MTRX2700 Major project

Assistive technology to support people with disabilities

> *"Assistive devices and technologies are those whose primary purpose is to maintain or improve an individual's functioning and independence to facilitate participation and to enhance overall well-being. They can also help prevent impairments and secondary health conditions."*
>
> *World Health Organisation (WHO)*
> *https://www.who.int/disabilities/technology/en/*

Technology has the potential to improve lives, and assistive technology has the potential to improve a person's ability to function or gain independence. The theme of this project is to develop a technology that could be used to assist a person with a disability.

Some background reading on this topic
https://www.who.int/publications/i/item/9789241564182
https://at-aust.org/home/assistive_technology/assistive_technology

Objective

The objective is to work together as a group to design and implement a proof-of-concept assistive technology. A design team of 3 students (4 for the Tuesday PM session) will work together to complete this project; each member of the group should be assigned responsibility for one or more modules of the system, and should be able to present their work during the demonstration week.

This project should follow good design practice, with focus on

- Understanding the requirements (what are you trying to achieve)
- Designing a modular system (the focus of the course content from weeks 6-8)
- Implementing test procedures where appropriate in the code (this is the focus of the course content for week 10)

As with the previous projects, the documentation and code is required to be submitted using a github repository

Assessment Criteria

Design quality

- Is there merit to the idea?
- Has the problem been broken down into appropriate modules?
- Is the software design reflective of the system modules?
- Is the final product/implementation of high quality?

Documentation

- Does the git repository reflect the project code and group work (include meeting minutes)?
- Does the git readme include a complete description of the system design and components?
- If the project was made public, could anyone easily use the code, or easily take a component from the project for use in a different project?

Testing and Demonstration

- Can the functionality of each module be established, is it possible to demonstrate each component?
- Is the code tested using test functions, or test data? (further discussion about this in the week 10 live tutorial)
- Is the implementation robust, and does it satisfy the requirements?

Testing is an important part of software engineering. Ideally, the testing plan should come before writing any code. What are the valid range of inputs/outputs that should be handled by each function. What are the constraints and limitations? You should get a different person to be involved in the construction of a testing plan for each software module. Any limitations should be described in the high level documentation, and should be included in the code comments for the public interface (the header file)

The marks will be scaled based on the difficulty of the problem being addressed, and the complexity of the modules used.

Please note - the tutors will be taking into account the challenges of working with remote group members. We understand that this can add another layer of complexity to the learning process. This is a very important challenge that requires good management and communication, well structured planning and strong leadership.

Potential technologies to include

This project will look to build on your understanding of mechatronics and microcontroller systems to develop an assistive technology. This could include (but is not limited to) ideas involving:

Lights/visual information	➢ This could include visual components on the dragon12 board ➢ It could also include other visual components on a PC controlled using the microcontroller through serial.
Sound information	➢ This could include direct sound driven through the dragon12 board ➢ It could also include sound from a PC that is controlled using the microcontroller using serial
Object Detection	➢ Objects can be detected using the lidar sensor ➢ Taking a scan with the lidar (using the pan/tilt) can be used to get a snapshot of the environment
Object Tracking	➢ Detected objects can be tracked over time to give an improved situation awareness.
Mapping	➢ Mapping of the environment could be useful to convey information
Guidance	➢ Detecting the state of the environment can lead to providing some kind of guidance.
Assistance	➢ helping with a specific task
Stabilising	➢ Compensating for the motion of the platform using inertial sensors

Project deliverables

The project involves designing and developing a proof-of-concept to be presented during the lab in week 13.

Deliverables for the week 13 lab

Presentation to the class

- Each group will make a presentation maximum 10 minutes
- The presentation should have contribution for each member
- Present the idea behind your project
- Give a demonstration of the proof-of-concept technology
- This can be done with the assistance of powerpoint slides (a projector will be available)
- During this presentation, your group should describe each of the main software elements and give a brief description of the module interfaces.

Before the demonstration, your group must tag the repo for release. Click on “tags” as shown in the image below, and create a new release called “assessment demonstration”.

Documentation

- Git repository including the code
- System level documentation in the repository outlining the code modules developed for the project.
- Meetings minutes.

