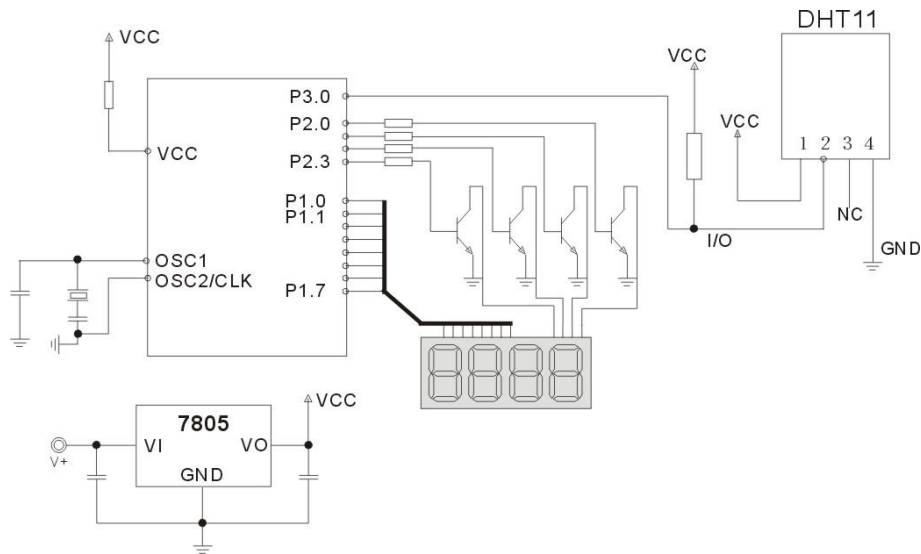
1, the VDD power supply 3.5 ~ 5.5V DC

2 DATA serial data, a single bus

3, NC, empty pin

4, GND ground, the negative power

### 6、 Typical circuit



Microprocessor and DHT11 of connection typical application circuit as shown above, DATA pull the microprocessor I / O ports are connected.

1. Typical application circuit recommended in the short cable length of 20 meters on the 5.1K pull-up resistor, the resistance of greater than 20 meters under the pull-up resistor on the lower of the actual situation.
2. When using a 3.5V voltage supply cable length shall not be greater than 20cm. Otherwise, the line voltage drop will cause the sensor power supply shortage, caused by measurement error.
3. Each read out the temperature and humidity values are the results of the last measurement For real-time data, sequential read twice, but is not recommended to repeatedly read the sensors, each read sensor interval is greater than 5 seconds can be obtained accurate data.

### 7、 Serial communication instructions (single-wire bi-directional)

#### ◎Single bus Description

DHT11 uses a simplified single-bus communication. Single bus that only one data line, the system of data exchange, control by a single bus to complete. Device (master or slave) through an open-drain or tri-state port connected to the data line to allow the device does not send data to release the bus, while other devices use the bus; single bus usually require an external one about 5.1kΩ pull-up resistor, so that when the bus is idle, its status is high. Because they are the master-slave structure, and only when the host calls the slave, the slave can answer, the host access devices must strictly follow the single-bus sequence, if the chaotic sequence, the device will not respond to the host.

#### ◎Single bus to transfer data defined

DATA For communication and synchronization between the microprocessor and DHT11, single-bus data format, a transmission of 40 data, the high first-out.

Data format:

The 8bit humidity integer data + 8bit the Humidity decimal data +8 bit temperature integer data + 8bit fractional temperature data +8 bit parity bit.

©Parity bit data definition

“8bit humidity integer data + 8bit humidity decimal data +8 bit temperature integer data + 8bit temperature fractional data” 8bit checksum is equal to the results of the last eight.

