Universal Motor Controller

Nathan Williams, Benjamin Degler, Timothy Thomas, Luke Alonso

University of Scranton

Abstract

A multitude of secondary and postsecondary engineering programs across the country and the world utilize electric motors for a variety of tasks, projects, and class work. The first time you learn how to control a direct current (DC) motor bi-directionally, for example, it is extremely beneficial as a learning experience to construct the DC control circuit using basic circuit components likes transistors, resistors, and capacitors. The next logical step, once you know how to make the circuit from scratch, is to use a digital H-bridge chip and simply connect your motor to it along with external power, controllable digital signals, and a multitude of prototyping supplies. This makes controlling your DC motor more manageable but still takes a significant amount of time to build your circuit, connect a microcontroller, and write your control program. This is where the path of easy motor controlling stops. There is no next step to make controlling your DC motor easier. What if there was a next step involving a device that you can connect a DC motor to, or for that matter, any type of motor and make controlling it simpler. The invention called the Universal Motor Controller is the answer to this challenge.

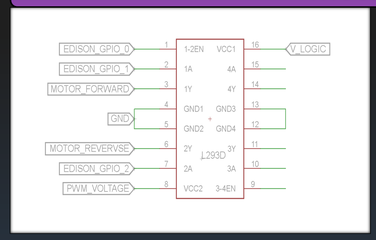
The motivation behind this project stemmed from the extensive use of motor operated devices in a multitude of our projects on the campus of The University of Scranton. We realized that we were spending a lot of time interfacing the same set of components and circuits in the same ways to control the same types of motors. The only difference seemed to be how they were being used, what they were being connected too, and what power requirements were for the application.

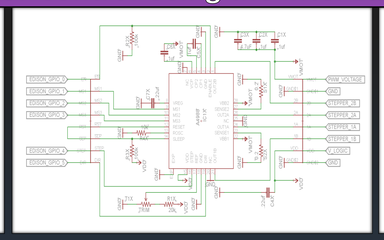
Requirements

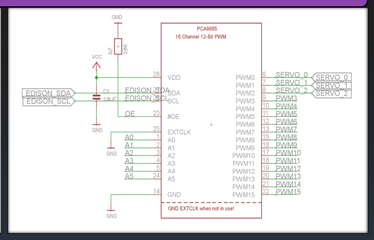
The Universal Motor Controller shall:

- Interface multiple motors including but not limited to DC motors, stepper motors, and servo controlled motors
- Connect the aforementioned motor types simultaneously
- Control and operate the aforementioned motor in the same set of sequential operations
- Connect the aforementioned motor types as “plug and play” devices
- Utilize the Intel Edison embedded microcontroller
- Utilize pre-composed python scripts for control as well as user defined controlling methods in python
- Provide a fixed range of power supplied to the motors specifically 5.00-12.0 volts at up to a maximum total current of 4.00 Amps
- Perform a multitude of tasks utilizing the same hardware
- Interface with an android mobile application graphical user interface
- Perform its functions on three different devices where motor control is a necessity.

Motor Control Using the Intel Edison



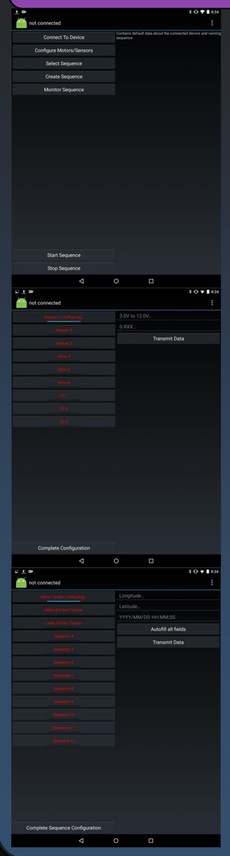




The Universal Motor Controller is designed to operate three motors simultaneously. These motors can be any combination of the three most common types of motors: DC, stepper, and servo motors. To connect these devices, there are a total of nine motor ports embedded into the controller device; three ports allocated to each type of motor. The circuitry involved utilizes L293D ICs for DC motor control, a PCA9685 I2C PWM generator for servo motor control, and A4988 micro stepping drivers for stepper motor control. Programming motor movements is done in a programming language called Python. Python is a high-level language with an emphasis on readability, generalized syntax, and open source nature.

Having a multitude of GPIO pins was essential in the decision to use the Intel Edison. Each subsystem of the controller requires signals from the embedded system in order to change its behavior based on a desired function. The GPIO pins allow for this type of communication by using the GPIO pins for HIGH/LOW signals and specific functions such as Inter-Integrated Circuit (I2C) communication for accessing additional hardware

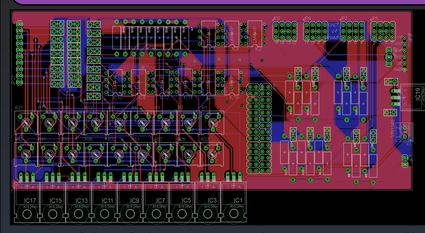
Mobile App

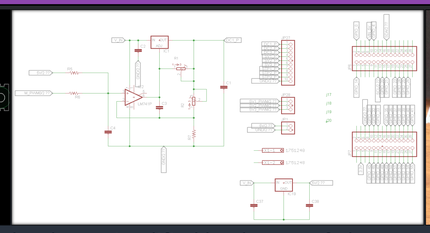


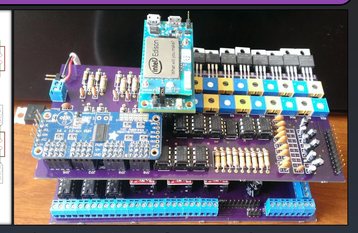
The Universal Motor Controller was developed in Android Studio for the Lollipop Android Operating System currently running on the Nexus 9 used for implementation. Java is the primary language used in programming many of the classes and interfaces within the application to handle the backend operations. Designing the UI for the application was done in XML.

