

Desarrollo de Prototipos Electrónicos para Desafíos Ambientales en Costa Rica:

**Los Casos de Incendios Forestales
y Tortugas Marinas**

Pasantía Profesional para
Ingeniería Electrónica

Luis Antonio Pérez Piedra

Periodo:

08 de Marzo al 08 de Junio del 2021

Índice

Introducción	3
Dispositivo Automático para el Monitoreo de Altas Temperaturas In situ para Incendios Forestales	4
<i>El Desafío</i>	5
<i>La Propuesta</i>	6
<i>Componentes del Sistema</i>	6
<i>Diagrama del Sistema</i>	7
<i>Código del Sistema</i>	8
<i>Evolución del Sistema</i>	9
<i>Datos Generados</i>	11
<i>Resultados, Recomendaciones y Próximos Pasos</i>	11
Monitoreo de Variables Críticas en el Proceso de Anidación de Tortugas Marinas en Peligro de Extinción	13
<i>El Desafío</i>	14
<i>La Propuesta</i>	15
<i>Componentes del Sistema</i>	15
<i>Diagrama del Sistema</i>	17
<i>Códigos del Sistema</i>	17
<i>Dirección de cada Sensor de Temperatura</i>	17
<i>Calibrar Sensor de Humedad</i>	18
<i>Arduino MKR1000</i>	18
<i>Google Script</i>	21
<i>Evolución del Sistema</i>	22
<i>Datos Generados</i>	24
<i>Gráficas Generadas</i>	25
<i>Resultados, Recomendaciones y Próximos Pasos</i>	26
Referencias	27
Bibliografía	27

Introducción

Actualmente en Costa Rica se enfrentan dos problemáticas ambientales. Una de estas está relacionada con los incendios forestales y la otra está ligada a las tortugas marinas. Ambos problemas son de alta importancia para el país ya que representa el 6% de la biodiversidad mundial, cuenta con 12 microclimas y actualmente tiene una cobertura forestal del 52,3%. Es por esto que el IICA decidió tomar acción para combatir estas situaciones que se enfrentan actualmente en el país con la creación de dos sistemas electrónicos realizados en el FabLab de la institución.

Para efectuar una mejor explicación de cada uno de los sistemas, se decidió dividir el documento en dos partes. Cada parte corresponde a uno de los sistemas desarrollados por el FabLab del IICA. De esta forma se puede desarrollar cada tema con mayor facilidad sin necesidad de hacer saltos drásticos de sistema y así solo se enfoca toda la atención en un tema. Primero se explicará el sistema dirigido al combate contra los incendios forestales y por último, el que está dirigido al monitoreo de variables para los nidos de tortugas marinas.

Dispositivo Automático para el Monitoreo de Altas Temperaturas *In situ* para Incendios Forestales

Los incendios forestales consisten en fuegos no programados o no controlados que afectan las tierras forestales como recurso protector, económico y recreativo. Dichos acontecimientos, además de afectar los suelos nacionales, llegan a afectar también la fauna y en algunos casos, dependiendo del área en la que se genere un incendio, se pueden ver perjudicadas las personas que residen en la zona.

Los pueden ser ocasionados por dos principales causas, las naturales y las antrópicas:

- En el caso de las causas naturales, comúnmente se generan gracias a la caída de rayos producidos por tormentas eléctricas que no están acompañadas de lluvia. Así mismo, otra causa que puede producir incendios en algunas regiones es la actividad volcánica.
- Las causas antrópicas son las que se producen gracias a la presencia del hombre. Los accidentes o negligencias en el uso de esos recursos y el deficiente manejo o mala intención en la aplicación del fuego generan que la mayoría de los incendios sean causados por el ser humano. Ejemplos de estos pueden ser campamentos, preparación de localizaciones para forestación, agricultura, ganadería o urbanización, también por vandalismo, fallas en transformadores de energía, incendios en vivienda e inclusive accidentes terrestres o aéreos.

Según un estudio realizado en el año 2020 por el *World Wild Fund* junto con el *Boston Consulting Group* indicó que en el 2019 se registró el segundo nivel más alto de incendios forestales a nivel global. Además, a esto se le puede sumar el

hecho de que para abril del 2020 había aumentado el número de alertas sobre incendios forestales un 13% en comparación al año previo. Entre los principales factores que se caracterizan por fomentar el aumento mencionado se puede encontrar la persistencia de climas más secos con mayores temperaturas y factores humanos como el pobre manejo forestal y la conversión de tierras a agricultura. En compensación, la pandemia mundial generada por la COVID-19 está ocasionando que los gobiernos redireccionen los recursos para ayudar a la lucha contra el virus, lo cual produce una escasez de recursos para el combate y la prevención de los incendios forestales, así como una disminución o erradicación de acciones tomadas por parte de Bomberos y Bomberas Forestales. Esto ha ocasionado que en algunos países se presente un incremento en la deforestación y en el aumento de incendios. (WWF. BCG, 2020).

El Desafío

Costa Rica actualmente cuenta con una muy limitada cantidad tanto de estudios e investigaciones como de instrumentos los cuales permitan estudiar la dinámica y el comportamiento del fuego durante un incendio forestal. Esto es de suma importancia ya que en un país como Costa Rica se cuenta con una gran variedad de ecosistemas. Es principalmente por esta razón que es complicado la mejora de la técnica de manejo de incendios durante un incidente.