Example 1: 40 data is received:

<u>0011 0101</u>	<u>0000 0000</u>	<u>0001 1000</u>	<u>0000 0000</u>	<u>0100 1101</u>
High humidity 8	Low humidity 8	High temp. 8	Low temp. 8	Parity bit

Calculate:

$$0011\ 0101 + 0000\ 0000 + 0001\ 1000 + 0000\ 0000 = 0100\ 1101$$

Received data is correct;

Humidity: 0011 0101=35H=53%RH

Temperature: 0001 1000=18H=24°C

Example 2: 40 data is received:

<u>0011 0101</u>	<u>0000 0000</u>	<u>0001 1000</u>	<u>0000 0000</u>	<u>0100 1001</u>
High humidity 8	Low humidity 8	High temp. 8	Low temp. 8	Parity bit

Calculate:

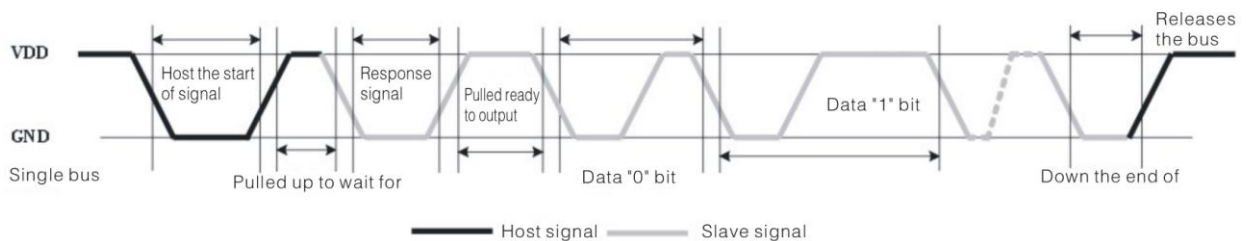
$$0011\ 0101 + 0000\ 0000 + 0001\ 1000 + 0000\ 0000 = 0100\ 1101$$

$$01001101 \neq 0100\ 1001$$

The received data is not correct, give up, to re-receive data.

©Data Timing Diagram

User host (MCU) to send a signal, DHT11 converted from low-power mode to high-speed mode, until the host began to signal the end of the DHT11 send a response signal to send 40bit data, and trigger a letter collection. The signal is sent as shown.



Data Timing Diagram

Note: The host reads the temperature and humidity data from DHT11 always the last measured value, such as twice the measured interval of time is very long, continuous read twice to the second value of real-time temperature and humidity values.



©Peripherals read steps

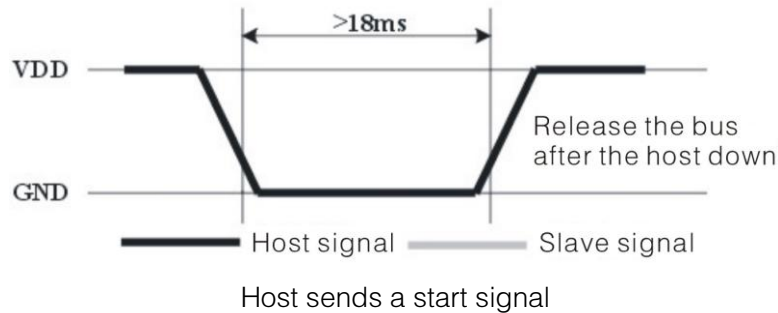
Communication between the master and slave can be done through the following steps (peripherals (such as microprocessors) read DHT11 the data of steps).

Step 1:

After power on DHT11 (DHT11 on after power to wait 1S across the unstable state during this period can not send any instruction), the test environment temperature and humidity data, and record the data, while DHT11 the DATA data lines pulled by pull-up resistor has been to maintain high; the DHT11 the DATA pin is in input state, the moment of detection of external signals.

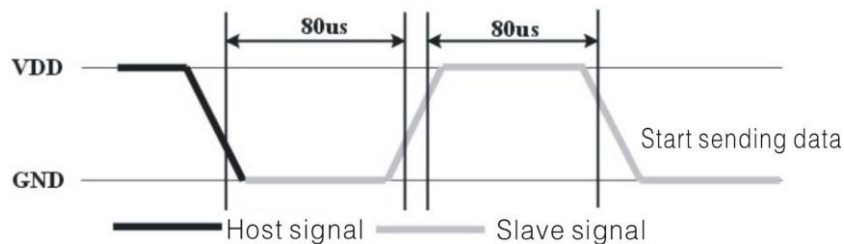
Step 2:

