



# **UNIVERSIDAD NACIONAL DEL LITORAL**

## **Facultad de Ingeniería y Ciencias Hídricas**

**Ingeniería en Inteligencia Artificial**

### **Introducción a los Sistemas Ciber-físicos 2024**

#### **Equipo docente**

- Prof. TITULAR: Carrique, Juan
- JTP 1: Borzone, Eugenio (EB)
- JTP 2: Zorzet, Bruno (BZ)
- JTP 3: Molas, José Tomas (TM)

#### **Alumnos**

Acosta, Milena - Basilio, Nicolas - Bonet, Francisco - Delgado, Santiago - Descalzo, Franco - Ismael, Bruno - Ramírez, Trinidad

Fecha de entrega: 20/11/24

## OJETIVOS

Desarrollar e implementar un sistema automatizado basado en Arduino para el monitoreo y control del proceso de compostaje en la FICH, que registre en tiempo real las condiciones de temperatura y aireación, garantizando el cumplimiento de las condiciones aeróbicas y termófilas necesarias para la descomposición óptima y la eliminación de patógenos.

Diseñar un prototipo que registre datos de temperatura y aireación de manera periódica y automática, activando alertas cada 86 horas para renovar el aire.

Garantizar un compost seguro y maduro que pueda utilizarse como abono para la producción agrícola local, promoviendo el aprovechamiento de los residuos orgánicos generados en la facultad y contribuyendo a prácticas de sostenibilidad.

Finalizar el desarrollo del sistema funcional (o representado mediante un gemelo digital) en el plazo del cursado de la materia, la cual cuenta con este proyecto como entrega de trabajo final.

Incluir indicadores visuales y sonoros que diferencien las fases del compostaje (mesófila, termófila, maduración) para que los usuarios puedan identificar fácilmente el estado actual del proceso, minimizando los requisitos técnicos para operar el sistema.

Asegurar que el sistema opere con un suministro eléctrico continuo independiente de la red local y cuente con un medio de almacenamiento para guardar los datos generados durante todo el proceso de compostaje.

## INTRODUCCION

El presente proyecto plantea una solución para abordar la creciente necesidad de gestionar los residuos orgánicos generados dentro de la facultad. Mediante la implementación de un sistema automatizado para el control y monitoreo del proceso de compostaje, buscamos optimizar la producción de compost y contribuir a la sostenibilidad ambiental de nuestra institución educativa.

La generación diaria de residuos orgánicos en las cantinas y áreas verdes de la facultad representa una oportunidad significativa para implementar prácticas sostenibles. Si bien el compostaje tradicional es efectivo, requiere supervisión constante y ajustes manuales, lo que puede resultar ineficiente y demandar recursos humanos considerables. Nuestro sistema automatizado

no solo optimizará este proceso, sino que también servirá como un ejemplo práctico y educativo para estudiantes de diversas carreras.

El sistema está diseñado para integrar sensores, actuadores y un microcontrolador Arduino que gestiona las distintas etapas del proceso biológico del compostaje. Su tarea principal es monitorear y controlar variables físicas críticas, como la temperatura, tomando muestras periódicas para asegurar el cumplimiento de las condiciones óptimas de cada fase. Por ejemplo, el sensor DS18B20 registra las temperaturas internas de la compostera, garantizando que superen los 50°C para eliminar bacterias anaeróbicas y patógenos. Una vez alcanzado este umbral en la etapa termófila, un LED rojo se activa como indicador del progreso del proceso.

Una vez la temperatura comienza a bajar, se debe esperar que dicho descenso de la temperatura quede por debajo del umbral de los 20°C estado en el cual se puede considerar que el compost llegó a la etapa de maduración. En este momento, un segundo actuador (LED verde) señala que el producto está listo para su retiro y uso como abono. Además, para mantener condiciones aeróbicas esenciales, el sistema monitorea los niveles de oxígeno y activa un tercer actuador (LED amarillo) si detecta que la materia orgánica no se ha aireado en un tiempo determinado.

El sistema también incluye un registro completo de los cambios y variables monitoreadas. Los datos son almacenados en una memoria SD portátil, lo que permite su análisis posterior. Este almacenamiento es gestionado mediante la sincronización de los sensores con un reloj interno (DS1307), y cada nuevo registro es notificado por un actuador buzzer (KY-012) que alerta sobre la actualización de la información.

En conjunto, este sistema representa una solución integral que optimiza el proceso de compostaje, al mismo tiempo que fomenta prácticas sostenibles y proporciona un recurso educativo valioso para nuestra comunidad universitaria.

## **DESARROLLO**

El sistema de control propuesto integra diversos sensores para monitorear parámetros críticos del proceso de compostaje, como la temperatura y el oxígeno en el compost. Utilizando un microcontrolador Arduino como unidad central de procesamiento, y tras una cuidadosa selección de sensores, hemos diseñado un sistema que no solo recopila datos en tiempo real, sino que también, mediante actuadores, emite avisos sobre las condiciones internas de la compostera. Esto facilita la supervisión del proceso por parte de los estudiantes o el personal de mantenimiento.

