ENEL 387
Design Project Proposal

Kyle Shaw
200 299 741
Jan 31th, 2014

Mounting flat screen TVs on a swing arm wall mount is great, that is until you want to adjust the position of the TV while lounging on your couch. My project will solve this problem by giving the user the ability to control the position of a swing arm wall mounted TV using your TV remote, three servo motors, and a STM32F100 microcontroller. I will read in commands from the remote with an IR signal receiver/demodulator to capture the IR signal sent from the remote. Then, depending on what code is received, the microcontroller will turn one of the servos either clockwise or counter clockwise, changing the position of the TV. The servomotor will continue to move until the user releases the remote button. To reduce costs I will build a prototype system made from small hinges and wood. The TV will be scaled to the size of an iPhone.

I will use either a positional rotation servo or a continuous rotation servo to move the swing arm. This choice will depend on the mechanical design of my swing arm, which has not been decided upon yet. If I use a positional servo, I will increment the position value sent to the servo for as long as the user holds the remote button or until the maximum/minimum angle has been reached (0^o-180^o). If I use a continuous servo either a clockwise or a counter clockwise signal and reasonable speed will be sent to the servo as constants. The user will act as feedback to ensure that the swing arm and TV do not crash into anything. In both cases, the signal sent to the servo will be a PWM signal, which can be generated by the STM32F100 microcontroller.

One of the challenges with this project will be to decode the signals received by the IR remote. There are a number of protocols in use and each protocol uses timed pulses to represent the ones and zeroes that make up each command. It may take some trial and error to determine the protocol the remote is using and then decode the signal sent by the remote. I have found a website that explains a dozen or so protocols to help with this process (http://www.sbprojects.com/knowledge/ir/nec.php). If this proves to be too difficult to complete with the STM32F100, I have found a demo on Sparkfun.com that describes how to use an IR library with an Arduino to easily decode the most common IR signals (https://learn.sparkfun.com/tutorials/ir-communication/all). Once I know which protocol the remote uses I will be able to decipher the signal the STM32F100 reads from the IR signal demodulator, and make decisions with the microcontroller based on which command has been received.

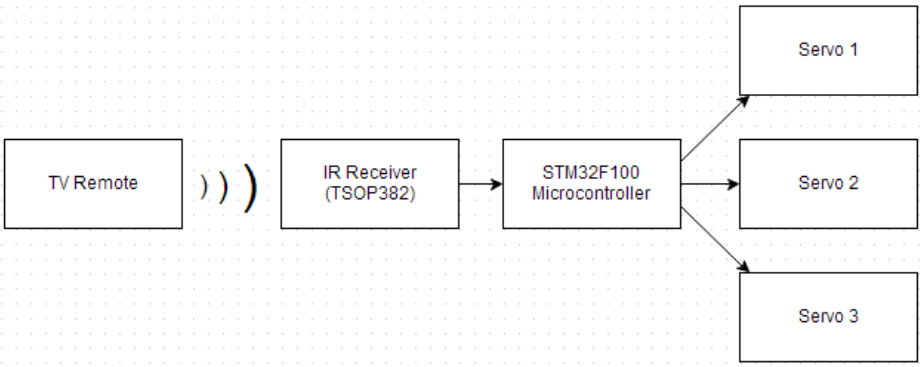

Figure 1 - Block Diagram

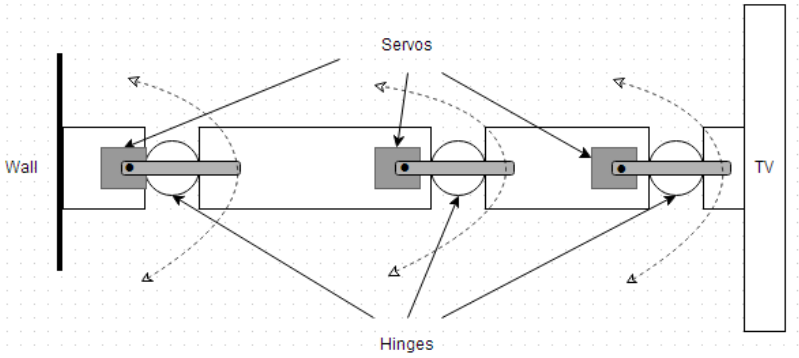

Figure 2 – Rough Schematic