

Guía Práctica - Transferencia de calor en aleta de cobre

Laboratorio de Propiedades Termodinámicas y de Transporte

Docente: Gustavo Adolfo Orozco Alvarado
gaorozcoa@unal.edu.co
Universidad Nacional de Colombia
Departamento de Ingeniería Química y Ambiental

1 Introducción:

El estudio de la transferencia de calor en superficies extendidas, como las aletas, es fundamental en la ingeniería térmica y de diseño de intercambiadores de calor. Una aleta es una extensión metálica cuya función es aumentar el área superficial disponible para disipar calor, y su comportamiento térmico puede modelarse mediante ecuaciones de conducción unidimensional y transferencia por convección.

Esta guía de laboratorio tiene como propósito que el estudiante pueda observar y analizar el perfil de temperatura en una aleta metálica utilizando sensores digitales DS18B20, integrados a un controlador ESP8266 con conexión Wi-Fi para registrar los datos en tiempo real en una hoja de cálculo de Google Sheets. El objetivo es comparar los datos experimentales con el modelo teórico, estimar parámetros térmicos y familiarizarse con la instrumentación electrónica.

2 Materiales y equipo:

Primero se presentará todo el material necesario para esta practica (Guías, códigos, links, etc) y posteriormente el equipo de laboratorio

2.1 Material de Apoyo

La gran parte de la información referida se encuentra en un repositorio para facilitar el acceso de manera pública a la información necesaria.

- [Se puede acceder al repositorio a través del siguiente link.](#)

2.2 Equipo de Laboratorio

Para el desarrollo de la practica se contó con diverso material que se presentará a continuación:

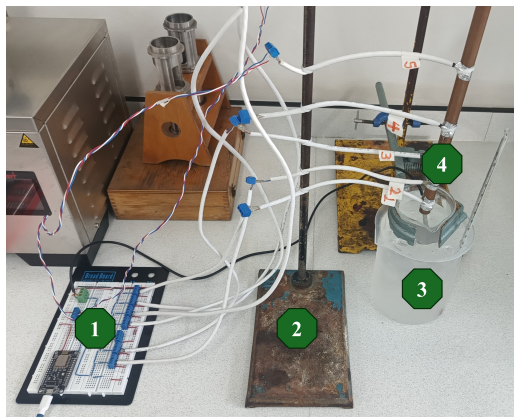
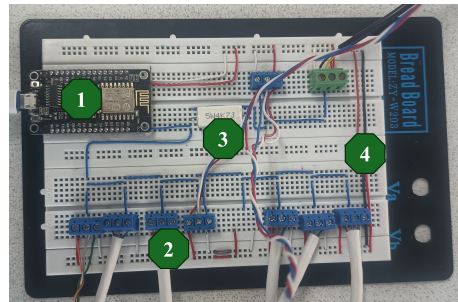
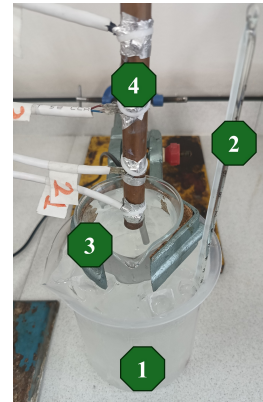


Figura 1: 1.Placa base 2.Soporte 3.Baño térmico 4.Aleta

Se puede apreciar el montaje general que se desarrolla a lo largo de la practica, consistiendo en una conexión con los sensores unidos a la aleta, la cual está absorbiendo o dispersando calor al ambiente (dependiendo de la temperatura del fluido de trabajo) y el baño térmico para mantener un control en las temperaturas. El montaje se encuentra suspendido gracias al conjunto del soporte y abrazaderas para la aleta.



(a) 1.Microcontrolador
sistencia 4.Placa Base
2.Sensores 3.Re-



(b) 1.Baño Térmico
2.Termómetro
3.Abrazadera 4.Aleta

Figura 2: Acercamiento

En la figura 2 se puede muestra más a detalle los componentes más relevantes del montaje para la práctica

3 Protocolo de medición:

En la practica se deben realizar preparaciones previas para llevar a cabo la práctica, estas se presentarán más adelante, junto con explicaciones adicionales y material de apoyo para la práctica.

3.1 Previo al Laboratorio

Los entornos de microcontroladores se suele trabajar en programas específicos, en este caso Arduino IDE. Sin embargo, para que Arduino reconozca, lea y compile en el microcontrolador que se usa en esta práctica, la ESP8266, se debe seguir una serie de pasos.