## COMPARACION DE SENSORES Y ACTUADORES

	Modelo	Ventajas	Rango	Comunicación	Precisión	
Sensores de Temperatura	DS18B20	Alta precisión, fácil de integrar en proyectos de múltiples sensores, resistente al agua	-55°C a +125°C	Protocolo 1-Wire, lo que permite conectar múltiples sensores en el mismo bus de datos	±0.5°C en el rango de -10°C a +85°C	Elegido
	DHT22	Mide también humedad (20% - 90% RH), buena precisión, pero limitada a temperaturas más bajas	-40°C a +80°C	Protocolo digital simple	±0.5°C	
	LM35	Muy sencillo de usar, ideal para aplicaciones donde solo se necesita una lectura de temperatura básica	0°C a +100°C	Señal analógica que depende de la tensión de referencia	±0.5°C en un rango estrecho	

Comparativa: El DS18B20 es superior cuando se necesitan múltiples sensores en un mismo sistema, gracias a su comunicación 1-Wire, y tiene un rango de temperatura amplio, lo que lo hace más versátil en aplicaciones de temperaturas extremas en comparación con el DHT22 y LM35.

	Modelo	Ventajas	Comunicación	Salida	
Sensores Magnéticos	A3144	Confiable para detectar campos magnéticos estables, resistente a ruidos electromagnéticos, ideal para detectar presencia de imanes en aplicaciones de RPM (revoluciones por minuto)	Lógico Digital	Digital (encendido/apagado)	Elegido
	AH49E	Ideal para medir la intensidad del campo magnético en lugar de solo su presencia	Lógico Digital	Analógica (proporcional a la intensidad del campo magnético)	
	Reed Switch	No requiere energía para activarse, muy confiable, pero es más lento y tiene una vida útil limitada comparado con sensores de efecto Hall	Lógico Digital	Digital (encendido/apagado)	

Comparativa: El A3144 es más preciso para detectar la presencia de campos magnéticos en proyectos donde se necesita respuesta rápida y alta durabilidad, superando al AH49E y al Reed Switch en aplicaciones que demandan precisión en la detección de manes en movimiento.

	Modelo	Ventajas	Comunicación	Tipo	Salida	
Botones	TTP223	Ofrece activación táctil sin desgaste mecánico, alta sensibilidad y fácil montaje en superficies no conductoras	Lógico Digital	Sensor táctil capacitivo	Digital (encendido/apagado)	Elegido
	Push Button	Económico y confiable, pero tiene un ciclo de vida limitado debido al desgaste mecánico	Lógico Digital	Botón mecánico	Digital (encendido/apagado)	
	KY-036	Permite leer diferentes niveles de presión, ideal para aplicaciones en las que se requiere respuesta ante diferentes grados de presión	Analógica	Sensor de presión	Analógica (proporcional a la fuerza aplicada)	

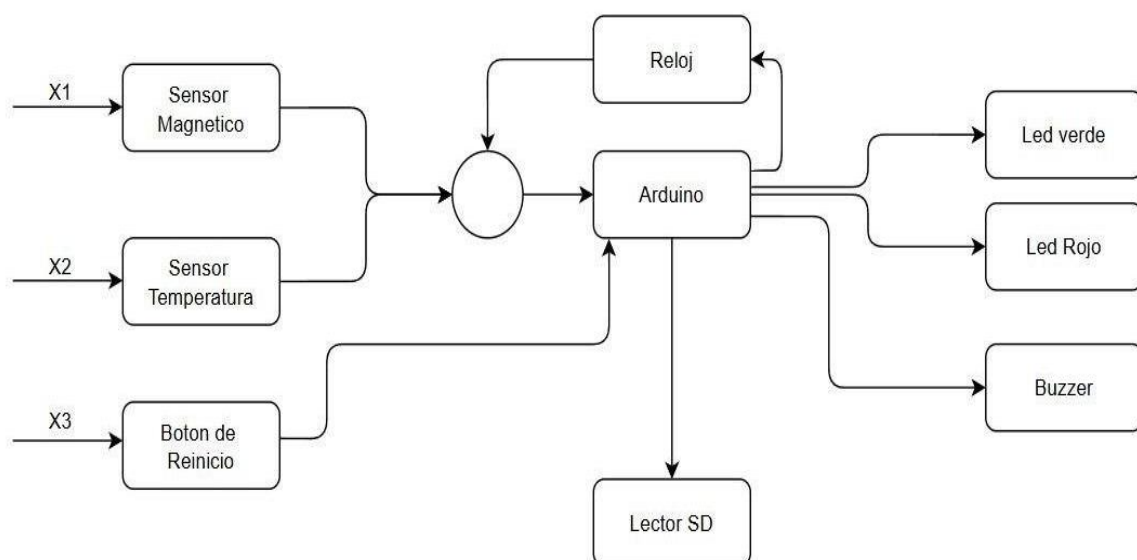
Comparativa: El TTP223 es una excelente opción para aplicaciones de Arduino que requieren un diseño sin desgaste y alta sensibilidad, donde el contacto es capacitivo. Al compararlo con el botón mecánico y el KY-036, el TTP223 sobresale en aplicaciones de interfaz táctil, especialmente en proyectos que demandan durabilidad.

