

Laboratorio transmisión y recepción de datos de la stm y curva del motor

Juan Sebastián Herrán Páez & Daniel Jimenez

I. RESUMEN

En este informe se presenta el diagrama de flujo del proyecto completo de stm32cubeIDE, teniendo nuevas funciones para enviar y recibir datos, así mismo la curva generada en python que representa las RPMs en el tiempo desde que se ejecuta el programa en python y los diagramas de flujo de python y qt para la app.

II. MATERIALES

- **STM32F411CEU6:** Microcontrolador base utilizado para el laboratorio.
- **Motoreductor DC:** Utilizado para ensamblar al cd y generar las RPM.
- **Placa de Prototipado:** Placa de circuito impreso para realizar conexiones temporales entre los componentes electrónicos.
- **Sensores de herradura:** Utilizados para detectar los pulsos y desarrollar los respectivos calculos a partir del cd.
- **Pantalla OLED:** Pantalla programable para mostrar los datos importantes y visualizar los cambios.
- **Cables de Conexión:** Cables eléctricos para realizar conexiones entre los diferentes componentes del circuito.
- **Fuente de Alimentación:** Fuente de alimentación de corriente continua para alimentar el motor.
- **Python:** Programa de alto nivel utilizado para generar la gráfica del motor.

III. DIAGRAMAS DE FLUJOS Y CURVA DE MOTOR

Se presenta el diagrama de flujo del proyecto con el programa STM32cubeIDE, en este se encuentra las variables, funciones creadas e interrupciones utilizadas en el proyecto de stm32cubeIDE

1. Curva del motor

Teniendo en cuenta el montaje del laboratorio, se presenta la grafica de la curva de velocidad del motor.

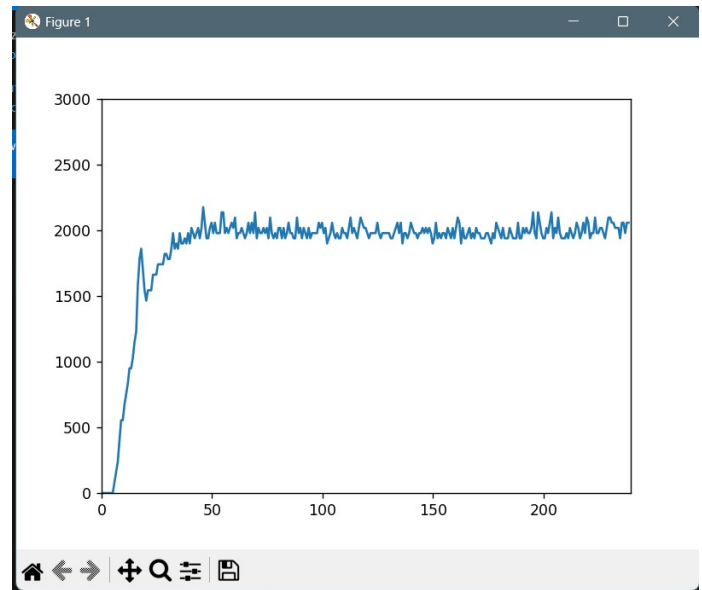
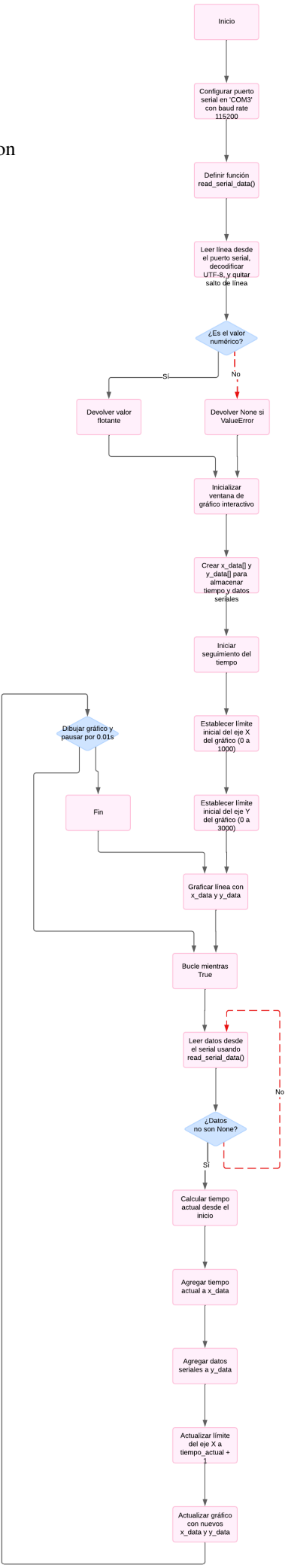


Fig. 1: Curva RPM motor

Se puede identificar por medio de la grafica que a traves del tiempo se estabiliza la velocidad por lo que esa es la velocidad máxima a la que llega el disco con un voltaje de 3.3v.