La Propuesta

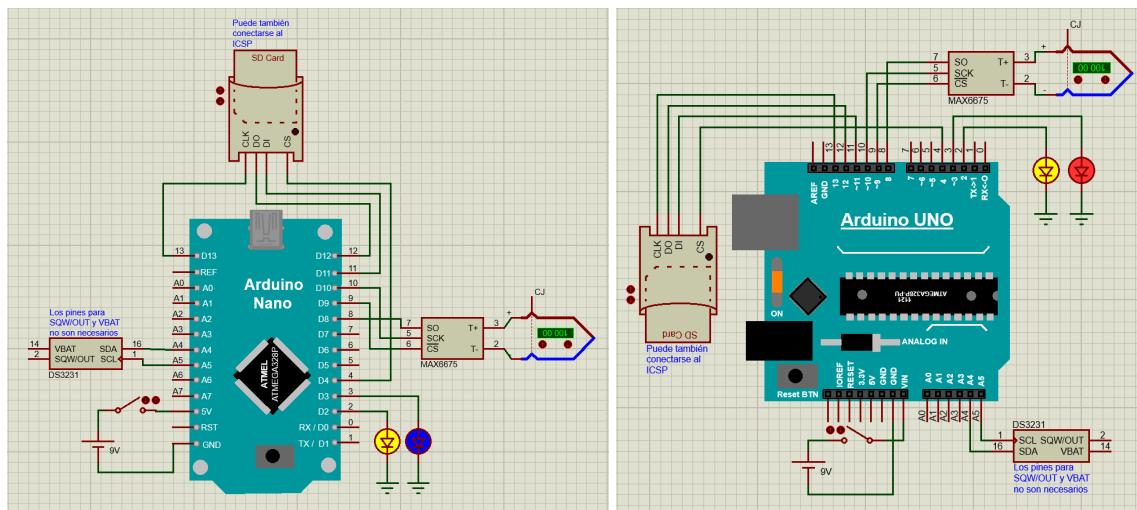
Es por esto que se creó un sistema el cual permite medir tanto los niveles como la evolución de la temperatura en todos los sistemas que se encuentran sujetos al riesgo de un incendio forestal. Este sistema cuenta con un sensor de temperatura y es enterrado cerca del incendio para que así no sufra daños a causa de las altas temperaturas una vez que el evento se propague. Dicho sensor queda expuesto en la superficie para así capturar los datos una vez que el incendio pase sobre este.

Componentes del Sistema

- Arduino Uno o Nano: Es el procesador del sistema que se encarga de la interconexión de todos los componentes del sistema y también de recibir los datos, interpretarlos y enviarlos al dispositivo de memoria.
- Sensor de Temperatura: El sensor de temperatura es una Termocupla tipo K que tolera valores de hasta 600°C. También se cuenta con un módulo MAX6675 el cual se encarga de convertir los valores tomados por el sensor en una señal que el Arduino pueda interpretar.
- Reloj de Tiempo Real: Se tiene un DS3231 el cual es un *Real Time Clock* de alta precisión que se encarga de registrar la fecha y hora en la que es tomado cada dato. Este es alimentado con una batería tipo moneda de 3 Voltios.
- Memoria: Para el almacenamiento de los datos se cuenta con un módulo para tarjetas MicroSD de la marca SparkFun. En este se crea un archivo de texto en el cual se encuentran todos los datos registrados por el sistema.

- Alimentación: Para hacer al sistema funcionar de forma completamente independiente, se implementó una fuente de alimentación usando una batería de 9 Voltios. Esto es ya que dichas baterías son muy fáciles de encontrar alrededor del país sin importar que tan remota sea la zona en la que se implemente el sistema.
- Encapsulado: El sistema es guardado en un tubo de Policloruro de Vinilo (PVC) para que de esta forma no sufra daños a causa de la tierra, la humedad de esta y las altas temperaturas a las que va a ser expuesto al estar enterrado.

Diagrama del Sistema



Se implementaron dos LEDs que se encargan de indicar cuando el sistema se encuentra encendido y cada vez que un dato esté siendo tomado. En el primer sistema creado se usó un Arduino Nano por su tamaño reducido aunque luego se crearon dos sistemas más usando Arduinos Uno al ver que entraban sin problema en los encapsulados de PVC.