Microprocessor I / O set to output at the same time output low, and low hold time can not be less than 18ms, then the microprocessor I / O is set to input state, due to the pull-up resistor, a microprocessor/ O DHT11 the dATA data lines also will be high, waiting DHT11 to answer signal, send the signal as shown:



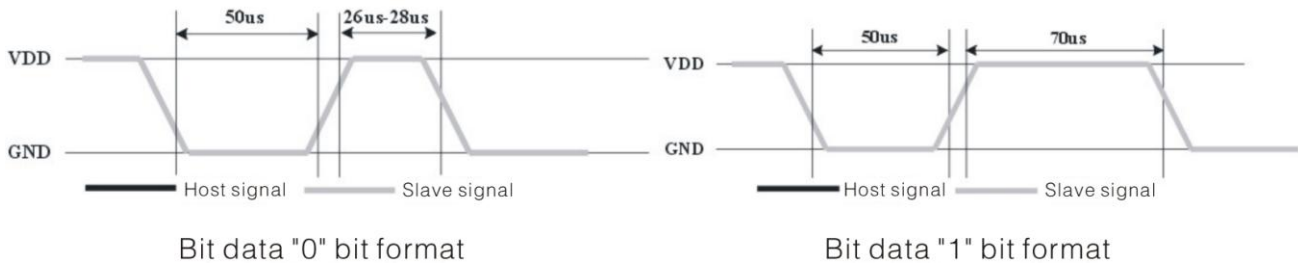
Step 3:

DATA pin is detected to an external signal of DHT11 low, waiting for external signal low end the delay DHT11 DATA pin in the output state, the output low of 80 microseconds as the response signal, followed by the output of 80 micro-seconds of high notification peripheral is ready to receive data, the microprocessor I / O at this time in the input state is detected the I / O low (DHT11 response signal), wait 80 microseconds high data receiving and sending signals as shown:



Step 4:

Output by DHT11 the DATA pin 40, the microprocessor receives 40 data bits of data "0" format: the low level of 50 microseconds and 26-28 microseconds according to the changes in the I / O levellevel, bit data "1" format: the high level of low plus, 50 microseconds to 70 microseconds. Bit data "0", "1" signal format as shown:



End signal:

Continue to output the low 50 microseconds after DHT11 the DATA pin output 40 data, and changed the input state, along with pull-up resistor goes high. But DHT11 internal re-test environmental temperature and humidity data, and record the data, waiting for the arrival of the external signal.

## 8、Application of information

### 1. Work and storage conditions

Outside the sensor the proposed scope of work may lead to temporary drift of the signal up to 300%RH. Return to normal working conditions, sensor calibration status will slowly toward recovery. To speed up the recovery process may refer to "resume processing". Prolonged use of non-normal operating conditions, will accelerate the aging of the product.

Avoid placing the components on the long-term condensation and dry environment, as well as the following environment.

A, salt spray

B, acidic or oxidizing gases such as sulfur dioxide, hydrochloric acid

Recommended storage environment

Temperature: 10 ~ 40 °C Humidity: 60% RH or less

### 2. The impact of exposure to chemicals

The capacitive humidity sensor has a layer by chemical vapor interference, the proliferation of chemicals in the sensing layer may lead to drift and decreased sensitivity of the measured values. In a pure environment, contaminants will slowly be released. Resume processing as described below will accelerate this process. The high concentration of chemical pollution (such as ethanol) will lead to the complete damage of the sensitive layer of the sensor.

### 3. The temperature influence

Relative humidity of the gas to a large extent dependent on temperature. Therefore, in the measurement of humidity, should be to ensure that the work of the humidity sensor at the same temperature. With the release of heat of electronic components share a printed circuit board, the installation should be as far as possible the sensor away from the electronic components and mounted below the heat source, while maintaining good ventilation of the enclosure. To reduce the thermal conductivity sensor and printed circuit board copper plating should be the smallest possible, and leaving a gap between the two.

### 4. Light impact

Prolonged exposure to sunlight or strong ultraviolet radiation, and degrade performance.

