ECE 477 Final Report – Fall 2010
Team 4 – Project Forget-Me-Not



Team Members:

#1: Mike Williams **Signature: _____ Date: _____**

#2: Andrew Sydelko **Signature: _____ Date: _____**

#3: Chris Cadwallader **Signature: _____ Date: _____**

#4: Craig Pilcher **Signature: _____ Date: _____**

CRITERION	SCORE	MPY	PTS
Technical content	0 1 2 3 4 5 6 7 8 9 10	3	
Design documentation	0 1 2 3 4 5 6 7 8 9 10	3	
Technical writing style	0 1 2 3 4 5 6 7 8 9 10	2	
Contributions	0 1 2 3 4 5 6 7 8 9 10	1	
Editing	0 1 2 3 4 5 6 7 8 9 10	1	
Comments:		**TOTAL**	

TABLE OF CONTENTS

Abstract 1
1.0 Project Overview and Block Diagram 1
2.0 Team Success Criteria and Fulfillment 2
3.0 Constraint Analysis and Component Selection 2
4.0 Patent Liability Analysis 7
5.0 Reliability and Safety Analysis 12
6.0 Ethical and Environmental Impact Analysis 16
7.0 Packaging Design Considerations 20
8.0 Schematic Design Considerations 23
9.0 PCB Layout Design Considerations 26
10.0 Software Design Considerations 30
11.0 Version 2 Changes 34
12.0 Summary and Conclusions 35
13.0 References 36
Appendix A: Individual Contributions A
Appendix B: Packaging B
Appendix C: Schematic C
Appendix D: PCB Layout Top and Bottom Copper D
Appendix E: Parts List Spreadsheet E
Appendix F: FMECA Worksheet F

Abstract

Project Forget-Me-Not is a child seat monitor for use in vehicles. It detects the presence of a child in a child safety seat and will sound an alarm if a child is left in a car. The device interfaces with the government-mandated OBD-II port to detect vehicle state, and is capable of activating vehicle features to assist in rescuing the child.

1.0 Project Overview and Block Diagram

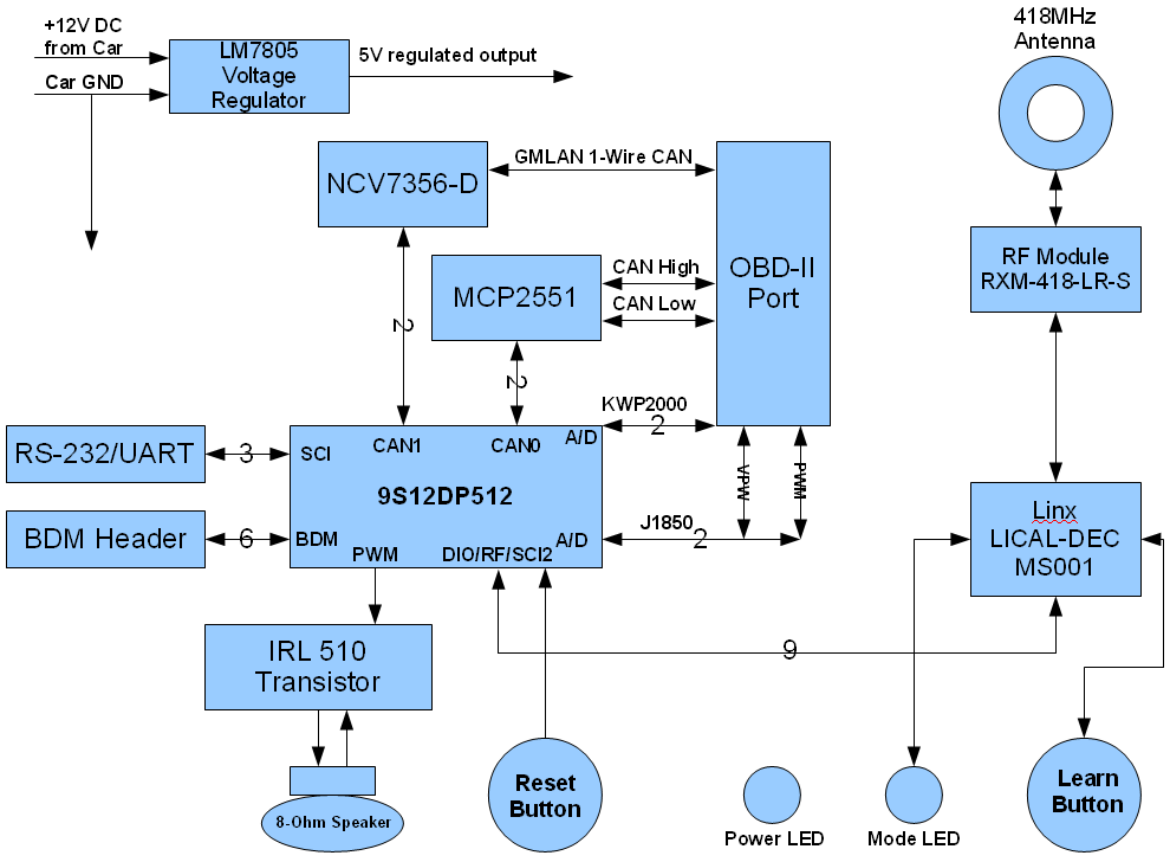


Figure 1.0 – Car Side Block Diagram

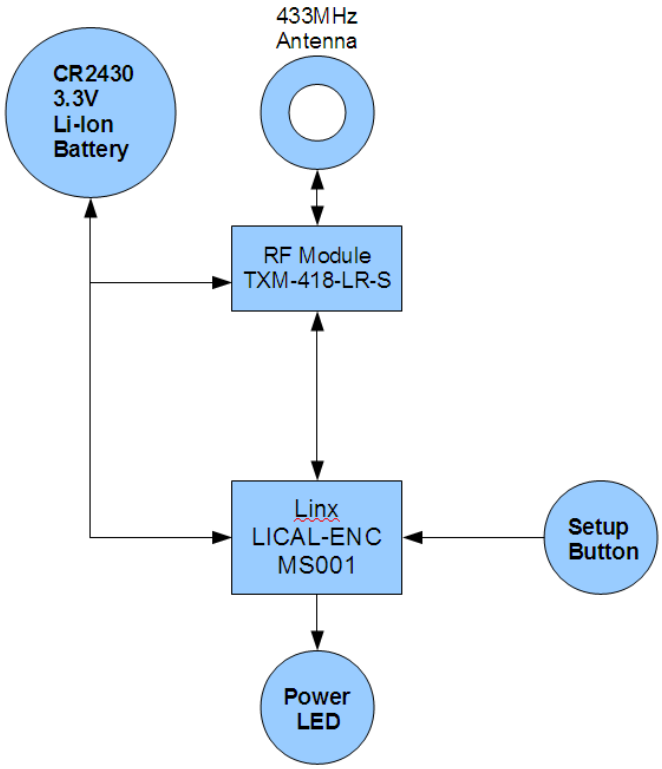


Figure 1.1 – Child Side Block Diagram

2.0 Team Success Criteria and Fulfillment

1. An ability to determine the operating state of the vehicle via the OBD-II port.
2. An ability to determine the presence of a child in a safety seat.
3. An ability to use multiple safety seats in one vehicle.
4. An ability to sound an audible alarm when the ignition is turned off.
5. An ability to interface directly to the internal vehicle CAN.

3.0 Constraint Analysis and Component Selection

3.1 Design Constraint Analysis

The major design constraints for this project are on- and off-chip peripherals, as well as cost. The microcontroller used should be able to interface with the vehicle's OBD-II port, and

ideally have built-in CAN processing. Flash memory will be utilized for audio samples, so the microcontroller should have at least 256K of flash, as well as PWM for audio output. The child-side board will be battery-powered, so power consumption is the biggest constraint on that side, especially with the wireless communications. Form factor is also a concern for the child-side, mainly with regards to the antenna. Ideally the final product will be marketable, as outlined in section 2.7, so keeping costs low is vital.

3.2 Computation Requirements

Three major requirements will need to be met. First, the system will need to continuously monitor the OBD-II port for any changes in vehicle state. It will need to determine which physical-level protocol is being used and use both physical-level and protocol-level techniques when available to track vehicle state. Second, the system will need to continuously listen to the wireless transmitters for any events where a child is put in a safety seat or removed from a safety seat to determine the number and presence of a child in the vehicle at any point in time. And finally, the system will need to playback audio from its flash memory when a child is present and the vehicle has changed from a running to an off state. Overall, reliability is more important than speed in regards to these computational tasks.