Código del Sistema

```
#include <max6675.h>
#include <DS3231.h>
#include <SD.h>
#include <SPI.h>

DS3231 rtc(SDA, SCL);

float temperatura=0;
int ktcS0 = 8;
int ktcCS = 9;
int ktcCLK = 10;
MAX6675 ktc(htcCLK, ktcCS, ktcS0);

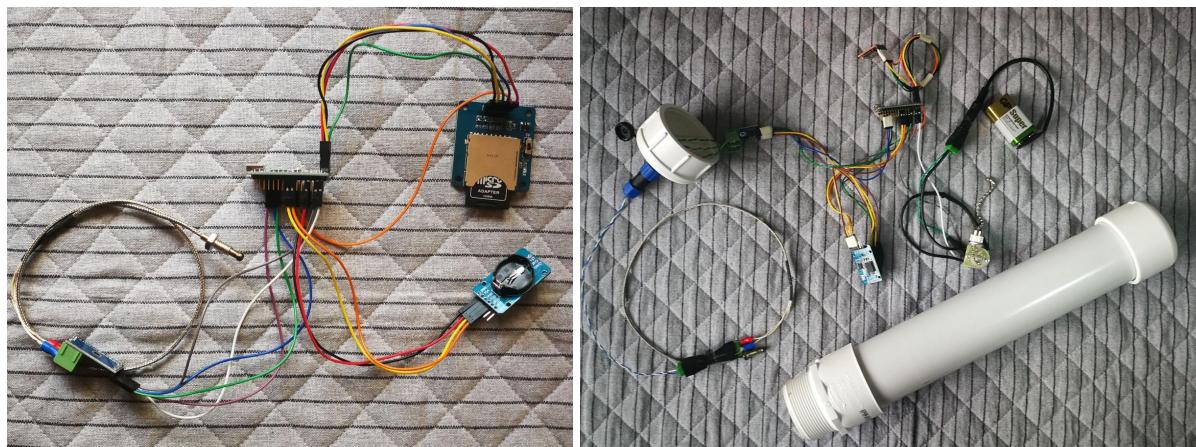
File archivo;
void setup() {
  Serial.begin(9600);
  pinMode(2, OUTPUT);
  digitalWrite(2, HIGH);
  pinMode(3, OUTPUT);
  rtc.begin();
  rtc.setTime(11, 52, 15);
  rtc.setDate(06, 03, 2021);
  while (!Serial) {
    ;
  }
  Serial.print("Iniciando tarjeta SD...\n");
  if (!SD.begin(4)) {
    Serial.print("No se pudo iniciar la tarjeta SD...");
    while (1);
  }
  archivo.println(" ");
  Serial.println("Tarjeta SD iniciada con éxito...");
}

void loop() {
  delay(55000);
  archivo = SD.open("datos.txt", FILE_WRITE);
  if (archivo) {
    archivo.print(rtc.getTimeStr());
    archivo.print(",");
    archivo.print(rtc.getDateStr());
    archivo.print(",");
    temperatura=ktc.readCelsius();
    temperatura=ktc.readCelsius();
    archivo.print(temperatura);
    archivo.print("°C");
    archivo.println(" ");
  }
}
```

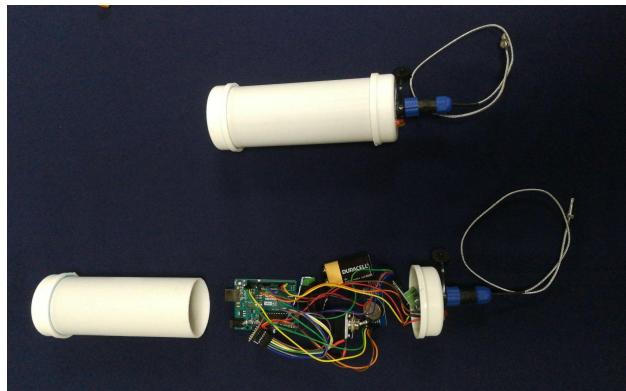
```

digitalWrite(3, HIGH);
delay(5000);
archivo.close();
digitalWrite(3, LOW);
} else {
  Serial.println("Error al abrir el archivo...");
}
Serial.print("HORA --> ");
Serial.print(rtc.getTimeStr());
Serial.print("    FECHA --> ");
Serial.println(rtc.getDateStr());
temperatura=ktc.readCelsius();
Serial.print("    TEMPERATURA --> ");
Serial.print(temperatura);
Serial.println("°C");
}
  
```

Evolución del Sistema



- En la foto izquierda se ve el primer montaje usando un Arduino Nano junto con el sensor de temperatura, el reloj y la memoria.
- En la foto derecha se ve el primer encapsulado usado para el sistema.

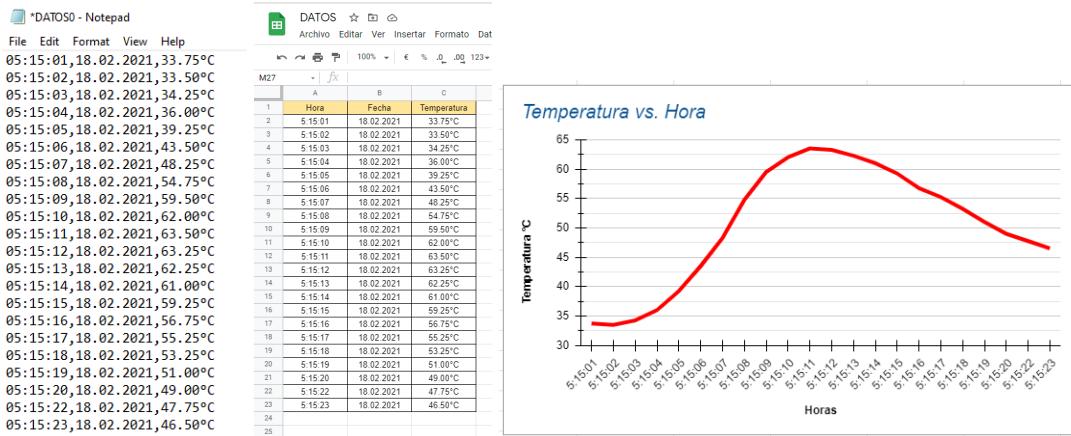


- Se redujo el tamaño del primer sistema y en base a esto se tomaron medidas para los otros dos sistemas.
- Ambos sistemas implementan el uso de Arduinos Uno.