### 5. Resume processing

Placed under extreme working conditions or chemical vapor sensor, which allows it to return to the status of calibration by the following handler. Maintain two hours in the humidity conditions of 45°C and <10% RH (dry); followed by 20–30°C and > 70% RH humidity conditions to maintain more than five hours.

### 6. Wiring precautions

The quality of the signal wire will affect the quality of the voltage output, it is recommended to use high quality shielded cable.

### 7. Welding information

Manual welding, in the maximum temperature of 300°C under the conditions of contact time shall be less than 3 seconds.

### 8. Product upgrades

Details, please the consultation Aosong electronics department.

### 9、 The license agreement

Without the prior written permission of the copyright holder, shall not in any form or by any means, electronic or mechanical (including photocopying), copy any part of this manual, nor shall its contents be communicated to a third party. The contents are subject to change without notice.

The Company and third parties have ownership of the software, the user may use only signed a contract or software license.

### 10、 Warnings and personal injury

This product is not applied to the safety or emergency stop devices, as well as the failure of the product may result in injury to any other application, unless a particular purpose or use authorized. Installation, handling, use or maintenance of the product refer to product data sheets and application notes. Failure to comply with this recommendation may result in death and serious personal injury. The Company will bear all damages resulting personal injury or death, and waive any claims that the resulting subsidiary company managers and employees and agents, distributors, etc. that may arise, including: a variety of costs, compensation costs, attorneys' fees, and so on.

### 11、 Quality Assurance

The company and its direct purchaser of the product quality guarantee period of three months (from the date of delivery). Publishes the technical specifications of the product data sheet shall prevail. Within the warranty period, the product was confirmed that the quality is really defective, the company will provide free repair or replacement. The user must satisfy the following conditions:

- ① The product is found defective within 14 days written notice to the Company;
- ② The product shall be paid by mail back to the company;
- ③ The product should be within the warranty period.

The Company is only responsible for those used in the occasion of the technical condition of the product defective product. Without any guarantee, warranty or written statement of its products used in special applications. Company for its products applied to the reliability of the product or circuit does not make any commitment.

**Part 2**

# lesson

# 10

**DHT11 Temperature  
and Humidity Sensor**

## Overview

In this tutorial we will learn how to use a DHT11 Temperature and Humidity Sensor.

It's accurate enough for most projects that need to keep track of humidity and temperature readings.

Again we will be using a Library specifically designed for these sensors that will make our code short and easy to write.

### Component Required:

- (1) x Elegoo Uno R3
- (1) x DHT11 Temperature and Humidity module
- (4) x F-M wires (Female to Male DuPont wires)

## Component Introduction

### Temp and humidity sensor:

**DHT11** digital temperature and humidity sensor is a composite Sensor which contains a calibrated digital signal output of the temperature and humidity. The dedicated digital modules collection technology and the temperature and humidity sensing technology are applied to ensure that the product has high reliability and excellent long-term stability. The sensor includes a resistive moisture sensor and a NTC temperature measurement devices, and connects with a high-performance 8-bit microcontroller.

**Applications:** HVAC, dehumidifier, testing and inspection equipment, consumer goods, automotive, automatic control, data loggers, weather stations, home appliances, humidity regulator, medical and other humidity measurement and control.