- **Instalar Drivers:** específicamente se necesitan dos drivers para que tanto la PC como arduino IDE reconozcan la placa.
 - **FTDI:** [link para instalación](#)
 - **CH340:** [link para instalación](#)
- **Instalar Bibliotecas** Arduino también requiere instalar una colección de bibliotecas, tanto para la lectura de los sensores, como para el manejo del microcontrolador. Se muestran a continuación
 - Plataforma ESP8266: en la sección de Archivo > Preferencias > Gestor de URLs Adicionales de Tarjetas, agregar: http://arduino.esp8266.com/stable/package_esp8266com_index.json
 - placa esp8266: En Herramientas > Placa > Gestor de tarjetas se debe instalar el paquete "ESP8266 by ESP8266 Community".
 - Onewire: En el gestor de bibliotecas buscar e instalar "OneWire de Paul Stoffregen".
 - DallasTemperature: También en el gesto de bibliotecas, instalar "DallasTemperature de Miles Burton"

El microcontrolador ESP8266 es el encargado de leer administrar y enviar los datos generados por los sensores. Para lograrlo se le debe cargar un código con las instrucciones adecuadas para ello. La gran parte de la estructura del código ya se encuentra completa. Sin embargo, Cada que se vaya a desarrollar una practica nueva se debe llenar unas secciones relevantes en el código, estas son las siguientes:

- **Credenciales WIFI**

En ellas se debe cargar tanto el nombre de la red WIFI a la cual se debe conectar el microcontrolador para enviar los datos a la nube, y también la respectiva contraseña de la red, en caso de que la red no posea contraseña, el espacio se deja vacío.

```
// 📶 WiFi Credentials
const char* ssid = "moto_g72_8732";
const char* password = "UJNyedv7841";
```

Figura 3: Ejemplo sobre cómo llenar la sección de credenciales

- **Direcciones de los sensores**

Cada sensor tiene un identificador único de tipo hexadecimal que fue obtenido previamente. Estos los usa el microcontrolador para nombrar y reconocer a qué sensor le pertenece su correspondiente dato de temperatura. Esta lista de direcciones se debe cargar en el código. En caso de reemplazar un sensor, se debe volver a obtener su identificador único y cargarlo al código.

```
// 📦 Direcciones HEX de los sensores
const char* hexAddresses[] = {
    "2861640B49BAFB7E", //Sensor 1
    "28E11548F6973CBE", //Sensor 2
    "2861640B4B38E7B1", //Sensor 3
    "2861640B4E1367EA", //Sensor 4
    "282BB748F65C3CC9", //Sensor 5
    "2861640B4E0E8827", //Sensor 6
    "286164098108F661", //Sensor 7
    "286164098105011F", //Sensor 8
    "2861640B4B372EFF", //Sensor 9
    "28ACB2833200121"   //Sensor 10
};
```

Figura 4: Ejemplo de listado para las direcciones

- **Dirección web de la nube**

Los datos deben ser almacenados en algún lado, puesto que la ESP no es capaz de almacenar tanta información por sí sola. En este caso se optó por crear un entorno de hoja de cálculo a la cual se puede acceder remotamente. En cualquier momento esta dirección se puede cambiar a una dirección web a su gusto.

```
// 🌐 Google Apps Script Webhook
const char* serverName = "https://script.google.com/macros/s/AKfycbyQSZIkZfSy1W0let3Mg";
```

Figura 5: Ejemplo de dirección web a donde se enviarán los datos

Una vez Ordenado estos tres aspectos principales, el código se encuentra en condiciones para ser cargado a la ESP. El procedimiento se realiza como un programa común en Arduino IDE. Una vez cargado el código al controlador, este permanece guardado dentro de la ESP, por tanto ya no es necesario volver a cargar el código, solo se requiere conectarle potencia al microcontrolador para que comience a buscar la red wifi para conectarse y cargar datos automáticamente.

Se puede comprobar que el código está funcionando en condiciones revisando el monitor serial de Arduino IDE para verificar que la ESP esté recibiendo los datos de todos los sensores; y al mismo tiempo revisar el entorno web para confirmar que la ESP se conecta a la red WIFI y envía los datos satisfactoriamente.

Ahora se repasarán varios casos adicionales que no involucran a la practica en sí, pero es de suma importancia conocerlos.

