

INFORME DE LABORATORIO III

Autores: Kevin David Martinez Zapata, Sebastián Bernal Cuaspa

*Laboratorio de Electrónica Digital 3
Departamento de Electrónica y de Telecomunicaciones
Universidad de Antioquia*

Resumen

Se propuso a realizar un sistema de telemetría que permitiera medir las variables básicas para el control de un drone con una IMU y un GPS, y, luego mandar esa información a una estación base por medio de wifi. Se logró calibrar todos los sensores y se logró transmitir desde el sistema hasta la estación base todos los datos necesarios para hacer después un análisis de los resultados mostrándolos en graficas. Aunque el sistema funciona en su totalidad, hay una limitante de alcance ya que depende de con que se cree la red para enlazar a los dispositivos.

Palabras clave: Telemetría, IMU, GPS, transmisión de información.

Introducción

La telemetría es un conjunto de tecnologías que se utilizan para la medición y transmisión de datos a distancia. Esta técnica se aplica en diversas áreas, como la industria, la aviación y la exploración espacial. En el caso de los drones, la telemetría es fundamental para monitorear su posición, velocidad, altitud, temperatura y otros parámetros importantes.

La telemetría se aplica a los drones de diversas maneras, desde la transmisión de datos a través de señales de radio hasta el uso de sistemas de satélites para la comunicación y el seguimiento de los drones.

Para llevar a cabo el sistema de telemetría es necesario contar con un microcontrolador, los

sensores necesarios y el módulo que permita transmitir la información, en este caso wifi.

Marco teórico

El microcontrolador usado en este caso es una Raspberry pi pico. El pinout de esta es el siguiente¹.

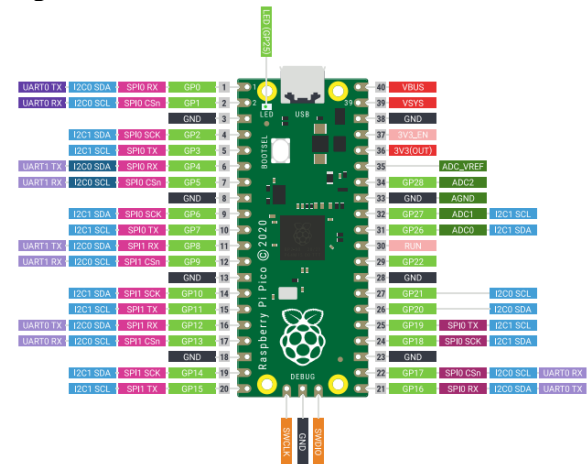


Fig 1. Pinout de la Raspberry pi pico.

Se van a usar los puertos uart0 y uart1 para conectar la IMU y el GPS respectivamente, ambos módulos se alimentan con 3.3V de la Raspberry y funcionan a 9600 baudios la IMU y el GPS, y, 115200 baudios el módulo wifi.

Para la IMU es necesario tener en cuenta que los valores que arroja deben ser divididos por 2^{15} ya que el ADC del módulo es de 16 bits y multiplicado por el rango. Para el modulo GY-91 se tiene que:

$$dato = \left(\frac{sensorValue}{2^{15}} \right) \cdot rango \quad (1)$$

- Rango Acelerómetro: $\pm 2g$, $\pm 4g$, $\pm 8g$, $\pm 16g$
- Rango Giroscopio: $\pm 250\text{Grad/Seg}$, $\pm 500\text{Grad/Seg}$, $\pm 1000\text{Grad/Seg}$, $\pm 2000\text{Grad/Seg}$
- Rango Magnetómetro: $\pm 4800\mu T$
- Conversor AD: 16 Bits (salida digital)
- Grados de libertad (DoF): 9

De la misma manera, el modulo ESP8266 que es el GPS se tienen las siguientes características:

- Voltaje alimentación: 3-5V
- Interfaz: UART (Serial)
- Precisión de posicionamiento 2.5 MCEP
- Tamaño de la antena: 25x25x08mm
- Tasa de baudios: 9600 bps

Finalmente, se tiene el modulo wifi que va a servir para transmitir los datos recogidos por los módulos anteriores. Las características del ESP8266 son las siguientes:

- Utiliza una CPU Tensilica L106 32-bit
- Voltaje de operación entre 3V y 3,6V
- Corriente de operación 80 mA
- Temperatura de operación -40°C y 125°C
- Soporta IPv4 y los protocolos TCP/UDP/HTTP/FTP

Procedimiento experimental y resultados

Como ya se dijo anteriormente, se va a usar una Raspberry pi pico para este proyecto. La conexión de la IMU a la Raspberry se muestra a continuación en la **figura 2**. Los pines ADO y NCS de la IMU se conectan a los pines GP6 y GP7 de la Raspberry pi pico respectivamente, de la misma forma, los pines SCL y SDA se conectan a los pines GP9 y GP10 de la Raspberry pi pico respectivamente³.