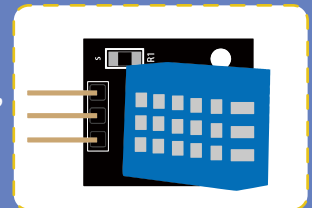
### Product parameters

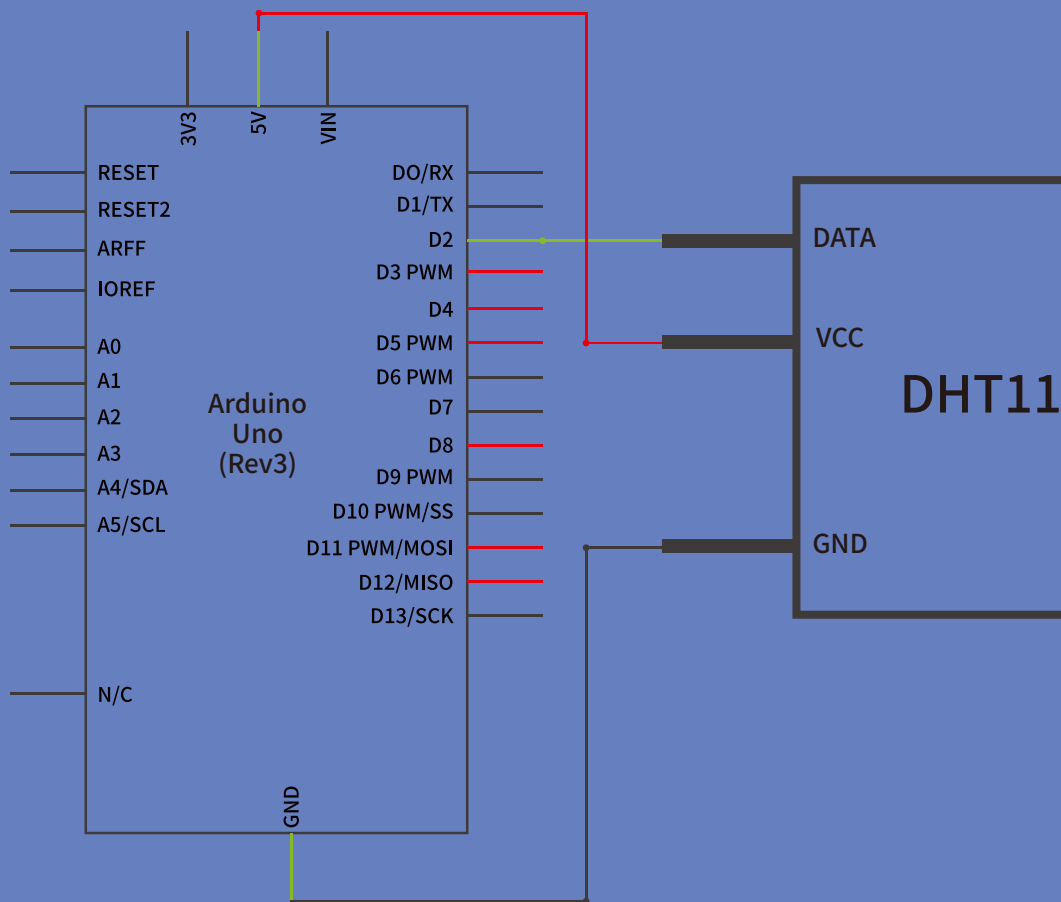
Relative humidity: 1m / s air 6s Hysteresis:  
Resolution: 8Bit <math>< \pm 0.3\% \text{ RH}</math>  
Repeatability:  $\pm 1\% \text{ RH}$  Long-term stability:  $< \pm 0.5\% \text{ RH} / \text{yr}$  in  
Accuracy: At  $25^\circ\text{C} \pm 5\% \text{ RH}$   
Interchangeability: fully interchangeable

Response time:  $1 / e$  (63%) of  $25^\circ\text{C}$  6s  
Temperature: Response time:  $1 / e$  (63%) 10S  
Resolution: 8Bit Electrical Characteristics  
Repeatability:  $\pm 0.2^\circ\text{C}$  Power supply: DC 3.5~5.5V  
Range: At  $0^\circ\text{C} \pm 50^\circ\text{C}$  Supply Current: measurement 0.3mA standby 60 $\mu\text{A}$   
Sampling period: more than 2 seconds