### Links, Datasheet y Sensores elegidos

Sensores	Datasheet	Link de compra
DS18B20 (Temperatura)	<a href="https://www.alldatasheet.com/html-pdf/58557/DALLAS/DS18B20/2227/12/DS18B20.html">https://www.alldatasheet.com/html-pdf/58557/DALLAS/DS18B20/2227/12/DS18B20.html</a>	<a href="#">Articulo En Mercado Libre</a>
DS1307 (Reloj)	<a href="https://www.alldatasheet.com/datasheet-pdf/pdf/58481/DALLAS/DS1307.html">https://www.alldatasheet.com/datasheet-pdf/pdf/58481/DALLAS/DS1307.html</a>	<a href="#">Articulo En Mercado Libre</a>
Lector Micro SD	<a href="https://components101.com/modules/micro-sd-card-module-pinout-features-datasheet-alternatives">https://components101.com/modules/micro-sd-card-module-pinout-features-datasheet-alternatives</a>	<a href="#">Enlace de Compra</a>
A3144 (Sensor Magnetico)	<a href="https://www.alldatasheet.com/datasheet-pdf/pdf/55092/ALLEGRO/A3144.html">https://www.alldatasheet.com/datasheet-pdf/pdf/55092/ALLEGRO/A3144.html</a>	<a href="#">Articulo En Mercado Libre</a>
TTP223 (Boton)	<a href="https://www.alldatasheet.com/datasheet-pdf/pdf/1179637/TTELEC/TTP223.html">https://www.alldatasheet.com/datasheet-pdf/pdf/1179637/TTELEC/TTP223.html</a>	<a href="#">Articulo En Mercado Libre</a>

Actuadores	Datasheet	Link de compra
LEDs (Verde, Rojo y Amarillo)	<a href="http://www.adafruit.com/datasheets/WP7113SRD-D.pdf">http://www.adafruit.com/datasheets/WP7113SRD-D.pdf</a>	<a href="#">Articulo En Mercado Libre</a>
KY 012 (Buzzer)	<a href="https://www.alldatasheet.com/html-pdf/1284500/JOY-IT/KY-012/250/1/KY-012.html">https://www.alldatasheet.com/html-pdf/1284500/JOY-IT/KY-012/250/1/KY-012.html</a>	<a href="#">Articulo en Mercado Libre</a>
Tarjeta Micro SD	<a href="https://components101.com/sites/default/files/component_datasheet/Micro%20SD%20Card%20Datasheet.pdf">https://components101.com/sites/default/files/component_datasheet/Micro%20SD%20Card%20Datasheet.pdf</a>	<a href="#">Articulo En Mercado Libre</a>

### DIAGRAMA DE BLOQUES



**SENSORES Y ACTUADORES UTILIZADOS (Detalles)**

Sensor/Actuador	Nombre de la variable	Comunicacion	Conexión
Temperatura (DS18B20)	sensor_temperatura	Protocolo 1-Wire, lo que permite conectar múltiples sensores en el mismo bus de datos	Pin digital 2, 5v y GND
Sensor Magnético	No utilizado (leer referencia * en el anexo)	Digital Simple	No utilizado
LED (Rojo)	led_red	Digital Simple	Pin digital 7 y GND mediante resistencia de 220 ohm
LED (Amarillo)	led_yellow	Digital Simple	Pin digital 9 y GND mediante resistencia de 220 ohm
LED (Verde)	led_green	Digital Simple	Pin digital 8 y GND mediante resistencia de 220 ohm
Lector SD	Sdpin	Protocolo de comunicación SPI (Serial Peripheral Interface)	Pin digital 10, SCK a pin digital 13, D0 a pin digital 12, 5v y GND
Reloj (RTC_DS1307)	rtc	I2C (Inter-Integrated Circuit)	Pines analógicos 4 (SDA) y 5 (SCL), 5v y GND
Boton Reset	reset_btn	Digital Simple	Pin digital 5 y GND
Boton Main	init_btn	Digital Simple	Pin digital 4 y GND
Buzzer	buzzer	Digital Simple	Pin digital 6 y GND

Las conexiones realizadas a los pines digitales entregan un valor de 5v cuando se encuentran en estado HIGH.