Calibración Sensores

Debido a la naturaleza de la fabricación de los sensores DS18B20, estos poseen un margen de error en sus lecturas, incluso márgenes de hasta $5^{\circ}C$, estos rangos son inaceptables para una practica de laboratorio el cual necesita cumplir ciertos requisitos de precisión. Por ello mismo se debe llevar a cabo una calibración de sensores.

$$T_{corregido} = m \cdot T_{sensor} + b \quad (1)$$

Para ello solamente basta con realizar un ajuste de parámetros simple para cada sensor. El comportamiento del aumento de temperatura ideal es lineal, por tanto el comportamiento que se plantea construir para cada sensor es el de una ecuación lineal de dos parámetros. Para determinar los parámetros ajustados experimentalmente, se comparará cada sensor con un patrón primario, como un termómetro de lectura confiable. Se tomarán los datos de lectura del sensor y se compararán con los datos del termómetro como valor real. Los parámetros ajustados para los sensores al momento del desarrollo de esta guía se encuentran en el repositorio.

Cabe destacar que los datos que se guardan en el repositorio se encuentran sin corregir, se debe aplicar los parámetros ajustados a cada cada sensor posteriormente para el tratamiento de datos.

Identificar HEX de los Sensores

En caso de tener que reemplazar algún sensor, además de recalibrarlo, se debe añadir su identificador único al código nuevamente para que el microcontrolador sea capaz de identificarlo. Para ello, se cuenta con un código sencillo para un controlador de tipo Arduino UNO el cual permite conocer fácilmente la dirección hexadecimal.

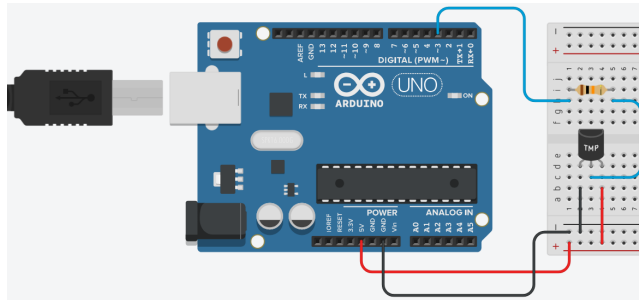


Figura 6: Esquema de conexión Arduino UNO

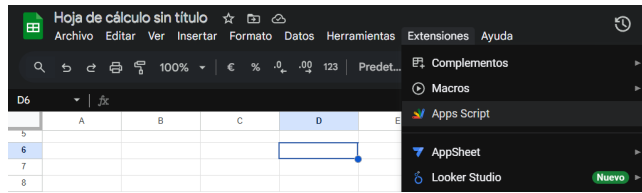
La figura 6 permite visualizar el montaje sencillo que se debe realizar a un Arduino para así obtener la serie de identificación para un sensor.

Crear el entorno en la Hoja de Cálculo

Siempre se puede cambiar la dirección final a la cual se reciben los datos, Se puede utilizar cualquier enlace con protocolo https para enviar y almacenar los datos, solo basta con crear el entorno web necesario para ello.

A continuación se muestra una guía rápida de cómo realizar el proceso, para este caso, crear una hoja de cálculo puesto que es la opción más rápida.

- En primer lugar se recomienda utilizar una cuenta de gmail no vinculada a la universidad, más adelante se explica el por qué.
- Una vez creada la nueva hoja de cálculo la cual será el destino de los datos, se debe cargar un código de appscript a la hoja. Para ello, en la hoja se debe dirigir a la opción de "Appscript" de las extensiones en la hoja de cálculo



- En esta nueva pagina se debe copiar el código necesario, el cual, automatizará la carga de datos a la hoja de cálculo (el código se encuentra en el repositorio).
- Posteriormente se debe dirigir a la sección de: Implementar > Nueva implementación > Seleccionar tipo > Aplicación web.



- Ahora se deben llenar los espacios con las debidas credenciales. En la sección "Ejecutar como" se debe establecer como "yo". Y en la sección "Quién tiene acceso se debe establecer como "Cualquier usuario". En este punto si se usa una cuenta universitaria, Appscript solo permitiría ingresar a cuentas vinculadas a la universidad, y el microcontrolador no tiene dicha capacidad de verificar ingresar con una cuenta.
- Procede a aceptar todos los permisos necesarios para permitir la implementación de appscripts
- Una vez completado, se mostrará un URL de la nueva implementación, esta será la dirección web que se ha de agregar en el código de la ESP.

Html para Graficar

Para visualizar los datos en tiempo real, su utilizó un codigo .html para cargar los datos directamente de la hoja de cálculo y así tener acceso a ellos desde cualquier dispositivo.

