Project Title: Military Master Light Switch

Team Members: Andy David, Carlos Gonzalez, Andy McCormick, Deepali Rauniyard, and Nick Sorg

Advisor: Dr. Wang, Dr. Zhao

Date: April 28/2008
## Table of Contents

Date: April 28/2008

0

Table of Contents ................................................................. 1
Table of Figures ........................................................................ 2
Acknowledgements ................................................................... 4
Abstract .................................................................................. 5

Section I: Final Design ................................................................. 6
  Housing Summary .................................................................. 7
  Pushbutton Summary ............................................................... 7
  Electrical Summary ................................................................. 8
  Housing Design ...................................................................... 9

Pushbutton Details ..................................................................... 13
  Pushbutton Operation .............................................................. 14
  Construction ......................................................................... 14
  Illumination .......................................................................... 16
  Pushbutton Selection .............................................................. 16

Electrical Design ....................................................................... 17
  De-bouncing Circuit and Schmitt Trigger .............................. 18
  Voltage Regulator ................................................................. 20
  Microcontroller Clock Generator .......................................... 22
  In-System Programmable CPLD ........................................... 24
  Light-Emitting Diodes (LEDs) ................................................ 30
  Xilinx HW-130 Programmer ................................................ 31

User Instructions ......................................................................... 39

Design Modifications ................................................................. 40

Section II: Building Process ......................................................... 42
  PCB Development ................................................................. 43
  Button Cover and Housing Construction ............................. 46
  Assembly .............................................................................. 49
  Difficulties ........................................................................... 51

Section III: Testing and Analysis ................................................... 55
  PCB Functionality Testing ..................................................... 56
  Pushbutton Functionality Testing .......................................... 58
  Vibration Test ....................................................................... 59
  Thermal Test ......................................................................... 61
  Analyses .............................................................................. 64
    Housing Analysis ............................................................... 64
    Button Cover Analysis ....................................................... 65

Section IV: Evaluation and Recommendations ................................ 67
  Design Evaluation ................................................................. 68
  Design Recommendations .................................................. 68
  Course Recommendations .................................................. 69

Conclusion ............................................................................... 70

References ............................................................................. 71
Table of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1:</td>
<td>Current Master Light Switch</td>
<td>9</td>
</tr>
<tr>
<td>Figure 2:</td>
<td>Current Switch Installed In Vehicle</td>
<td>10</td>
</tr>
<tr>
<td>Figure 3:</td>
<td>3-D Sketch of Housing Design</td>
<td>10</td>
</tr>
<tr>
<td>Figure 4:</td>
<td>Housing Dimensions for Casting</td>
<td>12</td>
</tr>
<tr>
<td>Figure 5:</td>
<td>Housing Dimensions for Machining</td>
<td>12</td>
</tr>
<tr>
<td>Figure 6:</td>
<td>Picture of Pushbutton</td>
<td>13</td>
</tr>
<tr>
<td>Figure 7:</td>
<td>Button Cover 3D Models</td>
<td>15</td>
</tr>
<tr>
<td>Figure 8:</td>
<td>Button Cover Dimensions, Side View</td>
<td>15</td>
</tr>
<tr>
<td>Figure 9:</td>
<td>Button Cover Dimensions, Bottom View</td>
<td>15</td>
</tr>
<tr>
<td>Figure 10:</td>
<td>Pushbutton Dimensions (inches)</td>
<td>16</td>
</tr>
<tr>
<td>Figure 11:</td>
<td>De-bouncing Circuit with Schmitt Trigger Schematic</td>
<td>18</td>
</tr>
<tr>
<td>Figure 12:</td>
<td>Schmitt Trigger Diagrams</td>
<td>19</td>
</tr>
<tr>
<td>Figure 13:</td>
<td>7805SR Voltage Regulator Diagrams</td>
<td>21</td>
</tr>
<tr>
<td>Figure 14:</td>
<td>Connection of MAX7389 and XC9572 CPLD</td>
<td>21</td>
</tr>
<tr>
<td>Figure 15:</td>
<td>MAX7389- Microcontroller Clock Generator Dimensions</td>
<td>23</td>
</tr>
<tr>
<td>Figure 16:</td>
<td>XC9572 In-System Programmable CPLD Connections</td>
<td>24</td>
</tr>
<tr>
<td>Figure 17:</td>
<td>CPLD Function Block</td>
<td>25</td>
</tr>
<tr>
<td>Figure 18:</td>
<td>XC9572 Architecture</td>
<td>26</td>
</tr>
<tr>
<td>Figure 19:</td>
<td>CPLD Dimensions</td>
<td>29</td>
</tr>
<tr>
<td>Figure 20:</td>
<td>Picture of CPLD</td>
<td>29</td>
</tr>
<tr>
<td>Figure 21:</td>
<td>LED Connection</td>
<td>30</td>
</tr>
<tr>
<td>Figure 22:</td>
<td>Xilinx HW-130 Programmer</td>
<td>31</td>
</tr>
<tr>
<td>Figure 23:</td>
<td>Flowchart</td>
<td>34</td>
</tr>
<tr>
<td>Figure 24:</td>
<td>Power Supply Schematic</td>
<td>35</td>
</tr>
<tr>
<td>Figure 25:</td>
<td>Clock and Schmidt Triggers Schematic</td>
<td>36</td>
</tr>
<tr>
<td>Figure 26:</td>
<td>Connectors Schematic</td>
<td>37</td>
</tr>
<tr>
<td>Figure 27:</td>
<td>CPLD Schematic</td>
<td>38</td>
</tr>
<tr>
<td>Figure 28:</td>
<td>Button Diagram</td>
<td>39</td>
</tr>
<tr>
<td>Figure 29:</td>
<td>Comparing chip dimensions (showing the size restriction)</td>
<td>40</td>
</tr>
<tr>
<td>Figure 30:</td>
<td>2D Picture of Ultiboard PCB</td>
<td>43</td>
</tr>
<tr>
<td>Figure 31:</td>
<td>3D picture of bottom side of Ultiboard PCB</td>
<td>44</td>
</tr>
</tbody>
</table>
Acknowledgements

We would like to thank our faculty advisors Dr. Wang and Dr. Zhao for all of their help during this senior design project. We would also like to thank Riverside Manufacturing and especially our contact there, Mr. Dimitrios “Jimmy” Gogos, for providing project funding and assistance. For the electrical side, we would like to thank Bob Tilbury at IPFW for helping with the PCB production. For the mechanical components, we thank Jason Davis at IPFW for helping us make parts in the rapid prototyping machine on campus at a low cost. Finally, we wish to thank all of the companies who provided free samples of components for us to use throughout the year, including Xilinx, OTTO Engineering, and Maxim.
Abstract

This senior design project involves designing a new Master Light Switch for military ground vehicles. The existing Master Light Switch routes power from the vehicle’s 28 volt battery directly to the vehicle lights. New military vehicles will instead require a 5 volt signal to be sent to a distribution center which will power the lights. Riverside Manufacturing, Inc. has sponsored this senior design project to develop a Master Light Switch that sends a signal to the distribution center instead of transmitting power directly. The final design chosen for this project uses a lighted pushbutton interface instead of the lever switches utilized in the current Master Light Switch. The vehicle’s battery power is lowered to a useable level with a voltage regulator and user input from the nine pushbuttons is interpreted and relayed by a logic controller inside the switch. This new Master Light Switch is physically interchangeable with the current switch, and is more user-friendly with an illuminated interface. This report will first summarize the final design that was developed and will then discuss the building of the prototype. Next, the testing and all of the analysis that was performed will be described. Finally, the report will give an evaluation of the design and recommendations along with conclusions and references.
Section I:
Final Design
This section summarizes our detailed design for the Master Light Switch senior design project. It gives a detailed description of the final design. We divided the components of our final design into three broad categories: details about the switch housing, details about the pushbuttons, and electrical details. The mechanical portions of the design include the switch housing and the pushbuttons. The electrical section describes the regulation of power and the implementation of a programmable chip that controls the signal relayed by the pushbuttons.

**Housing Summary**

The housing of the switch contains all of the electrical components and has the pushbuttons embedded in the front. The current switch housing is made of metal. A metal housing is also be used in our design because it is inherently fungus resistant and can withstand a wide temperature range. More details about the housing material will be included later in the report.

The front plate of the housing has four buttons positioned near the top, four buttons near the bottom, and one button in the middle. The buttons at the top control the exterior lights. The buttons displayed clockwise from the upper-left are the following: blackout drive, blackout marker, off, stop light, and service drive. The buttons at the bottom control the interior panel lights. The bottom buttons displayed counterclockwise from the lower-left are as follows: panel bright, dim, off, and park. The center button is the confirmation (“enter”) button.

**Pushbutton Summary**

Our switch design uses pushbuttons for operator input. The operator will use the pushbuttons to select which vehicle lights are turned on or turned off. The buttons needed to be large enough to be used easily, durable to last for a long time, and sealed to be waterproof. Using pushbuttons instead of levers simplified the mechanical design and gave the switch fewer moving parts.

The implementation of a confirmation or “enter” button replaces the function of the original switch’s locking mechanism to prevent accidental activation. The “enter” button should be pressed at the same time as the other buttons in order to change the vehicle’s exterior lighting condition. For the interior lighting, confirmation is not required; pressing the button once will change the interior lighting condition.
Electrical Summary

For the electrical portion of our design, two aspects of the light switch were considered: the power handling and the signal handling. The main change in the power handling is the addition of a voltage regulator. For the signal handling, a digital signal is interpreted and distributed by a logic controller. The design concepts for the electrical portion of the switch are summarized below.

Power Handling

The battery of the vehicle has an output of 28 volts. The present design for the Military Master Light Switch routes the 28 volts directly to each light. According to the new design requirements, the light switch should produce only a 5 volt digital signal. To accomplish this, we included a voltage regulator inside the switch. Since a voltage regulator is an electrical regulator designed to automatically maintain a constant voltage, we can use it to regulate the voltage coming into the switch down to 5 volts in order to meet the new requirements. The component that we used to accomplish this task is Murata Power Solution’s 7805SR voltage regulator. The properties of this component will be discussed in more detail later in the report.

Signal Handling

Digital signals perform as binary switches, yielding a 1 or 0. To manage the user input from the pushbuttons and control the signals, we used the XC9572 CPLD logic chip. In addition, we used a MAX7389 clock oscillator to control the timing of the signal, de-bouncing circuits and a Schmidt trigger to clean the signal, a voltage regulator, resistors, and capacitors. Each of these components will be discussed in more detail later in this report.
Housing Design

Most of the dimensions of the switch housing had to remain the same as those in the current switch design. This allows the new switch to be installed in the same place as the current switch inside the vehicles without having to redesign the vehicle’s interior. The threaded connector on the back of the switch and the pin configuration inside it could not be changed from the current design. This connector / back plate is made by Amphenol and the part number is MFR5A910. We used the current connector from a switch sample Riverside gave us in our prototype, which saved the cost of getting a new one manufactured. The things we were able to change on the housing included the depth, thickness, internal configuration, and front panel of the housing.

Shown below is a picture of the lever-style switch currently being used, followed by a picture of what the switch looks like installed in a vehicle. These pictures show that the current switch uses levers for the actuation of the switch. An unlocking lever must be turned in order to move the master switch lever to the desired position.

![Figure 1: Current Master Light Switch](Source: [12] Olive-Drab.com)
The following figure is a 3-D sketch of our housing design.

Figure 2:  Current Switch Installed In Vehicle
Source: [12] Olive-Drab.com

Figure 3:  3-D Sketch of Housing Design
In our design, we used pushbuttons instead of the levers for the actuation. The buttons are threaded into tapped holes on the front panel. They are arranged in a similar configuration to the current switch. An “ENTER” button is located in the center of the panel. Mounting holes were put on the inside of the housing for mounting the electronics.

The most inexpensive way to produce the housings in the future may be to have the housing with drilling dimples poured at a casting facility, then have the holes drilled and tapped and the mounting surfaces finished at a machine shop. 2-D prints of the housing are shown on the following two pages. For our prototype, we had a full-sized housing rapid prototyped at IPFW to test sizing, and had a metal housing machined at Johnson Precision Molds in Sturgis, Michigan. We made the housing out of a similar material to the current switch, since it has been proven to meet military specifications. The material that is used in the current switch is Aluminum alloy A206.0-T71. Johnson Precision Molds made the housing for our prototype out of 7075 mold quality hardened aluminum.

CAD drawings of the switch housing are shown below.
Figure 4: Housing Dimensions for Casting

Figure 5: Housing Dimensions for Machining
Pushbutton Details

Our design uses pushbuttons for the user’s input instead of the levers used on the original master light switch. Pushbuttons are electrical switches that close a circuit when the user pushes the button. They come in a variety of shapes, sizes, and styles and are available from several companies. The pushbuttons used in our design correspond to each of the different lighting configurations of the vehicle. In the original Master Light Switch, the user rotated levers to select which lights to turn on. In our design, each of the original lever positions is replaced by a pushbutton. There are several advantages to using pushbuttons instead of levers:

- Using pushbuttons simplifies the mechanical design of the light switch. If levers were used, separate devices such as rotary switches or sensors inside the switch housing would be required to “read” the position of the levers and relay their position to the logic controller. With pushbuttons, the switching is done within the pushbutton itself when the operator presses it and does not require additional hardware.
- With levers, there was the possibility of the levers breaking off of the light switch, as well as failure of the shafts and other mechanical devices inside. The only parts of our switch design that move are the pushbuttons, so there is less chance of mechanical failure.
- Waterproofing is made easier with pushbuttons since sealed buttons are available, eliminating the possibility of water entering the switch housing through the input controls.
- To operate the lever switch, the operator has to use both hands. Therefore, it could be potentially dangerous for the driver to use the switch when the vehicle is in motion as he or she would have to remove both hands from the steering wheel. Our pushbutton design makes it possible for the driver to use one hand to operate the switch.
- The current switch uses levers, shafts, and other mechanical devices that are all custom-made. We used standard pushbuttons and electrical devices. The only custom parts that we used for the pushbutton portion of the design were plastic covers for the buttons. This should lower the cost of production.

Figure 6: Picture of Pushbutton
Pushbutton Operation

The function of each of the original lever positions is replicated using pushbuttons. Nine buttons are required, including one button for confirmation of the user’s input. The original switch used a mechanical locking lever to prevent accidentally changing the lever positions. The pushbutton design must also provide protection against accidental activation, which is accomplished by requiring the operator to press the confirmation, or “enter,” button in addition to their lighting selection. The user will press and hold the button corresponding to the lights they wish to activate, and then simultaneously press and hold the confirmation button, keeping both held down for a short time to activate the vehicle’s exterior lights. The interior vehicle lights are not required to have protection, so the confirmation button is not used for those lights. Since a button combination is not required for the interior lighting selections, we were able to use a smaller CPLD with fewer gates that fit more easily into the housing. For the interior panel lighting the user simply presses the button corresponding to the desired lighting condition.