Project timeline/due dates

Week 10 present your proposed idea to the tutor for feedback

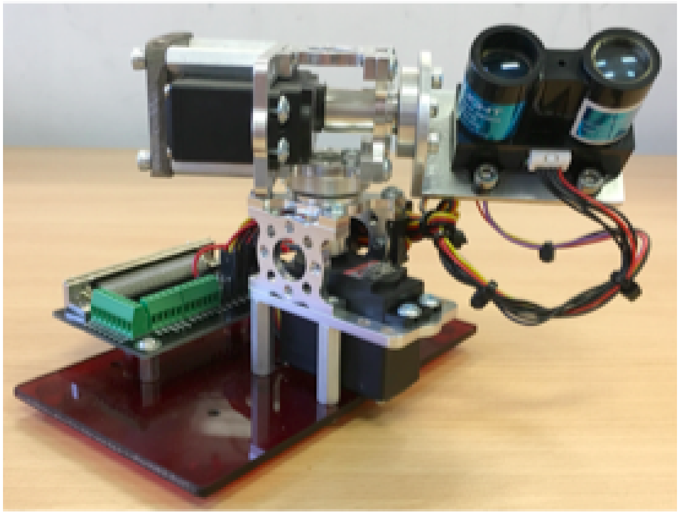
Week 12 progress report to the tutor during the lab

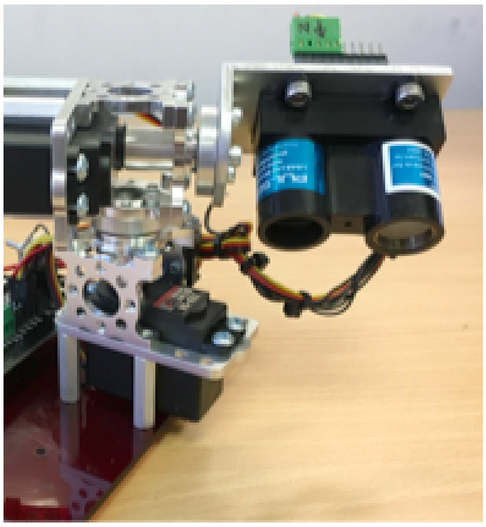
Week 13 presentations and technology demonstration.

Hardware/Software Components

The hardware consists of two major components: the dragon12 board and the pan/tilt unit (PTU)

Everyone should be familiar at this point with the sensors and interfaces that are available on the dragon board. This section of the document describes the additional hardware/software components that can be used in the major project.



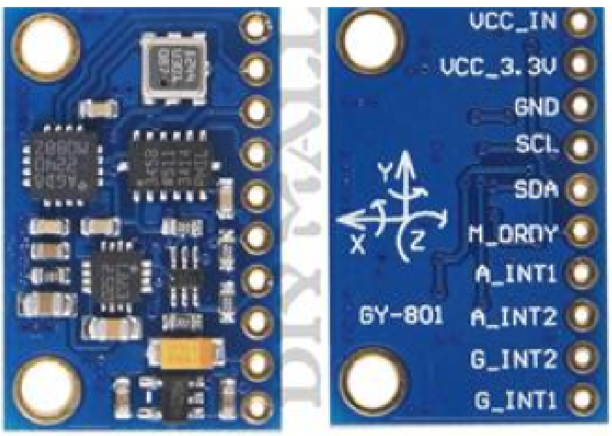


Servo motors

The PTU has two servo motors to change the heading and tilt of the laser. Each servomotor is connected to the PWM outputs **PP5** and **PP7** respectively. The servo module should be able to position the unit at any possible orientation in the range from 20 – 160 degrees.

Measuring the PTU Orientation

The PTU is fitted with a 10 Degree of Freedom (DOF) Inertial Measurement Unit (IMU) Module. The Module has a 3 axis gyro (L3G4200D), 3 axis accelerometer (ADXL345) and 3 axis magnetometer HMC5883L. It also has a pressure sensor BMP085. All sensors communicate with the main processor using the I2C bus. You will be provided with code to use I2C.



PTU Orientation using Gyro Information

A gyroscope sensor provides rate of rotation information. You will need to use the I2C functions provided to obtain the raw data. You will also need to use the proper constant to integrate the information of the gyros to obtain orientation. You may also need to consider removing any biases in the gyros.