## Estrategias de recarga

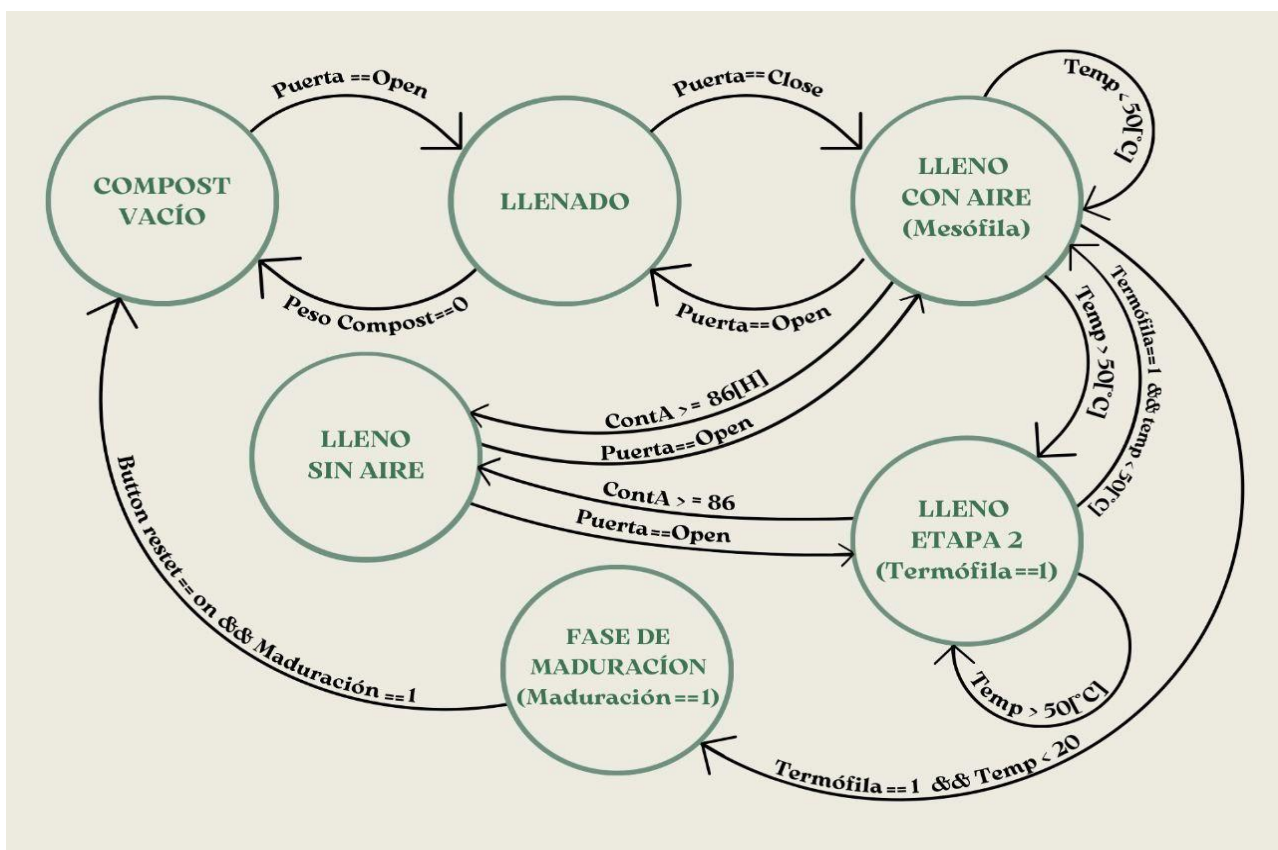
Planteamos dos estrategias de recarga, la primera es la implementación de dos baterías con ciclos de uso de una semana cada una, por consiguiente, mientras una batería esta en uso la otra batería se debería de estar cargando, pero este modelo presenta dos problemas muy grandes, uno es que el operador no debe olvidarse de cargar la batería y eso recarga al operador con mas responsabilidades. Y el otro problema es el costo de las baterías que es elevado.

Frente a esta situación planteamos la segunda estrategia de recarga de la batería que sería mediante el uso de un panel solar, aprovechando que el compostador se encuentra a la intemperie. El calculo hecho teniendo en cuenta el consumo máximo, nos dio como resultado que necesitaríamos un panel solar de 50 watts para mantener el sistema funcionando durante el día, cargando la batería y de noche con la batería, así desligaríamos de la responsabilidad de la carga de la batería al operador.