3.3 Interface Requirements

Table 2.2.1 Microcontroller I/O function

Microcontroller I/O Function	**Pins Req'd**
Audio output via PWM	1
Detection of vehicle state from OBD-II line activity detector on SAE-J1850 PWM & VPW, ISO 15765-4 2-wire CAN, and GM J2411 Single-Wire CAN	3
Connection to vehicle CAN network via ISO 15765-4 2-wire CAN	4
Learn button for adding safety seats to system	1
Factory-reset button for clearing all configuration data	1
System active display LED	1
Panic disable button	1
Serial port for debug messages	3
BDM connection port	6

Table 2.2.2 External Interface I/O

External Interface I/O Function	Electrical Operating Constraints
OBD-II J1850 PWM/VPW detection	-20V to 20V
OBD-II ISO 15765-4 detection/communication, 2-Wire CAN	2.75V to 4.5V for CAN-High 0.5V to 2.25V for CAN-Low +16V absolute max voltage
OBD-II ISO 15765-4 detection, 1-Wire CAN	0V-12V for 1-Wire protocol (12V used for wake-up only, 0V-4V used for data transmission)

On-Chip Peripheral Requirements

Table 2.3.1 Microcontroller Peripheral Requirements

Function	**Microcontroller Peripheral Requirement**
Audio output	1x PWM
Detection of vehicle state from OBD-II line activity detector on SAE-J1850 PWM & VPW, ISO 15765-4 2-wire CAN, and GM J2411 Single-Wire CAN	3x Digital I/O pins
Connection to vehicle CAN network via ISO 15765-4 2-wire CAN	1x CAN protocol controller
Learn button for adding safety seats to system	1x Digital I/O pins
Factory-reset button for clearing all configuration data	1x Digital I/O pins
System active display LED	1x Digital I/O pins
Panic disable button	1x Digital I/O pins
Serial port for debug messages	1x UART/SCI Controller
BDM connection port	1x BDM port

3.4 Off-Chip Peripheral Requirements

Table 2.4.1 Off-Chip Peripheral Requirements

Function	**Peripheral**
Wireless communication between car-side and child-side boards	1x RF Encoder 1x RF Decoder
Audio output	1x Speaker
Audio amplification	1x Class D Amplifier

3.5 Power Constraints

Power for the control unit will be delivered via the car's 12V system. A voltage regulator will be required to put the supply for the car-side in the 5V range. Initial prototype for the child safety seat will be battery powered. Heat-dissipation will not be an issue on either side.

3.6 Packaging Constraints

Packaging constraints are as follows: must be attractive to a consumer in-store, must fit comfortably under the dash of most vehicles. Control-unit will be a small box with included attachable Velcro strips to allow device to be hung from underside of the driver's control area. Compatible child-safety seat packing constraints will be individual to third-party manufacturers. Protocol level compatibility is all that is required to interoperate with our system properly.

3.7 Cost Constraints

As no known competing systems exist, the price-point for this product is self-determined. Via discussions with an interested manufacturer, pricing would be optimal near the $50 per unit level, with manufacturing costs near $20 per unit. Maximizing distribution to parents would also be morally desirable, because of the implication of saving lives.

3.8 Component Selection Rationale

The 9S12DP512 series [see 1] by Freescale exhibits all the peripherals needed for this project: CAN ports, PWM, and 512K of flash. The older version of this series was originally chosen but is no longer in production. The only difference between the current choice and its predecessor is that the 9S12DP512 has double the flash memory. Freescale was identified first as the manufacturer of choice because of the design team's familiarity with their products and with their development tools, such as their IDE, CodeWarrior. This tool also includes software libraries for the various OBD-II protocols, making design much easier. Atmel offers several comparable microcontrollers in their AVR series, such as the AT90CAN128 [see 2]. However, the most flash memory available in the series that includes a CAN interface is 128 KB, which will not be able to store enough audio data. Also, as previously stated, the team is familiar with Freescale's CodeWarrior, which will allow us to hit the ground running.

The RF modules chosen are the Linx LR series. These transmitters come with encoder/decoder pairs that are easily connected and designed to work together per FCC regulation. The decoder [see 3] contains built-in logic to remember devices that have communicated with it, making the process of pairing a child seat with a car-side unit simple. This also removes the need for a microcontroller on the child-side board to broadcast seat ID and other information, reducing power cost on the battery powered side significantly. The encoder

[see 4] can be powered by a button cell (Energizer CR2430, see [5]), which lasts 1000 hours with continuous drain. This product's application will not be in continuous use, so it should last much longer.

To support single and dual wire CAN, PCA82C250 CAN transceivers were chosen [see 6] for its high speed and reduced RFI and EMI. However, this part could not be found through any vendors. On the single wire side, it was replaced by the NCV7356 from ON Semi [see 7]. And for dual wire CAN, the Microchip MCP2551 was found to be a suitable replacement for its 1 Mb/s speed. For audio output, a simple LM386 audio amplifier was considered for cost reasons. After learning that it could only put out roughly 1W of audio, the switch was made to an IRL510N Class-D audio amplifier which can get 20W of amplification. If this is too loud, this item may be changed in the future. For voltage regulation, a LM7805 series linear regulator is currently being used instead of a switch-mode power supply for cost effectiveness and reliability.

4.0 Patent Liability Analysis

4.1 Introduction

Project Forget-Me-Not is a system to keep track of children in the child seat of a car. It alerts the driver or nearby people when a child is left in the car for an extended period. This product is a combination of two or more packages. The first package, from here on called the car-side package is the brains of the system. It contains the microcontroller and connects directly to the car through its government required, industry standard OBD-II port. The second package (or packages for vehicles containing multiple child seats), called the child-side, is the part that attaches to each child seat that is to be monitored.

Practically every part of this system except integration with the OBD-II port as a generic method of addition to a vehicle has been covered in numerous existing patents. Patents exist for concepts as simple as wireless switch detection, to “Occupant detection and notification system for use with a child seat” with many patents specifically claiming wireless transmission of seat status information. In addition, numerous actual products exist today which replicate some of the functionality of the system. While this does not prevent production of the system, it presents clear prior art for much of the functionality, and possible competitors with patents who would be willing to sue to protects its market advantage.

4.2 Results of Patent and Product Search

Existing commercial products and numerous patents exist. The products are listed first, and the patents are examined next.

4.3 Commercial Products

ChildMinder – http://www.babyalertinfo.com

This popular commercial product offering includes numerous devices that mimic some of the functionality of this design. The first is a system which clips into the buckle of the child safety seat and acts as the child detection system. An additional weight-detection pad system is sold as an alternative to the clip. Both of these encompass the essential functionality of the additions this project would need to detect a child.

Next, the child detection products are matched to a wireless transmission system which activates an alarm which the parent carries. While the purpose of these two systems may appear superficially different – the ChildMinder transmits bi-directionally to activate an alarm, while this system transmits one-way to indicate seat status – both systems essentially are wirelessly transmitting child seat states for the purposes of activating alarm functionality.

Halo Child Safety Seat System

This product is nearly identical to the ChildMinder and does not seem to have reached production, although it has been in develop in for quite some time. It represents publicly available knowledge of some aspects of this design, but not any more so than the ChildMinder.

"NASA Develops Child Car Seat Safety Device" - http://www.4rkidssake.org/NASAdevice.htm

NASA claims to have developed a system for detecting a child based on a "calibrated weighing pad", similar to the above products. Little other information is available.

4.4 US Patents

Only the two most potentially infringing patents are listed below and in the patent analysis because almost all aspects of this design are covered by these two. Further patents will be listed in the appendix.