Construction

The pushbuttons we used are waterproof and will prevent water from entering the housing and reaching the electronics once they are threaded in. We were unable to find suitable waterproof pushbuttons with large enough buttons to be user-friendly and easy to operate. All of the buttons we found with large pushing surfaces were too large to fit onto the front panel of the housing. Nine pushbuttons were needed on the front panel of the switch, so they could not be very large. To give the user a large and easy to find surface to press, we constructed covers to go over the small ends of the pushbuttons. The user will push the cover, which will in turn press on the pushbutton beneath it. The labels indicating which lighting function each button operates are molded into the button covers themselves. Some of the threads on the pushbuttons protrude from the front panel, and a cylinder to hold the button cover screws down over the exposed threads. A circular cap, like a key on a keyboard, pushes down into the cylinder and snaps into place. The operator can then press the larger, labeled cover to use the buttons. The covers should eventually be made from a translucent plastic to let light from the LEDs shine through. The prototype uses button covers made of Duraform PA created in the rapid prototyping machine at IPFW. Drawings of the button covers are shown below.
Figure 7: Button Cover 3D Models

Figure 8: Button Cover Dimensions, Side View

Figure 9: Button Cover Dimensions, Bottom View
Illumination

We are using illuminated pushbuttons for the redesigned Master Light Switch. This will allow the user to see the buttons in the dark without a flashlight. One member of our team spoke with a serviceman who said that one of the biggest problems with the current switch is that it cannot be seen at night. Illuminated pushbuttons have a built-in LED under the button cap, which provide illumination for the switch buttons. We were told that the lighting should be red or green to be less visible from outside the vehicle, so the pushbuttons have red LEDs. Although illumination was not required for our design, we feel that it is a good improvement to the current Master Light Switch.

Pushbutton Selection

The pushbuttons that were chosen are OTTO Engineering LP3-41D322R momentary, watertight, illuminated pushbuttons. These were the only watertight, illuminated pushbuttons we were able to find that were small enough to allow nine of them to fit on the front panel of the switch. Their operating range is -55°C to 85°C, which satisfies military specifications. They have a threaded connection to screw into the housing, along with a hex nut, lock washer, and panel seal gasket to hold them in and keep the connection watertight. These accessories were not used in construction of the prototype; sealant, glue, and the button covers replaced them for size reasons. These are momentary single pull single throw (SPST) normally-open pushbuttons, which means that they function as on-off switches that keep the circuit open until the button is pressed. Since the buttons are watertight, they should still operate even after being submerged in water, which is required by military specifications. Also, the pushbuttons include a panel seal that prevents water from entering the housing through the threaded holes where the pushbuttons are installed. These pushbuttons have a built-in red LED under the button cap that illuminates the buttons on our switch, allowing the user to operate the switch at night. Since the lights are built into the pushbuttons, we did not have to mount individual LEDs on the switch for illumination, which eliminated the potential problem of waterproofing external LED connections. A diagram of the pushbutton is shown below.

![Figure 10: Pushbutton Dimensions (inches)](source: [10] OTTO Engineering Catalog pp. 44)
Electrical Design

The main goal of this project was to redesign the Master Light Switch’s electrical and mechanical mechanisms to route a 5 volt signal to the power distribution center that controls power to the vehicle lights. The earlier model of the switch routes 28 volts directly from the vehicle’s battery to the lights. The electrical design of this project forms the main core of the design goal.

After doing extensive research and study on the working mechanism of the previous switch design, several factors were considered in creating the new electrical design for the Master Light Switch. These factors include the drawbacks of the old switches that use primitive technology that limit the function of the switch and give little space for future modification. The old switch design also fails to be user-friendly when the vehicle is operated in the dark.

The electrical mechanism in the new switch design overcomes the drawbacks of the old switch design and uses modern technology to efficiently distribute a signal (5 volts) to the distribution center. The design also provides flexibility in its operation and can be intelligently controlled and automated. Light emitting diodes (LEDs) have been implemented in the design, which are controlled by the logic controller device to make the switch more operator-friendly.

To achieve the objective of the electrical mechanism, several analog and digital components have been implemented. The major components include momentary pushbuttons, de-bouncing circuits using capacitor and resistors, Schmitt triggers (MM74HC14), a voltage regulator (7805SR), a microcontroller clock generator (MAX7389), an in-system programmable CPLD (XC9572), and light emitting diodes (LEDs) built into the pushbuttons.

The input to the electrical design is received from the nine momentary push buttons mounted on the front panel of the switch. The nine pushbuttons include the All Off, Stop Light, Service Drive, Panel Bright, Park, Panel Dim, Black Out Drive, Black Out Marker and Enter commands. Due to the mechanical nature of the momentary push button, the input signal transition is not clean which introduces error in acknowledging the input signal. To correct this, an analog de-bouncing circuit with a Schmitt trigger (MM74HC14) is placed after the pushbuttons to generate a clean and clear digital signal [0-Volt (0) or 5-Volt (1)]. The combination of the analog de-bouncing circuit with Schmitt triggers eliminates the error pulses generated when the button is pressed and released and gives a straight output, clean of any errors. The input signal is then directed to the logic controller device, the in-system programmable CPLD (XC9572-15PC44I) with 44 pins. This is used as a logic controller which is programmed using an HW-130 programming kit. The complex programmable logical device (CPLD) receives a total of ten input digital signals, nine from the push buttons and one from the input pin K (connected to the brake pedal). Depending on the combination of the input signals, the logic controller device directs the output signal of 5 volts to the respective output pins (A, B, C, D, E, H, J, L, M and N) connected to the 12-pin connector at the rear of the switch.

Regulating the power supply to the circuit is another important factor in the design. The switch receives 28 volts from the vehicle battery through pin F. A voltage regulator (7805SR) is used to control the voltage to different components in the circuit. Capacitors are used in the design of the voltage regulator to provide a smooth transient response to the supply voltage. The voltage regulator provides a 5 volt power supply to the de-bouncing circuit, Schmitt Trigger (MM74HC14), Microcontroller Clock Generator (MAX7389) and the In-System Programmable CPLD (XC9572).
A Microcontroller Clock Generator (MAX 7389) produces a square wave timing signal, which is connected to the global clock signal of the CPLD (Pin 5) for use in synchronizing the circuit’s logical operations.

**De-bouncing Circuit and Schmitt Trigger**

![Debouncing Circuit with Schmitt Trigger (MM74HC14)](image)

When the momentary pushbutton is operated, spikes of low and high voltages occur across the switch due to contact bounce, resulting in a series of high and low signals rather than one clean pulse. The input signal sent to the logic controller in this manner becomes difficult to interpret because of the random and uneven transition of the signal. Hence, an analog de-bouncing circuit with a Schmitt Trigger is implemented to obtain a clean and stable input signal which is sent to the logic controller device (CPLD).

---

Figure 11: De-bouncing Circuit with Schmitt Trigger Schematic
The capacitor plays an important role in the de-bouncing circuit. It prevents the output from changing too fast, which prevents random high and low pulses from appearing in the output. When the switch contact is made, the voltage on the capacitor falls rapidly to zero. When the switch is released the potential across the capacitor is charged up again. Although the capacitor makes the voltage rise smooth and clean while rising from 0 to 5 V, the voltage passes through a range where the output is unknown, causing an error pulse in the output. Hence, a Schmitt Trigger is implemented in the circuit to keep the output unchanged until a threshold voltage has been reached. This feature of the Schmitt Trigger helps to eliminate the error in the output that is introduced when the output voltage passes through the range of 0 to 5 V. The MM74HC14 Hex Inverting Schmitt Trigger used in the current design inverts the output. Therefore, the final output signal sent to the CPLD is a clean pulse of +5 V (logic 1 - when the push button is pressed) and 0 V (logic 0 - when the pushbutton is released).

**MM74HC14: Hex Inverting Schmitt Trigger**

Connection Diagram:

The Hex Inverting Schmitt Trigger (MM74HC14) is a comparator circuit that incorporates positive feedback. When the input is higher than a certain chosen threshold, the output is high; when the input is below another (lower) chosen threshold, the output is low; when the input is between the two, the output retains its value.
The MM74HC14 is a 14-pin chip that consists of six Schmitt triggers with one pin for the power (pin 14) and one pin for the ground (pin 7). The current circuit design requires implementation of nine Schmitt Triggers; therefore, two MM74HC14 chips are used. The chips are powered using the voltage regulator.

Some of the technical specifications of the MM74HC14 for which the chip was selected for the project design are shown in the appendix.

**Voltage Regulator**

A voltage regulator is an electrical regulator designed to automatically maintain a constant voltage level. The vehicle battery supplies a voltage of 28 V to the Master Light Switch through the Pin F located on the back of the connector. The 7805SR is a 3-pin adjustable regulator. It is used to regulate a voltage supply of 5 V to the electrical components used in the circuit. The voltage regulator provides a 5 V power supply to the de-bouncing circuit, Schmitt Trigger (MM74HC14), Microcontroller Clock Generator (MAX7389), LEDs and the In-System Programmable CPLD (XC9572).

Capacitors C10, C11 and C12 are used in the circuit to improve the transient response of the signal and provide improved output impedance. The 7805SR package TO-220 provides a typical output current of 0.8 A and a minimum of 0.5 A. The output current supplied by the voltage regulator meets our system design as shown by the following analysis:

**Table 1: Voltage Regulator Current Analysis**

<table>
<thead>
<tr>
<th>SNo.</th>
<th>Electrical Components</th>
<th>Quantity</th>
<th>Operating Supply Current</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>In-System Programmable CPLD (XC9572)</td>
<td>1</td>
<td>65 mA</td>
</tr>
<tr>
<td>2</td>
<td>Schmitt Trigger (MM74HC14)</td>
<td>2</td>
<td>100 mA (50mA × 2)</td>
</tr>
<tr>
<td>3</td>
<td>Microcontroller Clock Generator (MAX7389)</td>
<td>1</td>
<td>5.5 mA</td>
</tr>
<tr>
<td>4</td>
<td>Light Emitting Diodes (LEDs)</td>
<td>9</td>
<td>150 mA (max)</td>
</tr>
<tr>
<td>5</td>
<td>De-bouncing Circuit</td>
<td>9</td>
<td>10 mA (max)</td>
</tr>
</tbody>
</table>

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Total current required</strong></td>
<td><strong>330.5 mA &lt; 500 mA</strong></td>
</tr>
</tbody>
</table>

The above calculations show that the voltage regulator meets our current requirements.
Some of the technical specifications of the 7805SR 3-Terminal Adjustable Regulator for which the device was selected for the project are shown in the appendix. The dimensions and a connection diagram for the voltage regulator are shown below.

Figure 13: 7805SR Voltage Regulator Diagrams  

Figure 14: Connection of MAX7389 and XC9572 CPLD
Microcontroller Clock Generator

A microcontroller clock generator is a circuit that produces a timing signal (clock signal) for use in synchronizing a circuit's operation. The MAX7389 is a microcontroller clock generator that sends a timing signal in the form of a simple symmetrical square wave to the complex programmable logical device (CPLD). The output frequency provided by the MAX7389 clock generator ranges from 1 MHz to 16 MHz depending on the factory programming; however, for the current design a typical output frequency of 4.38 MHz was used.

The connection diagram for the MAX7389 silicon oscillator (8-pin) to the CPLD is shown in Figure 14 above. The MAX7389 chip is powered using the output terminal of the 7805SR voltage regulator, which supplies a voltage of 5 V to pin 1 of the clock generator. Pin 4 is used for grounding the chip. The output clock signal from the MAX7389 (pin 8) is connected to the clock input pin 5 and pin 39 and of the XC9572 CPLD.

The WDS1-Watchdog Timeout Select Input 1 (pin 2) and WDS2-Watchdog Timeout Select Input 2 (pin 3) are used in setting the watchdog timeout period. However, this feature is not required for our design and hence is connected to the ground. WDO*-Watchdog Output (pin 5) WDI –Watchdog Input (pin 6) provides a safety feature with the timing signal, which is connected to the input/output port of the CPLD and programmed as necessary. If WDI does not receive a rising edge within the watchdog timeout period, the set/reset asserts. However, connecting WDS1 and WDS2 to VCC disables the watchdog timer if the safety feature is not implemented in our design.

The MAX7389 silicon oscillator replaces ceramic resonators, crystals, and crystal-oscillator modules as the clock source for our CPLD. Since we are using 5 V, this oscillator is sufficient for our design. The MAX7389 features a factory-programmed oscillator and microprocessor power-on-reset supervisor. It also has watchdog timeout values in the range of 16 ms to 2048 ms. The watchdog output provides a status indicator to help control the safety elements in the system. One of the features for which we are choosing this oscillator is because it is resistant to vibration and EMI. The high-output drive current and absence of high-impedance nodes gives the oscillator less susceptibility to dirt and humid environments. Since military vehicles operate in dirty and humid environments, this oscillator should function under these conditions.

The dimensions of the microcontroller clock generator are shown below:
Figure 15: MAX7389- Microcontroller Clock Generator Dimensions
A complex programmable logic device (CPLD) is the heart of our electrical design. It performs the logical operations. It receives the input signal and directs the respective output signal to the power distribution center as well as controlling the operation of the LEDs. It is a programmable logic device with complexity between that of PALs and FPGAs, and includes architectural features of both. The building block of a CPLD is the macro cell, which contains logic implementing disjunctive normal form expressions and more specialized logic operations. The CPLD chosen for our design is the Xilinx XC9572, which is an In-System Programmable CPLD with 44-pins. The CPLD connection schematic is shown in Figure 16.