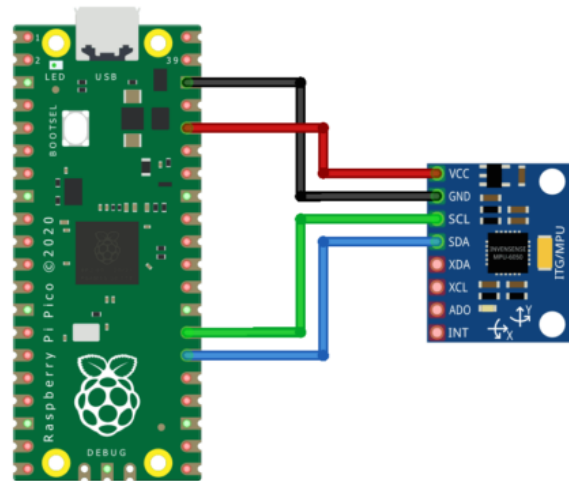


Fig 2. Conexión de la IMU al microcontrolador.

De la misma forma, se muestra la conexión del módulo GPS al micro en la **figura 3**. Los pines RXD y TXD de van a conectar a los pines GP16 y GP17 de la Raspberry pi pico respectivamente. En la figura se muestra una Raspberry diferente pero la conexión indicada en este caso es para la Raspberry pi pico. Procure ver el pinout del micro en la **figura 1**.

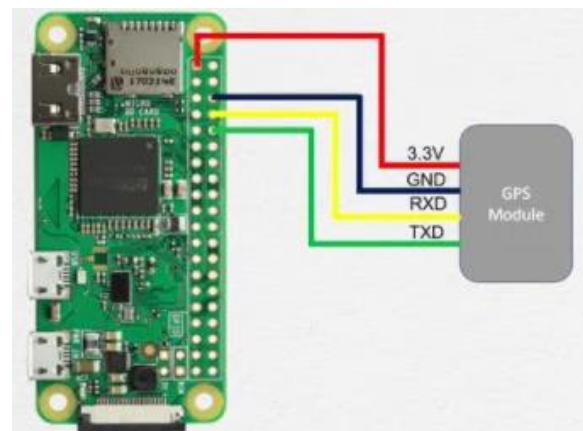


Fig 3. Conexión del módulo GPS al microcontrolador.

Una vez se tienen montados todos los sensores se procede a hacer la calibración de estos, en especial de la IMU. Se corre un programa en C que calcula los valores de offset de cada uno de los sensores: acelerómetro, giróscopo y magnetómetro. Estos valores de offset se llevan a programa principal para así evitar el proceso de

calibración en tiempo de ejecución. Los resultados se muestran a continuación en la **figura 4**.

```
Offset accl. X=1, Y=8, Z=9
Offset mag. X=1, Y=4, Z=0
Offset gyro. X=-11, Y=-6, Z=-15
PS C:\Users\martinez\Downloads\Pico-MPU-9250-main>
```

Fig 4. Valores de calibración para los sensores de la IMU.

El siguiente paso, una vez verificado que la IMU y el GPS funcionan correctamente en conjunto esto por medio del puerto serial, se procede a montar el módulo wifi. Como ya se tienen todos los datos es simplemente mandar toda la trama de datos separa por comas para finalmente en la estación base guardar los datos como un archivo CVS para posteriormente graficar los datos usando un programa de Python.

La conexión del módulo wifi a la Raspberry es la siguiente^{4,5} que se muestra en la **figura 5**.

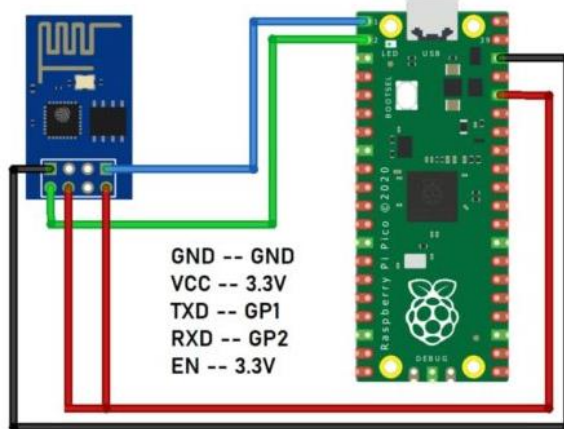


Fig 5. Conexión del módulo wifi ESP8266 a la Raspberry.