- Primera prueba en la cual se tuvo que cavar un agujero de 50 cm para enterrar todo el sistema. Una vez enterrado se dejó descubierto el sensor.

Datos Generados



- Como se mencionó previamente, el sistema se encarga de generar un archivo de texto el cual puede ser fácilmente exportado a Google Sheets o Excel para así tener mejor ordenados los datos e inclusive para ver un análisis gráfico.

Resultados, Recomendaciones y Próximos Pasos

Entre los resultados se obtuvieron tres sistemas funcionales y compactos capaces de almacenar datos por más de 12 horas continuas.

Como recomendaciones se tiene realizar una mejor administración del cableado del sistema para así poder reducir el tamaño del sistema para hacerlo aún más compacto.

Finalmente, como próximos pasos se puede implementar un módulo o emplear el uso de un Arduino que cuente con conexión a internet para así enviar datos en tiempo real a los usuarios ya sea que estén combatiendo el incendio o monitoreando la situación desde afuera.

Monitoreo de Variables Críticas en el Proceso de Anidación de Tortugas

Marinas en Peligro de Extinción

Actualmente se calcula un aumento en las temperaturas globales de 1,8°C a 4°C para el último siglo. Esto es un tema alarmante para las tortugas marinas ya que, al igual que muchos reptiles, estas son altamente sensibles a cambios de temperatura. Una de las principales etapas en las que estas especies se pueden ver afectadas a causa del aumento de temperaturas es durante el desarrollo embrionario. El sexo de las tortugas marinas, al igual que muchas otras especies, es determinado por medio de la temperatura y esto sucede durante el segundo tercio del periodo de incubacion. A temperaturas más elevadas se producen más hembras, mientras que para temperaturas más bajas se producen más machos. Es por esto que para una incubación exitosa los nidos deben encontrarse en una temperatura que vaya desde los 25°C hasta los 31°C ya que esto produce una proporción de sexos 1:1. Además, se ha visto que las altas temperaturas pueden llegar a acelerar la duración de la incubación y esto da como resultado neonatos más pequeños y débiles. (Baker, Fish & Drews. 2009)

Según Santidrián (2011) la estabilidad de las playas, la productividad del océano y la temperatura de la playa y de los mares son factores que pueden potencialmente afectar a todas las especies de tortugas marinas ya que, gracias al aumento de las temperaturas, los ecosistemas se están viendo afectados y dichos factores son los que se encargan de determinar una serie de factores involucrados en su desarrollo.

Baker, Fish y Drews (2009) establecen que una serie de características que influyen en las temperaturas deben ser analizadas en viveros, así como en nidos no reubicados. Entre los factores que se deben considerar se encuentran las características de la arena como el color, el tamaño de las partículas y el contenido de humedad ya que estos influyen en la transferencia de calor que recibe el nido. Además, se debe tener en cuenta la radiación reflejada o emitida por un objeto ya que en playas más oscuras las temperaturas son más elevadas y que un mayor tamaño de partículas de arena y un mayor contenido de agua generan una mayor conductividad térmica. Otra característica que determina la temperatura de la arena sería el sombreado, en términos de cantidad de horas, en áreas específicas de las playas que se encuentren expuestas o no al sol. (Baker, Fish & Drews. 2009)

El Desafío

Las elevadas temperaturas generadas gracias al cambio climático es un factor de alto impacto en las especies de tortugas marinas alrededor del mundo. Como se mencionó anteriormente, las altas temperaturas causan que nazcan neonatos débiles y pequeños o que inclusive produzcan la muerte de los neonatos. A esto se le puede sumar el hecho de que de cada 1000 tortugas marinas solo 1 llega a la adultez, lo cual es una evidencia muy clara de que esta especie realmente se está viendo afectada a un nivel global. Es por esto que es de suma importancia tener algún tipo de dispositivo con el cual se pueda monitorear una serie de variables relacionadas con los nidos o viveros de tortugas marinas.

La Propuesta

Es de suma importancia tener, previo al establecimiento de un vivero, un monitoreo de tanto la humedad como la temperatura del nido. Es por esto que se creó un sistema el cual cuenta con tres sensores de temperatura y uno de humedad. Uno de los sensores se debe encontrar a lo largo del vivero, otro posicionado de cara al mar, uno en el centro y el final debe estar posicionado hacia la tierra para así cubrir por completo la zona del nido. Esto es complementado por una estructura que, tomando en cuenta las dimensiones específicas de la distribución de los nidos establecida para cada especie, cuente con aspersores en la parte superior que cubren un área de al menos tres nidos para controlar tanto la temperatura como la humedad del vivero.