The XC9572 is a high performance CPLD that provides advanced in-system programming and testing capabilities. It is comprised of eight 36V18 Function Blocks, providing 1600 usable gates with propagation delays of only 7.5 ns. The function block of the CPLD is shown below.
<table>
<thead>
<tr>
<th>Function Block</th>
<th>Macro-cell</th>
<th>PC.44</th>
<th>Function Block</th>
<th>Macro-cell</th>
<th>PC.44</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>3</td>
<td>1</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>9</td>
<td>3</td>
<td>1</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>1</td>
<td>11</td>
<td>3</td>
<td>1</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>1</td>
<td>13</td>
<td>3</td>
<td>1</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
<td>3</td>
<td>1</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>1</td>
<td>17</td>
<td>3</td>
<td>1</td>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>4</td>
<td>2</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>4</td>
<td>2</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>11</td>
<td>4</td>
<td>2</td>
<td>12</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>13</td>
<td>4</td>
<td>2</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>2</td>
<td>14</td>
<td>4</td>
<td>2</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>4</td>
<td>2</td>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>2</td>
<td>16</td>
<td>4</td>
<td>2</td>
<td>17</td>
<td>17</td>
</tr>
<tr>
<td>2</td>
<td>18</td>
<td>4</td>
<td>2</td>
<td>18</td>
<td>18</td>
</tr>
</tbody>
</table>

Figure 17: CPLD Function Block

Source: [19] Xilinx, Inc. XC9572 CPLD Datasheet
We chose this CPLD because it offers 3.3V and 5V I/O capability. Since we need 5 volts as our output, this CPLD works well. Some of the other features of the XC9572 In-System Programmable CPLD for which the chip was selected are shown in the appendix.
The pin configuration for the XC9572 is shown in the table below:

**Table 2: CPLD Pin Diagram**

<table>
<thead>
<tr>
<th>PIN Number</th>
<th>Connection</th>
<th>Connected To</th>
<th>Function</th>
<th>PIN Number</th>
<th>Connection</th>
<th>Connected To</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 9</td>
<td>Output-C</td>
<td>23</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
</tr>
<tr>
<td>2</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 10</td>
<td>Output-D</td>
<td>24</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 11</td>
<td>INPUT-PANEL DIM</td>
</tr>
<tr>
<td>3</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 8</td>
<td>Output-E</td>
<td>25</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 13</td>
<td>INPUT-PARK</td>
</tr>
<tr>
<td>4</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 7</td>
<td>Output-H</td>
<td>26</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 15</td>
<td>INPUT-PANEL BRT</td>
</tr>
<tr>
<td>5</td>
<td>GCK1</td>
<td>U2 MAX7389 - PIN 8</td>
<td>CLOCK</td>
<td>27</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 17</td>
<td>INPUT-ENTER</td>
</tr>
<tr>
<td>6</td>
<td>GCK2</td>
<td>OPEN</td>
<td>CLOCK</td>
<td>28</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 2</td>
<td>LED-ALL OFF</td>
</tr>
<tr>
<td>7</td>
<td>GCK3</td>
<td>OPEN</td>
<td>CLOCK</td>
<td>29</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 4</td>
<td>LED-STOP LIGHT</td>
</tr>
<tr>
<td>8</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 6</td>
<td>Output-I</td>
<td>30</td>
<td>TDO</td>
<td>U2 HDRX6 - PIN 4</td>
<td>Programming Pin</td>
</tr>
<tr>
<td>9</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 5</td>
<td>Output-L</td>
<td>31</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
</tr>
<tr>
<td>10</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>32</td>
<td>VCC IO 5V</td>
<td>U3 7805SR-ND - PIN 2</td>
<td>VCC</td>
</tr>
<tr>
<td>11</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 3</td>
<td>Output-M</td>
<td>33</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 6</td>
<td>LED-B.O. MARKER</td>
</tr>
<tr>
<td>12</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 4</td>
<td>Output-N</td>
<td>34</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 8</td>
<td>LED-B.O. DRIVE</td>
</tr>
<tr>
<td>13</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 1</td>
<td>INPUT-K (Brake pedal)</td>
<td>35</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 10</td>
<td>LED-SER.Drive</td>
</tr>
<tr>
<td>14</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 1</td>
<td>INPUT-ALL OFF</td>
<td>36</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 12</td>
<td>LED-PANEL DIM</td>
</tr>
<tr>
<td>15</td>
<td>TDI</td>
<td>U2 HDRX6 - PIN 3</td>
<td>Programming Pin</td>
<td>37</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 14</td>
<td>LED-PARK</td>
</tr>
<tr>
<td>16</td>
<td>TMS</td>
<td>U2 HDRX6 - PIN 2</td>
<td>Programming Pin</td>
<td>38</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 16</td>
<td>LED-PANEL BRT</td>
</tr>
<tr>
<td>17</td>
<td>TCK</td>
<td>U2 HDRX6 - PIN 1</td>
<td>Programming Pin</td>
<td>39</td>
<td>GSR</td>
<td>U2 MAX7389 - PIN 7</td>
<td>CLOCK SET/RESET</td>
</tr>
<tr>
<td>18</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 3</td>
<td>INPUT-STOP LIGHT</td>
<td>40</td>
<td>GT52</td>
<td>OPEN</td>
<td>CLOCK</td>
</tr>
<tr>
<td>19</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 5</td>
<td>INPUT-B.O. MARKER</td>
<td>41</td>
<td>VCC INT 5 V</td>
<td>U3 7805SR-ND - PIN 2</td>
<td>VCC</td>
</tr>
<tr>
<td>20</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 7</td>
<td>INPUT-B.O. DRIVE</td>
<td>42</td>
<td>GT51</td>
<td>OPEN</td>
<td>CLOCK</td>
</tr>
<tr>
<td>21</td>
<td>VCC INT 5V</td>
<td>U3 7805SR-ND - PIN 2</td>
<td>UCC</td>
<td>43</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 11</td>
<td>Output-A</td>
</tr>
<tr>
<td>22</td>
<td>I/O</td>
<td>CONNEC2 HDRX10- PIN 9</td>
<td>INPUT-SER. DRIVE</td>
<td>44</td>
<td>I/O</td>
<td>CONNEC1 HDRX6- PIN 12</td>
<td>Output-B</td>
</tr>
</tbody>
</table>

These are the connections for the two connectors that are on our PCB. CONNEC1 is attached to the back of the light switch and CONNEC2 is attached to the pushbuttons.
Table 3: PCB Connection Description – CONNEC1

<table>
<thead>
<tr>
<th>PIN Number</th>
<th>Connection from</th>
<th>Connection to</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PIN-K (Brake Pedal) - Input</td>
<td>U5 MM74HC14 - PIN 13</td>
</tr>
<tr>
<td>2</td>
<td>PIN-F (Battery) - Input</td>
<td>U3 7805SR-ND - PIN 1</td>
</tr>
<tr>
<td>3</td>
<td>CPLD- PIN 11</td>
<td>PIN M - Output</td>
</tr>
<tr>
<td>4</td>
<td>CPLD- PIN 12</td>
<td>PIN N - Output</td>
</tr>
<tr>
<td>5</td>
<td>CPLD - PIN 9</td>
<td>PIN L - Output</td>
</tr>
<tr>
<td>6</td>
<td>CPLD - PIN 8</td>
<td>PIN J - Output</td>
</tr>
<tr>
<td>7</td>
<td>CPLD- PIN 4</td>
<td>PIN H - Output</td>
</tr>
<tr>
<td>8</td>
<td>CPLD - PIN 3</td>
<td>PIN E - Output</td>
</tr>
<tr>
<td>9</td>
<td>CPLD- PIN 1</td>
<td>PIN C - Output</td>
</tr>
<tr>
<td>10</td>
<td>CPLD - PIN 2</td>
<td>PIN D - Output</td>
</tr>
<tr>
<td>11</td>
<td>CPLD - Pin 43</td>
<td>PIN A - Output</td>
</tr>
<tr>
<td>12</td>
<td>CPLD - PIN 44</td>
<td>PIN B - Output</td>
</tr>
</tbody>
</table>

Table 4: PCB Connection Description – CONNEC2

<table>
<thead>
<tr>
<th>PIN Number</th>
<th>Connection from</th>
<th>Connection to</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ALL OFF- Input Push Btn</td>
<td>U5 MM74HC14 - PIN 11</td>
</tr>
<tr>
<td>2</td>
<td>CPLD- PIN 28</td>
<td>ALL OFF- LED</td>
</tr>
<tr>
<td>3</td>
<td>STOP LIGHT- Input Push Btn</td>
<td>U5 MM74HC14 - PIN 9</td>
</tr>
<tr>
<td>4</td>
<td>CPLD-PIN 29</td>
<td>STOP LIGHT- LED</td>
</tr>
<tr>
<td>5</td>
<td>B.O.Marker- Input Push Btn</td>
<td>U5 MM74HC14 - PIN 5</td>
</tr>
<tr>
<td>6</td>
<td>CPLD - PIN 33</td>
<td>B.O. Marker LED</td>
</tr>
<tr>
<td>7</td>
<td>B.O. Drive- Input Push Btn</td>
<td>U5 MM74HC14 - PIN 3</td>
</tr>
<tr>
<td>8</td>
<td>CPLD - PIN 34</td>
<td>B.O Drive-LED</td>
</tr>
<tr>
<td>9</td>
<td>SER.DRIVE- Input Push Btn</td>
<td>U5 MM74HC14 - PIN 11</td>
</tr>
<tr>
<td>10</td>
<td>CPLD - PIN 35</td>
<td>SER. DRIVE-LED</td>
</tr>
<tr>
<td>11</td>
<td>PANEL DIM- Input Push Btn</td>
<td>U4 MM74HC14 - PIN 11</td>
</tr>
<tr>
<td>12</td>
<td>CPLD - PIN 36</td>
<td>PANEL DIM-LED</td>
</tr>
<tr>
<td>13</td>
<td>PARK- Input Push Btn</td>
<td>U4 MM74HC14 - PIN 5</td>
</tr>
<tr>
<td>14</td>
<td>CPLD - PIN 37</td>
<td>PARK-LED</td>
</tr>
<tr>
<td>15</td>
<td>PANEL BRT- Input Push Btn</td>
<td>U4 MM74HC14 - PIN 3</td>
</tr>
<tr>
<td>16</td>
<td>CPLD-PIN 38</td>
<td>PANEL BRT-LED</td>
</tr>
<tr>
<td>17</td>
<td>ENTER- Input Push Btn</td>
<td>U4 MM74HC14 - PIN 1</td>
</tr>
<tr>
<td>18</td>
<td>VCC</td>
<td>ENTER- LED</td>
</tr>
<tr>
<td>19</td>
<td>GND</td>
<td>GND to PUSH BTN’s</td>
</tr>
<tr>
<td>20</td>
<td>VCC</td>
<td>VCC to LED’s</td>
</tr>
</tbody>
</table>

*LED for the ENTER button will always be turned on.
The physical dimensions of the CPLD as well as a picture are shown below.

**Figure 19: CPLD Dimensions**
Source: [15] The Programmable Logic Data Book pg. 630

**Figure 20: Picture of CPLD**
Source: [1] EAS Electronics
Light-Emitting Diodes (LEDs)

A light-emitting diode (LED) is a semiconductor diode that emits incoherent narrow-spectrum light when electrically biased in the forward direction of the p-n junction. This effect is a form of electroluminescence. The electrical design for the project implements nine light emitting diodes (LEDs) connected to the output port of the CPLD. The LEDs are built into the pushbuttons to make the buttons visible in the dark. This feature makes the Master Light Switch more user friendly than the older version of the switch.

LEDS are implemented by connecting the anode end of the LED to the 5 V power supply sent by the voltage regulator while connecting the cathode end to the input/output port of the CPLD with a 330 Ω resistor in the middle (for protection of the circuit).
Xilinx HW-130 Programmer

The HW-130 Programmer is the programming kit that was used to program our CPLD to perform the necessary logical operations. A picture of the kit is shown below.

![Figure 22: Xilinx HW-130 Programmer](source)

The HW-130 Programmer comes with the following items:

- RS-232 9-to-9-pin data cable and 9-to-25-pin adapter
- 25-pin female adapter for PC and IBM workstation users
- 25-pin male adapter for Sun workstation users
- Universal Power Supply
- Power Cord
- Software
- User Guide
- Vacuum Handling Tool Kit
The HW-130 Programmer consists of a base with a 96-pin connector on top of the case to attach the different socket adapters for each device and package type.

On the device, only 5 pins are required for the programming interface. The JTAG pins TDI, TDO, TCK, and TMS of the XC9572 CPLD are used for addressing the elements to be programmed, for shifting the programming data, for verification, and for testing. The VCC pin provides a high voltage during programming and is used to measure the resistance of the programmed anti-fuse. The logical operations that are programmed in the CPLD are defined by the following truth table:

**Table 5: Input-Output Truth Table**

<table>
<thead>
<tr>
<th>INPUTS</th>
<th>OUTPUTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>VEHICLE BRAKE PEDAL</td>
<td>STOP LAMP SWITCH</td>
</tr>
<tr>
<td>SER. DRIVE †</td>
<td>SERVICE STOP LAMP</td>
</tr>
<tr>
<td>STP LIGHT</td>
<td>BLACKOUT DRIVING</td>
</tr>
<tr>
<td>B.O. MARKER</td>
<td>B.O. LAMP</td>
</tr>
<tr>
<td>B.O. DRIVE</td>
<td>SER. TAIL/CL. LAMP</td>
</tr>
<tr>
<td>ALL OFF</td>
<td>SER. TRN. IND.</td>
</tr>
<tr>
<td></td>
<td>B.O. STOP LAMP</td>
</tr>
<tr>
<td>INPUT FROM THE PUSH BUTTON</td>
<td>A</td>
</tr>
<tr>
<td>1 1 0 0 0 0 0 0</td>
<td>0 1 1 0 0 0 1 1 0</td>
</tr>
<tr>
<td>1 0 1 0 0 0 0 0 0</td>
<td>0 1 1 0 0 0 1 0 0</td>
</tr>
<tr>
<td>1 0 0 1 0 0 0 0 0</td>
<td>0 1 0 0 1 0 0 1 0 1</td>
</tr>
<tr>
<td>1 0 0 0 1 0 0 0 0</td>
<td>0 1 0 1 1 0 0 0 1 0</td>
</tr>
<tr>
<td>X 0 0 0 0 0 1 0 0 0</td>
<td>0 1 0 0 0 0 1 0 0 0</td>
</tr>
<tr>
<td>0 1 0 0 0 0 0 1 0 0</td>
<td>0 1 0 0 0 1 0 0 0 0</td>
</tr>
<tr>
<td>0 0 1 0 0 0 0 0 1 0</td>
<td>0 1 0 0 0 0 1 0 0 0</td>
</tr>
<tr>
<td>0 0 0 1 0 0 0 0 0 1</td>
<td>0 1 0 1 1 0 0 0 0 0</td>
</tr>
<tr>
<td>0 0 0 0 0 1 0 0 0 0</td>
<td>0 1 0 1 1 0 0 0 0 0</td>
</tr>
</tbody>
</table>
**Notes:**

SER DRIVE † → Service Drive Button in the main switch also controls the functions/behaviour of the auxiliary switch

X → Does not matter if the brake pedal is pressed or not.

1* → Master Light Switch is digital in nature, which means Pin B would be receiving either 0 V (0-low) or 5 V (1-high). Hence the Brt, Dim and Park input button would result in the same output (more details about the input-output interference is needed to solve this technical problem).
A flowchart showing the switch behavior is shown below.