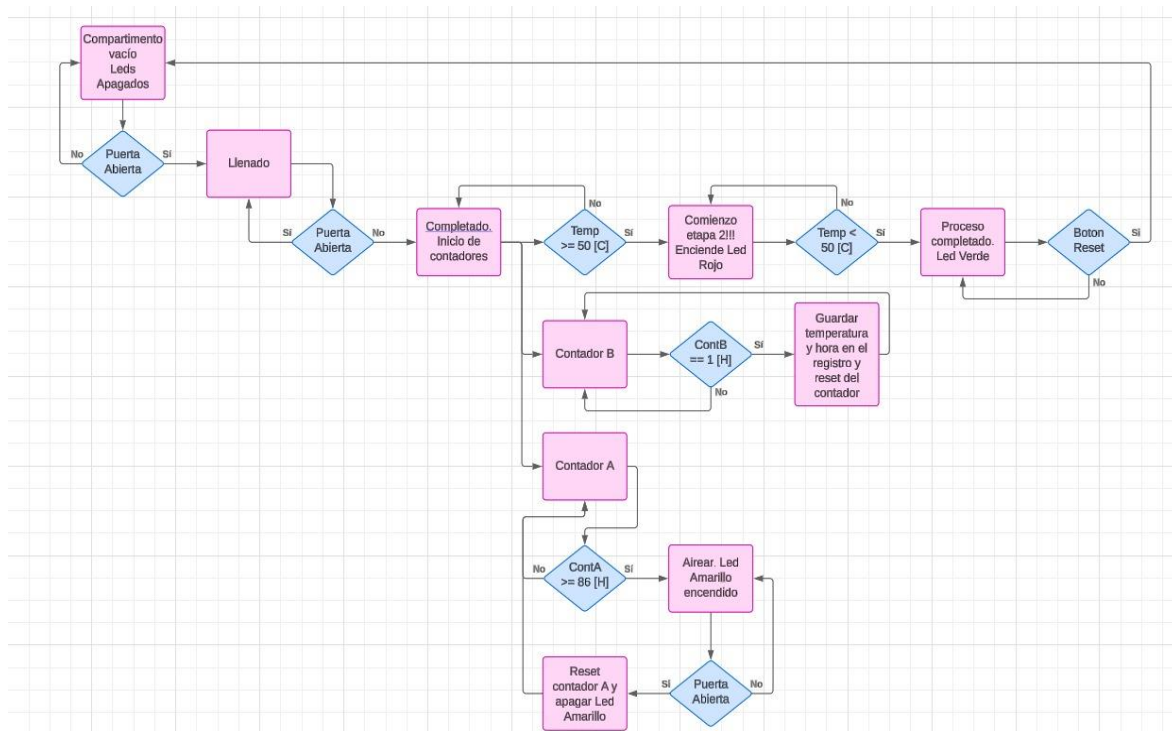
El costo del panel solar y de la batería aproximadamente es de unos 70 USD cada uno, por lo tanto, el gasto del panel solar se amortiza teniendo en cuenta el gasto del consumo de la energía al cargar las baterías y como punto ecológico favorable estamos usando energía limpia.

## ALGORITMOS PRINCIPALES

### DIAGRAMA DE ESTADOS



## DIAGRAMA DE FLUJO



## CÓDIGO

```

// LIBRERIAS
#include <OneWire.h> // Comunicación con sensores OneWire
#include <DallasTemperature.h> // Lectura de sensores de temperatura
#include <Wire.h> // Comunicación I2C entre dispositivos
#include <RTClib.h> // Manejo de tiempo con reloj RTC
#include <SD.h> // Control de tarjetas SD

// CONEXIONES
#define sensor_temperatura 2 // Pin para DS18B20 (temperatura)
#define init_btn 4 // Pin para el botón
#define reset_btn 5 // Pin para el botón reset
#define buzzer 6 // Pin para el buzzer
#define led_red 7 // Pin para LEDrojo
#define led_green 8 // Pin para LEDverde
#define led_yellow 9 // Pin para LEDamarillo
#define SDpin 10 // Pin CS para tarjeta SD

OneWire oneWire(sensor_temperatura); // Configuración de comunicacion en el pin 2
DallasTemperature sensors(&oneWire); // Instanciacion de la clase DallasTemperature

RTC_DS1307 rtc; // Configuración del RTC DS1307

File dataFile; // Instanciacion de la clase File

// Variables de tiempo necesarias para la lógica
unsigned long lastTime = 0; // Última vez que se registraron datos
unsigned long lastTimeaire = 0; // Última vez que se registraron datos
const unsigned long intervalreg = 1000; // Intervalo de 1 hora (3.600.000 ms)
const unsigned long intervalaire = 8000; // Intervalo de 1 hora (309.600.000 ms)
bool timerStarted = false; // Bandera para saber si el temporizador debe comenzar
int contadorBoton=0;

// Banderas globales para indicar el paso por los estados
bool mesofila=false;
bool termofila=false;
bool maduracion=false;
bool sinaire = false;

float temperatura = 0; // Variable global de lectura de temperatura

```

## Setup

```
void setup() {
  Serial.begin(9600); //Iniciamos la comunicacion con el puerto serial

  // Configuración de pines
  pinMode(init_btn, INPUT_PULLUP);
  pinMode(reset_btn, INPUT_PULLUP);
  pinMode(buzzer, OUTPUT);
  pinMode(led_red, OUTPUT);
  pinMode(led_green, OUTPUT);
  pinMode(led_yellow, OUTPUT);

  sensors.begin(); // Inicializar sensor de temperatura

  if (!rtc.begin()) {
    Serial.println("Error al inicializar el RTC");
    while (1);
  }
  if (!rtc.isrunning()) {
    Serial.println("RTC no está en marcha, configurando la hora...");
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  }

  if (!SD.begin(SDpin)) {
    Serial.println("Error al inicializar la tarjeta SD");
    while (1);
  }
  Serial.println("Tarjeta SD inicializada correctamente");
}
```