US Patent 5949340 - "Warning system for detecting presence of child in an infant seat"

Filed July 28, 1998, Granted Sep. 7, 1999

Abstract: *"An apparatus is provided for warning when a child has been left in an infant seat and a vehicle has been turned off. The apparatus includes an occupant detection mechanism for detecting the presence of an occupant within an infant seat located within a vehicle; an ignition detection mechanism for detecting the state of the vehicle's ignition system; a control unit for generating an alarm signal when the occupant detection mechanism detects the presence of an occupant within the infant seat and the ignition detection mechanism detects that the vehicle's ignition system has been turned from an "on" state to an "off" state; and an alarm units for generating an alarm in response to the alarm signal. The components of the apparatus can be located within the infant seat, within the vehicle or combined within the infant seat and the vehicle."*

US Patent 7733228 - "Wireless system to detect presence of child in a baby seat"

Filed Aug. 26, 2008, Granted Jun. 8, 2010

Abtract: *"A wireless system that detects the presence of a child in a safety seat located in the passenger cabin of a vehicle includes a controller responsive to signals generated by sensors monitoring predefined functions of the vehicle, RFID tag device attached to the safety seat and RFID tag reader mounted in the cabin. The system generates control signals which activate an alarm, open the doors of the vehicle and roll down windows if the child is left in the safety seat of an unattended vehicle."*

4.5 Analysis of Patent Liability

Literal Infringement, US Patent 5949340 - *"Warning system for detecting presence of child in an infant seat"*

The very first claim covers most aspects of this system. Claim 1 includes as follows:

"An apparatus comprising: means for detecting a presence of an occupant within an infant seat located within a vehicle; means for detecting a state of a vehicle's ignition system; means for generating an alarm signal when the detection means detects the presence of an occupant within the infant seat and the means for the detecting the state of the vehicle's ignition system has been turned from an "on" state to an "off" state."

Claim 2 describes "an occupant sensor located within the infant seat" [literal infringement]. Claim 3 describes "An apparatus as claimed in Claim 1, wherein the means for generating an alarm signal comprises a voice generator" [literal infringement]. Claim 5 describes "An apparatus … wherein the means for generating an alarm signal controls the operation of a theft alarm of the vehicle after a predetermined time period …" [literal infringement]. Claim 10 describes "An apparatus … wherein the mean for generating an alarm signal controls the operation of power windows of the vehicle after a predetermined time period …" [literal infringement]

The primary aspects that this patent does not cover include the wireless system, or the use of the internal CAN network connected via the OBD-II network. They also do not specify a

generic approach instead of an internal vehicle approach. No patent that this group is aware of specifies a generic approach.

Possible Doctrine of Equivalents, US Patent 7733228 - *"Wireless system to detect presence of child in a baby seat"*

Claim 3 describes "*The apparatus of claim 2 wherein the RFID reader includes an antenna; a switch operatively coupled to said antenna; and RF circuit module operatively coupled to said switch."* It is possible that this would infringe under the doctrine of equivalents, as substantially the same function, indicating the state of the child seat, is done substantially the same way, by means of wireless transmission. Claims 11 and 13 describe connecting said wireless system to the vehicle alarm system, and rolling down windows, respectively. It does not describe attachment to the vehicle's OBD-II system, and it does not describe the same system of wireless communication as this project.

4.6 Action Recommended

Were this system to reach production, extensive legal protection would be an absolute requirement. Many aspects of this system are neither original nor patentable, while potentially infringing upon numerous other patents at the same time. Licensing of the genuine patents would be an essential avenue to consider. It appears that it will be impossible to design a wireless system for transmitting the child seat status without infringing patents using the current design.

4.7 Summary

While few actual products exist that perform similar functions to this project, a very large number of patents that cover similar functionality do. These patents individually cover many aspects of the system, and together cover essentially all of the functionality. All alternatives, such as not using wireless switch closures to indicate child safety seat status, defeat the essential purpose of the project. As such, numerous patents will likely need to be licensed or cross-licensed.

5. Reliability and Safety Analysis

5.1 Introduction

Project Forget-Me-Not is an aftermarket safety system for automotive applications to prevent parents and caretakers from accidentally leaving children in vehicles. The product will first alert the user audibly if a “child left” event occurs. If the child is still present after a time threshold, the device will interact with the host vehicle to alert other nearby people or reduce the interior temperature of the vehicle.

Safety and reliability are primary concerns as the intended use of this device implies additional safety to child occupants. Several failures to key components can cause the device to fail, prevent detection of a child, or inhibit control of the host vehicle. These issues of reliability and sources of false negatives need to be thoroughly analyzed and mitigated.

5.2 Reliability Analysis

Several components vital to the operation of Forget-Me-Not have been chosen. This analysis focuses on the components that are most likely to fail. Among these devices are the Freescale 16-Bit microcontroller, the LM805 linear regulator, and the power MOSFET driving the audio circuitry. The microcontroller was chosen as it is the most complex component in the design. The linear regulator and power MOSFET are included as they are the largest sources heat dissipation on the board. Worth mentioning are the Linx wireless transceiver modules due to their criticality. While these components do not have a high failure rate compared to the ones listed, they are essential to the operation of the device. The inability to detect a failure of these components makes them a critical safety concern.

Component: *Freescale 9S12DX256*

Model Used: Microcircuits, Microcontroller

$$\lambda_p = (C_1*\pi_T + C_2*\pi_E)*\pi_Q*\pi_L$$

Assumptions: Worst case operating condition of 54°C, 16MHz Clock

Parameter Name	*Description*	*Qualifier*	*Value*	*Comments*

C1	Die Complexity	16-bit MOS	0.280	-
π_T	Temperature Factor	Average	1.6	T_j = 54 °C + (5V)(85mA)(41°C/W) = 71.425 °C
C2	Package Fail Rate	SMT	0.0320	112-pin QFP
π_Q	Quality Factor	Commercial	10.0	-
π_E	Environmental Factor	Ground Mobile	4.0	-
π_L	Learning Factor	>2 years	1.00	-
Entire Design:	λ_p = 5.76 failures per 10^6 hours		MTTF = 1.736E+05	

Component: *LM 7805 Linear Regulator*

Model Used: Microcircuits

$$\lambda_p = (C_1*\pi_T + C_2*\pi_E)*\pi_Q*\pi_L$$

Assumptions: Worst case operating condition of 54°C

Parameter Name	*Description*	*Qualifier*	*Value*	*Comments*
C1	Die Complexity	Linear Gate Array	0.01	-
π_T	Temperature Factor	Average	0.75	T_j = 54 °C + (12V)(110mA)(5°C/W) = 60.6°C
C2	Package Fail Rate	SMT	0.00047	3-pin SMT
π_Q	Quality Factor	Commercial	10.0	-
π_E	Environmental Factor	Ground Mobile	4.0	-
π_L	Learning Factor	>2 years	1.00	-
Entire Design:	Λ_p = 0.0938 failures per 10^6 hours		MTTF = 1.066E+07	

Component: IRL510

Model Used: Power transistor, low frequency

$$\lambda_p = \lambda_b*\pi_{T*}\ \pi_{A*}\ \pi_{Q*}\ \pi_E$$

Assumptions: Worst case operating condition of 54°C, Continuous use.

Parameter Name	*Description*	*Qualifier*	*Value*	*Comments*
λ_b	Base Rate	MOSFET	0.012	-
π_T	Temperature Factor	Max	3.7	Factor based on continuous use
π_A	Application factor	Power FET	2.0	2W < Pr < 5W
π_Q	Quality Factor	Commercial	10.0	-

π_E	Environmental Factor	Ground Mobile	4.0	-
π_L	Learning Factor	>2 years	1.00	-
Entire Design:	λ_p = 3.552 failures per 10^6 hours		MTTF = 2.815E+05	

As expected, the power MOSFET and high complexity microcontroller are the components with the highest failure rate. The long MTTF for the linear regulator can be attributed to the low current draw of our present design. Also the MTTF of the power MOSFET is most certainly an underestimation due to its relatively infrequent use. Future designs can improve reliability by switching to a 3.3V microcontroller with fewer pins and mounting the power MOSFET onto a heat sink for better thermal dissipation. Additionally, a fuse should be added to the incoming battery source to prevent the high current-sourcing lead-acid battery from generating excessive heat should a bypass capacitor become shorted.

5.3 Failure Mode, Effects, and Criticality Analysis (FMECA)

Project Forget-Me-Not can be decomposed into functional blocks. These blocks include the power supply system, audio amplifier circuitry, CAN communications system, and wireless communications system. Each functional block is prone to a set of failures that are detailed in Appendix B. Each failure mode is assigned a criticality rating that is defined by the following:

Low: Failure causes loss of functionality but no risk to personal injury. (1 failure per 10^6 hours acceptable)

Moderate: Failure causes significant loss of functionality or complete failure of product. No risk of personal injury due to user awareness that the product is inoperational**.** (1 failure per 10^6 hours acceptable)

High: Failure cannot be detected and/or poses significant risk to personal injury. User is unaware of the fault. (1 failure per 10^9 hours acceptable)

The critical failures present in the design primarily include undetectable failures that can lead to false negatives and a false sense of security by the user. Other failures such as those of the CAN transceivers or audio power MOSFET are less critical, but still reduce functionality.

