Theme Report 3: Act

Diego Verdin
400504430
6 April 2025

As a future member of the engineering profession, the student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is our own and adheres to the Academic Integrity Policy of McMaster University and the Code of Conduct of the Professional Engineers of Ontario.

Theme

The final theme in this course is Act. In the context of intelligent systems, acting refers to the system's ability to respond to inputs in real time. This response can be anything, from toggling an LED to navigating a robot through an environment. Acting is the output phase of an intelligent system, where sensed and processed information drives actions or decisions that have real world impacts.

Background

In intelligent systems, action is what connects computation to the real world, the mechanism which makes the system visible. Unlike passive systems, intelligent systems actively respond to outside conditions.

Labs 7 and 8 as well as the final project all embody this theme. In lab 7, interrupt driven programming was explored which allowed the microcontroller to respond immediately to time sensitive events without having to be stuck in a polling loop. This kind of action is critical in real world applications. Lab 8 introduced the ToF sensor showing how intelligent systems act on environmental distance measurements to build spatial awareness. Together these labs allowed me to create the final project for the course and emphasized the ability to react efficiently and autonomously.

Theme Exemplars

Exemplar 1: Periodic Interrupt with GPIO Output (Lab 7)

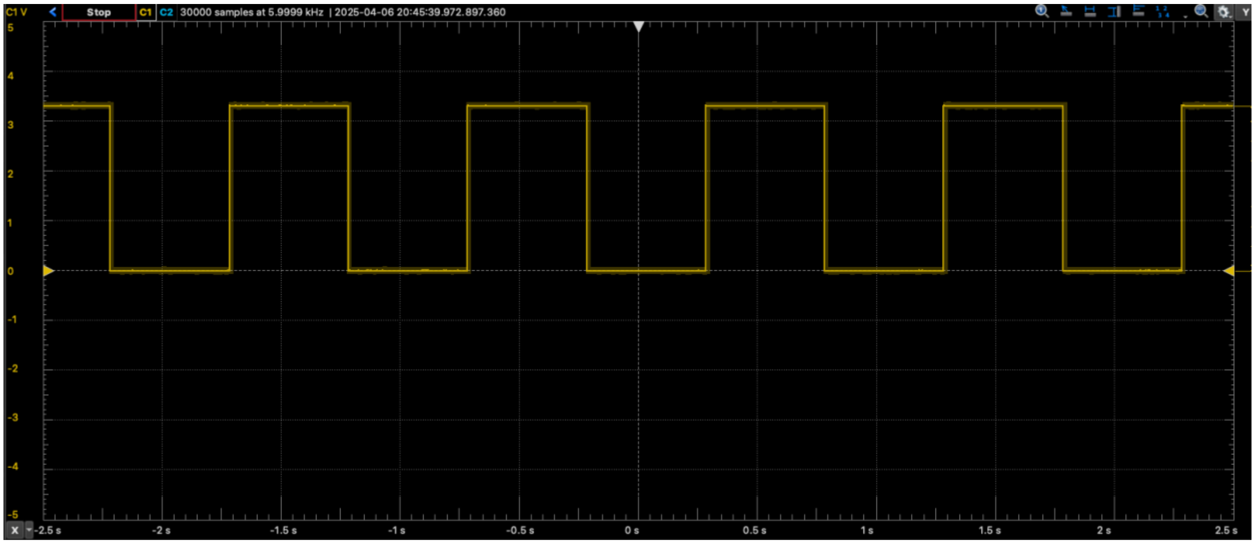
In this lab, we configured a periodic timer to toggle a GPIO output. This allowed an LED to blink independently of the main program loop.

Method:

- Configured timer 0A interrupt to fire at a fixed interval.
- Wrote an ISR to toggle PN1 and PM1 each time the interrupt occurred.
- Enabled global interrupts and confirmed the output waveform with the logic analyzer.

Validation:

The LED toggled every 0.5 seconds, and this behaviour was confirmerd using the logic analyzer on PM1. The image shown below is the waveform produced with a constant period of one second.



