Neeraj Basu & Carlton Duffett

EC 450 Final Project

4/29/15

Professor Giles

Dynamic Elevator Control System

Purpose:

The purpose of this project was to use the MSP430 microcontroller to create a four-floor model elevator. The elevator responds to calls to a specific floor on the structure, followed by a destination selected by the passenger once they enter the elevator.

Features:

On-structure Call Buttons

The purpose of these 6 pushbuttons is to call the elevator car to a floor on the structure so that a passenger may then enter the car. Since we were severely limited by the number of GPIO pins on the MSP430, each button was connected to a priority encoder. This allowed us to use only four pins (enable and 3 bits of address) to determine which button was pressed.

In-Elevator Destination Buttons

Once the elevator car has arrived at the selected floor, the user then selects their desired destination (floors 1-4). This feature also ensures that the user's destination is consistent with the direction they selected using the call buttons. For example if the user calls the elevator from the second with the "up" button but then presses the button for the first floor, they will not be able to select the first floor as a destination. Each of these 4 buttons was also sent to a priority encoder, using only 3 GPIO pins in total.

Limit Switches and Seven-Segment Display

For the control algorithm to work, the system must know exactly where the elevator car is at all times. This is done using limit switches that are depressed by the car as each floor is passed. When a switch is depressed, an address is sent to the seven-segment display decoder, which in turn displays the current location of the car to the user.

Motor/Gear Box and Pulley Mechanism

Depending on the desired direction, the elevator car is raised or lowered by a motor and gearbox assembly. The motor is driven by an H-bridge circuit using PWM to control motor speed. Motor direction (clockwise or counterclockwise) is selected using two additional GPIO pins.

Collectively, all of our I/O devices use the entirety of port 1 and port 2, all 16 GPIO pins.