The power circuit of the design has two moderate failures that disable the device. These failures are an overvoltage of VCC and no voltage on VCC. The former can be attributed to a failure of the linear regulator to scale down the incoming 12V from the battery to an acceptable 5V. This failure will likely cause permanent damage to many components such as the microcontroller, and communication transceivers. Conversely if VCC has no voltage relative to ground then a short has likely occurred between the incoming battery and ground. This can be a fire hazard if a fuse is not installed on the incoming battery rail. An additional failure of the power circuit is with the power LED. This is a cosmetic failure of low criticality and does not effect the operation of the device.

Some failures in the communications circuits of the device can be considered critical. If one of the wireless transceivers fail a child side event will not be detected. Even worse, it is not currently known how this type of failure can be detected within the current design, making a wireless transceiver failure one of high criticality. Other communication failures such as an inability to snoop the CAN bus are only of moderate criticality provided the audio circuit is still functional and can warn the occupant. These failures are likely caused by the loss of a CAN transceiver.

Failure of the microcontroller circuitry will likely disable the device or cause repeated resets. In many cases this type of failure is considered of moderate criticality since the user will be aware of its malfunction. Critical failures in this block would be caused by intermittent behavior that makes the device appear to be working properly. Common causes of this failure are random resets of the microcontroller and software bugs.

5.4 Summary

The goal of Forget-Me-Not is to provide additional safety mechanisms to help prevent accidental harm or death. This implied security makes it essential that our product be able to detect all child events present, and in the case of a critical failure, alert the user that the product

is no longer able to function correctly. To improve upon the existing design, component changes can be made that will increase reliability. Safety critical changes such as the development of a detection method for wireless transceiver failures are also warranted to prevent false negatives. By taking these steps in future revisions of the design, Forget-Me-Not will drastically improve in safety and reliability.

6. Ethical and Environmental Impact Analysis

6.1 Introduction

This project has many more ethical issues than environmental issues. The project is a set of devices that are meant to provide a layer of added safety by attempting to prevent children being left behind in a car in unsafe conditions. There are two main parts to the design, a car-side device that directly interfaces with the car's OBD-II port and a child-side device that transmits the state of the seat back to the car-side device. A failure of almost any component in either device can be detrimental to the overall function of the project. This product will be used by non-technical consumers who rely on the device always working. The product must attempt to inform the consumer that it is no longer functional. False positives are also a major concern because in an attempt to protect the child left in the car, we do things like roll down the windows. A false positive situation could cause the windows to be rolled down when no one is in the car leaving the car vulnerable.

Environmentally, this project does have some issues. These issues are related to standard problems with any electronic device. The project includes PCBs, batteries and plastic cases. All of these require proper disposal measures to mitigate the environmental issues.

6.2 Ethical Impact Analysis

When implementing the design of this project, we have many failure modes that need to be handled in order to mitigate some of the ethical concerns. Audio feedback and LEDs are the mechanisms used to communicate with the end user of device functionality. The LEDs are used to show communication between the child-side device and the car-side device. Audio is used to

broadcast when a child seat is buckled and unbuckled as well as announcing an alarm when the car-side device detects that a child has been left in the car in an unsafe condition.

Circuit design is a key element in the mitigation of failures. The components chosen and their reliability play a major role in the operation of the final product. The main components of this project are the microcontroller, the linear regulator used for step-down voltages, the power MOSFET used for audio, the two CAN transceivers and the Linx wireless transceiver modules. The environment for this product also needs to be taken into account as the temperatures in a car can be extreme, from freezing conditions to over 140 degrees when parked in the sun. Failure of any of the previously mentioned components will cause the device to not audibly announce state changes or show functionality through the LEDs.

Another major ethical component of this project is the direct car interface. The car-side device is meant to attach to the OBD-II port of the car. This port provides access to the internal CAN bus of the car. Misuse of this interface could potentially impact a wide variety of car functions and possibly even cause damage to the car. Through good software design we should be able to mitigate any problems with this interface. Our use of the CAN bus is limited to sending specific commands. These commands will be used to get back specific values related to the car state and requesting certain functions to be performed.

Product testing is another key element in designing a product that limits its ethical impact. This testing will need to include full environmental testing in an environment similar to inside a car. The car-side device will likely fit under the dash of the car and will be subject to temperature extremes, possibility of water exposure and dirty conditions. Battery life also needs to be tested. The child-side device has a coin-cell CR2032 3V battery that powers the transmitter. The transmitter is in a sleep state except when there is a button press causing transmission of data. Determining the number of button presses possible before the battery is exhausted is important in knowing how often the battery will need to be replaced.

Possible sources of interference of the wireless transmission between the child-side device and the car-side device can be found through testing in an environment similar to a car. Through

limited initial testing, with the antenna that we chose, the range of the communication is about 8 feet. While this should suffice in most cars, testing will show true operating conditions.

On the child-side device we are currently only using a buckled and unbuckled state of detecting a child in the seat. Further improvements could help with better child detection and help with both the false negative and false positive problems that could arise through misdetection of the child seat state. Weight and heat sensors placed in the child seat might help with this detection. This project is meant to be an after-market device and hence it needs to be as simple as possible for the user to add to existing car seats.

Since the child-side device will be near the child, somehow attached to the child seat, it is possible that a child could get a hold of the device and possibly take it apart or tamper with it. The battery, if removed by a child, could easily become a choking hazard. Design of the enclosure for the child-side device needs to take into account that there is a battery inside that needs to be easily replaced by an end-user, but also not so easy to take apart as to cause harm to a child.

An easily added feature that wasn't originally planned for this device would be to also detect a child out of the seat while the car is in motion. The CAN interface to the car can provide the necessary information to know whether the car is moving and we already know the state of the child seat. On detection of transition from buckled to unbuckled while the car is motion, the product could also audibly announce that the child has unbuckled their seatbelt.

6.3 Environmental Impact Analysis

In today's society, environmental impact has become very important when discussing product lifecycle. This includes manufacture, normal use and disposal/recycling. All stages of the product's life must be analyzed before bringing a product to market.

Almost all electronic devices run into similar environmental issues. In this project, we have two (or more) PCBs, lots of components on the PCBs, a battery in the child-side device, the

enclosures for both the child-side and car-side devices as well as an OBD-II cable for the car-side device.

Perhaps the largest environmental impact comes from the manufacture of the PCBs. PCB manufacturing involves the use of many different hazardous chemicals. These hazardous chemicals, once used, must be disposed of safely. Once the PCB is manufactured, the disposal of the PCB must then be looked at. PCBs can be recycled to get the lead and copper back out of them to be safely reused. An incentive could be provided with the product to make sure that the PCB is correctly disposed of.

The European Union adopted the Restriction of Hazardous Substances Directive (RoHS) in 2003 [1]. The RoHS directive limits the use of lead and other hazardous substances in the manufacture of electronic components. While this directive is not enforced in the United States, the EU is a large enough consumer of electronic components that all manufactures have moved to make components that are in compliance with the RoHS directive. All of the components in our design are RoHS compliant.

The next component with an environmental impact is the battery in the child-side device. This battery is a coin cell CR2032 battery. This battery is made by a wide variety of manufacturers. Panasonic has information on their website about the disposal procedures for these batteries [2]. While the battery can be disposed of as normal trash, there is a precaution that the battery contains more Perchlorate than allowed by the State of California [3]. The proper way to deal with this is to make sure the product documentation has an appropriate label warning the consumer of this risk. In California, these batteries must be disposed of at a hazardous waste landfill [4].

The final parts of the system that can have an environmental impact are the ABS plastic enclosures for both the child-side and car-side devices and the cable used to connect the car-side device to the car's OBD-II port. The ABS enclosure can be recycled through standard recycling practices. A recycle logo with the appropriate plastic type number can be stamped on the

enclosure to remind the consumer to recycle it. The cable can also be recycled but not through standard recycling practices.