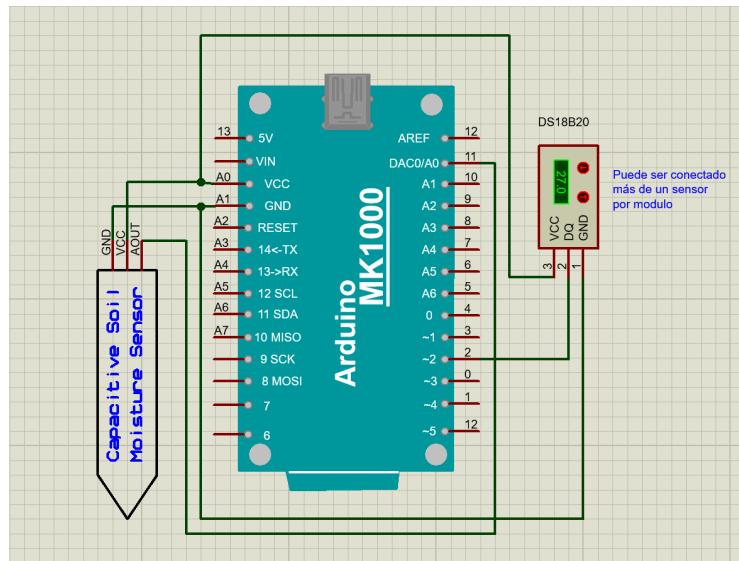
Componentes del Sistema

- Arduino MK1000: Es el sistema embebido del sistema el cual tiene la capacidad de ser conectado a una red Wi-Fi, por lo que se encarga de la interconexión de todos los componentes del sistema y también de recibir los datos, interpretarlos y enviarlos por medio de internet.
- Sensores de Temperatura y Humedad: El sensor de temperatura es un DS18B20 el cual tolera temperaturas desde los -55°C hasta los 125°C y tiene la particularidad de soportar la conexión de más de un sensor en su entrada lo que permite reducir la complejidad del sistema. En el caso de la humedad, se cuenta con un Sensor Capacitivo de Humedad de Suelo el cual es

resistente a la corrosión. A este se le adaptó un encapsulado imprimido en 3D para proteger su circuitería de la humedad.

- Almacenamiento de Datos: Como los datos pueden ser enviados por medio de internet no hay necesidad de un dispositivo físico de almacenamiento. En este caso los datos son enviados directamente en tiempo real a un documento de Google Sheets lo cual permite una mejor lectura e interacción con los datos.
- Conexión: Se cuenta con un punto de acceso Wi-Fi Trinity de la marca Nexxt el cual cuenta con una batería recargable con una duración de 6 horas y soporta conexiones 3G y 4G. Este es usado para poder acceder al internet de forma remota y así poder enviar los datos tomados por el dispositivo.
- Alimentación: Para hacer al sistema funcionar de forma completamente independiente, se debe contar con adaptabilidad con sistemas de alimentación solar para que así este funcione las 24 horas del día.

Diagrama del Sistema



- Originalmente se había realizado un prototipo inicial usando un Arduino Uno y un módulo de tarjetas SD pero este fue luego intercambiado por un Arduino MKR1000 para simplificarlo por completo.

Códigos del Sistema

Dirección de cada Sensor de Temperatura

```
#include <OneWire.h>

OneWire ourWire(2);

void setup(void) {
  Serial.begin(9600);
}

void loop(void) {
  byte addr[8];
  Serial.println("Obteniendo direcciones:");
  while (ourWire.search(addr))
  {
    Serial.print("Address = ");
    for (int i=0; i<8; i++)
      Serial.print(addr[i], HEX);
    Serial.println();
  }
}
```

```

for( int i = 0; i < 8; i++) {
  Serial.print(" 0x");
  Serial.print(addr[i], HEX);
}
Serial.println();
}

Serial.println();
ourWire.reset_search();
delay(2000);
}

```

Calibrar Sensor de Humedad

```

float ap = A0;
float valor;
float calculo;
float Porcentaje;

void setup() {
  Serial.begin(9600);
}

void loop() {
  valor = analogRead(ap);
  delay(1000);
  Serial.println(valor);
  calculo = ((valor-505)*100L)/(952-505);
  //Los valores de 505 y 952 son sustituidos por
  //los obtenidos por la variable 'valor' al
  //encontrarse el sensor seco y luego húmedo
  Porcentaje = 100-calculo;
  Serial.print(" ");
  Serial.print(Porcentaje);
  Serial.println("%");
}

```

Arduino MKR1000

```

#include <WiFi101.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <SD.h>
#include <SPI.h>