Luego de tener todo el montaje listo se procedió a hacer las pruebas pertinentes, en primera instancia se evidenció que el alcance la transmisión es muy corto debido que se estaba generando la red con un celular, lo cual tiene una fuerte limitante en cuanto a distancia.

Las pruebas se hicieron con esta corta distancia para probar que todos los sensores funcionaran

correctamente. Las **figuras 6 y 7** muestran los resultados de la IMU y del GPS respectivamente.

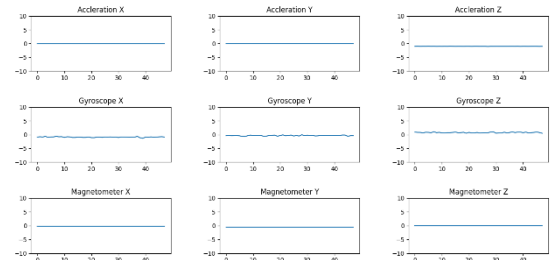


Fig 6. Resultados de las pruebas de la IMU.



Fig 7. Resultados de las pruebas del GPS.

Se puede notar que los resultados obtenidos por la IMU, mas específicamente el acelerómetro muestra una correcta calibración ya que muestra aproximadamente 0 en los ejes X e Y, y, muestra aproximadamente 1 en el eje Z, o sea, 1g que significa una vez la gravedad.

Para el GPS se tienen buenos resultados también ya que las pruebas que se hicieron en el bloque 18 de la Universidad de Antioquia muestran que los datos, aunque tengan una pequeña variación, no se salen mucho de la zona de prueba.

Discusión de resultados

La transmisión resultó ser exitosa usando wifi a pesar de que existen muchas otras técnicas como radio frecuencia. Este método resultó ser muy estable en este caso y permitió mandar una gran cantidad de datos en una sola trama, técnicamente se pudieron mandar todos los datos necesarios de la IMU que eran nueve y los dos datos de latitud y longitud del modulo GPS, o sea, una trama bastante larga con éxito total y estabilidad. En la **figura 8** se muestra el proceso de transmisión de información.

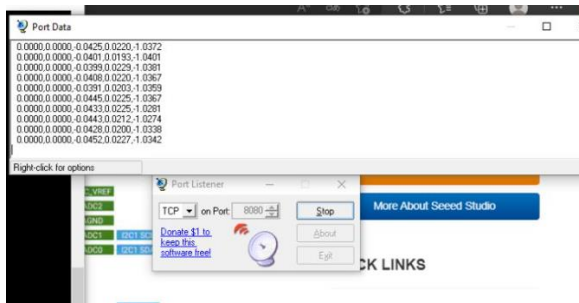


Fig 8. Prueba de funcionamiento de la transmisión usando el programa Listener para leer la información que pasa por el puerto 8080 que es por donde se comunica el sistema con la estación base.

A continuación, en la **figura 9**, se muestra la tabla de datos obtenidos de la transmisión por medio del modulo wifi a la estación base.

```
Latitud,Longitud,accx,accy,accz,gyroX,gyroy,gyroz,magX,magY,magZ
6.2676,-75.5676,-0.0269,0.0093,-1.0311,-0.8850,-0.3891,0.8926,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0311,0.0098,-1.0283,-0.7706,-0.3433,0.7935,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0323,0.0059,-1.0386,-0.8621,-0.3967,0.7782,-0.1465,-0.5859,0.0000
6.2675,-75.5676,-0.0296,0.0044,-1.0255,-0.6409,-0.3738,0.6332,-0.1465,-0.5859,0.0000
6.2675,-75.5676,-0.0323,0.0100,-1.0306,-0.9842,-0.3510,0.8316,-0.1465,-0.5859,0.0000
6.2675,-75.5676,-0.0274,0.0066,-1.0169,-0.8316,-0.4272,0.7935,-0.1465,-0.5859,0.0000
6.2675,-75.5676,-0.0316,0.0100,-1.0318,-0.8240,-0.6790,0.6790,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0296,0.0061,-1.0250,-0.6409,-0.6866,1.0376,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0301,0.0054,-1.0413,-0.7858,-0.3891,0.7019,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0298,0.0066,-1.0301,-0.7401,-0.1755,0.8011,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0267,0.0078,-1.0381,-1.0300,-0.3128,0.6332,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0258,0.0120,-1.0286,-0.7782,-0.2899,0.5722,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0269,0.0063,-1.0355,-0.8163,-0.2518,0.7019,-0.1465,-0.5859,0.0000
6.2677,-75.5675,-0.0284,0.0083,-1.0176,-1.0452,-0.3433,0.7019,-0.1465,-0.5859,0.0000
6.2677,-75.5676,-0.0330,0.0093,-1.0279,-1.0376,-0.6180,0.7782,-0.1465,-0.5859,0.0000
6.2677,-75.5676,-0.0306,0.0063,-1.0386,-0.9384,-0.5188,0.8926,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0269,0.0093,-1.0306,-0.9613,-0.2975,0.6790,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0333,0.0085,-1.0328,-1.0376,-0.2289,0.7172,-0.1465,-0.5859,0.0000
6.2676,-75.5676,-0.0316,0.0122,-1.0389,-1.0147,-0.1907,0.8240,-0.1465,-0.5859,0.0000
```