To make sure that the products disposal is most environmentally friendly, we can provide documentation that explains to the consumer that the entire product can be taken to any electronic waste (E-waste) recycler. A web link pointing at various E-waste recycling center locators should make it easy for the consumer to properly dispose of this product after they are done using it.

6.4 Summary

In conclusion, much consideration and study must be done to mitigate both ethical and environmental concerns that arise from the manufacture and use of this project. Since this design is meant to protect kids from the dangers of being left in a car, perhaps the most work needs to be done on the reliability and function of the product. Through careful software testing and design most failures that would have an ethical impact can be dealt with. Making the product simple and fool-proof to use will also help. As for the environmental impact, RoHS compliance *and proper documentation and labeling of recycling procedures should go a long way towards being green.*

7 Packaging Design Considerations

7.1 Introduction

Project Forget-Me-Not is a system to keep track of children in their child seat of a car. It alerts the driver or nearby people when a child is left in the car for an extended period. This product is combination of two or more packages. The first package, from here on called the car-side package is the brains of the system. It contains the microcontroller and connects directly to the car through it's government required, industry standard OBD-II port. The second (or more, depending on number of child seats) package, called the child-side, is the part that attaches to each car seat that is to be monitored.

7.2 Commercial Product Packaging

Project Forget-Me-Not is a car safety system and while other car safety systems exist, like airbags (with passenger seat detection), traction control, assisted braking, etc., these other products don't really compare with this project. This project works closely with the child car seat typically purchased and installed by the consumer separately. Other car seat safety systems include LATCH (an industry standard seat tethering system), child safety locks and driver controlled window lockouts. Since the product for this project is initially going to be designed as an after-market device, it probably makes more sense to compare commercial devices that have similar packaging requirements and interfaces. The following two comparisons are useful only in user interface and OBD-II connection ideas.

7.3 Product #1

One product that has a similar car interface is the ecoRoute HD [1] from Garmin. This device is basically an OBD-II to Bluetooth adapter. The GPS head unit pairs with the ecoRoute HD and gets car diagnostic data to help aid in deciding the most economical route to guide the user to their destination. The product connector consists solely of the OBD-II port and also includes holes for tying the device under the dash. The device also consists of 2 LEDs to help indicate current working status and a reset button.

7.4 Product #2

Another OBD-II product on the market is the ElmCan [2]. The ElmCan line of products take the OBD-II interface and convert it to a choice of serial, USB, Bluetooth or WLAN. The OBD-II side of the device is a DB-9 connector for using a DB-9 to OBD-II cable. The ElmCan device is enclosed in a plastic case that is just big enough to house the PCB. The PCB for each user side interface is slightly different. There are 5 LED indicators on the top of the device. Three of the LEDs show OBD-II functionality and two of the LEDs show PC traffic.

7.5 Project Packaging Specifications

The packaging design for this project consists of two parts. The car-side device will consist of a 5" x 3" x 2" case. This case will house the 4" x 2.5" PCB, the OBD-II cable, 1 external button, 2 external LEDs and a speaker. The 2" height of the case is due to the mounting of the speaker. The OBD-II cable, buttons and LEDs, speaker will all be mounted directly to the

box with cables connecting to headers on the PCB. The LEDs and push button will be on the top side of the case to allow easy access for pairing of the child-side devices to the car-side device. The speaker will mount under the top of the case and holes will be drilled into the case to allow the sound to be heard. To mount under the dash, the box bottom is empty of any components and Velcro can be used for mounting. A conceptual drawing of the car-side device is included in Appendix B.

The child side packaging design is focused on size. It contains the 1.75” x 3” PCB, two external push buttons for child seat state and one LED. The small size is necessary to make sure that it can easily be attached to the car seat without getting in the way. One other factor in the child-side device is that the case must be easily opened to facilitate battery replacement that can be handled by the consumer. The child-side device can also be mounted via Velcro to the backside of the child seat. A conceptual drawing of the child-side device is included in Appendix B.

7.6 PCB Footprint Layout

The car side device has the most components as it includes the microcontroller, RF receiver, RF encoder, two CAN transceivers, audio amplifier. The only size requirements of this board are that it can fit in a box small enough to mount easily near the car’s OBD-II port. The largest component is the microcontroller. Headers will be available for the various parts of the microcontroller that we will be using. Also, headers for the CAN transceivers and RF receiver will be included on the PCB. The current estimate for the car-side PCB is 2.5” by 4”.

The child side device is very simple in comparison to the car-side device. The PCB layout is mostly taken up by the battery holder, antenna and header pins for the RF transmitter. The ICs for the RF components take up very little space. The PCB for the child-side is 1.75” by 3”.

7.7 Summary

This report consists of the packaging specifications for our project devices. It also includes a list of the necessary components and general PCB layout of these components. The

appendices include conceptual drawings of the enclosures, possible PCB layout and the listing of components.

8 Schematic Design Considerations

8.1 Introduction

Forget-Me-Not is an aftermarket vehicle safety system designed to prevent parents and caretakers from accidentally leaving their children in a vehicle. Every year children are forgotten in vehicles and are injured or die as a result. The Forget-Me-Not system comes in two parts in order to monitor both the vehicle state and the presence of a child. The vehicle unit will connect to the federally-mandated OBD-II port where it will be able to receive information from the vehicle and listen for communication from the child unit. The child unit will sit on a modified car seat and transmit wirelessly to the vehicle unit when it detects an insertion or removal of a child. If the vehicle unit determines that the car has been turned off with a child present it will produce an audible message after a given amount of time as a reminder. If the child is still not removed the vehicle unit will attempt to send messages on the vehicle CAN bus in order to draw attention to the vehicle or contact someone who can help.

8.2 Theory of Operation

The main subsystems of Forget-Me-Not include a voltage regulator, wireless communications circuitry, dual and single wire CAN nodes, a serial RS232 debugging port, and an audio amplifier. Each of these subsystems integrate with a Freescale 9S12DP512 with the exception of the child side hardware that does not use a microcontroller.

All components on the vehicle unit operate at +5V with the exception of the CAN transceivers which operate at +12V. Power is provided by the vehicle battery which is fed to a 7805 linear regulator. The wireless transceivers are designed to operate off of a battery and require an extremely clean signal hence the 7805 was chosen over a switching regulator. The child side board uses a +3.3V coin cell battery to power the Linx encoder and wireless transmitter. The power consumption of the child side board is very low and has an estimated battery life of 3 to 5 years.

The vehicle unit employs a Linx MS series decoder[1] which directly interfaces with a Linx LR receiver[2]. These modules were designed to be paired together making the interface very

simple. Similarly, the child unit uses a Linx encoder[3] that serially feeds a Linx transmitter[4]. The RF receiver and transmitter modules operates at 418MHz[2] at a rate of 2,400 baud set by the encoder and decoder. The unobstructed range of the transmitter is roughly 30 ft, but provisions were made for a PCB mounted antenna if additional signal stength is deemed necessary.

The Forget-Me-Not vehicle unit has two CAN interfaces to the OBD-II port. One is a single wire CAN node that is driven by a NCV7356 SWCAN[5] tranceiver, and the other is a high-speed, dual wire interface via a MCP2551 CAN[6] tranceiver. The SWCAN module will connected to the OBD-II pin designated for the manufacturers descretion. The rate of operation depends on the vehicle it is connected to. The high-speed CAN tranceiver will interface with the vehicles main CAN bus at a rate of either 500kb/s or 1Mb/s[6] depending on the vehicle.

The RS232 circuitry uses a MAX232 transceiver[7] with a built in level translator to output to a PCB mounted D-sub connector. The transceiver operates off of +5V and requires external capacitors supplied in order to double and invert the supply voltage resulting in ±10V[7]. An additional RS232 driver is available in the transceiver if a need for it arises.

The last main subsection of the Forget-Me-Not system is the audio amplifier. The amplifier circuit is driven by the microcontroller where it is low-pass filtered above 4KHz and fed into a LM386 power amplifier[8]. The LM386 is configured for a gain of 20[8] and used to drive an 8 Ohm speaker mounted on the PCB. Later designs have opted to replace the LM386 with an IRL510N power MOSFET to reduce cost and complexity.

