

Mohamed Hawas

Embedded Software Engineer

☎ +201009922839 ✉ muhamed.muhamed.hawas@gmail.com in mohamed-hawas0
🌐 Mohamed-Hawas 🧑 Military Status : Exempted 📧 @muhamed-hawas

Education

BE Faculty of Engineering, Alexandria University
Electronics and Communication Engineering
Graduation Project Grade : A+

Graduate in 2024

Experience

nfrti X, IoT & Embedded Software Engineer, Intern

(On-Site - 2 months)

- Worked on embedded solutions for **predictive maintenance systems**.
- Developed and optimized embedded software for Nordic nRF5340
- Integrated a **26.7 kHz MEMS accelerometer** for vibration analysis.
- Debugged the **official sensor driver** to ensure full functionality.
- Developed **BLE** communication between Nordic and Raspberry Pi.
- Integrated **MQTT** protocol between Raspberry Pi and platform.

Projects

ADAS Algorithms 🎥 [Video Demo 📄]

(GP Mentored By **Valeo**)

- Implemented four features in ADAS

- Auto Parking - Reversing Assistant - Personal Parking - Collision Avoidance

- Secured **5th place** 📄 at Valeo Tech Talents Demo Day.
- Led the algorithm team in developing features and designing the architecture of the application layer.
- Embedded Systems and ARM Architecture Expertise :
 - Implemented **MCAL drivers** (RCC, GPIO, EXTI, NVIC, TIMER, SYSTICK, UART, I2C, SPI, FPU)
 - Developed **HAL drivers** (Motor, Encoder, Servo, Ultra Sonic, Compass, IMU, EEPROM)
 - Facilitated interfacing with the application layer using the implemented MCAL & HAL APIs.
- Gained practical knowledge in **Autonomous Solutions**
 - Implemented an **Odometry** 📄 module to localize the vehicle position with an accuracy of **90%** using IMU and Encoder sensors
 - Implemented a **Motion Planning** module to detect allowable areas for parking and plot the path from pos-ok to target in real-time execution.
 - Implemented a **Vehicle Tracking** module to enable the vehicle to follow the path using the **PID Controller**

OOP - Library Management System 🎥 [Video demo 📄]

- Developed a Library Management System in **C++** with book and user management.
- Implemented features like search, rating, inserting, deleting, and **dynamic terminal interface**.

Balancing Ball System with PID Controller [Video demo].

- Derived the system **transfer function** and analyzed it through **step response** plots to assess stability.
- Designed a PID-controlled system on **STM32**, integrating ultrasonic sensors and servo motor.
- Applied **low-pass filtering** for stability, reducing noise, and achieving smooth ball positioning.

IoT - Soil Monitoring System [Video demo].

- Developed a wireless soil monitoring system using an ESP32 microcontroller, integrating a 7-in-1 **Modbus** sensor.
- Collected key soil metrics (Nitrogen, Phosphorus, Potassium, conductivity, pH, humidity, and temperature) via **RS485 communication**.
- Transmitted real-time data to **ThingSpeak** using WiFi for remote monitoring and analysis.

CMake

- Developed a CMake file to make the project independent of the operating system
- Generalized the CMake file to be suitable for any project using the **Arm toolchain**.
- Used **st-flash** program to flash the code to the board.
- Added a section for **GDB** debugging to enable debugging from the terminal.

Startup, Linker and Makefile Implementations

- Created startup and linker scripts and automated build processes using Makefile.
- Gained in-depth knowledge of the build process and booting sequence.

Socket Programming

- Developed TCP and UDP client-server applications in Python.

Computer Architecture Labs

- **32-Register File in VHDL**
- **Reorder Buffer**
- **Using GPU in Matrix Operations**

Operating Systems Labs

- **Simple Shell**
 - Developed a UNIX shell in C capable of executing basic commands
- **Multi-Threading**
 - Achieved concurrent matrix processing via thread-per-row, and thread-per-element methods.

Technologies & Tools ---

Languages: C, Assembly, Bash Script, familiar with Java and python

Debugging: GNU GDB

Software Design: TTool , Draw.io , UML

Documentation: LaTeX , Markdown , MS Office

Others: Arm Toolchain , Make , Linux

Languages ---

Arabic : Mother Tongue

English : Excellent