## Loop

```
void loop() {

  if (digitalRead(init_btn) == LOW){ // Emulador de boton
    contadorBoton= contadorBoton+1;
    delay(500);
  }

  sensors.requestTemperatures(); // Leer temperatura
  temperatura = sensors.getTempCByIndex(0);

  if (temperatura>=50){ // Encendido de alarma
    termofila=true;
    digitalWrite(led_red, HIGH);
  }

  if (termofila==true && temperatura<50){ // Encendido de alarma
    mesofila=true;
  }

  if (mesofila==true && temperatura<20){ // Encendido de alarma
    maduracion=true;
    digitalWrite(led_green, HIGH);
  }

  if (contadorBoton==2 && !timerStarted) {
    timerStarted = true; // Iniciar temporizador
    lastTime = millis(); // Escribir hora de inicio
    lastTimeaire = millis(); // Escribir hora de inicio
    registrarDatos(); // Registrar datos
    delay(1000); // Esperar 1 segundo
  }

  if (timerStarted) {

    unsigned long currentMillis = millis();

    if (currentMillis - lastTime >= intervalreg) {
      lastTime = currentMillis;
      registrarDatos();
    }
  }
}
```

```
if(sinaire=true && digitalRead(init_btn) == LOW){
  lastTimeaire = currentMillis;
  sinaire=false;
  digitalWrite(led_yellow, LOW);
}

// FUNCION DE RESET
if (maduracion==true && digitalRead(reset_btn) == LOW){
  unsigned long lastTime = 0;
  unsigned long lastTimeaire = 0;
  bool timerStarted = false;
  int contadorBoton=0;
  bool mesofila=false;
  bool termofila=false;
  bool maduracion=false;
  bool sinaire = false;
  float temperatura = 0;
  digitalWrite(led_red, LOW);
  digitalWrite(led_green, LOW);
  digitalWrite(led_yellow, LOW);
  asm volatile ("jmp 0"); // Utilizamos Assembler
}
}
```

## Función de registro de datos

```

void registrarDatos() {

    sensors.requestTemperatures();
    temperatura = sensors.getTempCByIndex(0);    // Leer temperatura

    DateTime now = rtc.now();    // Leer hora del RTC para el registro

    Serial.print("Fecha y hora: ");
    Serial.print(now.year(), DEC);
    Serial.print('/');
    Serial.print(now.month(), DEC);
    Serial.print('/');
    Serial.print(now.day(), DEC);
    Serial.print(" ");
    Serial.print(now.hour(), DEC);           // Horas
    Serial.print(':');
    Serial.print(now.minute(), DEC);
    Serial.print(':');
    Serial.print(now.second(), DEC);
    Serial.print(" | Temperatura: ");
    Serial.print(temperatura);
    Serial.println(" °C");

    dataFile = SD.open("data.txt", FILE_WRITE);    // Escritura
    if (dataFile) {
        dataFile.print(now.year(), DEC);
        dataFile.print('/');
        dataFile.print(now.month(), DEC);
        dataFile.print('/');
        dataFile.print(now.day(), DEC);
        dataFile.print(" ");
        dataFile.print(now.hour(), DEC);
        dataFile.print(':');
        dataFile.print(now.minute(), DEC);
        dataFile.print(':');
        dataFile.print(now.second(), DEC);
        dataFile.print(" | ");
        dataFile.print("Temp: ");
        dataFile.print(temperatura);
        dataFile.println(" °C");
        dataFile.close();
        Serial.println("Datos guardados en la tarjeta SD");
    } else {
        Serial.println("Error al abrir el archivo en la tarjeta SD");
    }

    tone(buzzer, 1000); // Encender LEDs y sonar buzzer como indicación
    delay(500);
    noTone(buzzer);
}

```

Para poder apreciar mejor el código y mejorar el seguimiento con los comentarios recomendamos leerlo desde la página de Wokwi en este [enlace](#).

## GABINETE

Nuestro objetivo principal al diseñar el gabinete fue el entorno en donde se instalará además de dotarlo de durabilidad. Las condiciones climáticas, el polvo y posibles golpes bruscos serían la idea principal del diseño y el punto débil a cubrir, lo que nos llevó a decantarnos por la impresión en 3D utilizando el filamento PET-G. Este material es conocido por su gran resistencia a factores externos como la humedad, los rayos UV y las variaciones de temperatura, lo que garantiza la protección de los componentes frente a las hostilidades del ambiente. Esto no asegura que sea hermético ya que presenta las conexiones propias del Arduino UNO a la vista para su uso.