PTU orientation using magnetometer Information

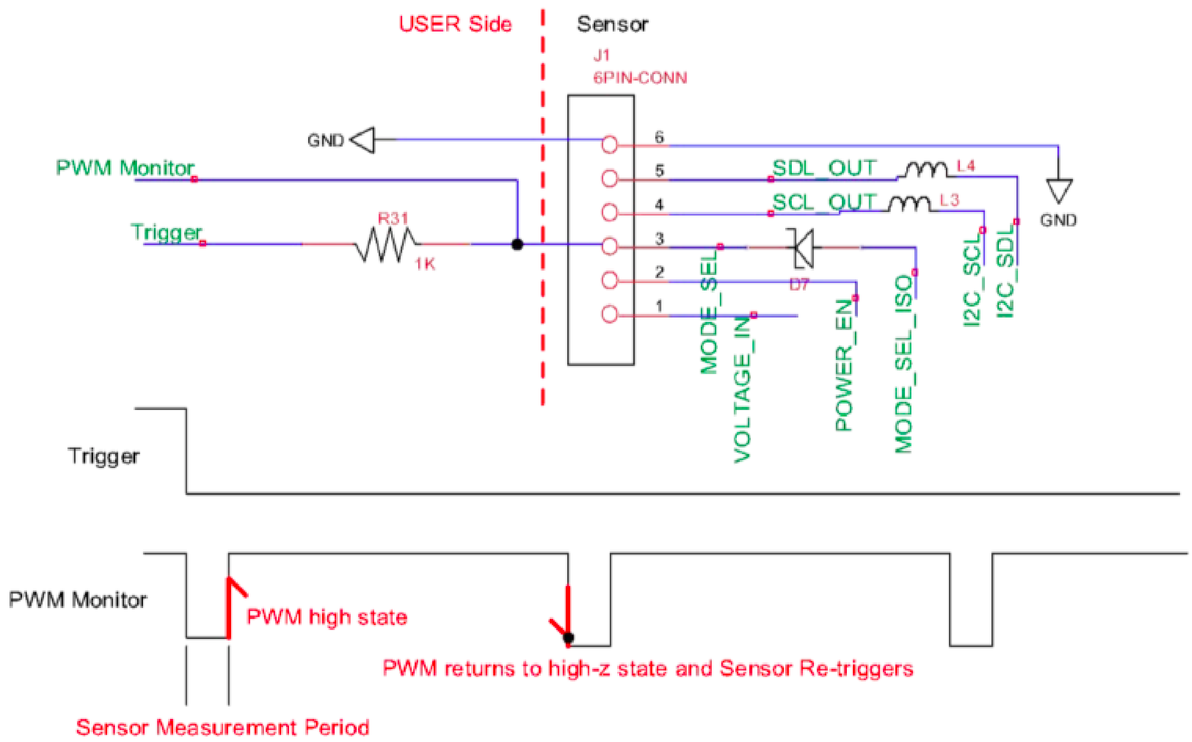
The magnetometer provides information of the direction of the earth's magnetic field. You can use this information to estimate the orientation of the unit. You will need to use the functions provided to obtain the raw data and use the proper constants to obtain the orientation of the unit.

PTU Tilt using Accelerometer Information

Accelerometers provide information about the absolute acceleration. When the unit is stationary, the only acceleration measured is gravity. You can use this information to estimate the inclination (elevation) of the unit. You will need to use the functions provided to obtain the raw data and use the proper constants to obtain the elevation in degrees.

Laser Range Observation

The PTU has a Lidar-Lite sensor version 2. This sensor provides the range information to the first object to reflect the laser light. The range can be observed by triggering the laser and measuring the width of a PWM signal. The interface to the laser is displayed below:



The laser can be interfaced to the microcontroller board using two pins: Trigger and PWM Monitor. A low-going transition on the mode control pin (Trigger) will trigger a single measurement, and the pin (PWM Monitor) will be actively pulled high with a pulse width proportional to distance. The time this pin is high is proportional to the distance to the first object in the orientation of the laser. The pulse width follows a 1 msec / metre relationship to the measured distance. This implies that a maximum range of 40 metres will require 40 msec.

The interface board with the PTU has the Monitor Pin connected to PT1 (Enhanced Capture Timer) and the trigger connected to PH0. There is also a jumper that connects the trigger signal to zero to trigger the laser continuously.

The PWM_Monitor Signal with the PWM Waveform is connected to Pin2 of P8 Header. The jumper is currently connected for continuous triggering (pin TRG-CONT). It can be changed to (Pin TGR – PH0) to be triggered by output Port PH0.

Controlling the Pan and Tilt Unit Interface Board

The orientation of the pan/tilt unit can be controlled open-loop, i.e. the position is set using PWM to control the servo motors. If you know the PWM signal, you can determine the orientation. An important caveat is that there is some time delay with the movement of the PTU, the servos are not instantaneous.

An alternate strategy for determining the current orientation of the PTU is to use the inertial sensors. The gyros can be used to determine the actual movement of the PTU, and the magnetometer can be used to determine the heading/orientation relative to north.