Figure 23: Flowchart
A complete electrical schematic of the design is broken up into several figures and shown below. The complete schematic in one figure is shown in the appendix.

Figure 24: Power Supply Schematic
Figure 25: Clock and Schmidt Triggers Schematic
Figure 26: Connectors Schematic
Figure 27: CPLD Schematic
User Instructions

The new Master Light Switch is installed in a vehicle the same way as the old switch. The back connector should be threaded in and four bolts at the corners secure it to the dashboard. Each button on the new switch duplicates the function of a switch position in the old switch. Instead of using the “Unlock” lever of the old switch, the main lights are selected by pressing the lighting button and the “Enter” button at the same time. To select auxiliary lights, the “Enter” button does not need to be used; only the lighting button is pressed. To turn off the main lights, the “All Off” button should be used with the “Enter” button. To turn off the auxiliary lights, press the chosen auxiliary lighting button a second time. The switch indicates which lights are turned on by illuminating the button with a red glow.

Red indicates main buttons – require simultaneous pressing of “Enter” button
Blue indicates auxiliary buttons – press once for on, again for off

Figure 28: Button Diagram
Design Modifications

Over the course of the semester, several modifications had to be made to our initial design. These changes are summarized here.

- **Size restriction**
  - The initial planned electrical design was supposed to use an 84-pin CPLD as the logic controller device. While designing the basic electrical schematic using Multisim, the size and dimensions of the parts were not seriously considered in the design factor and with basic research it was assumed that the size of the parts would meet our design requirements. However, with detailed design of the printed circuit board using Ultiboard it was realized that using an 84-pin CPLD would occupy too much space, leaving very little space for the other parts to fit on the circuit board and minimum space for routing traces (as shown in the diagram below).

![Comparing chip dimensions (showing the size restriction)](image)

- To solve the problem, the 44-pin CPLD was reconsidered in the electrical design with certain modifications. The reason 44-pin CPLD was not utilized in the first place is that it would be short of one input/output pin. Using a 44-pin CPLD would mean a certain compromise in the initial planned electrical design. After group discussion, the input/output connection of the LED for the ‘Enter’ button in the updated design was removed. The main motive behind using the LED’s is to make the switch user friendly by making them aware of the switch status. However the CPLD controlled LED for the ‘Enter’ button is only used to confirm the activation of other buttons to prevent accidental pushing of the switch buttons.
• Change in the Voltage Regulator and Potentiometer
  o The voltage regulator used in our initial design was an LM317AH-ND. The voltage regulator was tested in the breadboard for its performance and functionality. Connections were made and it was observed that the voltage regulator was able to maintain a steady output voltage level of around 5 V with the input voltage being 28 V and the potentiometer adjusted to 725Ω-735Ω (although the theoretical value for the potentiometer was expected to be around 820Ω-830Ω). The test results were successful as the input voltage of 28 V was dropped down to 6.8 V, and the output voltage was still maintained at 5 V. However with the construction of the PCB, the voltage regulator did not function as expected. The new voltage regulator 7805SR was used in place of LM317AN-ND. Also, the potentiometer used in the earlier design 490-2031-1-ND (1 KΩ, 2 mm) was removed from the system because the 7805SR did not require one.

• The print that was provided to us was different from the actual model being used.
  o The outer edge of the housing which was provided to us went to the extents of the mounting flange, whereas the print that was provided to us had an outer edge that was about 1/8” from the extent of the mounting flange. If we would’ve known that we could’ve made our outer dimensions larger, we could’ve fit the electrical components in the switch better and reduced the overall height of the switch.

• Since we designed our housing based on the outdated print, the smaller size made it difficult to fit all of our electrical components.
  o As explained earlier, an 84-pin CPLD would not fit well, so a 44-pin chip had to be used. Also, the pushbuttons were very long, so the housing had to be elongated to be sure there would be room for them, the PCB board, and the wires connecting them. All of the electrical parts had to be placed on a board that would fit inside the housing, which was a challenge.
Section II:

Building Process
PCB Development

The PCB design was more complicated and in-depth than what was originally planned. The first step in designing the PCB was to use a program called Multisim. The components were added from the master database and put on the schematic page. The data packages for each component were selected in Multisim according to the size of the Military Master Light Switch. Since we couldn’t change the housing of the light switch, we had to design the PCB to be as small as possible. Most of the components that were assembled in Multisim were surface mount parts. After the parts were placed in Multisim, the wires were attached as depicted by Figure 24.

From here, the file was imported into a program called Ultiboard. Ultiboard placed the components on a PCB, which the user chose the size and shape of, and then connected the traces and vias accordingly. This was the biggest challenge in the design. Some of the traces needed to be smaller than others since there was limited space on the PCB. Also, the traces that connected to our pushbuttons were made as large as possible so that a large current would flow through. After all of the components were placed and attached, a DRC Netlist and Connectivity check was run to ensure all of the components were attached properly and error-free. From here, Gerber files were created and the files were sent to a machine on the second floor of the Engineering Building. The figures below show 3D views and 2D views of the PCB.

After the PCB was created, the components needed to be placed on it. The tools that were involved were a microscope, tweezers, pliers, glue, a solder station, and an exact-o knife. After these parts were placed on the PCB, it was baked using a special oven and temperature. Any parts that were not baked were soldered on manually. Figure 35 shows the finished version of the PCB. While the PCB was being created, a VHDL code was being generated for the Military Master Light Switch operation. Once the PCB was finished, the VHDL code was programmed on our XC9572 CPLD. The next step was the functionality testing process which will be explained more in section III of the report.
Figure 31: 3D picture of bottom side of Ultiboard PCB

Figure 32: Picture of Bottom of PCB
Figure 33: 3D view of top side of Ultiboard PCB

Figure 34: Another 3D view of top side of Ultiboard PCB

Figure 35: Finished PCB
Button Cover and Housing Construction

The button covers were designed in AutoCAD to fit over the pushbuttons and operate similarly to a keyboard key. There is a cylindrical base that screws onto part of the pushbutton that protrudes from the housing and a top section that snaps into the base. The snaps slide in a groove in the base, allowing the button cover to be pressed by the user to actuate the pushbutton underneath. The pushbutton then moves the top of the cover out so it is ready to be pressed again. The button covers for our prototype were made in the Engineering Building at IPFW using the rapid prototyping machine. The technician, Jason Davis, used our CAD files to program the machine to make the button covers as well as a first prototype of the housing. He was able to fit these in with other builds in the machine, greatly reducing our cost. The rapid prototyped parts are made from DuraForm PA, a material that starts as a powder but hardens in the rapid prototyping machine into the shapes of the programmed parts. When the rapid prototyped parts were finished, we had to scrape out the residual powder in the grooves and holes in the parts. The button covers were used in our finished prototype, but the rapid prototyped housing was only used to make sure all of the electrical components fit inside and verify the design before having a metal housing made.

Figure 36: Button Cover Pictures
Our sponsor arranged for us to have a housing made out of aluminum by Johnson Precision Molds in Sturgis, Michigan. This was the most substantial expense for our project, costing $2500. It took several weeks to get in contact with them, give them our CAD drawings, and have the part made. When it was finished, we still had to purchase a tap to thread the pushbutton holes and drill out and tap some holes on the inside to attach the PCB board. For the back connector of the switch, we were able to reuse a back connector from an old lever-style switch given to us by our sponsor. We designed the switch housing using a military specification sheet (MS 5113), but found out that the size of the rear of the housing in the old switch we were given was slightly larger than the size specified in MS 5113. Therefore, we had to machine the back connector down to fit our housing, which is slightly smaller in the rear than the old switch.

Figure 37: Rapid Prototyped Housing

Figure 38: Final Metal Housing
Figure 39: Machined Back Plate

Figure 40: Inside and Outside Views of Housing

Figure 41: PCB Inside Housing and Standoff Used to Support PCB
Assembly

To make installation easier, the ribbon cable’s wires for the pushbuttons each had sliding connectors soldered to the ends that were pushed over the terminals on the backs of the pushbuttons. The pushbutton threads were covered with sealant and threaded into the housing. Using a threaded bar, standoffs were attached to the inside of the housing to hold the PCB. These were cut to be approximately 1.5 inches tall. The PCB board was trimmed using a Dremel rotary tool to fit into the housing because the rounded edges were slightly too long and fit very tightly. A ribbon cable was attached to the inner wires of the back terminal and attached to the PCB. The pushbutton ribbon cable was also attached to the PCB. A ground wire was soldered from the PCB to the housing, and the PCB was put in place inside the housing and secured to the standoffs with screws. The back connector was put in place and sealed shut with sealant. Finally, the button covers were glued over the buttons in their proper positions. A picture of the housing with the pushbuttons uncovered shows that adding button covers makes the buttons much easier to use. Also, pictures of the finished prototype are shown below.

Figure 42: Housing Without Button Covers

Figure 43: Pushbuttons Attached to Ribbon
Figure 44: Finished Prototype Angled View

Figure 45: Finished Prototype Front and Side Views
Difficulties

There were many difficulties involved throughout the senior design project. Presented below is a list of some of the difficulties that our group faced, summaries of why we thought they were difficult, and changes or actions required to overcome them.

- Getting familiar with Ultiboard V.10
  - Ultiboard is the PCB layout application of National Instruments Circuit Design Suite, a suite of EDA (Electronics Design Automation) tools that assists in carrying out the major steps in the circuit design flow. Ultiboard is used to design printed circuit boards and prepare them for manufacturing. The Printed Circuit Board (PCB) is designed using Ultiboard V.10. However, like other engineering software, this particular software was not a part of our course requirements. Also we had limited resources to learn more about the software and get familiar with its features.
    - Professor Peter Goodman and his online notes about Ultiboard were of tremendous help to get familiar with the software.

- Creating/Updating footprints of the parts in Ultiboard
  - Footprint refers to the physical dimensions of digital or analog parts used in creating the PCB design in Ultiboard. In other words, it is a set of copper pads on the PCB physically laid out to match up with a particular component package. The master and corporate database of the software contains footprints of several parts which are assigned by special codes. However several parts that were ordered in the initial batch lacked proper datasheets and footprints of these parts were designed by mere physical inspection. Later it was realized that a few parts didn’t match the proper/exact physical dimensions. Although it did not create a major problem in the design, it took extra time to make proper adjustments and changes.

- Surface mount parts too small for breadboard testing
  - For every electrical design, it important to carry out a preliminary test to make sure that the final solid model of the design works properly. In our initial design plan, breadboard testing was part of our design procedure. However, with lack of resources for testing surface mount parts, breadboard testing was not feasible. It was then planned that the overall circuit would be tested after the circuit board was built in the IPFW lab.
• Getting proper information about the input/output interface
  o This project was concerned with re-designing the electrical mechanism of the Military Master Light Switch to efficiently send a signal (5-volt) to the distribution center. The basis of the project and its input-output interference details were planned by analyzing the old switch design which was analog in nature. The new switch design which uses pushbuttons is digital in nature and routes 0 V or 5 V to the respective distribution center as needed. However, after going through the truth table of the old switch design functions to make a flowchart description about input-output interference for the current pushbutton switch, a new problem was noticed. The old switch uses one pin to control the panel lights (Pin B). However, Pin B is controlled by 3 buttons in the auxiliary switch (Panel Dim, Panel Bright, and Park). Owing to this fact the system has to have a way to make the panel lamp dim or bright with only 1 pin controlling this function. Since our design is different in its internal mechanism (sending a digital signal of either 0 or 5 V), pin-B would be receiving the same information when either the panel bright button or the panel dim button was pressed.

![Potential problem in input-output interference for the new design](image)

Figure 46: Panel Bright/Dim Problem
Source: [16] MIL-PRF-11021H

• Intrinsics details about the components overlooked
  o Although ample time and effort was put forth by the electrical engineering team working on the project, some fundamental information about the parts went unnoticed which created some problems in the electrical design. However necessary steps were taken by the team to rectify the problems.

• Programming the CPLD
  o One of the major concerns in the design process was programming the complex programmable logic device (CPLD) using the VHDL code. The electrical engineering team involved in the project did not have much experience with VHDL programming. The team could have used a computer engineering
member. However, with Dr. Guoping Wang’s generous help, time, and effort, the CPLD was programmed using the VHDL code and the electrical design of the project was carried forward.

- Placing digital/analog components on the printed circuit board
  - The dimensions of the board were comparatively small with a width of around 2 inches and length approximately of 3 inches. The delicate components placed on the board were mostly surface mount with some of their dimensions being around 1 mm or 2 mm or more. Also, owing to size restrictions, traces at some parts of the circuit board were only 10-15 mils thick. This added an additional challenge to properly place the parts on the PCB. A microscope was used to see the traces on the PCB to put the parts on.

- Getting the parts made by a certain date.
  - We dealt with multiple problems with the components having a small space to put them on. It was a challenge to find a small enough data package with the certain parameters that were necessary. For each PCB to be made, the person had to have a full board in order to make it. This caused the PCB to take a lot longer than expected. Also, different bits on the machine used to make them broke, causing a bigger delay in the PCB construction. It also took several weeks to have the switch housing machined.

- Getting the parts shipped to us on time.
  - Every order had to go through the secretary of the engineering department. This meant we could only order the parts during the times she was at school. Ordering them and getting them shipped to us took some time also because the companies had to process the order and ship it out.

- Deep going back to Nepal.
  - When Deep left to go back to Nepal, this left just one electrical engineer to complete the remaining electrical sections for the Military Master Light Switch. He was still able to help through email, but the separation left a lot of the hands-on work to one person.

- Communication with Jimmy and Riverside MFG.
  - It seemed like it was a challenge to get information and communicate with Jimmy and Riverside MFG when we really needed it.

- Communication with the companies that we got free samples from.
  - Most companies were more than happy to give us samples of their products; unfortunately, it took a long time for our parts to be delivered. For example, the pushbutton switches were ordered in December, but did not arrive until March.
The company that makes the pushbuttons, OTTO Engineering, said they would send us the pushbuttons in early December, but we had to repeatedly email and telephone the company to remind them to send us the pushbuttons. They were not shipped to us until March.

- Assembling the PCB.
  - Most of the components that we used were surface mount. They were very hard to solder and put on the PCB. A microscope had to be used to see the traces on the PCB to put the parts on.

- Finding meeting times that our whole group could agree on.
  - Everybody in our senior design group participates in activities other than school. Most of our group works at least 30 hours a week and one member is part of the tennis team. With this situation and attending class, it was a challenge to find meeting times that worked for our whole group. Most of the time we had to meet on the weekends to work on our senior design project.