Se presenta el diagrama de flujo del archivo de python para generar la grafica de la curva del motor.

Fig. 2: Diagrama de flujo Python



Se presenta el diagrama de flujo del archivo del STM32CubeIDE para enviar y recibir los datos por el circuito realizado.

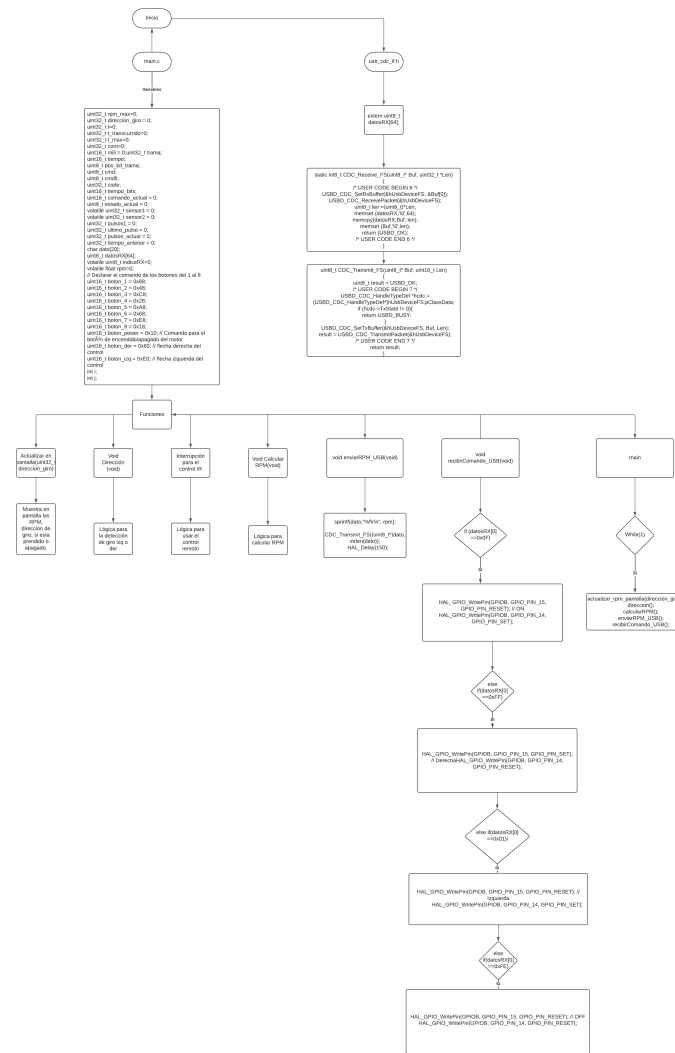


Fig. 3: Diagrama de flujo stm32cubeIDE

Se presenta el diagrama de flujo del archivo de qt para enviar y recibir los datos por medio de la app diseñada.

IV. CONCLUSION

Mediante el uso de un cable USB se pudo hacer la comunicación serial entre la stm32F411CU6 y el computador para generar la gráfica en python utilizando funciones ya creadas en el archivo usbdcdcf.c.

Las variables más importantes en este laboratorio son *dato[20]* y *datosRX[64]* puesto que son las variables utilizadas como buffer para enviar y recibir datos respectivamente con la función de transmisión del archivo usbdcdcf.c y la función de recepción creada para la recepción de datos.

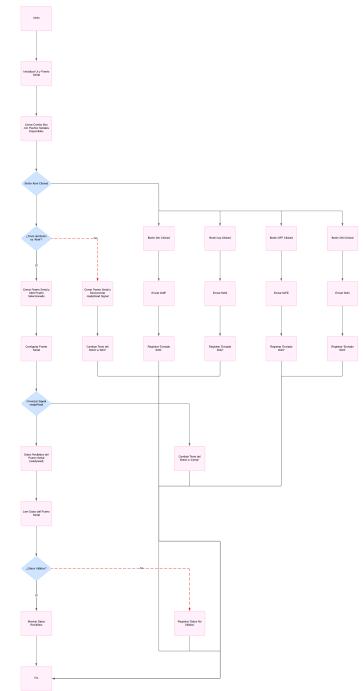


Fig. 4: Diagrama de flujo qt

Gracias a las herramientas como Qt se pudo realizar el laboratorio de una manera más didáctica, haciendo que con el uso de botones y con tablas podamos controlar el motor y monitorear la velocidad del mismo de una manera más fácil.

Se pudo realizar el laboratorio con éxito, resaltando lo práctico que es hacer el widget en el programa QT y también el uso de herramientas que nos ayudaban a rectificar el envío y recepción de datos como el software HERCULES.

Se pudo rectificar el funcionamiento de las funciones utilizadas y del cálculo de correcto de las RPM, esto con ayuda de la gráfica hecha en Python, esto mostrando un comportamiento real del motor al momento de accionar cualquier botón.