Computer Interface

The microcontroller board can be connected to a PC using the serial port. By connecting to the PC, the capabilities of the microcontroller board can be greatly expanded. A program can be developed for the PC in any language (python, c++, even matlab) that can connect to the serial and both receive information, or send information to the microcontroller board.

some ideas for the computer interface

- display visual information from the microcontroller on the PC monitor
- make the PC play sound files/etc directed by the microcontroller
- have the computer interface with the microcontroller by sending commands
- a computer interface could include other inputs (mouse, joystick, gamepad, etc) to give commands to the microcontroller
- the computer can make use of the internet to find information

For more information on the computer interface, the lecture on serialisation and the live tutorial from week 9 are recommended viewing for a starting point. There is example code that shows how to send data to/from the PC using serial implemented using python.

Appendix 1: Electrical connections

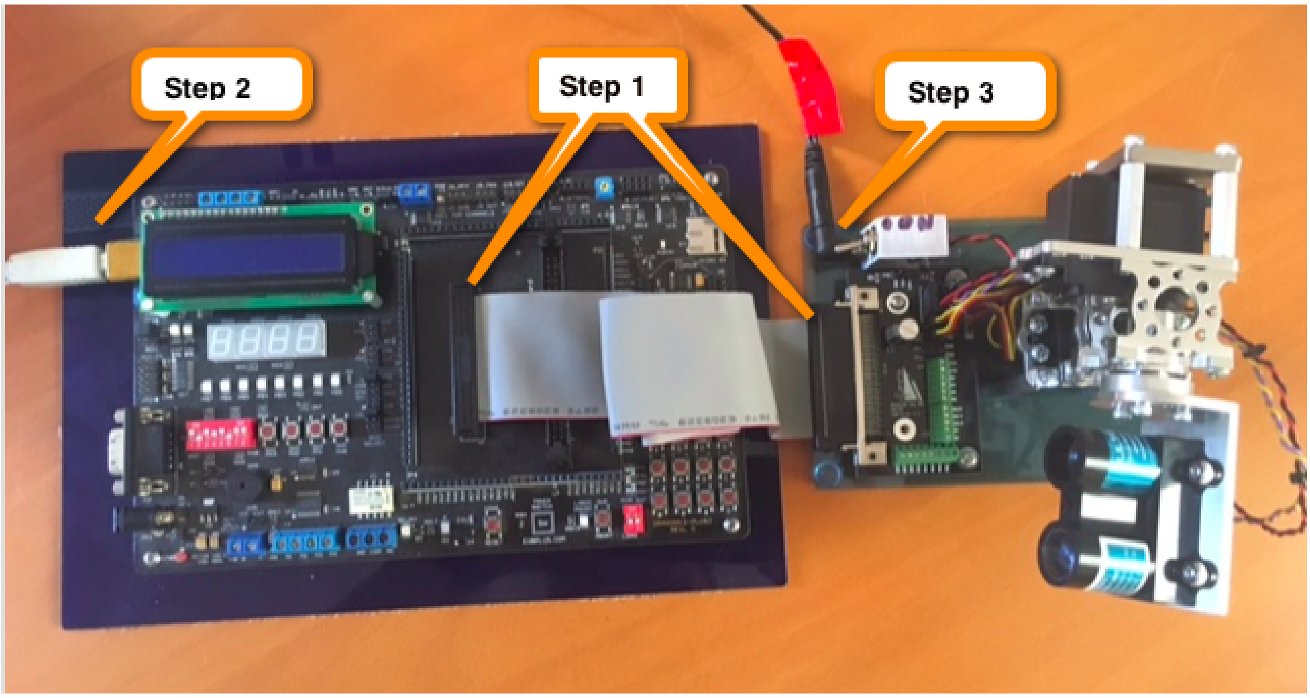
It is very important to follow the correct procedure when connecting the pan and tilt hardware to the microcontroller board.

The following procedure in essential to prevent damage to the board:

Step 1
Connect the 69 Pin Ribbon Cable interconnecting the board and the Pan and Tilt Unit

Step 2
Power the Dragon 12 Board (Connect USB)

Step 3
Connect the Power to the Pan and Tilt Unit



Appendix 2: Schematic for Pan/Tilt Unit interface board

The PWM_Monitor Signal with the PWM Waveform is connected to Pin2 of P8 Header. The jumper is currently connected for continuous triggering (pin TRG-CONT). It can be changed to (Pin TGR – PH0) to be triggered by output Port PH0.