- The rapid prototype machine left a thick residue on the button covers and initial housing that we had made.
  - This meant that we had to clean and file down the button covers so they would fit together properly. Any grooves or holes designed into our parts were filled with the powder used in the rapid prototyping machine which had to be removed.

- The pushbuttons used a non-standard thread size, which made finding a tap to thread holes for them difficult.
  - It took a very long time to find a place to buy a tap that would thread holes for the pushbuttons, which delayed the construction process. The buttons required a $15/32 \times 32$ tap which was very hard to find.
Section III:
Testing and Analysis
The first step for the PCB functionality testing was to program our CPLD. Using the HW-130 programmer and a VHDL code, the CPLD was programmed. The second step was to apply a 28 volt source to the PCB. Here we used a power supply from the ET 307 lab. Using LEDs to simulate the exterior lighting of the military vehicle, each pushbutton was pressed to check and see if the VHDL code and the PCB worked properly. Since we had the final two PCB’s professionally made, testing each trace and via was already done by the company that made them. After turning on the power, it seemed that two of the push buttons lit up automatically. This was not how our project was designed to operate. The first step to fix this was to check the VHDL code to make sure that no mistake was made. It was determined that the software part was correct. The next step was to solder a ground wire onto the PCB and hook it up to an oscilloscope. Once hooked up to the oscilloscope and turned on, the problem was fixed for the time being. The next day the PCB was checked again to ensure it was working properly. Unfortunately, the same two lights turned on when the voltage was applied. In order to fix this problem, a 10 uF capacitor was soldered from a certain trace of the PCB and then grounded. Also, a trace was cut out based on the reset of the clock oscillator that was not functioning properly. Figure 47 below shows the functionality testing stage of the PCB and VHDL code.

![Figure 47: PCB Functionality Testing 1](image-url)
Figure 48: PCB Functionality Testing 2

Figure 49: Cut Trace
Pushbutton Instructions

This is how the pushbuttons should be operated:

- Initialize
  - Press the desired mode function and press the ENTER button at the same time.
  - After the mode is turned on, press one of the three auxiliary switches (DIM, PARK, or BRIGHT).
    - Press the DIM, PARK, or BRIGHT buttons again to turn off the auxiliary lights
  - When ready to disable all of the lights, press the ALL OFF button and press the ENTER button at the same time.

Pushbutton Functionality Testing

In order to determine if the pushbuttons would work properly under a 5 volt load, some functionality testing needed to be done. We connected a pushbutton to a 5 volt voltage source. Then we added a 330 Ω resistor to the breadboard and grounded the pushbutton. After turning on the function generator to 5 volts, the pushbutton lit up. Figure 50 shows the functionality testing of the pushbuttons.

Figure 50: Pushbutton Functionality Testing
Vibration Test

Components in military vehicles are subject to a lot of vibration. Although vibration testing that would simulate military tests is beyond what we are capable of, a simple vibration test was performed on the PCB board. The board was bounced on a string at high speeds for an hour, and checked for damage. After the test, all parts on the PCB were still secure, so the test showed that the electrical board’s components were attached well enough to withstand some vibration.

Several setups for a vibration testing station were experimented with. First, a small DC motor from an old tape player was used. The end of a craft stick was glued to the pulley attached to the motor shaft. A string was through a hole in the craft stick and passed through a hole in the motor’s mounting bracket above and away from the pulley. The PCB board would then be hung on the string to shake up and down. This setup is shown in Figure 51. The idea was for the string going out of the bracket to be bounced up and down repeatedly as the motor ran, without the string becoming wrapped around the motor shaft. However, with a light load attached to the string, the string became twisted so much that it shortened significantly, and the test was halted. In addition, the motor was not strong enough to move the PCB, so another setup was created.

![Initial Vibration Testing Device](image)

For a second test setup, a small exhaust fan motor was rigged using clamps and wood blocks the same way as the DC motor. However, the string became twisted the same way as with the DC motor. The string could not be tied to the piece of wood being rotated by the motor. To vibrate the PCB, the string had only to move up and down. The string was stretched in front of the spinning arm and around a dowel. The arm was supposed to push the string each time it went around, which would pull it over the dowel slightly and vibrate the PCB. However, the string caught on the arm and became wrapped around it. Several variations, including a plastic cam and horizontal and vertical mounting of the fan were unsuccessful. A problem with this fan motor that could not be solved was that it ran far too fast for good results; the string would still be bouncing when the motor spun around again and it did not work well.

The solution was found using a rock tumbler for the motor source. The rock tumbler motor had a shaft extending from it intended for a grinding wheel that ran at 60 rpm. A wooden arm was cut using a scroll saw and holes were drilled in the ends for mounting to the motor and attaching a small dowel. A string was stretched in front of the arm, secured at one side of the
rock tumbler, and draped over a dowel supported on the other side with the PCB tied at the hanging end. A small dowel extending from the arm lifted the string with each revolution, making the PCB bounce up and down. The dowel was covered with tape to make it smooth and reduce friction, making it easier for the dowel to lift the string without getting slowed down. A lot of tuning was needed to position everything so the string did not bounce and get caught or tied up on any of the parts. The rock tumbler’s electrical plug was put into a remote control device so it could be easily turned on and off and the system was run for one hour. The PCB bounced up and down and shook from side to side as it moved, but all parts remained secure. This test showed that the electrical board could withstand some vibration. The final PCB board used in the prototype was not used in the vibration test because there would not be time to fix any damage before the end of the semester, but the board used in the test was a previous version of the PCB that was very similar. The entire switch could not be tested this way because the motors the group had access to were not strong enough to lift the weight of the whole prototype. A picture of the final setup is shown below.

![Final Vibration Test Station](image)

Figure 52: Final Vibration Test Station
Thermal Test

One of the concerns with our design was that the PCB would put off a lot of heat. The PCB has a voltage regulator that takes 28 volts and converts it to 5 volts. This component gets very hot when power is running through it, so we wanted to ensure that it would not affect the functionality of the switch.

To test the thermal characteristics of the switch, we used type J Omega thermocouples. We thermal pasted one thermocouple to the hottest part of the PCB, the voltage regulator. Another thermocouple was mounted to the chip and a final thermocouple was placed in a location in the housing where the buttons would be to see how hot the inside of the housing would get. We did the thermal testing with the rapid prototype model. Since the thermal conductivity of this material is very low compared to the aluminum housing, it could be considered a worst case scenario. The following figure shows the PCB with the thermocouples and power wires mounted to it.

![Figure 53: TC Layout](image)

Once the thermocouples were mounted to the PCB, we inserted the PCB in the housing and sealed up the housing to trap the heat. We connected the positive and negative wires to the power supply. We used a DAQ signal conditioner and WinDAQ software to collect the thermal data. Before we did any testing we had to calibrate the channels. We did this with an Omega Thermocouple Simulator and the calibration mode in the WinDAQ software (refer to the following figure).
When the calibration was complete, we connected the thermocouples to the appropriate channels and then began recording data. To record data we simply started recording when all of the thermocouples were at ambient temperature (~69 °F), turned on the power supply then continued to record until all of the thermocouples reached steady state. We repeated this process six times. After the data collection, the wdq files, which were the raw data files, were converted to csv files, which was used to process the data through Excel, and the following plot was generated.
Figure 55: Temperature graph

One can see from the plot that the voltage regulator gets very hot (~300 °F), and the chip gets somewhat warm (120 °F) but neither of these generated enough heat to cause the inside of the housing to get hot. The hottest that the inside of the housing got after 40 minutes of testing was 82 °F.

Based on these tests, we can say that the inside of the housing, where the wires and buttons are located, will not get hot enough to cause any damage.
Analyses

Housing Analysis

Due to the properties of the material used for this military master light switch, the housing is the part of the switch that will have the least risk of failure.

Face Plate Analysis

The face plate analysis was done by using Ansys, a program that helps to solve FEA problems. The first step in this analysis was to draw an area equal to the face plate of the switch. This drawing was based on the dimensions of our prototype Figure # 4. Once the figure was drawn, the physical characteristics of the material were loaded into this program (i.e. modulus of elasticity and Poisson’s ratio). Consequently, the loads were applied, in this case a concentrated load of 25 lbs (in compression) was applied to the top and bottom of the plate. The results of the calculations are shown below in Figure # 56. This figure shows the location and the magnitude of the stresses in the face plate.

Figure 56 Stress Analysis Result
Button Cover Analysis

The button covers designed for this project were analyzed mathematically to determine the loads they can be expected to withstand in use. Five loading conditions were analyzed: a load on the edge of the top of the cap, a compressive load on the cap only, a compressive load on the base only, a bending moment on the button cover assembly, and shear force in the button cover assembly. All calculations are shown in the appendix. A summary of the results is presented in the table below. Note that these values are for the prototype button covers made of DuraForm PA plastic in the rapid prototyping machine at IPFW.

<table>
<thead>
<tr>
<th>Sketch Analysis</th>
<th>Failure Load (lb)</th>
<th>Failure Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cap Edge Load</td>
<td>16</td>
<td>Bending</td>
</tr>
<tr>
<td>Cap Compressive Strength</td>
<td>286</td>
<td>Buckling</td>
</tr>
<tr>
<td>Base Compressive Strength</td>
<td>356</td>
<td>Buckling</td>
</tr>
<tr>
<td>Assembly Bending Moment</td>
<td>108</td>
<td>Bending at Section 1</td>
</tr>
<tr>
<td>Assembly Shear Force</td>
<td>76</td>
<td>Shear Between 3 and 4</td>
</tr>
</tbody>
</table>
The allowable load on the edge of the cap is only 16 lb because the top part of the cap is very thin. However, it does not extend far past the inner cylinder and the load would be distributed mostly over the center portion of the cap in normal use so there is little chance of breakage. All of these values correspond to the rapid prototyping material, so a stronger material could be used in production that would allow higher loads. An analysis was not performed on the situation of pulling on the button covers (an axial tensile load) when they are attached to the housing. This would involve the threads in the base, any glue used to help attach them to the housing, shear of the angled snap holding the cap in the base, friction helping to prevent bending of the snap, and displacement and tension of the flexing portion of the snap. This analysis is too complicated to be accurate, so physical testing is recommended. We did not perform this test this semester on the prototype we constructed because we only have one set of button covers. This test could break one, and we need the entire prototype intact at the end of the semester. For the same reason, a test was not performed to determine the maximum pushing force the buttons can withstand. The company that manufactures the buttons was contacted, and they reported that they have not tested their buttons for the maximum push they can withstand without breaking. They said that when their buttons are pressed, the plastic cap will hit a stop in the bezel when it is pushed down fully. The full force is not put on the internal mechanism, which protects it. Therefore, the plastic button will break before the mechanism is damaged, so the pushbuttons themselves should withstand a large force.

If this switch were to go into production, a different plastic would be used for the button covers. A transparent high-temperature polycarbonate might work well. If a polycarbonate that is stronger than the rapid prototyping material was used, the button covers could withstand higher loads. They would also let more light from the pushbuttons through, which would make the switch easier to see in the dark.
Section IV:

Evaluation and Recommendations
Design Evaluation

The main advantage of the current design over the old design is that the amount of power going through the switch was greatly reduced. This eliminated the need for fuses within the switch and the risk of an electrical fire within the switch. Another advantage of the current design is that it utilizes pushbuttons instead of the levers which were used previously. This makes operation of the switch easier for the user, since they will only have to use one hand instead of two. Also, we made use of dim led lights in the buttons, so that the user can see the switch when it is dark as they were unable to previously. Finally, the old switch had many mechanical devices that had the opportunity to fail whereas the new design has very few moving parts.

Design Recommendations

Some future design recommendations are as follows:

- New plastic for button covers
  - The material that we used for the button covers is not intended to be the material of the final design. A transparent high-temperature polycarbonate might work well. This would make the covers more durable and allow more light to shine through. The machine shop that made our housing, Johnson Precision Molds, specializes in making parts like these and would likely be a good choice for a supplier of the covers.

- Shorten pushbutton connections
  - Since we did not have intricate soldering tools at our disposal during assembly we decided to make connectors that would fit to the terminals on the back of the buttons to make assembly easier for us. These connectors took up roughly ¾” in the housing that would otherwise be freed up. We recommend that the wires be soldered directly to the terminals in the final assembly process. This would give more room inside the housing for the wires and PCB board.

- Find supplier for back-plate / connector assembly
  - For our design we used the connector from the old design. We machined the outer rim of the back-plate so that it would fit in the housing. The fact that we had the connectors on the back of the buttons made it impossible for the connector to screw directly onto the mounting holes that we designated for mounting the connector. With the aforementioned step taken of soldering the wires directly to the button terminals there will be enough room for the connector to mount in the correct location.

- Button recommendations
  - We thought that using a two stage button may be beneficial, so that every button is always illuminated, making the switch easier to see. When a certain button is selected that button would go to a brighter or different color stage.
  - Smaller or simpler buttons may also be an advantage in the future. The buttons that we used consumed a significant amount of space inside of the housing.
Using smaller buttons would allow for the use of a smaller housing. Also, the buttons that we used were fairly expensive, roughly $9 apiece. Using a cheaper button could greatly reduce the cost of production.

Course Recommendations

- We needed to have one more electrical engineer.
  - Since we couldn’t change the size of the housing, this made the mechanical engineers have a limited role in this project. What needed to be changed for this project was based around an electrical background. Our group only consisted of two electrical engineers, one of which had to leave to Nepal before the project was finished. Our group would have felt more comfortable if we had three electrical engineers instead of two.
- Senior design should start in the spring.
  - If senior design would start in the spring, this would give students the whole summer to work on the project as well as the fall. In order to produce the best prototype for each project, a sufficient amount of time is needed. Adding the summer into senior design would give each project more time.
Conclusion

Now that we have researched, detailed, and built a prototype of our final design, the next step will be to give Riverside MFG all of our information for the project and let them start the next phase of the design. The next phase of the design will be to do further testing and hopefully get the Military Master Light Switch into production. We had some difficulties during this project, but we were able to solve them and construct a complete, working prototype. With hard work and a lot of time and effort, we were able to successfully complete our prototype. Hopefully our sponsor will be able to use our prototype as a model for future production. We are proud of the work we have done and hope that it gives Riverside and the military a new product superior to previous designs.
References


Appendix I: Analysis of Button Covers

Button Cover: Cap Edge Load

- The calculations show that the maximum allowable load that can be applied pushing down on the very edge of the cap is 16.5 lb. The edge will first break off due to bending stress.

