Final Project Report

Robotic Arm

Members:

Kody Byrd
Luis Mata Moreno
Michael Stevenson

CDA 3631 Section 1

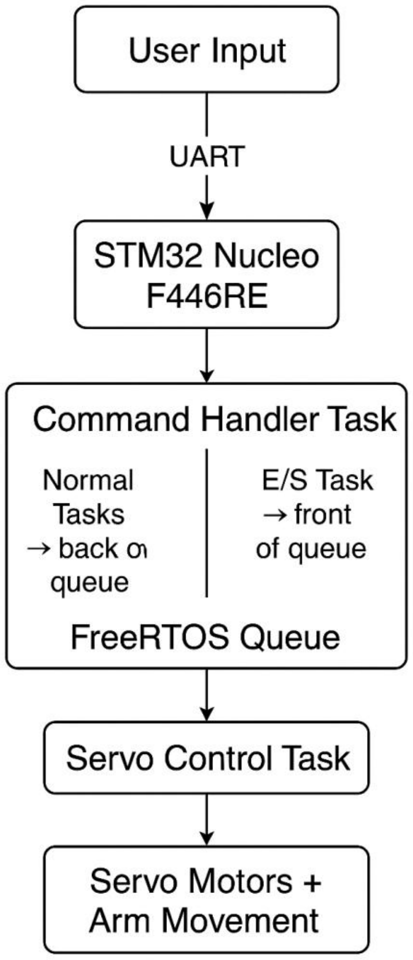
Introduction

For our Final Project, we designed, coded, and implemented a robotic arm system controlled by an STM32 Nucleo-F446RE board running FreeRTOS. The robot can perform five tasks: left rotate, right rotate, rotate to center, emergency stop, and start, which are triggered by keys sent over UART from a laptop. Incoming tasks are added to a queue and executed in order. The controls are the following: 'l' triggers the left rotate task, 'r' triggers the right rotate task, the spacebar triggers the rotate to center task, 'e' triggers the emergency stop task, and 's' issues the start task. When a rotate task is received, the arm first moves to the specified position (if not already there), then grabs the object. The following rotation task moves the arm to the specified position (if not already there), then places the object. The arm will repeat the process of grabbing and then placing the object based on the rotation task it receives. The emergency stop task is inserted at the head of the queue and immediately disables all PWM signals, stopping the arm. Similarly, the start task is placed at the head of the queue and re-enables all PWM outputs, allowing the arm to resume operation.

Materials

- PLA plastic for the 3D printed arm parts
- Power supply
- Three MG996R Servos
- Three SG90 Micro-Servos
- STM32 Nucleo-F446RE board
- Breadboard
- Assorted jumper wires
- 2 x 6 wood board (robot base)

Methods



The system architecture consists of three primary elements: the STM32 microcontroller, the Servo Control Task, and the user input interface. The keys the user enters are transmitted to the STM32 via UART, where a FreeRTOS task interprets incoming characters and distributes instructions to the appropriate servo-control routines. The robot arm's operations: rotation, gripping, and placement are controlled through timed PWM signals.

The FreeRTOS architecture uses a queue to ensure execution. When a command arrives, it is placed at the end of the queue unless it is an emergency stop or start command, in which case it is inserted at the head of the queue. This approach ensures safety-critical priority handling.

Results & Discussion

During testing, the robotic arm successfully executed all programmed tasks under no-load conditions. However, when a small load (a small pack of Kleenex tissues) was introduced, the arm encountered an issue. The robotic arm's claws would grab the object and attempt to lift it, but it would fail to lift the load off the ground and would then stop trying. The behavior indicated a possible mechanical or software issue, as if the motors weren't strong enough or if the code controlling the claws' and the arm's position and movement with a load wasn't functioning correctly. After analyzing and troubleshooting, it was found that insufficient current was the issue. After supplying more current, it worked successfully under a load, confirming that the hardware and software were functioning correctly and that the problem was a human error of insufficient current. Overall, the project went smoothly and was completed within the allotted time.

Conclusion

In conclusion, the robotic arm performed all the functions we wrote for it, with minor setbacks, including grabbing, lifting, moving, and placing an object, while responding accurately to user commands through the STM32 Nucleo-F446RE board running FreeRTOS. Although we encountered a temporary setback when the robot was unable to lift an object, the issue was quickly resolved, and the robotic arm operated reliably and consistently, completing all the assigned operations. Ultimately, this project showed the effectiveness of our design, our knowledge of hardware/software integration, and demonstrated our real-world ability to adapt and overcome challenges that require a practical engineering skillset.

Acknowledgements

We extend our thanks to Professor Ngo for providing the STL files used to construct the robotic arm.

References

Assembly Instructions