8.3 Hardware Design Narrative

The main purpose of our uC is to act upon a child forgotten event in an intelligent manner. To accomplish this it interfaces with the Linx wireless decoder using 8 GPIO pins for data reception, and one SCI port for serial reception of the transmitted ID. This allows the uC to keep track of when a child is present. Meanwhile, the uC uses two ADC ports to read voltages on the OBD-II port to determine if the car is running and two CAN ports for the high-speed and

single wire CAN transceivers to sniff the CAN busses. Depending on the vehicle, our design may be able to issue commands to the vehicle to further warn the driver. These commands can perform functions such as honking the horn, sounding the car alarm, and rolling down the windows. One routine that is run when a child forgotten event occurs is an audible warning. A PWM signal with varying duty-cycle is used to drive our audio amplifier circuit which drives an 8 Ohm speaker. The RS232 tranceiver communicates via an SCI module that is used to connect to a PC for debugging purposes and additional CAN, PWM, ADC and GPIO pins are routed to headers in case there is a need for them. The child side board has no microcontroller and therefor its only interface to the DP512 is through the wireless communications link.

8.4 Summary

The Forget-Me-Not child safety system employs two units to prevent the accidental injury or death of a child from being left unattended in a vehicle. This is accomplished by pairing a wireless transmitter module with a mainboard receiver where the mainboard has the ability to warn the driver or passers-by of the possibly dangerous situation. By connecting directly to the vehicle CAN bus, the vehicle side unit can issue commands to the car that allows various functionality depending on the vehicle, in addition to the audio warning system.

9 PCB Layout Design Considerations

9.1 Introduction

Project Forget-Me-Not is a system to keep track of children in child seats in a car. It alerts the driver and other nearby people when a child is left in the car for an extended period. This product is a combination of two or more packages, which will subsequently require two or more PCBs. The first package, called the 'car-side', is the more complex and contains a microcontroller, audio amplifier, RF circuitry, power supply, debugger, and other minor elements on its layout. The second package/layout, called the 'child-side', is the part that attaches to each car seat that is to be monitored, and contains a battery supply system and RF circuitry matched to the car-side.

9.2 PCB Layout Design Considerations – Overall

Overall PCB layout considerations are split into two separate categories: technical requirements and manufacturing requirements. Technical requirements include antenna placement and optimization, generation of accurate timing sources, EMI reduction for passing FCC requirements, and extensibility. Technical requirements will be discussed in detail in subsequent sections.

Manufacturing requirements are as important as the technical requirements in this design. As the end product is designed to be produced in large quantities and sold at retail, a number of considerations arise. The primary consideration is price. Each component in the layout, whether it be a capacitor, power supply, microcontroller, or other, has been optimized for total cost in large quantities. After component cost, board manufacturing costs have been considered. Essentially, every device that can be a SMD, is. The final product should have only one PTH connection for the OBD-II cable (excluding development debug headers, which are not part of the final product) such that every other component can be configured via a pick-and-place machine for automated assembly. Finally, board cost and manufacturability itself is another issue. The use of no smaller than 10 mil traces allows the use of less expensive processes, and fewer continuity errors. High margins for keep-out areas around planes and vias have been employed to reduce the percentage of bad boards.

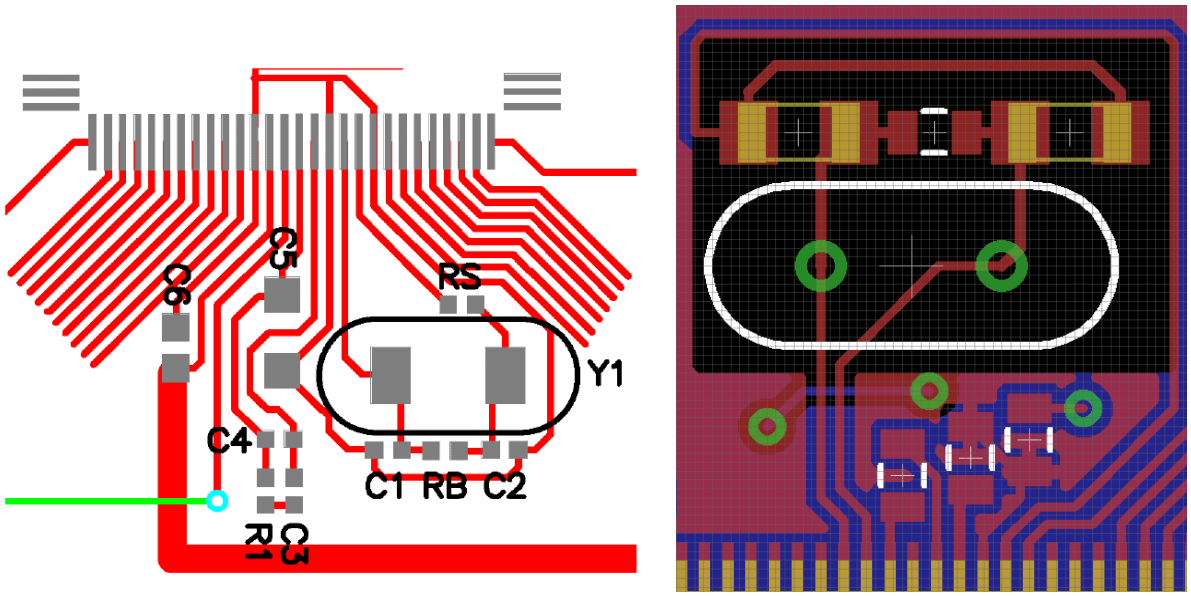
9.3 PCB Layout Design Considerations – Microcontroller

The three primary design considerations for the 9S12 microcontroller are the crystal oscillator layout, proper isolation of the internal 2.5V supply, and correct placement of bypass capacitors. Most importantly, this design requires a very accurate timing source in order to meet the requirements of the High-speed CAN protocol on the OBD-II port.

To meet the timing requirement, a noise-resistant Pierce oscillator was chosen over a Colpitts-based design. Load capacitors were calculated and chosen from the 1% tolerance C0G series, with the larger of the closest value capacitors chosen to account for the tendency of Pierce oscillators to be a few hundred kilohertz higher than their Colpitts counterparts. Removal of the ground and power plane nearby was required to avoid any parasitic capacitance effects, and all traces were routed without vias. Finally, a ground isolation loop was run around the entire oscillator and back to an isolated connection to the ground plane at one point. These design considerations exceed the Freescale recommendations. In the figure below, note that Rs is missing: it is only recommended for lower-frequency crystals than the 16MHz part used.

Isolation of the 2.5V internal supply was simple: the external bypass capacitors were added with only an external connection to the ground plane. Other bypass capacitors for the 5V supplies, PLL, and ATD converters were placed as closely to the pins as possible with direct connections to the ground plane.

Figure 1.0 - Recommended Oscillator Layout and Actual Layout



9.4 PCB Layout Design Considerations - Power Supply

Power supply layout in this design is fairly simple because of the requirement for a very clean source. The Linx receivers are designed to be run from batteries and are not very noise tolerant, therefore a linear regulator was chosen over a switch-mode regulator. Power supply trace routing is handled by using discrete power and ground planes. A ground plane was chosen to reduce EMI, act as a heat-sink for the audio amplifier and linear regulator, and to simplify the layout. Because the entire back of the LM7805 regulator is a ground, effective keep-out polygons were defined to avoid grounding signals.

Bypass capacitors are extensively used throughout the layout. The linear regulator has a large bulk capacitor on the input side, as well as a smaller bypass capacitor on the output. The microcontroller has seven pure bypass capacitors, and each other IC has at least one bypass capacitor to provide current as necessary during switching. The RF circuitry has both bypass capacitors and supply-filtering capacitors. Because of the power and ground plane pours, there are no power traces. However, care was taken to avoid ground loops or high impedance paths in the current supply and return paths.

9.5 PCB Layout Design Considerations – Antenna Placement and Decoder Operation

As wireless reception is paramount to the operation of this system, placement and routing of the antenna, receiver, and decoder were completed first. The antenna requires placement with no copper pours under it, so restricts were placed to remove these planes automatically. The receiver requires no planes, traces or vias on the same layer on which its placed, but strongly suggests the use of a ground plane under it on a separate layer. Both of these requirements were met again with restricts to remove the automatic pour instead of resorting to a polygonal path.