- The allowable bending stress is 0.6 times the ultimate strength and the allowable shear stress is 0.4 times the ultimate strength (Hamrock, et. al, 109).

\[
c = 0.03125 \text{ in} \\
d = 0.0925 \text{ in} \\
S_{u,bending} = 6962 \text{ psi} \\
\sigma_{all,bending} = 0.6 \ S_{u,flexural} \\
\sigma_{all,shear} = 0.4 \ S_{u,tensile}
\]

Bending:
\[
I = \frac{1}{12} \ bh^3 = \frac{1}{12} (0.56 \text{ in})(0.0625 \text{ in})^3 \\
= 1.1393 \times 10^{-5} \text{ in}^4
\]

\[
M = Pd \\
\sigma_{max} = \frac{Mc}{I} = \left(\frac{Pd}{I}\right)c \\
P_{all,bending} = \frac{0.6S_u I}{dc} = \frac{0.6(6962 \text{ psi})(1.1393 \times 10^{-5} \text{ in}^4)}{(0.0925 \text{ in})(0.03125 \text{ in})} = 16.5 \text{ lb}
\]

Shear:
\[
A = bh = (0.56 \text{ in})(0.0625 \text{ in}) = 0.035 \text{ in}^2
\]

\[
\tau_{max} = \frac{2P}{A} \\
P_{all,shear} = \frac{0.4S_u A}{2} = \frac{0.4(6237 \text{ psi})(0.035 \text{ in}^2)}{2} = 43.7 \text{ lb}
\]

\[
t_{all} = 0.4S_u
\]
Button Cover: Cap Compressive Strength

Notes: Rapid prototyping material has no yield strength data, so ultimate strength is used in the calculations. Not accounting for vertical grooves. Analyzing the vertical section only, excluding top. Treat as a hollow cylindrical column. End constraints: fixed-free.

\[ d_i = 0.25 \text{ in} \quad E = \text{tensile modulus of elasticity} = 230 \text{ ksi} \]
\[ d_o = 0.375 \text{ in} \quad S_u = \text{ultimate tensile strength} = 6237 \text{ psi} \]
\[ l = 0.3259 \text{ in} \quad l_e = \text{effective length (fixed-free ends)} = 2.1l \]

Area:
\[ A = \frac{\pi}{4} (d_o^2 - d_i^2) = \frac{\pi}{4} \left[ (0.375 \text{ in})^2 - (0.25 \text{ in})^2 \right] = 0.06136 \text{ in}^2 \]

Area Moment of Inertia:
\[ I = \frac{\pi}{64} (d_o^4 - d_i^4) = \frac{\pi}{64} \left[ (0.375 \text{ in})^4 - (0.25 \text{ in})^4 \right] = 7.79 \times 10^{-4} \text{ in}^4 \]

Radius of Gyration:
\[ r_g = \sqrt{\frac{I}{A}} = \sqrt{\frac{7.79 \times 10^{-4} \text{ in}^4}{0.06136 \text{ in}^2}} = 0.1127 \text{ in} \]

Effective Length:
\[ l_e = 2.1l = 2.1(0.3259 \text{ in}) = 0.6844 \text{ in} \]
\[ \frac{l_e}{r_g} = \frac{0.6844 \text{ in}}{0.1127 \text{ in}} = 6.0741 \]

\[ \left( \frac{l_e}{r_g} \right)_y = \sqrt{\frac{2\pi^2 E}{S_u}} = \sqrt{\frac{2\pi^2 (230,000 \text{ psi})}{6237 \text{ psi}}} = 26.98 \]

\[ \frac{l_e}{r_g} < \left( \frac{l_e}{r_g} \right)_T \Rightarrow \text{Use Johnson equation} \]

From Fundamentals of Machine Elements, 2nd ed. by Hamrock, et. al (Eqn 9.16 pg 382),
\[ (\sigma_{cr})_J = S_y - \frac{S_y^2}{4\pi^2 E} \left( \frac{l_e}{r_g} \right)^2 = S_u - \frac{S_u^2}{4\pi^2 E} \left( \frac{l_e}{r_g} \right)^2 = 6237 \text{ psi} - \frac{(6237 \text{ psi})^2}{4\pi^2 (230,000 \text{ psi})} \left( 6.0741 \right)^2 = 4656.38 \text{ psi} \]
\[ (P_{cr})_{compression} = (\sigma_{cr})_J A = (4656.38 \frac{\text{psi}}{\text{in}^2}) \left( 0.06136 \text{ in}^2 \right) = 286 \text{ lb} \]
Button Cover: Base Compressive Strength

Notes: Rapid prototyping material has no yield strength data, so ultimate strength is used in the calculations
Not accounting for vertical groove
Used widest inner diameter for safety
Treat as a hollow cylindrical column
End constraints: fixed-free

\[ d_i = 0.49 \text{ in} \quad E = \text{tensile modulus of elasticity} = 230 \text{ ksi} \]
\[ d_o = 0.56 \text{ in} \quad S_u = \text{ultimate tensile strength} = 6237 \text{ psi} \]
\[ l = 0.50 \text{ in} \quad l_e = \text{effective length (fixed-free ends)} = 2.1l \]

Area:
\[ A = \frac{\pi}{4} \left( d_o^2 - d_i^2 \right) = \frac{\pi}{4} \left[ (0.56 \text{ in})^2 - (0.49 \text{ in})^2 \right] = 0.0577 \text{ in}^2 \]

Area Moment of Inertia:
\[ I = \frac{\pi}{64} \left( d_o^4 - d_i^4 \right) = \frac{\pi}{64} \left[ (0.56 \text{ in})^4 - (0.49 \text{ in})^4 \right] = 3.608 \times 10^{-3} \text{ in}^4 \]

Radius of Gyration:
\[ r_g = \sqrt{\frac{I}{A}} = \sqrt{\frac{3.08 \times 10^{-3} \text{ in}^4}{0.0577 \text{ in}^2}} = 0.25 \text{ in} \]

Effective Length:
\[ l_e = 2.1l = 2.1(0.5 \text{ in}) = 1.05 \text{ in} \]
\[ \frac{l_e}{r_g} = \frac{1.05 \text{ in}}{0.25 \text{ in}} = 4.20 \]
\[ \frac{l_e}{r_g} > \frac{2\pi^2E}{S_u} = \frac{2\pi^2(230,000 \text{ psi})}{6237 \text{ psi}} = 26.98 \]

\[ \frac{l_e}{r_g} < \left( \frac{l_e}{r_g} \right)_T \quad \Rightarrow \text{Use Johnson equation} \]

From Fundamentals of Machine Elements, 2nd ed. by Hamrock, et. al (Eqn 9.16 pg 382),
\[ \sigma_{cr} = S_y - \frac{S_y}{4\pi^2E} \left( \frac{l_e}{r_g} \right)^2 = S_u - \frac{S_u}{4\pi^2E} \left( \frac{l_e}{r_g} \right)^2 = 6237 \text{ psi} - \frac{(6237 \text{ psi})^2}{4\pi^2(230,000 \text{ psi})} (4.20)^2 \]
\[ = 6161.43 \text{ psi} \]
\[ (P_{cr})_{\text{compression}} = (\sigma_{cr}J)A = \left( 6161.43 \frac{\text{lb}}{\text{in}^2} \right) (0.0577 \text{ in}^2) \]
\[ = 356 \text{ lb} \]
Button Cover: Allowable Bending Force

- The calculations show that the maximum allowable force applied as shown is 108 lb, which would cause the cover to break near section 1.
- The inner diameter of the base is 0.195 in while the outer diameter of the cap’s inside section is 0.1875 in so they fit together.

\[
\sigma_{\text{max}} = \frac{Mc}{I} = \frac{(dP)c}{I} = \frac{0.6 S_u I}{dc} \quad P_{\text{all}} = \frac{0.6 S_u I}{dc}
\]

**Section 1:**

\[
\begin{align*}
I &= \frac{\pi}{4} \left[ (0.280\text{ in})^4 - (0.195\text{ in})^4 \right] = 3.6919 \times 10^{-3} \text{ in}^4 \\
P_{\text{all}} &= \frac{0.6(6962 \text{ psi})(3.6919 \times 10^{-3} \text{ in}^4)}{(0.51\text{ in})(0.280\text{ in})} = 108 \text{ lb}
\end{align*}
\]

**Section 2:**

\[
\begin{align*}
I &= \frac{\pi}{4} \left[ (0.280\text{ in})^4 - (0.195\text{ in})^4 \right] = 3.6919 \times 10^{-3} \text{ in}^4 \\
P_{\text{all}} &= \frac{0.6(6962 \text{ psi})(3.6919 \times 10^{-3} \text{ in}^4)}{(0.30\text{ in})(0.280\text{ in})} = 183 \text{ lb}
\end{align*}
\]

**Section 3:**

\[
\begin{align*}
I &= \frac{\pi}{4} \left[ (0.1875\text{ in})^4 - (0.125\text{ in})^4 \right] = 7.7898 \times 10^{-4} \text{ in}^4 \\
P_{\text{all}} &= \frac{0.6(6962 \text{ psi})(7.7898 \times 10^{-4} \text{ in}^4)}{(0.10\text{ in})(0.1875\text{ in})} = 173 \text{ lb}
\end{align*}
\]

**Section 4:**

\[
\begin{align*}
I &= \frac{\pi}{4} \left[ (0.1875\text{ in})^4 - (0.125\text{ in})^4 \right] = 7.7898 \times 10^{-4} \text{ in}^4 \\
P_{\text{all}} &= \frac{0.6(6962 \text{ psi})(7.7898 \times 10^{-4} \text{ in}^4)}{(0.0625\text{ in})(0.1875\text{ in})} = 277 \text{ lb}
\end{align*}
\]
- The calculations show that the maximum allowable force applied as shown is 76 lb and breakage will first occur between sections 3 and 4.
- The inner diameter of the base is 0.195 in while the outer diameter of the cap’s inside section is 0.1875 in so they fit together.
- The allowable shear stress is 0.4 times the ultimate strength (Hamrock, et. al, 109).

\[
\tau_{\text{all}} = \frac{0.4S_u A}{2}
\]

\[
\tau_{\text{max}} = \frac{2V}{A}
\]

**Section 1:**
\[
 r_i = 0.195 \text{ in} \quad r_o = 0.280 \text{ in} \\
 A = \pi \left(0.280 \text{ in}^2 - 0.195 \text{ in}^2\right) = 0.12684 \text{ in}^2 \\
 V_{\text{all}} = \frac{0.4(6237 \text{ psi})(0.12684 \text{ in}^2)}{2} = 158 \text{ lb}
\]

**Section 2:**
\[
 r_i = 0.195 \text{ in} \quad r_o = 0.280 \text{ in} \\
 A = \pi \left(0.280 \text{ in}^2 - 0.195 \text{ in}^2\right) = 0.12684 \text{ in}^2 \\
 V_{\text{all}} = \frac{0.4(6237 \text{ psi})(0.12684 \text{ in}^2)}{2} = 158 \text{ lb}
\]

**Section 3:**
\[
 r_i = 0.125 \text{ in} \quad r_o = 0.1875 \text{ in} \\
 A = \pi \left(0.1875 \text{ in}^2 - 0.125 \text{ in}^2\right) = 0.06136 \text{ in}^2 \\
 V_{\text{all}} = \frac{0.4(6237 \text{ psi})(0.06136 \text{ in}^2)}{2} = 76 \text{ lb}
\]

**Section 4:**
\[
 r_i = 0.125 \text{ in} \quad r_o = 0.1875 \text{ in} \\
 A = \pi \left(0.1875 \text{ in}^2 - 0.125 \text{ in}^2\right) = 0.06136 \text{ in}^2 \\
 V_{\text{all}} = \frac{0.4(6237 \text{ psi})(0.06136 \text{ in}^2)}{2} = 76 \text{ lb}
\]
Figure 57 Load Applied
## Appendix III: Budget Summary