### Imágenes del diseño



En esta imagen podemos observar los 3 leds de estado, el lector de tarjeta SD y el orificio para la salida del buzzer.

Mientras que en esta imagen se pueden apreciar los botones de inicio y de reset. El archivo de formato 'f3d' se encuentra en el repositorio de GitHub en las referencias por si se desea observar con más detalle.



## APRENDIZAJES Y EVIDENCIAS

Confeccionamos un PowerPoint con el contenido de esta sección. Incluimos presupuesto total, un resumen del trabajo, objetivos y metas, una descripción del desarrollo práctico, el calendario de evolución del proyecto y los tres diagramas, el link se encuentra en el repositorio de GitHub.

## **CONCLUSIÓN**

La implementación de este sistema de control para la compostera ofrece una solución práctica y educativa que integra los conocimientos adquiridos en diversas materias, al mismo tiempo que promueve un compromiso con el cuidado del medio ambiente. Se espera que el desarrollo preliminar demuestre una mejora notable en la calidad del compost generado, así como una reducción en el tiempo necesario para completar el proceso y la dependencia de quien sea responsable de hacer el compost, dándole tiempo para enfocarse en otras áreas del proceso para agregarle valor agregado a este o permitirle gestionar fácilmente una cantidad mayor de composteras.

Este proyecto no solo ha permitido aplicar conceptos teóricos, sino prácticos y también establece una base para futuros desarrollos en sostenibilidad dentro de nuestra universidad, mostrando cómo la tecnología puede combinarse con otras disciplinas para generar soluciones innovadoras y cada vez mejores.

## ANEXO

### Cálculos de Batería

Consumo de los Componentes

- **Arduino Uno**

$$50 \text{ mA} \times 5 \text{ v} = 250 \text{ mWh}$$

- **A3144**

$$3 \text{ mA} \times 5 \text{ v} = 15 \text{ mWh}$$

- **DS18B20**

$$1.5 \text{ mA} \times 5 \text{ v} = 7,5 \text{ mWh}$$

- **Lector SD**

$$60 \text{ mA} \times 5 \text{ v} = 300 \text{ mWh}$$

$$\text{Como se escribe 1 vez por hora } (60 \text{ mA} \times 5 \text{ v} \times 1/3600 \text{ [s]}) = 0.111 \text{ mWh}$$

$$\text{Cuando está inactivo } (4 \text{ mA} \times 5 \text{ V} \times 3599 / 3600 \text{ [s]}) = 19.995 \text{ mWh}$$

- **Led**

$$20 \text{ mA} \times 5 \text{ v} = 100 \text{ mWh}$$

- **KY-012**

$$25 \text{ mA} \times 5 \text{ v} = 125 \text{ mWh}$$

- **RTC DS1307**

$$1.5 \text{ mA} \times 5 \text{ v} = 7,5 \text{ mWh}$$

Por lo tanto:  $0.111 \text{ mWh} + 19.995 \text{ mWh} = 20 \text{ mWh}$  ← Este es el consumo aproximadamente

### Consumos Totales

Consumo mínimo

$$250 + 15 + 7,5 + 100 + 7,5 + 20 = 400 \text{ mWh}$$

Consumo promedio

$$250 + 15 + 7,5 + 2 \times 100 + 0.3 \times 125 + 7,5 + 20 = 537 \text{ mWh}$$

Consumo máximo

$$250 + 15 + 7,5 + 3 \times 100 + 125 + 7,5 + 20 = 725 \text{ mWh}$$

### Consumo en una semana

$$537 \text{ mWh} \times 24 \text{ [h]} \times 7 \text{ [d]} = 90216 \text{ Wh}$$

$$90216 \text{ Wh} / 5 \text{ V} = 18043 \text{ mAh}$$

### Margen de seguridad del 20%

$$18043 \text{ [mAh]} \times 1.2 = 21651 \text{ [mAh]}$$

### Cálculos del Panel Solar

Consumo máximo del sistema: 725 mWh (0.725 Wh) por hora.

Consumo diario del sistema: 17.4 Wh (725 mWh  $\times$  24 horas).

Capacidad de la batería: 273.6 Wh (22,800 mAh a 12 V).