9.6 Summary

As discussed earlier, primary layout concerns were the functionality of the RF circuitry, the accuracy of the crystal oscillator, and the proper bypass and filtering of the power supply. Functionality of the RF circuitry was accomplished by precisely following the constraints of the design documentation and routing these elements first to ensure absolute shortest trace length. Crystal oscillator accuracy requirements were met by using a noise-resistant Pierce oscillator with compensation for negative tendencies of this circuit, as well as by additional ground loops beyond the default design notes by Freescale. Extensive bypass capacitors coupled to the ground and power plane ensure proper supply to each circuit element, as well as overall EMI reduction and impedance for supply and return currents. Together, these design elements should be sufficient to ensure proper operation of the device.

10 Software Design Considerations

10.1 Introduction

Project Forget-Me-Not is a warning system for when a child is left in a vehicle. It communicates wirelessly via RF signal to a child safety seat to detect when a child is in the vehicle, and detects vehicle state via the driver side OBD-II port. It then audibly warns the driver of the child still in the vehicle, much like when the car door is opened and the keys are still in the ignition or the headlights are still on. The main board (hereafter referred to as “car-side”) operates using a Freescale 9S12D microcontroller, while the board in the child seat

("child-side") does not need a microcontroller. The 9S12D runs an interrupt-based software configuration, and utilizes several on-chip peripherals such as PWM for audio output and SCI for debugging.

10.2Software Design Considerations

10.2.1 Memory Constraints

There are not any serious constraints on memory for this project. The 9S12 programs to flash, and there is more than enough available. Initially the microcontroller chosen had 256 KB of flash, but because that specific package was no longer in production, a package with 512 KB of flash was chosen. The spacious memory is needed for audio samples. However, assuming half of the available flash memory is devoted to audio samples (which it will not be), that would yield 23.78 seconds of 8-bit uncompressed audio [1]. This will be plenty for a few simple voice confirmations, like "(number) child(ren) detected" as well as buckle/unbuckle and vehicle state confirmation. There are no computation-heavy functions either, so the 14 KB of RAM is sufficient [2].

10.2.2 External Interface Mapping

Audio output is generated by the PWM channel 0 on port P, pin 0. CAN receiving and transmitting is done on port M; pins 1 and 2 are CAN channel 0 and pins 3 and 4 are CAN channel 1. The single wire CAN transceiver's mode of operations is controlled by I/O pins on port K, pins 2 and 3. Port S is serial communications; RF messages are received on pin 2 and debug data is transmitted and received over pins 1 and 0 [2].

10.2.3 Peripheral Utilization

For this project, the 9S12D will utilize its PWM, CAN modules, ATD for sensing vehicle state, and SCI for debugging. Initialization is handled by memory mapped registers; the memory map for the 9S12DP512 is shown in appendix C.

10.2.4 Code Organization

The main code for the microcontroller is interrupt-driven. This is because all actions taken by the microcontroller only need to happen after activity on the CAN bus or the RF decoder. This implementation will also save power on the car-side.

10.2.5 Debugging Provisions

A test module has been written and tested on the current prototype which shows ATD values and sends junk CAN messages, but it is yet to be decided whether the module will be necessary in the final version of the software. The microcontroller has a serial interface through which debugging data can be obtained, so it may still be necessary. At present there is only one car available to the design team that supports CAN actions and has the necessary CAN messages available; many manufacturers keep this information private.

10.3 Software Design Narrative

The overall sequence of software events is simple. The car-side device powers up and initializes. It then processes events as they come, which are outlined below. The microcontroller will go into a sleep state when no relevant actions are happening to save power. When a child is in a seat and the vehicle is turned off, a timer starts. If the child is still in the seat when the timer expires, the alarm state is triggered until the child is unbuckled or the vehicle is turned back on.
Status: Written, working

10.3.1 Initialization

Once initialized, the microcontroller will wait for interrupts from one of the several relevant functions. The main loop will run when the microcontroller is not in a sleep state.

10.3.2 Child Seat Event

This function will be called from the main loop in the event that a message is received from the RF transceiver. It will set a flag to indicate whether a child is present in the child seat or not. If there is no child present, the main board can go into a sleep state, and the timer can be disabled if it is running.
Status: Written, working

10.3.3 Car Ignition Event

When the vehicle ignition is turned on or off, this function will handle setting the ignition flag in code. When the car is turned from on to off, it will start a timer depending on the child seat flag. This section has been combined with the CAN event in section 3.5. For vehicles that do not support CAN, ignition can be monitored via an ATD channel, which will toggle this flag and start the timer accordingly.

Status: Written, working

10.3.4 Timer Expired Event

The timer is started when there is a child present and the vehicle ignition is turned off. The initial timer setting will be between 30-40 seconds, after which this event will trigger a series of other functions. This may also be used in conjunction with the door open event triggered by CAN, because if the ignition is off but the door has not been opened, the driver would still be in the car with the child. In the event that the microcontroller cannot detect the door opening, this event will start the audio warning function as well as other optional CAN functions.

Status: Written, working

10.3.5 CAN Event

This event is triggered by a CAN message, which in this case will be received when the car is turned on or off. However this is not supported in all vehicles. Compatibility will need to be determined before the event handler is initialized, most likely by pinging the vehicle's CAN network to see what version is supported. For the vehicles that do support it, though, the car ignition message functionality should be present as discussed in section 3.3. The handler would call a function which would in turn set the ignition flag and check the child seat flags and start the timer accordingly. It is denoted in appendix B as CANmsgRcvd ().

Status: Written, working

10.3.6 Sound Alarm

When deemed appropriate by the timer and the other interrupts detected, the car-side board will need to sound an audible warning. This function will load a sample from flash memory and generate it using the PWM. As discussed in section 2.1, utilizing just half of the

flash memory would yield over 23 seconds of audio samples. This would be plenty for a verbal warning of "There is a child in the car." There is also an indication of which child seat is occupied, in the case that there is more than one seat.

Status: Written, working

10.3.7 CAN Actions

After the audio warning, other actions may need to be taken to save the child. The device currently triggers the windows rolling down and the doors unlocking, followed by a short delay and then the car alarm being activated. These actions are all accessed from the car-side board via CAN commands. Currently only newer GM vehicles are supported, because manufactures keep their proprietary CAN commands confidential. This function is denoted in appendix B as sendCANmsg () [3].

Status: Written, working

10.4 Summary

Project Forget-Me-Not has a very simple software structure. Only one board has a microcontroller, and the actions of that board are all triggered by events. Thus, an interrupt-driven code organization is used, reliant on flags for child-seat and vehicle ignition state. The interrupts then check these flags and perform the appropriate action, such as sounding an audible warning and performing various actions on the vehicle via the CAN bus. All other functions are handled in hardware in the various components on both the child- and car-side boards.

11. Version 2 Changes

Continued development of this project is following an iterative path. The primary objective is to drastically reduce production costs to make this device affordable for the masses. To accomplish this goal, the following changes are being made, in order:

Phase 1: Removal of High-Speed CAN transceiver, removal of RS232 level translator, removal of the Linx decoder, switching from the 9S12 series to the 9S08 series microcontrollers, and changing RF receivers to the SiLabs 4312. Even with these changes, the main board will

continue to interoperate with the original child boards, as the RF architecture remains a 433.92 MHz OOK system. Additionally, the integrated helical antenna will be changed to a PCB trace antenna, as the range requirements are minimal.

Phase 2: Changing the child side boards to match the reduction in costs on the main board. The Linx transmitter and encoder will be replaced with a single SiLabs 4010 integrated transmitter and programmable 8051 processor. This reduces our production costs to below $5 for each child side board, while allowing us to integrate 128-bit AES encryption, low battery detection and indication, and other features.

12. Summary and Conclusions

Project Forget-Me-Not is a safety seat monitor for use in vehicles by parents. It is made up of two parts: a seat-side and a car-side device. The seat-side device detects the presence of a child in a child safety seat and transmits its state to the car-side device. The car-side will sound an alarm, roll down windows, and unlock the doors if a child is left in a car when the car is off. The car-side device interfaces with the government-mandated OBD-II port to detect vehicle state.

The seat-side PCB does not require a microcontroller and consists of an RF transmitter with 2 buttons to determine seat state. The car-side has a microcontroller that utilizes an RF encoder, a single-wire CAN transceiver, and PWM audio output. The car-side is packaged with a speaker to give an audible warning in the event that the vehicle's CAN is not supported.