### Table 7: Budget Summary

<table>
<thead>
<tr>
<th>Item</th>
<th>Part #</th>
<th>No. Parts</th>
<th>Unit</th>
<th>Total Cost</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mechanical Parts</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Housing</td>
<td></td>
<td></td>
<td></td>
<td>$2,500</td>
<td>Johnson Precision Molds</td>
</tr>
<tr>
<td>Back Connector</td>
<td></td>
<td></td>
<td></td>
<td>$0.00</td>
<td>Reuse from old switch</td>
</tr>
<tr>
<td>Button Covers &amp; Plastic Housing</td>
<td></td>
<td></td>
<td></td>
<td>$40.00</td>
<td>Rapid Prototyping at IPFW</td>
</tr>
<tr>
<td><strong>Electrical Part Donations</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PCB board</td>
<td></td>
<td>2</td>
<td></td>
<td>$0.00</td>
<td>Free at IPFW</td>
</tr>
<tr>
<td>PCB Assembly</td>
<td></td>
<td>2</td>
<td></td>
<td>$0.00</td>
<td>Done at IPFW</td>
</tr>
<tr>
<td>Schmitt Trigger MM74HC14</td>
<td></td>
<td>7</td>
<td></td>
<td>$0.00</td>
<td>Donation - Nick Sorg</td>
</tr>
<tr>
<td>Clock Generator MAX7389</td>
<td></td>
<td>8</td>
<td></td>
<td>$0.00</td>
<td>Donation - Maxim</td>
</tr>
<tr>
<td>Pushbuttons LP3-4</td>
<td></td>
<td>11</td>
<td></td>
<td>$0.00</td>
<td>Donation - OTTO Eng.</td>
</tr>
<tr>
<td>CPLD XC9572</td>
<td></td>
<td>40</td>
<td></td>
<td>$0.00</td>
<td>Donation - Xilinx</td>
</tr>
<tr>
<td>HW-130 Programmer</td>
<td></td>
<td>1</td>
<td></td>
<td>$0.00</td>
<td>Free at IPFW</td>
</tr>
<tr>
<td>JTAG CPLD Development kit</td>
<td></td>
<td>1</td>
<td></td>
<td>$0.00</td>
<td>Free at IPFW</td>
</tr>
<tr>
<td><strong>Digikey Invoice</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Voltage Regulator LM317</td>
<td>LM317AH-ND</td>
<td>5</td>
<td>7.56</td>
<td>$37.80</td>
<td>Digi-key</td>
</tr>
<tr>
<td>Shipping</td>
<td></td>
<td></td>
<td></td>
<td>$4.95</td>
<td></td>
</tr>
<tr>
<td>Order Total</td>
<td></td>
<td></td>
<td></td>
<td>$42.75</td>
<td></td>
</tr>
<tr>
<td><strong>Digilent Inc. Order # 105586 (2-8-08)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JTAG USB Programming Cable</td>
<td>JTAG-USB</td>
<td>1</td>
<td>37.95</td>
<td>$37.95</td>
<td>Digilent Inc.</td>
</tr>
<tr>
<td>Shipping</td>
<td></td>
<td></td>
<td></td>
<td>$12.15</td>
<td></td>
</tr>
<tr>
<td>Order Total</td>
<td></td>
<td></td>
<td></td>
<td>$50.10</td>
<td></td>
</tr>
<tr>
<td><strong>Electronics Plus Order (2-8-08)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>47 μF Capacitor 10V</td>
<td>C0137</td>
<td>6</td>
<td>3.6</td>
<td>$21.60</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>0.1 μF Capacitor 50V (10 pack)</td>
<td>C1686</td>
<td>6</td>
<td>3.95</td>
<td>$23.70</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>48&quot; 40 Contact IDE Ribbon</td>
<td>FLT-1650-48</td>
<td>1</td>
<td>19.95</td>
<td>$19.95</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>1 kΩ 1/2 Watt Knob Potentiometer</td>
<td>HTP1K</td>
<td>3</td>
<td>1.95</td>
<td>$5.85</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>40 Pin (2x20) Solder Pin Header 1/10&quot;</td>
<td>IDH40S</td>
<td>2</td>
<td>5.63</td>
<td>$11.26</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>PLCC-84 Pin Solder Socket</td>
<td>PLCC-84</td>
<td>2</td>
<td>5.95</td>
<td>$11.90</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>240 Ω Surface Mnt Resistor (20 Pack)</td>
<td>SR1-1206-124</td>
<td>1</td>
<td>2.4</td>
<td>$2.40</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>333 Ω Surface Mnt Resistor (20 Pack)</td>
<td>SR1-1206-133</td>
<td>1</td>
<td>2.4</td>
<td>$2.40</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>4700 Ω Surface Mnt Resistor (20 Pack)</td>
<td>SR1-1206-247</td>
<td>1</td>
<td>2.4</td>
<td>$2.40</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>47 kΩ Surface Mnt Resistor (20 Pack)</td>
<td>SR1-1206-347</td>
<td>1</td>
<td>2.4</td>
<td>$2.40</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>6 Contact Male Header Strip .100&quot; (3-pk)</td>
<td>WMLX-109</td>
<td>2</td>
<td>5.25</td>
<td>$10.50</td>
<td>Electronics Plus</td>
</tr>
<tr>
<td>Shipping</td>
<td></td>
<td></td>
<td></td>
<td>$13.00</td>
<td></td>
</tr>
<tr>
<td>Order Total</td>
<td></td>
<td></td>
<td></td>
<td>$127.3</td>
<td></td>
</tr>
<tr>
<td><strong>Digi-Key Invoice #25094320 (2-26-08)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLCC Socket - 44 pos thru hole</td>
<td>ED90008-ND</td>
<td>4</td>
<td>1.16</td>
<td>$4.64</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Header 12POS STR TIN</td>
<td>WM8124-ND</td>
<td>6</td>
<td>1.23</td>
<td>$7.38</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Header 20POS .100&quot; STR TIN</td>
<td>WM8128-ND</td>
<td>6</td>
<td>2.07</td>
<td>$12.42</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Potentiometer 1kΩ 2mm</td>
<td>490-2031-1-ND</td>
<td>5</td>
<td>0.152</td>
<td>$0.76</td>
<td>Digi-Key</td>
</tr>
</tbody>
</table>

79
<table>
<thead>
<tr>
<th>Description</th>
<th>Quantity</th>
<th>Unit Price</th>
<th>Total</th>
<th>Supplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header Cable 12POS .100&quot; DL Crimp</td>
<td>6</td>
<td>$6.96</td>
<td>$41.76</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Header Cable 20POS .100&quot; Dual</td>
<td>6</td>
<td>$10.86</td>
<td>$65.16</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Standoff Hex 4-40 THR 1&quot;L Alum</td>
<td>10</td>
<td>$5.64</td>
<td>$56.40</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Standoff Rnd 4-40 THR 1&quot;L Alum</td>
<td>10</td>
<td>$5.64</td>
<td>$56.40</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Shipping</td>
<td></td>
<td></td>
<td>$12.57</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>$66.87</td>
<td></td>
</tr>
</tbody>
</table>

**Digi-Key Invoice #25190773 (3-10-08)**

<table>
<thead>
<tr>
<th>Description</th>
<th>Quantity</th>
<th>Unit Price</th>
<th>Total</th>
<th>Supplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resistor 4.7 kΩ 1/4 W 5% 1206 SMD</td>
<td>20</td>
<td>$1.88</td>
<td>$37.60</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Capacitor 0.1 μF 50V Ceramic X7R 0805</td>
<td>20</td>
<td>$3.22</td>
<td>$64.40</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Shipping &amp; Handling</td>
<td></td>
<td></td>
<td>$13.60</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>$18.70</td>
<td></td>
</tr>
</tbody>
</table>

**Digi-Key Invoice #21672835 (3-20-08)**

<table>
<thead>
<tr>
<th>Description</th>
<th>Quantity</th>
<th>Unit Price</th>
<th>Total</th>
<th>Supplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 Position Socket</td>
<td>3</td>
<td>$3.36</td>
<td>$9.84</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>20 Position Low-Profile Header</td>
<td>3</td>
<td>$8.52</td>
<td>$25.56</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>14 Position Socket</td>
<td>3</td>
<td>$3.36</td>
<td>$10.08</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>IC Regulator 1A 5V - 120V</td>
<td>5</td>
<td>$3.60</td>
<td>$18.00</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Ceramic Capacitor 47μF 6.3V</td>
<td>10</td>
<td>$36.30</td>
<td>$363.00</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Ceramic Capacitor 50μF 50V</td>
<td>20</td>
<td>$3.22</td>
<td>$64.40</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>10 kΩ Resistors</td>
<td>10</td>
<td>$1.17</td>
<td>$11.70</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>330 Ω Resistors (Carbon Film)</td>
<td>20</td>
<td>$1.04</td>
<td>$20.80</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Shipping</td>
<td></td>
<td></td>
<td>$8.60</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>$69.17</td>
<td></td>
</tr>
</tbody>
</table>

**Digi-Key Invoice #21743880 (4-1-08)**

<table>
<thead>
<tr>
<th>Description</th>
<th>Quantity</th>
<th>Unit Price</th>
<th>Total</th>
<th>Supplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>IC Trigger</td>
<td>25</td>
<td>$8.13</td>
<td>$203.25</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Resistors 4.7 kΩ 1/4 W</td>
<td>50</td>
<td>$2.53</td>
<td>$126.50</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Resistors 4.7 kΩ 1/4 W</td>
<td>50</td>
<td>$2.53</td>
<td>$126.50</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Resistors 330 Ω 1/4 W</td>
<td>50</td>
<td>$2.53</td>
<td>$126.50</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Ceramic Capacitor 0.1 μF 50V</td>
<td>40</td>
<td>$6.44</td>
<td>$257.60</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>44 Position Socket</td>
<td>10</td>
<td>$9.79</td>
<td>$97.90</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Radial Capacitor 47 μF 35V</td>
<td>20</td>
<td>$1.88</td>
<td>$37.60</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>IC Oct Buf/Line Driver 20 DIP</td>
<td>10</td>
<td>$8.80</td>
<td>$88.00</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>IC Array 8 Darlington 18 DIP</td>
<td>10</td>
<td>$9.10</td>
<td>$91.00</td>
<td>Digi-Key</td>
</tr>
<tr>
<td>Shipping</td>
<td></td>
<td></td>
<td>$8.60</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>$60.33</td>
<td></td>
</tr>
</tbody>
</table>

**PCBexpress Order #8H E1 0152 (4-2-08)**

<table>
<thead>
<tr>
<th>Description</th>
<th>Quantity</th>
<th>Unit Price</th>
<th>Total</th>
<th>Supplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>3&quot;x2&quot; PCB Board</td>
<td>2</td>
<td>$63.00</td>
<td>$63.00</td>
<td>PCBexpress</td>
</tr>
<tr>
<td>Shipping</td>
<td></td>
<td></td>
<td>$0.00</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>$63.00</td>
<td></td>
</tr>
</tbody>
</table>

**Cutting Tools, Inc (4-7-08)**

<table>
<thead>
<tr>
<th>Description</th>
<th>Quantity</th>
<th>Unit Price</th>
<th>Total</th>
<th>Supplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>15-32 x 32 Plug Tap</td>
<td>1</td>
<td>$27.85</td>
<td>$27.85</td>
<td>Cutting Tools, Inc.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Fort Wayne, IN</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>$27.85</td>
<td></td>
</tr>
</tbody>
</table>

**Totals**

<table>
<thead>
<tr>
<th>Description</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Budget</td>
<td>$4,000</td>
</tr>
<tr>
<td>Estimated Cost of Parts</td>
<td>$3,066</td>
</tr>
<tr>
<td>Remaining Budget</td>
<td>$934</td>
</tr>
</tbody>
</table>
Appendix IV: Component Specifications

This appendix contains technical specifications for the components used in our design.

**MM74HC14: Hex Inverting Schmitt Trigger Specifications**
- Typical propagation delay: 13 ns
- Wide power supply range: 2–6V
- Low quiescent current: 20 PA maximum
- Low input current: 1 PA maximum
- Fan-out of 10 LS-TTL loads
- Typical hysteresis voltage: 0.9V at VCC 4.5V
- Low power dissipation and high noise immunity
- All inputs are protected from damage due to static discharge by internal diode clamps to VCC and ground.
- Supply Voltage (VCC) -0.5 to 7.0V
- Operating Temperature Range (TA) -55 °C to 125 °C

**7805SR Voltage Regulator Specifications**
- Guaranteed 0.5-1.0A output current to the CPLD
- Output voltage of 5 Volts
- Thermal overload protection
- Output transition SOA protection
- 2% Output voltage tolerance
- Guaranteed in extended temperature range
- Operating range temperature: -40°C ≤ T ≤ 125°C (SOT-220)

**MAX7389 Silicon Oscillator Specifications**
- Robust Microcontroller Clock and Supervisor in a Single Package
- Integrated Reset and Watchdog Functions
- Pin-Programmable Watchdog Timeout
- Speed Select
- +2.7V to +5.5V Operation
- Factory-Trimmed Oscillator
- Reset Valid Down to 1.1V Supply Voltage
- ±10mA Clock-Output Drive Current
- ±4% Total Accuracy for -40°C to +125°C
- ±2.75% Total Accuracy for 0°C to +85°C
- 5.5mA Operating Current (12MHz Version)
- -40°C to +125°C Temperature Range
- 8-Pin μMAX Surface-Mount Package
- 1MHz to 16MHz Factory Preset Frequency

**XC9572 In-System Programmable CPLD Specifications**
- 7.5 ns pin-to-pin logic delays on all pins
- $F_{CNT}$ to 125 MHz (16-bit counter frequency)
- 72 macrocells with 1,600 usable gates
- Up to 72 user I/O pins
- 5V in-system programmable
  - Endurance of 10,000 program/erase cycles
  - Program/erase over full commercial voltage and temperature range
- Enhanced pin-locking architecture
- Flexible 36V18 Function Block
  - 90 product terms drive any or all of 18 macrocells within Function Block
  - Global and product term clocks, output enables, set and reset signals
- Extensive IEEE Std 1149.1 boundary-scan (JTAG) support
- Programmable power reduction mode in each macrocell
- Slew rate control on individual outputs
- User programmable ground pin capability
- Extended pattern security features for design protection
- High-drive 24 mA outputs
- 3.3V or 5V I/O capability
- Advanced CMOS 5V Fast FLASH™ technology
- Supports parallel programming of more than one XC9500 concurrently
- Available in 44-pin PLCC, 84-pin PLCC, 100-pin PQFP, and 100-pin TQFP packages
- Operating temperature condition ranges from -40°C to +85°C
Appendix V: Complete Electrical Schematic

Figure 58: Complete Electrical Schematic
Appendix VI: VHDL Code

ASWITCHLATCH.vhd

-------------------------------------------
-- Company: 
-- Engineer: 
--
-- Create Date:  13:17:50 03/10/2008
-- Design Name: 
-- Module Name:  ASWITCHLATCH - Behavioral
-- Project Name: 
-- Target Devices: 
-- Tool versions: 
-- Description: 
--
-- Dependencies: 
--
-- Revision: 
-- Revision 0.01 - File Created
-- Additional Comments: 
--
-------------------------------------------

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

-- Uncomment the following library declaration if instantiating
-- any Xilinx primitives in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity ASWITCHLATCH is
  Port ( CLK : in STD_LOGIC;
         RST : in STD_LOGIC;
         AIN : in STD_LOGIC_VECTOR (2 downto 0);
         MSTATUS: in STD_LOGIC_VECTOR(2 downto 0);
         LAOUT : out STD_LOGIC_VECTOR (2 downto 0));
end ASWITCHLATCH;

architecture Behavioral of ASWITCHLATCH is

begin
  process(CLK,RST,AIN)
  variable LSW,TSW: std_logic_vector(2 downto 0);
  variable LSWED: std_logic;
  variable TVV: std_logic;
  begin
    if MSTATUS ="000" or RST='0' then
      TSW:="000";
      LSW:="000";
    elsif rising_edge(CLK) then
      for i in 0 to 2 loop
        LSWED:=(LSW(i) xor AIN(i)) and AIN(i);
        LSW(i):=AIN(i);

end process;

end Behavioral;
if LSWED='1' then
TVV:=TSW(i);
TSW:="000";
TSW(i):=not TVV;
end if;
end loop;
end if;
LAOUT<=TSW;
end process;
end Behavioral;
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

-- Input: CLK 10 Mhz (xxx) global clock , RST = '1' reset signal active high input
-- Output: FCLK: flash clock 1 hz, 0.5 seconds on, 0.5 seconds off

entity CLK_FLASH is
  Port (CLK : in std_logic;
        RST : in std_logic; -- Reset signal active HIGH
        FCLK : out std_logic);
end CLK_FLASH;

architecture Behavioral of CLK_FLASH is
begin
  process(CLK,RST)
  begin
    variable CNT: integer range 0 to 5000000; --for 10 Mhz
    variable CNT: integer range 0 to 2190000; -- for 4.38Mhz
    variable CNT: integer range 0 to 21900; -- for 4.38Mhz
    variable CNT: integer range 0 to 50000000; -- for 100 Mhz
    variable CNT: integer range 0 to 5;  -- for testing
    variable FCLK_INT: std_logic:= '0';
    begin
      if RST='0' then
        FCLK_INT:= '0';
        CNT:= 0;
      elsif rising_edge(CLK) then
        if CNT>=5000000 then -- 0.5 seconds for 10 Mhz
          if CNT>=2190000 then -- 0.5 seconds for 4.38Mhz
            FCLK_INT:= not FCLK_INT;
            CNT:= 0;
          ELSE
            CNT:=CNT+1;
          end if;
        end if;
      end if;
      FCLK<= FCLK_INT;
    end process;
  end Behavioral;
f07cpld.ucf