```

```

OneWire ourWire(2);
DallasTemperature sensors(&ourWire);
DeviceAddress address1 = {0x28, 0x1C, 0x8F, 0x79, 0xA2, 0x0, 0x3, 0xFC};
DeviceAddress address2 = {0x28, 0x89, 0xC5, 0x79, 0xA2, 0x0, 0x3, 0xF2};
DeviceAddress address3 = {0x28, 0xD7, 0x47, 0x79, 0xA2, 0x0, 0x3, 0x87};

float EA = A0;
float humedad;
float calculo;
float porcentaje;

const char WEBSITE[] = "api.pushingbox.com";
const String devid = "v00DB1FF6420FBAE";
const char* MY_SSID = "Familia Perez";
const char* MY_PWD = "familiaPEREZ";

int status = WL_IDLE_STATUS;

void setup() {
    Serial.begin(9600);
    sensors.begin();
    if (WiFi.status() == WL_NO_SHIELD) {
        while (true);
    }
    while (status != WL_CONNECTED)
    {
        Serial.print("Intentando conectarse al dispositivo de internet: ");
        Serial.println(MY_SSID);
        status = WiFi.begin(MY_SSID, MY_PWD);
        delay(5000);
    }
    Serial.println("\nConectado!!!\n");
    printWifiStatus();
}

void printWifiStatus() {
    Serial.print("(SSID: ");
    Serial.print(WiFi.SSID());
    Serial.print(" / ");

    IPAddress ip = WiFi.localIP();
    Serial.print("Dirección IP: ");
    Serial.print(ip);
    Serial.print(" / ");

    long rss = WiFi.RSSI();
    Serial.print("Fuerza de la señal (RSSI): ");
    Serial.print(rss);
    Serial.println("dBm)\n");
}

```

```

void loop() {
    delay(900000);
    Serial.println(".....\n");
    humedad = analogRead(EA);
    calculo = (humedad-505)*100L/(952-505);
    porcentaje = 100-calculo;
    sensors.requestTemperatures();
    float temp1= sensors.getTempC(address1);
    float temp2= sensors.getTempC(address2);
    float temp3= sensors.getTempC(address3);
    Serial.print("Temperatura Sensor 1: ");
    Serial.print(temp1);
    Serial.println(" °C");
    Serial.print("Temperatura Sensor 2: ");
    Serial.print(temp2);
    Serial.println(" °C");
    Serial.print("Temperatura Sensor 3: ");
    Serial.print(temp3);
    Serial.println(" °C");
    Serial.print("Humedad: ");
    if (porcentaje > 100) {
        Serial.println("100%");
    } else if (porcentaje < 0) {
        Serial.println("0%");
    } else if (porcentaje > 0 && porcentaje < 100) {
        Serial.print(porcentaje);
        Serial.println("%");
    }
    WiFiClient client;
    if (client.connect(WEBSITE, 80)){
        client.print("GET /pushingbox?devid=" + devid
        + "&TemperaturaS1=" + (String) temp1
        + "&TemperaturaS2=" + (String) temp2
        + "&TemperaturaS3=" + (String) temp3
        + "&Humedad=" + (String) porcentaje
        );
        client.println(" HTTP/1.1");
        client.print("Host: ");
        client.println(WEBSITE);
        client.println("User-Agent: MKR1000/1.0");
        client.println();
        Serial.println("Datos Enviados!!!\n");
        client.stop();
    }
}

```

Google Script

```

function doGet(e) {
  Logger.log( JSON.stringify(e) );

  var result = 'Ok';
  if (e.parameter == undefined) {
    result = 'No Parameters';
  }
  else {
    var id = '1gNBDofW0-h0bLTEZ6Wp7GFe9J0DBhkUFVo8GT2tnHg8';
    var sheet = SpreadsheetApp.openById(id).getActiveSheet();
    var newRow = sheet.getLastRow() + 1;
    var rowData = [];
    rowData[0] = new Date();

    for (var param in e.parameter) {
      Logger.log('In for loop, param='+param);
      var value = stripQuotes(e.parameter[param]);
      switch (param) {
        case 'TemperaturaS1':
          rowData[1] = value;
          break;
        case 'TemperaturaS2':
          rowData[2] = value;
          break;
        case 'Temperaturas3':
          rowData[3] = value;
          break;
        case 'Humedad':
          rowData[4] = value;
          break;
        default:
          result = "unsupported parameter";
      }
    }
    Logger.log(JSON.stringify(rowData));

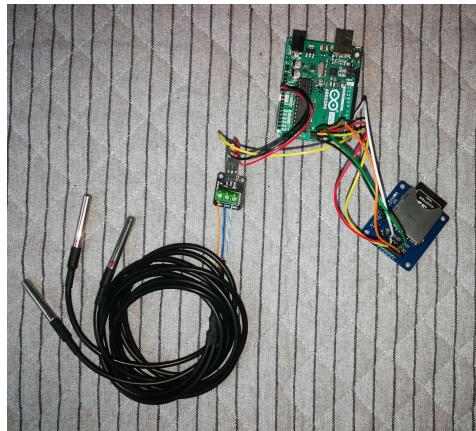
    var newRange = sheet.getRange(newRow, 1, 1, rowData.length);
    newRange.setValues([rowData]);
  }

  return ContentService.createTextOutput(result);
}

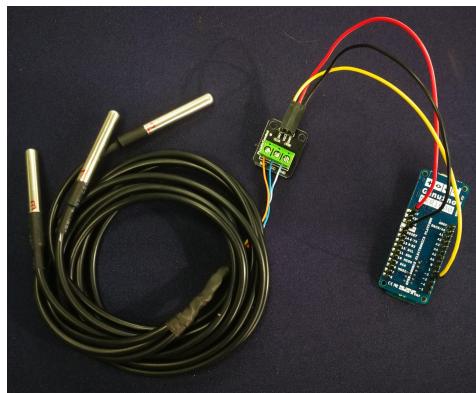
function stripQuotes( value ) {
  return value.replace(/^["]|[""]$/g, "");
}