- Crear un nuevo repositorio en Github
- Subir el archivo .html
- Dentro del repositorio, ir a Settings > Pages
- En la sección "Build and deployment", seleccionar
 - **Source:** Deploy from a branch
 - **Branch:** main (o master) y carpeta /root
- Esperar unos segundos y GitHub entregará una URL como: https://TU_USUARIO.github.io/NOMBRE_DEL_REPOSITORIO/

Lo anterior permitirá visualizar los datos compilados de la hoja de calculo en tiempo real. Cabe destacar que el código html tiene por predeterminado el ID de la hoja de cálculo original, Si se quiere cambiar la hoja de calculo a la cual se quiere visualizar sus datos, se debe cambiar la dirección ID en el código html.

3.2 En el Laboratorio

Antes de iniciar la medición, se deben verificar minuciosamente todas las conexiones eléctricas entre los sensores DS18B20, las borneras, la protoboard y del microcontrolador. Es indispensable comprobar que cada cable esté correctamente insertado, que los tornillos de las borneras estén firmemente ajustados, y que los colores de los conductores coincidan a lo largo de la conexión (rojo o naranja para VCC, blanco para GND y azul o verde para DQ). No deben existir filamentos expuestos ni interferencias entre canales. Una vez verificado el montaje eléctrico y estructural, se procede a energizar el microcontrolador mediante una fuente de 5 V por conexión USB. El controlador ha sido previamente programado para conectarse automáticamente a una red Wi-Fi específica y transmitir las lecturas de los sensores a una hoja de cálculo de Google Sheets. Se recomienda confirmar la correcta conexión a la red y la visualización en tiempo real de los datos en la hoja digital. Para realizar la prueba térmica, se prepara el sistema según el tipo de condición deseada:

- En **condición de enfriamiento**; se emplea un beaker pequeño con agua destilada, sostenido por una pinza en un soporte universal. Este beaker se introduce dentro de otro más grande con una mezcla de hielo y agua evitando así gradientes bruscos o interferencias por choque térmico. Se debe monitorear y realizar extracción de agua con una jeringa y adición de hielo periódicamente con el propósito de mantener una temperatura próxima a 0°C. Adicionalmente, se recomienda registrar la temperatura del baño externo con un termómetro adicional para mayor control.
- En **condición de calentamiento**; se utiliza un único beaker con agua colocado sobre una placa calefactora. Este beaker debe ser cubierto con una tapa de icopor o papel aluminio, material que actúa como aislante térmico y barrera física frente al vapor. Esta cubierta contribuye a minimizar el flujo convectivo ascendente y la condensación sobre la superficie de la aleta, condiciones que podrían alterar la convección natural del entorno y generar perturbaciones en el perfil térmico medido. Durante el ensayo, se debe adicionar agua caliente de manera periódica para compensar las pérdidas por evaporación y asegurar que el extremo inferior de la aleta permaneciera constantemente sumergido, garantizando una condición de contorno estable en la base, fundamental para preservar la validez del modelo empujado.

Con el sistema en funcionamiento y estabilizado, se inicia la adquisición de datos. La lectura se mantiene durante un intervalo de tiempo suficiente (30 a 45 minutos) para asegurar la llegada al régimen estacionario o estudiar el comportamiento transitorio. Al finalizar la prueba, se interrumpe la alimentación eléctrica de la placa, y los datos se descargan desde la hoja digital para su posterior análisis.

4 Problemas frecuentes:

Por lo general, se suelen presentar errores de conexión de los sensores, típicamente reflejados como "Nan" o el valor "85" en la hoja de datos de los sensores. Esta puede ser causada por diferentes razones, en esta sección se repasarán las más comunes.

1. Una mala conexión en la protoboard suele ser el problema más común, tanto que algún cable se haya desconectado parcialmente, como que directamente se encuentren conectados de forma incorrecta. Se soluciona típicamente volviendo a revisar todo el cableado, y volviendo a conectar todo el montaje.
2. Sensor dañado. Generalmente para identificar algún fallo del sensor se suele verificar midiendo el voltaje del sensor con un multímetro, así logrando el punto donde se genera el corto o si directamente alguna pata del sensor se encuentra defectuosa. Lo recomendable suele ser reemplazar el sensor, con el trabajo que lleva a cabo desmontar, calibrar, identificar y volver a montar el nuevo sensor.
3. Puede que en la lista de Identificación HEX No se encuentre la lista completa de los sensores y por tanto el microcontrolador no identifica uno o varios de los sensores.
4. Se genera una conexión cruzada entre varios sensores generando un corto y perdiendo la información de varios sensores.

Por lo general el driver convertidor de USB a serie "CH340" suele presentar problemas al momento de compilar y cargar datos a la ESP. Se suele solucionar desinstalando y reinstalando nuevamente el driver para que Arduino IDE vuelva a identificar el puerto COM de salida del controlador.