NET "CLK" IOSTANDARD = TTL;
NET "CLK" LOC = "5";

NET "RST" IOSTANDARD = TTL;
NET "RST" LOC = "39";

NET "AIN<2>" IOSTANDARD = TTL;
NET "AIN<2>" LOC = "26";  # BRIGHT

NET "AIN<1>" IOSTANDARD = TTL;
NET "AIN<1>" LOC = "24";  # DIM

NET "AIN<0>" IOSTANDARD = TTL;
NET "AIN<0>" LOC = "25";  # PARK

NET "MSERVE" IOSTANDARD = TTL;
NET "MSERVE" LOC = "22";

NET "MSTOP" IOSTANDARD = TTL;
NET "MSTOP" LOC = "18";

NET "MOFF" IOSTANDARD = TTL;
NET "MOFF" LOC = "14";

NET "MBOMARKER" IOSTANDARD = TTL;
NET "MBOMARKER" LOC = "19";

NET "MBODRIVE" IOSTANDARD = TTL;
NET "MBODRIVE" LOC = "20";

NET "MENTER" IOSTANDARD = TTL;
NET "MENTER" LOC = "27";

NET "MLED<3>" IOSTANDARD = TTL;
NET "MLED<3>" LOC = "35";

NET "MLED<2>" IOSTANDARD = TTL;
NET "MLED<2>" LOC = "29";

NET "MLED<1>" IOSTANDARD = TTL;
NET "MLED<1>" LOC = "33";

NET "MLED<0>" IOSTANDARD = TTL;
NET "MLED<0>" LOC = "34";

NET "ALED<2>" IOSTANDARD = TTL;
NET "ALED<2>" LOC = "38";

NET "ALED<1>" IOSTANDARD = TTL;
NET "ALED<1>" LOC = "36";

NET "ALED<0>" IOSTANDARD = TTL;
NET "ALED<0>" LOC = "37";
NET "MMOUT<9>" IOSTANDARD = TTL;
NET "MMOUT<9>" LOC = "43";
NET "MMOUT<8>" IOSTANDARD = TTL;
NET "MMOUT<8>" LOC = "44";
NET "MMOUT<7>" IOSTANDARD = TTL;
NET "MMOUT<7>" LOC = "1";
NET "MMOUT<6>" IOSTANDARD = TTL;
NET "MMOUT<6>" LOC = "2";
NET "MMOUT<5>" IOSTANDARD = TTL;
NET "MMOUT<5>" LOC = "3";
NET "MMOUT<4>" IOSTANDARD = TTL;
NET "MMOUT<4>" LOC = "4";
NET "MMOUT<3>" IOSTANDARD = TTL;
NET "MMOUT<3>" LOC = "8";
NET "MMOUT<2>" IOSTANDARD = TTL;
NET "MMOUT<2>" LOC = "9";
NET "MMOUT<1>" IOSTANDARD = TTL;
NET "MMOUT<1>" LOC = "11";
NET "MMOUT<0>" IOSTANDARD = TTL;
NET "MMOUT<0>" LOC = "12";

NET "K_IN" IOSTANDARD = TTL;
NET "K_IN" LOC = "13";
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity MAINSWITCH is
  Port ( CLK : in STD_LOGIC;
         RST : in STD_LOGIC;
         MSERVE : in STD_LOGIC;
         MSTOP : in STD_LOGIC;
         MOFF : in STD_LOGIC;
         MBOMARKER : in STD_LOGIC;
         MBODRIVE : in STD_LOGIC;
         MENTER : in STD_LOGIC;
         MSTATUS : out STD_LOGIC_VECTOR (2 downto 0));
end MAINSWITCH;

architecture Behavioral of MAINSWITCH is
begin
  process(CLK,RST,MSERVE,MSTOP,MOFF,MBOMARKER,MBODRIVE,MENTER) is
    variable VMSTATUS:std_logic_vector(2 downto 0);
    begin
      if RST='0' then
        VMSTATUS:="000";
      elsif rising_edge(CLK) then
        if MENTER='1' then -- Enter is pressed
          if MSERVE = '1' then -- Service Drive is pressed
            VMSTATUS:="010";
          elsif MSTOP='1' then -- Stop is pressed
            VMSTATUS:="001";
          end if;
        end if;
      end if;
    end process;
  end;
end Behavioral;
elsif MOFF='1' then  -- All off is pressed
    VMSTATUS:="000";
elsif MBOMARKER='1' then  -- Blackout marker is pressed
    VMSTATUS:="011";
elsif MBODRIVE='1' then  -- Blackout Drive is pressed
    VMSTATUS:="100";
    END IF;
end if;
if VMSTATUS>"100" then
    VMSTATUS:="000";
    end if;
end if;
MSTATUS<=VMSTATUS;
end process;
end Behavioral;
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

---- Uncomment the following library declaration if instantiating
---- any Xilinx primitives in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity TOP_SWITCH is
  Port (CLK : in STD_LOGIC;
        RST : in STD_LOGIC;
        AIN : in STD_LOGIC_VECTOR (2 downto 0);
        MSERVER : in STD_LOGIC;
        MSTOP : in STD_LOGIC;
        MOFF : in STD_LOGIC;
        MBOMARKER : in STD_LOGIC;
        MBODRIVE : in STD_LOGIC;
        MENTER : in STD_LOGIC;
        MLED : out STD_LOGIC_VECTOR (3 downto 0);
        ALED : out STD_LOGIC_VECTOR (2 downto 0);
        MMOUT : out STD_LOGIC_VECTOR (9 downto 0);

        -- LEDG: out STD_LOGIC, -- for FPGA testing
        K_IN : in STD_LOGIC);
end TOP_SWITCH;

architecture Behavioral of TOP_SWITCH is

  component BUFG
    port (O : out STD_ULOGIC;
          I : in STD_ULOGIC);
  end component;

  component BUFGSR
    port (O : out STD_ULOGIC;
          I : in STD_ULOGIC);
  end component;

end component;

cOMPONENT CLK_FLASH IS
    PORT (CLK : IN std_logic;
           RST : IN std_logic; -- Reset signal active HIGH
           FCLK : OUT std_logic);
END COMPONENT CLK_FLASH;

SIGNAL CCLK, FCLK: STD_LOGIC;

COMPONENT ASWITCHLATCH IS
    PORT (CLK : IN STD_LOGIC;
          RST : IN STD_LOGIC;
          AIN : IN STD_LOGIC_VECTOR (2 DOWNTO 0);
          MSTATUS : IN STD_LOGIC_VECTOR (2 DOWNTO 0);
          LAOUT : OUT STD_LOGIC_VECTOR (2 DOWNTO 0));
END COMPONENT ASWITCHLATCH;

SIGNAL LAOUT: STD_LOGIC_VECTOR (2 DOWNTO 0);

COMPONENT MAINSWITCH IS
    PORT (CLK : IN STD_LOGIC;
          RST : IN STD_LOGIC;
          MSERVE : IN STD_LOGIC;
          MSTOP : IN STD_LOGIC;
          MOFF : IN STD_LOGIC;
          MBOMARKER : IN STD_LOGIC;
          MBODRIVE : IN STD_LOGIC;
          MENTER : IN STD_LOGIC;
          MSTATUS : OUT STD_LOGIC_VECTOR (2 DOWNTO 0));
END COMPONENT MAINSWITCH;

SIGNAL MSTATUS: STD_LOGIC_VECTOR (2 DOWNTO 0);

COMPONENT ASWITCH IS
    PORT (MSTATUS : IN STD_LOGIC_VECTOR (2 DOWNTO 0);
          BRT : IN STD_LOGIC;
          DIM : IN STD_LOGIC;
          PKR : IN STD_LOGIC;
          K_IN : IN STD_LOGIC;
          MOUT : OUT STD_LOGIC_VECTOR (9 DOWNTO 0);
          MFOUT : OUT STD_LOGIC;
          MLED : OUT STD_LOGIC_VECTOR (3 DOWNTO 0);
          ALED : OUT STD_LOGIC_VECTOR (2 DOWNTO 0));
END COMPONENT ASWITCH;

SIGNAL MFOUT: STD_LOGIC;

SIGNAL MOUT: STD_LOGIC_VECTOR (9 DOWNTO 0);

SIGNAL FFCLK: STD_LOGIC_VECTOR (9 DOWNTO 0);

SIGNAL CRST: STD_LOGIC;
BEGIN
    U1:CLK_FLASH PORT MAP (CCLK, CRST, FCLK);
    U2:ASWITCHLATCH PORT MAP (CCLK, CRST, AIN, MSTATUS, LAOUT);
    U3:MAINSWITCH PORT MAP (CCLK, CRST, MSERVE, MSTOP, MOFF, MBOMARKER, MBODRIVE, MENTER, MSTATUS);
    U4:ASWITCH PORT MAP (MSTATUS, LAOUT (2), LAOUT (1), LAOUT (0), K_IN, MOUT, MFOUT, MLED, ALED);
U5:BUFG port map(CCLK,CLK);
U6:BUFGSR port map(CRST,RST);

-- CRST<=RST;
FFCLK<=(others=>FCLK);
process(FCLK,MFOUT,MOUT,FFCLK) -- flash control
begin
  if MFOUT='1' then -- flash
    MMOUT<=MOUT and FFCLK;
  else -- No flash
    MMOUT<=MOUT;
  end if;
end process;

-- LEDG='1';
end Behavioral;
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

-- Input: MSTATUS: Main switch status
-- K_IN: brake signal input
-- BRT,DIM,PARK: auxiliary switch after key debouncing and latch (toggle switch) using clock
-- MOUT: signal outputs to back connectors
-- MFOUT: flash signal control signals, '1' flash, '0' no flash
-- MLED: switch panel LED indicator for main switch status: service dirve, stop light, blackout marker, and blackout drive
-- ALED: auxiliary panel LED indicator for bright, dim and park,
entity ASWITCH is
Port ( MSTATUS : in STD_LOGIC_VECTOR (2 downto 0);
   BRT : in STD_LOGIC;
   DIM : in STD_LOGIC;
   PRK : in STD_LOGIC;
   K_IN : in STD_LOGIC;
   MOUT : out STD_LOGIC_VECTOR (9 downto 0);
   MFOUT : out STD_LOGIC;
   MLED : out STD_LOGIC_VECTOR (3 downto 0);
   ALED : out STD_LOGIC_VECTOR (2 downto 0));
end ASWITCH;

architecture Behavioral of ASWITCH is
begin
   process(MSTATUS,BRT,DIM,PRK,K_IN) is
      begin
      MLED<="0000";
      case MSTATUS is
         when "010" => -- Service drive mode
            MLED<="1000";
            if BRT='1' then
               if K_IN='1' then -- K='1', BRT = '1', Service Drive mode
                  MOUT<="1110001011";
                  MFOUT<='0';
               else -- K='0'; BRT='1'
                  MOUT<="1100001011";
                  MFOUT<='0';
               end if;
            elsif DIM='1' then
               if K_IN='1' then
                  MOUT<="1110001011";
                  MFOUT<='0';
               else -- K='0'; DIM='1'
                  MOUT<="1100001011";
                  MFOUT<='0';
               end if;
            elsif PRK='1' then -- service mode, park = '1'
               if K_IN='1' then
                  MOUT<="1110001101";
                  MFOUT<='0';
               else
                  -- else case
               end if;
            end case;
      end process;
end Behavioral;
MOUT <="1100001101";
MOUT<= '0';
end if;
else -- default position
if K_IN='1' then
  MOUT<="1110001101"; MOUT<='0';
else
  MOUT<="1100001101"; MOUT<='0';
end if;
end if;
when "001" => -- STOP light;

MLED<="0100";
if BRT='1' then
  if K_IN='1' then -- K='1', BRT = '1', stop light
    MOUT <="1110000001";
    MOUT<= '0';
  else -- K='0', BRT='1'
    MOUT <="1100000001";
    MOUT<= '0';
  end if;
elsif DIM='1' then
  if K_IN='1' then
    MOUT <="1110000001";
    MOUT<= '1';
  else -- K='0', DIM='1'
    MOUT <="1100000001";
    MOUT<= '1';
  end if;
elsif PRK='1' then -- park = '1'
  if K_IN='1' then
    MOUT <="1110000001";
    MOUT<= '0';
  else
    MOUT <="1100000001";
    MOUT<= '0';
  end if;
else -- default position
  if K_IN='1' then
    MOUT<="1010000001"; MOUT<= '0';
  else
    MOUT<="1000000001"; MOUT<= '0';
  end if;
end if;
when "000" => -- ALL OFF
  MOUT<= (others=>'0'); MOUT<= '0';
when "011" => -- Blackout marker
  MLED<="0010";
  if BRT='1' then
    if K_IN='1' then -- K='1', BRT = '1', stop light
      MOUT <="1101010000";
      MOUT<= '0';
    else -- K='0', BRT='1'
      MOUT <="1100010000";
    end if;
  end if;
end if;
elseif DIM='1' then
  if K_IN='1' then
    MOUT <="1101010000";
    MFOUT<='1';
  else
    -- K='0'; DIM='1'
    MOUT <="1100010000";
    MFOUT<='1';
  end if;
elseif PRK='1' then -- park = '1'
  if K_IN='1' then
    MOUT <="1101010000";
    MFOUT<='0';
  else
    MOUT <="1100010000";
    MFOUT<='0';
  end if;
else -- default position
  if K_IN='1' then
    MOUT <="1001010000"; MFOUT<='0';
  else
    MOUT <="1000010000"; MFOUT<='0';
  end if;
end if;
when "100" => -- Blackout drive
  MLED<="0001";
  if BRT='1' then
    if K_IN='1' then -- K='1', BRT = '1', stop light
      MOUT <="1101110000";
      MFOUT<='0';
    else
      -- K='0'; BRT='1'
      MOUT <="1100110000";
      MFOUT<='0';
    end if;
  else -- default position
    if K_IN='1' then
      MOUT <="1001110000";
      MFOUT<='0';
    else
      MOUT <="1000110000";
      MFOUT<='0';
    end if;
  end if;
else -- default position
  if K_IN='1' then
    MOUT <="1001110000";
    MFOUT<='0';
  else
    MOUT <="1000110000";
    MFOUT<='0';
  end if;
end if;
end if;
end if;

when others => -- other invalid status
    MOUT<= (others=>'0'); MFOUT<= '0'; MLED<= "0000";
end case;
ALED<=BRT&DIM&PRK;
end process;
end Behavioral;