Horas de luz solar al día: 8 horas (cantidad mínima en invierno en Argentina)

Potencia mínima del panel solar = (Consumo diario del sistema + Energía para recargar la batería) / Horas de luz solar al día



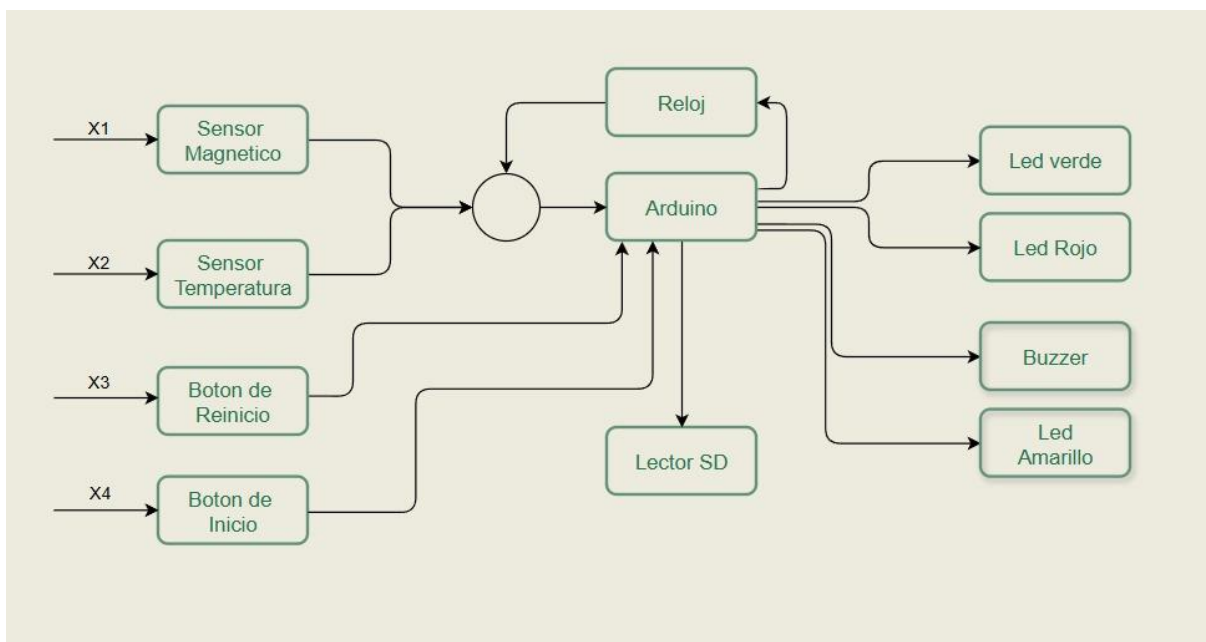
Por lo tanto:

Potencia mínima del panel solar =  $(17.4 \text{ Wh} + 273.6 \text{ Wh}) / 8 \text{ horas} = 291 \text{ Wh} / 8 \text{ horas} = 36.37\text{W}$ , aproximadamente 40W.

Teniendo en cuenta el 20% de seguridad recomendado:  $40 \text{ W} * 1.2 = 48 \text{ W}$ .

Por lo que necesitaríamos un panel solar de potencia de 50 Watts.

**Referencia \*:** Durante el desarrollo del gemelo digital no encontramos el sensor magnético para su implementación en Wokwi, por lo tanto, reemplazamos este por un botón del mismo modelo que el utilizado para el reset, así entonces, en el desarrollo lógico teórico del trabajo practico incluyó dicho sensor, pero en cuanto al desarrollo practico utilizamos el botón 'inicio' o 'init' como se ve en el código. De esta forma su uso es el mismo y su aplicación también ya que ambos utilizan el mismo tipo de comunicación y el mismo tipo de conexión con el Arduino.



Acá se pueden observar las dos entradas, el botón de inicio y la entrada magnética, como ven, ambos van al Arduino y se pueden intercambiar el uno con el otro.

## REFERENCIAS Y BIBLIOGRAFÍAS CONSULTADAS

Bonet, F., Acosta M., Delgado S., Basilio N., Descalzo F., Ismael B. y Ramírez T. (2024, noviembre 19). *Repositorio compostera*. GitHub.

[https://github.com/FranciscoJoseBonet/the\\_university\\_paradigm.git](https://github.com/FranciscoJoseBonet/the_university_paradigm.git)

Autosolar Energía del Perú S.A.C. (2022, agosto 12). *Cómo cargar una batería con un panel solar*.

<https://autosolar.pe/aspectos-tecnicos/como-cargar-una-bateria-con-un-panel-solar>

OpenAI. (2024, noviembre 19). *ChatGPT: Your AI assistant*.

<https://chatgpt.com/share/673c92c2-625c-8003-9a7f-c30279b91b41>

ALLDATASHEET. (2018, septiembre). *Electronic Components Datasheet Search*.

<https://www.alldatasheet.com/>

Components101. (2024, noviembre). *Datasheets*.

<https://components101.com/>

PIEDRA ARTIFICIAL BALAUSTRÉS MARTÍNEZ (2020 ). *¿Cómo funciona el compostaje?*. <https://www.piedra-artificial.es/blog/jardineria/como-functiona-el-compostaje/>

Alicia E. Castillo, Silvio H. Quarín y María C. Iglesias (2000, enero). *Vermicompost chemical and physical characterization from raw and mixed organic wastes*.

SciELO. [https://www.scielo.cl/scielo.php?script=sci\\_arttext&pid=S0365-28072000000100008](https://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0365-28072000000100008)