There exist several layouts within the Android Studio hierarchy that contain the basic mockup of what each screen for the Universal Motor Controller should look like. Each xml layout is paired with a unique class within the java hierarchy that allows a view to be created during the initial startup of the application, along with any relevant button presses that occur on a currently inflated view. Each layout has several buttons, text boxes, and container views that help a user interact with the Universal Motor Controller. In addition to the xml files, much of the content within each layout was altered programmatically to meet specific constraints on what a user should and should not be able to do on specific screens.

The Design

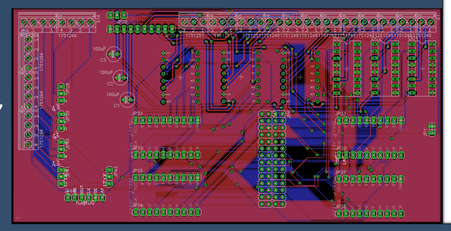


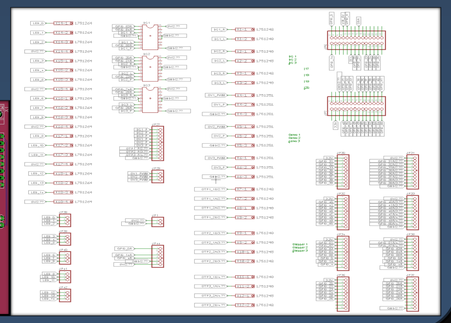




Assembly of the Universal Motor Controller has been simplified by the use of electronic design CAD software. The Mother and Sister Board designs are shown in two aspects for easy recreation. The Mother Board was designed to contain all of the power regulation circuitry. In addition, there is pin headers to slot on the pulse-width modulation generator attachment board for easy replacement. Contained on this board is a set of potentiometers for calibrating the regulation circuits based on the input power source. The circuits are laid out in a group-like fashion for simplified placement and soldering. Attached to the board is the Intel Edison Mini Breakout Board via expansion pin headers. These pins extend beyond the bottom of the board for easy attachment to the Sister Board. The Sister Board was designed to contain a majority of the replaceable and slotted components for motor and logic control. This board has the logic level converters for control of the motor drivers. The motor drivers and logic level converters are attached via IC sockets and pin headers for easy replacement. Also attached to this board is a pulse-width modulated LED driver for RGB LEDs. There are five LED ports for various status indicators, programmable by the user. Located around the outside of the Sister Board are two and three pin screw terminals for motor wire connections. These terminals can be used to format a motor attachment of the users preference. The Universal Motor Controller uses USB-like connections for easy interfacing. This was chosen because the power requirements of the motors attached to this version of the Universal Motor Controller will not exceed twelve watts.

The overall aesthetic design of the Universal Motor Controller can be completely custom based on the user deploying the technology, or the environment in which it will be used. The key components of the Universal Motor Controller are all housed on the PCB itself, making it easy to port to other casings. In our design, we used acrylic walls with 3D printed brackets and mounts.





Next Steps

- Voltage regulation circuit dissipates too much power
- Voltage regulation circuit is large and inefficient
- Overall size of the controller is too large
- Cost of the controller can be reduced
- Graphic User Interface can be difficult to use for implementing new sequences
- Uploading of sequences can be more efficient and done through the app to eliminate the need of server based execution hosted on the Edison

The current layout of the voltage regulation circuit individually regulates each motor port. This results in nine total regulation circuits each requiring a large amount of PCB real-estate. Developing a more robust circuit to regulate three motor ports rather than a circuit for each individual port would reduce the overall component requirement and layout space, thus reducing the production costs of the PCB.

Redesigning this circuit would take into account the power dissipation, and power consumption concerns. With each port having its own regulation circuit, there are times where three motors would be dissipating power from a voltage drop of twelve to three volts with up to three amps of current draw. The redesign would develop a circuit that is capable of consolidating voltage use, and building lower level regulations off of regulators that are already in use. This type of design will increase the efficiency of the circuit by reducing the power consumption and dissipation.

To remedy the graphical user interface ease of use concern, it is planned to remove the existing sequence execution hierarchy and replace it with a more intuitive system. This system will include integrating the mobile app with an integrated development environment which will compile the sequence on the tablet itself before uploading it to be executed by the Edison.

References

Attitude control. (n.d.). Retrieved April 15, 2015, from <http://en.wikipedia.org/wiki/Attitude_control>
Ephemeris. (n.d.). Retrieved April 15, 2015, from <http://en.wikipedia.org/wiki/Ephemeris>
Intel Edison Product Brief. 2014. Intel Corporation. 30 Mar. 2015. Web. <http://download.intel.com/support/edison/sb/edison_pb_331179001.pdf>
Smith, Anthony. “Control A High-Power Load With A Low-Power Microcontroller” Electronic Design. 3 Aug. 2006. Web. 30 Mar. 2015. <http://electronicdesign.com/electromechanical/control-high-power-load-low-power-microcontroller>
Williams, Nathan D. and Truncale, Nicholas P. (2014). “Reflective Solar Tracking: Calculating motor run intervals using astronomical data using a Raspberry Pi”. The MagPi, 22 (April), pages 4-7

Acknowledgements

Dr. Joseph Dreisbach, Dr. Brian Conniff, Dr. Tabbi Miller-Scandle, Dr. Christine Zakzewski, Mr. Majid Mohktari, Mrs. Nancy Laffey, Mr. Bill Gunshannon, Intel Cornell Cup, OSH Park, Backyard Ale House

‘Raspberry Pi is a trademark of the Raspberry Pi Foundation