```

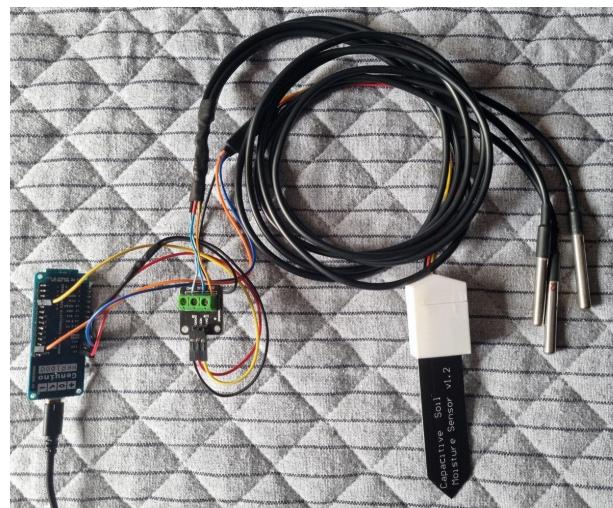
Evolución del Sistema



- Prototipo inicial usando un Arduino Uno en vez del Arduino MKR1000 de la versión final.



- Primero se realizó una prueba utilizando los sensores de temperatura para probar el funcionamiento del Arduino MKR1000.



- Sistema con los sensores de temperatura y el sensor de humedad siendo alimentado por medio de conexión USB a un cargador de 25 Watts.



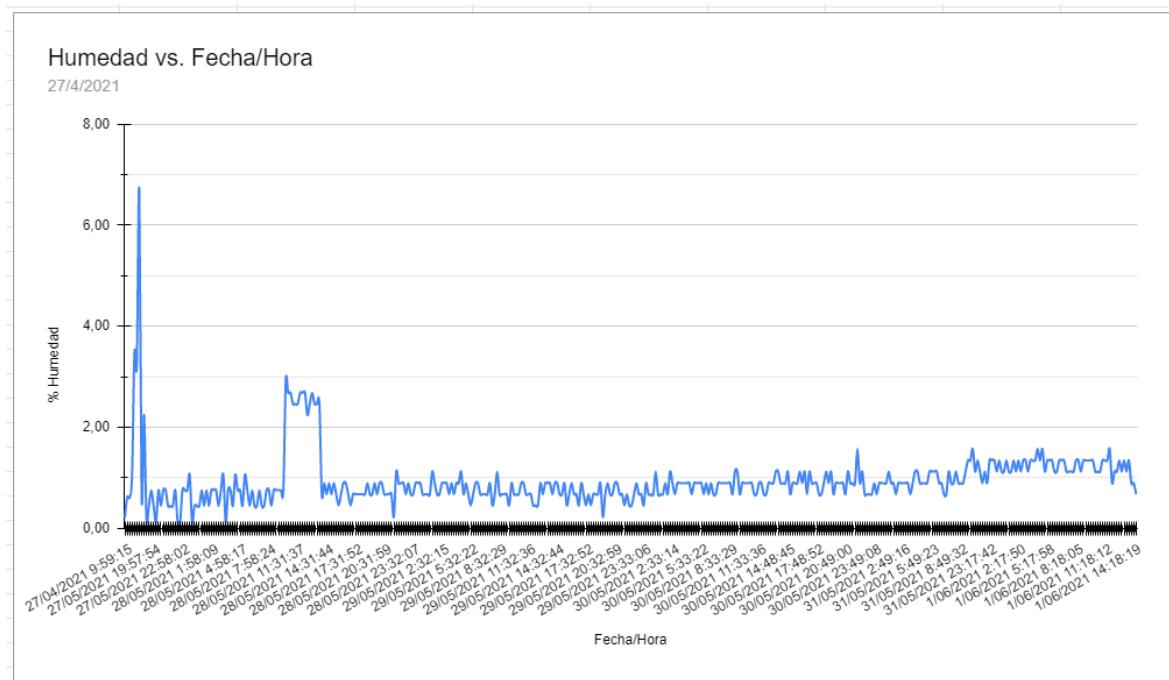
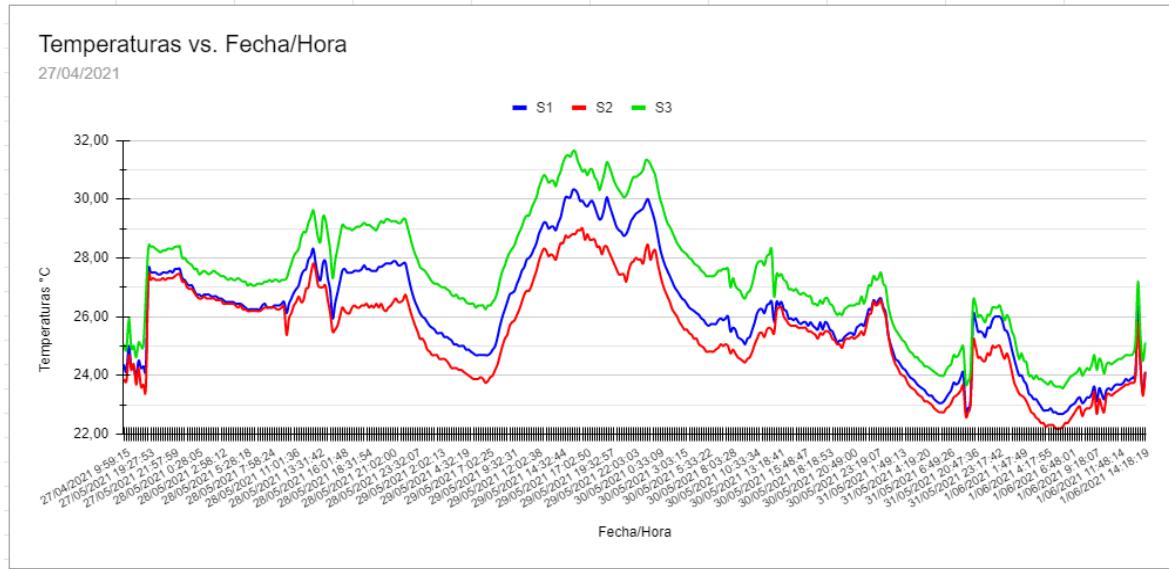
- Para fines de simulación, se realizó una maqueta en la cual fueron posicionados los diferentes sensores de forma distribuida para tener una idea del comportamiento que estos podrían llegar a tener.

