ECSE426 - Microprocessor Systems
Fall 2013
Final Project

Introduction

For your final project, you will invent and implement a robotic hand to be controlled wirelessly from a distance. The robotic arm will mimic the movements of your own hand (the one holding the transmitting board) in all possible degrees of freedom. That is, you will use the tilt information of one board in 3D space to be reproduced remotely by the robotic hand on the other end. Advanced designs are in actual development in various prototyping and implementation stages to allow for various applications. Such applications include remote surgery operations (i.e. battlefield surgery) or handling dangerous or radioactive items remotely for higher levels of safety. The project will merge many of the elements you've built up in the labs through the semester into a cohesive and practical objective with the addition of a few new modules which you will have to develop.

This project is intended to have a large creative component, and hence no two projects should be alike. However, there are a few core requirements that we will lay out accompanied by simple suggestions for other features you may want to implement. There is, of course, a minimum amount of extra work that must be done to make your project unique. Your final project should consist of the core requirements and at least three of the special requirements listed in this document. You are also highly encouraged to replace the special requirements with some of your own provided they are equally challenging and at the same level of complexity.

Project Details and Design

Wireless interfacing

In designing the simple robotic hand, you will need to start with breaking up your project into several modules to work on. The highest priority is establishing wireless communication. To this end, each group will have two TI MSP430-CC2500 modules to work with. Initially, you have to physically connect the chipset to the F4 Discovery board. You need to connect the 4-wire SPI and power to the chipset (and possibly configure and connect any interrupt pins). Consult F4 Discovery board and the MSP430 for schematics and pin layout to carry on the wire-wrapping. Secondly, you need to write the drivers for communicating with the transceiver via SPI interface. The drivers need be modular in design and function correctly at all times. The drivers are similar in concept but slightly different in implementation than ST ones as described in the tutorial. For details on interfacing the CC2500 via SPI, refer to the cc2500 design note. Thirdly, you need to write the high level functions to setup (configure) the cc2500 and operate it. Finally, to test your drivers, you attempt to read any of the internal status registers like VERSION or PARTNUM. You can also write into a register and then read it back (also test for burst transmissions).

For testing wireless operation, a beacon preconfigured to transmit the numeric codes 0 – 9 is going to be installed in the lab. You might wish to test your receiver code against it (use the provided configurations to set up the chipset). Then you can develop your transmitter. If you are having any troubles with the beacon for no apparent reason, you can skip this part and test your transceivers against each other (though it might be harder to debug).

A document detailing the beacon operation and configuration is also attached. You will also use the same configuration in your project with few changes (i.e. channel, IOCFG for interrupts ... etc.) as described in the tutorial. Configuration details are in the abovementioned document.

Robotic hand design and control

After establishing the wireless communication, the next step after sending tilt information is to make us of it. You need to do experiments and analyse those signals to control the robotic hand. The hand in its very basic setup is based on two motors, one to reproduce the rotational movement of your wrist and the other the movement of your palm in two directions. The combination of the two should precisely mimic the hand movement at the transmitter's end. You might need to construct your platform of hard cardboard or any other materials of your choice (tubes, strings, Legos ... etc). You might also wish to cover your model with a glove to have a more realistic display of a hand. This video shows the basic idea http://treehouseprojects.ca/robohand/. The servo motors you will use are Hitec HS-422 models, the datasheets of which is uploaded on myCourses. The motors are simple to interface, with two lines for power and ground directly taken from the board, the third pin is for PWM control input.

Core Requirements

There are a few basic design requirements for all groups:

- You must have a fully working servo motors.
- You must make use of the accelerometer readings in a significant way as a control input.
- You must use two boards communicating wirelessly.
- You must continue to use the RTOS. This is for your own good. There will be a lot of tasks to synchronize.
- Controls should be responsive and this should be a functional device.
- Your design must be safe in the sense that all deadlines are met and that it exhibits no unintended behaviour.