Fig 9. Archivos CVS con los datos almacenados de la transmisión del sistema.

Los datos están especificados por columnas, se puede evidenciar que los datos de la columna 1 y 2 que son del GPS se mantienen bastante estables, la variación es breve ya que las mediciones se

hicieron en una habitación debido a limitante de distancia del router, en este caso un celular.

En cualquier caso, la transmisión fue totalmente exitosa y muy estable, las tramas llegan siempre y se pueden transmitir bastante largas pero la limitante está en la distancia. Se debe optar por usar un elemento adicional como un repetidor de señal en el punto central a donde se van a hacer las pruebas para garantizar el largo alcance y que la transmisión no se vea interrumpida.

Conclusiones

En la práctica es esencial el conocimiento del datasheet de los componentes para realizar una calibración correcta, a partir de los datos y las mediciones de los sensores leídas, dicha calibración se realiza una vez para no sobrecargar al programa y dejar que fluya lo más rápido posible.

Los puertos UART0 y UART1 son muy importantes debido a que ellos son el puente para comunicar datos, como lo es datos de sensor y datos de Wifi, estos puertos UART proporcionan una forma estándar de comunicación serial asincrónica, lo que significa que se pueden utilizar para intercambiar datos entre dispositivos de manera confiable. La comunicación UART es ampliamente utilizada en una variedad de aplicaciones, como comunicación entre microcontroladores, periféricos y otros dispositivos electrónicos. Los puertos UART pueden admitir diferentes velocidades de transmisión de datos, lo que permite una flexibilidad en la configuración según las necesidades del sistema. Además, ofrecen una buena capacidad para la transmisión de datos a distancias moderadas, lo que los hace adecuados para aplicaciones tanto locales como de larga distancia, es por esto por lo que el GPS es enviado por UART0 y el WIFI por UART 1.

El wifi es una tecnología ampliamente adoptada y de fácil acceso. Esto hace que la implementación de sistemas de telemetría basados en wifi sea más sencilla en diferentes entornos, sin embargo, hay una gran limitante que

es el alcance que esta puede llegar a tener, Esto puede dificultar la transmisión de datos en áreas de cobertura limitada o en entornos con muchas obstrucciones, como paredes o interferencias electromagnéticas.

Bibliografía

- [1] Ground, P., Uart, G. U. /., & Uart, U. /.. (s/f). Raspberry Pi Pico Pinout. Raspberrypi.com. Recuperado el 8 de mayo de 2023, de <https://datasheets.raspberrypi.com/pico/Pico-R3-A4-Pinout.pdf>
- [2] (S/f-b). Raspberrypi.com. Recuperado el 8 de mayo de 2023, de <https://datasheets.raspberrypi.com/pico/getting-started-with-pico.pdf>
- [3] (S/f-a). Electronilab.co. Recuperado el 8 de mayo de 2023, de <https://electronilab.co/tienda/tarjeta-gy-80-imu-10dof/>
- [4] Connect a Raspberry Pi Pico with Ubidots using an ESP8266. (s/f). Ubidots.com. Recuperado el 8 de mayo de 2023, de <https://help.ubidots.com/en/articles/5097358-connect-a-raspberry-pi-pico-with-ubidots-using-an-esp8266>
- [5] Kenttä, J. (s/f). Pico-ESP8266-C: Setup the Pico-ESP8266 Wifi expansion module with Raspberry Pico.