Contributions

Team Members: Gabriel Sosa, Mahdi Mazloumi and Randy Herrera

Phase 1:

- Initial research and testing
  - Researching different IDEs (VS Code PlatformIO, Arduino, Espressif)
  - Successfully flashed “blink” program through Espressif and Arduino IDE
  - Learned how to manage manage libraries in Arduino IDE
  - All of the above in PlatformIO, including making changes to PlatformIO configuration files (platform.ini)
  - Understanding how a lathe works as no member had experience in CNC
  - Understand how to use the devices peripherals such as (OLED, bluetooth, multidirectional toggle switch)
  - Research on how embedded systems work
  - Understanding how to read the schematic
- **Contributors: (Gabriel, Randy, Mahdi)**

Phase 2:

- **(Mahdi)**
  - Created Asana Task Board
- (**Gabriel)**
  - Created LED Strip Control program using FastLED.h
    - Creating chasing effect (rainbow pattern)
- (**Gabriel & Randy**)
  - Successfully Connected appropriate GPIO pin with boards display and LED strip
  - Successfully created test program to display text and images on OLED display
- (**Gabriel, Mahdi, Randy**)
  - Discovered the use for serial ports as each connect to an individual axis (X,Y,Z)

Phase 3:

- **(Gabriel)**
  - Implemented test web server using CGI
- **(Randy)**
  - Gained eduroam access using boards Wifi capabilities using https://github.com/martinius96/ESP32-eduroam **repo**

- **(Mahdi)**
    - Testing and troubleshooting wifi (home and eduroam)

- **(Gabriel & Randy)**
    - Tested flashing the same program on Arduino,Platformio and Espressif to compare how they compile
    - Found that Arduino is the easiest IDE to start with as it has lots of community support and easy to use
    - Utilize libs such as (Wire.h, FastLED.h, Wifi.h, Adafruit_SSD1306.h & Adafruit_GFX.h) as it taught us how to leverage existing resources
    - Gained a better understanding of why C/C++ is a foundational language as it is used widespread for firmwares and drivers (memory, precise control, processing power, ability to interact with hardware peripherals)
    - Learned how the jumpers work to emulate lathe
    - Created a functional Menu system using the boards multi directional toggle switch and boards Display

---

Phase 4 :

- **(Mahdi)**
    - Managing Asana and researching OLED programs

- **(Gabriel)**
    - Researched the Binary Principle
    - Utilized two outward signals (pins) to provide direction and position feedback
    - Gained access to the the boards encoder and created test programs in combination with OLED display
    - Implemented LED toggle ability through wifi using web server using /on, /off GET request

- **(Gabriel & Randy)**
    - Used Boards wifi capabilities to successfully connect to eduroam but still dealing with connectivity issues

---

Phase 5:

- **(Gabriel)**
    - Transitioned from using CGI to Flask Application for back end
- **(Gabriel , Randy , Mahdi)**
    - Migrated Arduino program to Platformio
    - Tested created

        - Troubleshooting eduroam and discovered it uses a certificate
        - Research how DRO’s are designed to start comping up with prototypes of the GUI

Phase 6:

- **(Gabriel)**
    - Split code into different files
    - Implemented Individual tasks for WIFI connection
    - Further research for encoder to further increase its accuracy
    - Research Z pin use case
    - Redesigned Menu System with navigation ability via multidirectional switch
- **(Randy)**
    - Hosted Flask server on pythonanywhere

  (Gabriel & Randy)
    - Communication established between ESP32 and Flask server
    - Translated CGI routes to Flask
    - Tested hosting flask server online

Phase 7:

- **(Gabriel)**
    - Updated Website by being able to send X and Y coordinate values every few seconds or via button
    - Redesigned the way we handles our pins by creating an Encoder struct
    - Improved Encoder by implementing Debouncing removing unwanted noise
    - Adapt dynamically based on Speed
- **(Gabriel & Randy)**
    - Worked on formatting data sent via GET requests as JSON
    - Updated JavaScript on web interface to request and process JSON
    - Wrote request using AJAX to display data on web interface

- **(Gabriel, Mahdi, Randy)**
    - Initial discussions regarding UX design of web interface

---

Phase 8:

- **(Gabriel)**

    - Fully transitioned project from Python Flask to AsyncWebServer library (server-hosted app -> embedded web server on device) Reducing complexity of firmware
- **(Randy)**
    - Researching ToAuto DRO and relevant user manual
    - Created several use cases using user manual
- **(Gabriel & Randy)**
    - Rewrote our routes from Flask to accommodate for AsyncWebServer

---

Phase 9:

- **(Mahdi)**
    - Researching the design of UI to make the best web design possible
    - Implementing user accessibility
    - Managing github
- **(Gabriel)**
    - Designed GUI for web server
    - Added ability to toggle between ABS/INC coordinates
    - Added functionality to zero out coordinates
    - Modified Randy's functionality to toggle between MM/INCH in either mode ABS/INC
    - Updated the boards display to accurately show our coordinates in any combination of ABS/INC or INCH/MM
- **(Randy)**
    - Added functionality to toggle between MM/INCH mode
    - Researched midpoint functionality on a DRO such as use case and implementation for the board
    - Researching use case for F1-F6 buttons
    - Researched and implemented calculator feature

Phase 10:

- **(Gabriel)**
    - Added ability to use multiple coordinate planes, storing each ABS/INC values independently for each plane
    - Added ability to toggle between planes/setups using the boards multi-directional switch
    - Added functionality to save current Coordinate positions
    - Display the saved coordinates
- **(Randy)**

        - Added ability to dynamically change the function F1-F6 buttons to calculator mode to avoid overwhelming UI

- **(Mahdi)**
    - Modifying the GUI design and the web server for better user accessibility

---

Phase 11:

- **(Gabriel)**
    - Refactored Code
    - Fixed Bugs after implementing Coordinate Planes feature
    - Finalized GUI
- **(Randy)**
    - Added feature to input user defined factors for millimeter and/or inches
- **(Gabriel , Randy , Mahdi)**
    - Collaborated on Presentation
    - Collaborated on README file
    - Collaborated on Slides