Datos Generados

Datos del Nido (MKR1000)					
	A	B	C	D	E
1	Fecha/Hora	Sensor 1	Temperaturas (°C)	Sensor 3	% Humedad
2					
3	27/04/2021 9:59:15	24,38	23,88	25,06	0,20
4	27/04/2021 10:14:15	24,12	23,81	24,88	0,61
5	27/04/2021 10:29:17	25,00	24,69	25,94	0,61
6	27/04/2021 10:44:15	24,25	24,19	24,94	1,02
7	27/04/2021 10:59:13	24,31	24,38	25,00	3,46
8	27/04/2021 11:14:10	23,94	23,69	24,62	3,25
9	27/04/2021 11:29:09	24,50	24,25	25,12	6,71
10	27/04/2021 13:58:59	24,25	23,62	25,00	0,61
11	27/04/2021 14:13:58	24,31	23,69	25,06	2,24
12	27/04/2021 14:28:58	24,31	23,62	26,69	0,20
13	27/05/2021 19:27:53	27,44	27,25	28,31	0,44
14	27/05/2021 19:42:54	27,50	27,25	28,38	0,75
15	27/05/2021 19:57:54	27,50	27,31	28,38	0,44
16	27/05/2021 20:12:55	27,50	27,25	28,31	0,12
17	27/05/2021 20:27:55	27,44	27,25	28,25	0,75
18	27/05/2021 20:42:56	27,44	27,25	28,19	0,44
19	27/05/2021 20:57:57	27,50	27,31	28,25	0,75
20	27/05/2021 21:12:57	27,50	27,25	28,25	0,75
21	27/05/2021 21:27:58	27,50	27,31	28,31	0,44
22	27/05/2021 21:42:59	27,56	27,31	28,31	0,44
23	27/05/2021 21:57:59	27,50	27,31	28,31	0,44
24	27/05/2021 22:13:00	27,62	27,38	28,38	0,75

- Los datos son enviados directamente a Google Sheets y el sistema es simplificado ya que no se necesita un Reloj de Tiempo Real físico porque el sistema se encarga de obtener la fecha y hora por medio del servidor que usa para enviar los datos. Además, en el documento generado se pueden agregar gráficas para visualizar mejor el comportamiento de las variables que están siendo medidas.

Gráficas Generadas



Resultados, Recomendaciones y Próximos Pasos

Entre los resultados se obtuvo la realización de un sistema capaz de enviar datos al usuario en tiempo real de forma inalámbrica por medio de una conexión Wi-Fi.

Como recomendaciones, por la simplicidad de su conexión, al sistema se le pueden implementar aún más sensores y usar cables más largos para así poder tener un mejor monitoreo de los nidos o del vivero. Se puede implementar un máximo de 7 sensores de humedad y mínimo 15 de temperatura ya que en su módulo se puede conectar más de uno y el Arduino MKR1000 cuenta con 15 pines digitales.

Por último, como próximos pasos se puede adaptar el sistema para funcionar con el dispositivo Trinity y encontrar fuentes de alimentación ecológicas y efectivas para el sistema.

Referencias

Baker, J., Fish, M. y Drews, C. (2009). *Manual de monitoreo de temperatura: Guía para monitorear temperaturas de arena e incubación de anidación de tortugas marinas.* wwf.panda.org.

https://wwfint.awsassets.panda.org/downloads/manual_temperatura_resumen.pdf

WWF y BCG. (27 de Agosto del 2020). *Fires, Forest and The Future: A crisis ranging out of control?*. wwf.panda.org.

https://wwfeu.awsassets.panda.org/downloads/wwf_forestfiresreport2020_final.pdf

Bibliografía

Secretaría de Ambiente y Desarrollo Sostenible. (s.f.). *Incendios Forestales*. noticias-librodar.com.ar.

https://noticias-librodar.com.ar/index.php?option=com_content&view=article&id=870:incendios-forestales&catid=36:informacion-util-en-medio-ambiente&Itemid=56