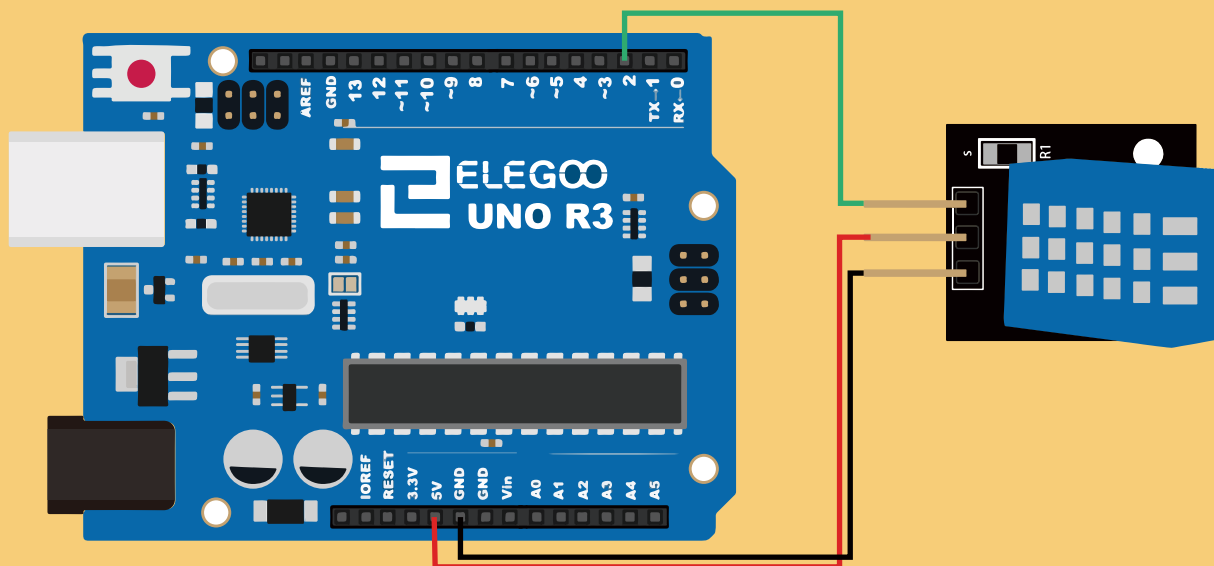
### Pin Description:

1. DATA serial data, a single bus.
2. the VDD power supply 3.5~5.5V DC.
3. GND ground, the negative power.





Connection Schematic



The connections are: Signal, Voltage and Ground which can be connected to any Pin on our UNO.

Wiring diagram

## Code

After wiring, please open the program in the code folder- **DHT11\_Example** and click UPLOAD to upload the program. See Lesson 5 in part 1 for details about program uploading if there are any errors.

Before you can run this, make sure that you have installed the **<DHT>** library or re-install it, if necessary. Otherwise, your code won't work.

### static

#### [Variable Scope & Qualifiers]

##### Description

The static keyword is used to create variables that are visible to only one function. However unlike local variables that get created and destroyed every time a function is called, static variables persist beyond the function call, preserving their data between function calls.

Variables declared as static will only be created and initialized the first time a function is called.

```
static const int DHT_SENSOR_PIN = 2;
```

### float

#### [Data Types]

##### Description

Datatype for floating-point numbers, a number that has a decimal point. Floating-point numbers are often used to approximate analog and continuous values because they have greater resolution than integers. Floating-point numbers can be as large as 3.4028235E+38 and as low as -3.4028235E+38. They are stored as 32 bits (4 bytes) of information.

```
float temperature;  
float humidity;
```

##### Syntax

`float var = val;`

##### Parameters

**var**: variable name.

**val**: the value you assign to that variable.

## Bool

#### [Data Types]

##### Description

Range:0~2<sup>32</sup>.

##### Syntax

`bool var = val;`

##### Parameters

**var**: variable name.

**val**: the value to assign to that variable.

### unsigned long

#### [Conversion]

##### Description

Converts a value to the unsigned long data type.

##### Parameters

**x**: a value of any type

##### Returns

unsigned long

```
static bool measure_environment( float *temperature, float *humidity )  
{  
    static unsigned long measurement_timestamp = millis( );  
  
    /* Measure once every four seconds. */  
    if( millis( ) - measurement_timestamp > 3000ul )  
    {  
        if( dht_sensor.measure( temperature, humidity ) == true )  
        {  
            measurement_timestamp = millis( );  
            return( true );  
        }  
    }  
  
    return( false );  
}
```

- Upload the program then open the monitor, we can see the data as below:  
(It shows the temperature of the environment, we can see it is 27 to 25 degrees with 45.0% humidity)
- Click the Serial Monitor button to turn on the serial monitor. The basics about the serial monitor are introduced in details in part 2 Lesson 4.