Special Features

The above tasks don't necessarily imply any creative design. As such, we are requiring that you implement at least **three** special technical features. Examples (with difficulty estimates) may be:

- Add a character LCD (see below) to show the active angle positions (medium),
- Use the keypad input to move the hand to predefined position (medium-low),
- Use the keypad to input a set of movements for the hand to cycle through (medium),
- Manual adjustments based on keypad input (medium-low),

- Use a set of manual adjustments to learn and better calibrate the hand movement. This should serve automatic and precise reproduction for future inputs (high),
- Use distance estimation (effectively) as a control parameter (i.e. have a sub-mode which uses distance as input for a certain hand function (medium),
- Hand movement versatility (high). Be creative in your physical design of the hand (physical motor setup) to allow for interesting hand movements (i.e. allow to simulate the motion of picking up things, controlling single fingers … etc.). ***We have an extra 3 motors which can be given to groups on a first come first serve basis for those who wish to implement this part. Check with the TA before you go to ECE labs to inform them of your request.***
- Create a USB interface (high), more details below,
- Optimization of power consumption in the device through management of clock rates, sleep modes and efficient design practices (high).

These are just some quick examples and additional ideas are encouraged. Difficulty ratings are provided to give an idea how much work might be involved. You should aim for **a "medium" average** across three features. If all of your features are on the low side, you may need a fourth or you will face higher expectations of your overall implementation. If you have an idea but don't yet know how to implement it, ask a TA and we'll be happy to look at it with you. Further, if you have some crazy ideas that you would like to implement but think it might qualify as several features, let us know. We can accommodate ambition and creativity.

Your features should be central to your design. For example, don't just write an LCD driver and add that as a feature. You must do something meaningful with it. Design around functionality rather than just a checklist of features.

Components

Onboard:

USB

The STM32F4 chips have a USB 2.0 OTG peripheral with a full-speed PHY. The Discovery board exposes this through a micro-USB connector, allowing for USB-based applications to be developed. ST provides drivers for CDC (serial port) and HID (human interface device) classes but USB can be very difficult to implement and you are encouraged to explore the implementation details further before proposing it as a feature in your application. Also keep in mind that it is likely that you would have to write some PC software to accommodate.

Extras

HD44780 Character LCD

You will be provided with a Hitachi HD44780-compatible 2x16 character LCD that can be used in one of the special features requirements. They follow a simple protocol which is well documented for outputting text and very basic graphics. You are free to use these in your project.

Keypad

We have a lot of telephone-style keypads which are very useful for providing events for arm control or for precise movement input / correction.

Application notes on using HD44780 LCD and keypad have been written for you to assist in writing your drivers.

RF2500 Wireless Kits

The TI wireless kits are a required component of your final project. They don't have enough bandwidth to move large amount of data but simple control signals can be sent to coordinate the boards. One nice feature is that they can indicate the received signal power and this can provide an estimate of distance between the two systems.

Anything Else

You can add your own parts as well. There are plenty of free I/O pins to add whatever you want. For fairness, you can only be judged based on innovative design and the difficulty of implementation for whatever you may add, as of course not everybody has access to the same resources.

Demonstration

The demonstrations will be just like your previous lab demos. However, this time all of you will be presenting at the same day. The schedule of the final presentation will be posted on the web. Presentations will take place in one of the last days of the semester. Usually, the professor not the TAs demos the projects and reads your reports, so make sure that your presentation is clear and comprehensive. The TAs could show up for the demos as well.

Report

The final project report is supposed to be more formal than the lab reports which you wrote during this semester. You should consider all feedback you've received this semester while preparing your report.

In particular, we would like to stress the need for the extra following points in your report:

- The components in your system
- A timeline of work and a breakdown between team members
- A block diagram of your firmware, showing roughly how modules interact

As always, you should explain the reasoning behind your design choices.

Resources

- Datasheets/reference manuals for all listed devices from their respective manufacturers website
- Firmware package for STM32F4-Discovery platform for examples using all on-board peripherals