Exemplar 2: UART/I2C Communication (Lab 8)

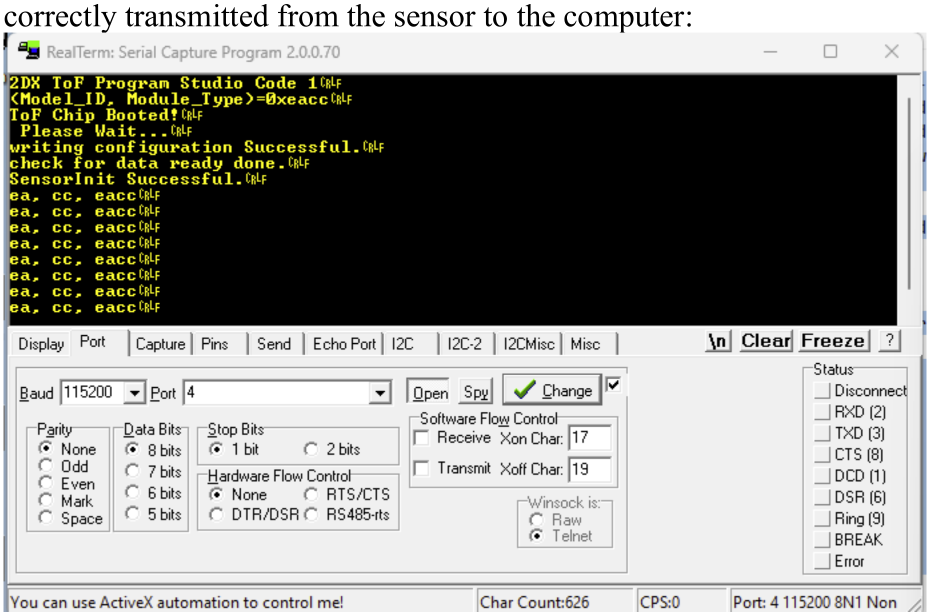
In this lab, the microcontroller connected to the ToF sensor through I2C and then to our computer using UART this is valuable in the act section as it allowed us to view results in real time and to receive info from the microcontroller in real time.

Method:

- Initialized I2C communication with ToF sensor.
- Initialized UART communication with microcontroller.
- Sent data from ToF sensor to computer (model ID and module type)

Validation:

The image below shows a screenshot of real term which shows that the data is being correctly transmitted from the sensor to the computer:



Debugging Exemplar

While trying to verify the UART transmission of sensor data, no output appeared on the real term terminal. The code compiled and ran without errors, but the terminal remained blank.

Debugging Method:

- First, confirmed that the UART peripheral was correctly initialized in the code (baud rate, TX and RX pins)
- Verified that the code reached UARTprintf() calls using breakpoints in the keil debugger
- Checked the port to which the UART was connected.

Resolution:

When device manager was opened it was identified that the UART was assigned to COM port 4 and real term was listening to COM port 3, once this was updated, the program ran without errors and real term outputted the correct data.

This debugging process showed the importance of verifying not just code but also external tools and configurations.

Synthesis

The two selected exemplars demonstrate how intelligent systems respond dynamically to external and internal inputs which is the main idea behind the Act theme.

In the first exemplar, we used a periodic internal timer to toggle GPIO outputs, allowing the system to function independently of the main loop. The ability to perform timely actions without constant polling allows our system to do other tasks while still performing required actions when certain events occur. The periodic toggling of an LED simulated a real time system that must perform consistent actions such as sampling sensors or updating something. It is an internally driven mechanism that prompts an action.

The second exemplar introduced a different time of action, real time communication with users. By integrating I2C and UART the system did not only read the sensor but also sent it to a readable display in real time. This allows for decision making and external action be able to happen at the same time. It also expands the concept of acting beyond just toggling something but now it includes sharing data which is critical to systems that must report/log data.

Together both of these exemplars show two sides of acting in intelligent systems. Autonomous action, characterized by internal logic which enables physical responses and interactive action which allows the system to interact with users or other devices in a meaningful way. Both of them required hardware and software integration and synchronization, reminding us that action in intelligent systems is not just about doing something but doing it at the right time.

Reflection

Through these labs I have developed appreciation for the role of action in intelligent systems. I now understand how microcontrollers can handle events without relying on blocks or inefficient loops, in addition I also now know how I can communicate with sensors and receive data that I can then use for another purpose in real time.

The debugging process taught me how important it is to be aware of how my hardware is connected to each other when working with these types of systems. Something as simple as being looking in the wrong port can make my code look completely wrong.

Overall, these experiences showed me how acting is the final step that gives intelligent systems presence in the real world.

Reference:

1. Texas Instruments. “MSP432E401Y SimpleLink™ Microcontrollers Technical Reference Manual.”
2. STMicroelectronics. “VL53L1X Time-of-Flight Ranging Sensor Datasheet.”
3. Valvano, J. W. *Embedded Systems: Introduction to ARM Cortex-M Microcontrollers*, 5th ed.
4. McMaster University. *2DX3 Laboratory Manual*, 2025.