The final deliverables of Project Forget-Me-Not include two types of pieces, the car-side device and the child-side device. The car-side device consists of a project box that has an OBD-II cable coming out of it, a speaker, a button and two LEDs. This box is the main component of the system and only one will exist per car. The button is for pairing of new child-side devices. The LEDs show power and learn status. The power and car interface all connect through the OBD-II cable. The child-side device(s) consists of a small box with two buttons and an LED. Multiple child-side devices can exist, we made two for this project. The buttons are for communicating the current seat state (buckled or unbuckled) to the car-side device. The LED shows when the device is transmitting to help troubleshoot communication problems. The child-side device is

powered by a single CR-2032 coin cell battery that is easily replaced. All aspects of both the car-side and child-side devices work. The car-side device communicates with the car getting current car status as well as sending commands to the car to roll down the windows and turn on the car alarm when it detects that one of the child-side devices is still in the buckled state while the car has been off for a set period of time.

13. References

Packaging References

[1] ecoRoute HD, “ecoRoute HD” [Online]. Available: https://buy.garmin.com/shop/shop.do?pID=38354. [Accessed: September 24, 2010].

[2] ElmCan, “OBD-II Productinfo ElmCan ELM327” [Online]. Available: http://obd-ii.de/kabelec.html. [Accessed: September 24, 2010].

Software References

[1] AN2250: Audio Reproduction on HCS12 Microcontrollers. [Online] Available: http://cache.freescale.com/files/product/doc/AN2250.pdf?fsrch=1&sr=1 [Accessed 27 October 2010]

[2] MC9S12DP512 Device Guide. [Online] Available: http://cache.freescale.com/files/microcontrollers/doc/data_sheet/9S12DT256_ZIP.zip?fsrch=1&sr=1 Also available: https://engineering.purdue.edu/477grp4/Datasheets/9S12DP512DGV1.pdf [Accessed 16 September 2010

[3] MSCAN. [Online] Avaliable: http://www.embeddedrelated.com/groups/68hc12/show/6819.php [Accessed 22 September 2010]

Schematic and Hardware Design References

[1] Linx Technologies, “MS Series Decoder Data Guide”, Linx Technologies. [Online]. Available: **https://engineering.purdue.edu/477grp4/Datasheets/Linx%20XCVRs/LICAL-DEC-MS001_Data_Guide.pdf**.

[2] Linx Technologies, “LR Series Receiver Module Data Guide”, Linx Technologies. [Online]. Available: **https://engineering.purdue.edu/477grp4/Datasheets/Linx%20XCVRs/RXM-xxx-LR_Data_Guide%5b1%5d.pdf**.

[3] Linx Technologies, “MS Series Encoder Data Guide”, Linx Technologies. [Online]. Available: **https://engineering.purdue.edu/477grp4/Datasheets/Linx%20XCVRs/LICAL-ENC-MS001_Data_Guide.pdf**.

[4] Linx Technologies, “LR Series Transmitter Module Data Guide”, Linx Technologies. [Online]. Available: **https://engineering.purdue.edu/477grp4/Datasheets/Linx%20XCVRs/TXM-xxx-LR_Data_Guide%5b1%5d.pdf**.

[5] On Semiconductor, “NCV7356 Single Wire CAN Transceiver”, On Semiconductor. [Online]. Available: **https://engineering.purdue.edu/477grp4/Datasheets/Single%20Wire%20CAN%20-%20NCV7356-D.PDF**.

[6] Microchip Technology, “MCP2551 High-Speed CAN Tranceiver”, Microchip Technology. [Online]. Available: **https://engineering.purdue.edu/477grp4/Datasheets/High%20Speed%20CAN%20-%20MCP2551.pdf**.

[7] MAXIM, “Multichannel RS-232 Drivers/Receivers”, MAXIM. [Online]. Available: **https://engineering.purdue.edu/477grp4/Datasheets/MAX220-MAX249.pdf**.

[8] National Semiconductor, “LM386 Low Voltage Audio Power Amplifier”, National Semiconductor. [Online]. Available: **https://engineering.purdue.edu/477grp4/Datasheets/Audio%20Amplifiers/LM386%5b1%5d.pdf**.

Patent References

[1] US Patent 5949340 - “Warning system for detecting presence of child in an infant seat”

[2] US Patent 6998988 - “Infant alarm system for automobile”

[3] US Patent 5793291 - "Child alert system for automobile"

[4] US Patent 7733228 - "Wireless system to detect presence of child in a baby seat"

[5] US Patent 7012533 - "Occupant detection and notification system for use with a child seat"

[6] *ChildMinder* – http://www.babyalertinfo.com

PCB Layout References

[1] Freescale, "Designing Hardware for the HCS12 D-Family" [Online]. Available: http://www.freescale.com/files/microcontrollers/doc/app_note/AN2727.pdf. [Accessed: August 15, 2010].

[2] Linx Technologies, "RXM-xxx-LR Data Guide" [Online]. Available: http://www.linxtechnologies.com/Documents/RXM-xxx-LR_Data_Guide.pdf. [Accessed: August 15, 2010].

[3] Linx Technologies, "TXM-xxx-LR Data Guide" [Online]. Available: http://www.linxtechnologies.com/Documents/TXM-xxx-LR_Data_Guide.pdf. [Accessed: August 15, 2010].

[4] Linx Technologies, "LICAL-DEC-MS001 Data Guide" [Online]. Available: http://www.linxtechnologies.com/Documents/LICAL-DEC-MS001_Data_Guide.pdf. [Accessed: August 15, 2010].

[5] Linx Technologies, "LICAL-ENC-MS001 Data Guide" [Online]. Available: http://www.linxtechnologies.com/Documents/LICAL-ENC-MS001_Data_Guide.pdf. [Accessed: August 15, 2010].

Reliability and Safety References

[1] United States Department of Defense, "Military Handbook on Reliability Prediction of Electronic Equipment," Dec. 1991 [online]. Available: **https://engineering.purdue.edu/ece477/Homework/CommonRefs/Mil-Hdbk-217F.pdf**. [Accessed: 10 November 2010].

[2] MC9S12DP512 Device Guide. [Online] Available: **http://cache.freescale.com/files/microcontrollers/doc/data_sheet/9S12DT256_ZIP.zip?fsrch=1&sr=1** Also available:

https://engineering.purdue.edu/477grp4/Datasheets/9S12DP512DGV1.pdf [Accessed: 10 November 2010].

[3] MSCAN. [Online] Available: **http://www.embeddedrelated.com/groups/68hc12/show/6819.php** [Accessed: 10 November 2010]

[4] International Rectifier. "IRL510," [online]. Available: **http://www.datasheetcatalog.org/datasheet/irf/irl510.pdf**. [Accessed: 10 November 2010].

[5] Fairchild Semiconductor. "LM7805 3-Terminal 1A Positive Rectifier". April. 2010. [online]. Available: **http://www.fairchildsemi.com/ds/LM/LM7805.pdf**. [Accessed: 10 November 2010].

Appendix A: Individual Contributions

A.1 Contributions of Mike Williams:

My personal contributions were primarily technical. My earliest work started with component selection. First was determining exactly what sort of wireless devices we needed: it needed to be extremely low power, one-way, primarily designed for transmitting simple switch closures, not data, and be easily attainable. Next, microcontroller selection was the second most important aspect. Nearly every large microcontroller manufacturer could provide a system which satisfied our requirements, but Freescale had competitive costs and large amounts of available flash for audio samples. Because of prior familiarity from ECE 362, personal work experience, and possession of debugging equipment, it seemed like a logical choice. After this point, the remaining selections were simply picking components that matched our basic requirements, such as choosing a linear voltage regulator that outputs 5V from a 12V-25V input.

After component selection was completed, I began working on schematic capture and prototyping. I ordered development kits for both the Linx MS series chips and the 9S12DP256 kit with built-in CAN transceivers. These together with jumper wiring allowed software development to begin at the very start of the project. Our development setup matched our eventual schematic and PCB layout pin-for-pin. With the schematic capture completed and the development board setup proving the feasibility of the design, I proceeded to lay out the PCBs.

The first PCB was the child-side board primarily because I had no experience with PCB layout and decided it would be better to start with the simplest piece. This also allowed me to test the skillet-reflow technique on a smaller board before moving on to the 112-pin LQFP 9S12. Soldering went well, and the child boards were actually completed and communicating with the prototyped other half long before schematic capture was completed on the car board.

The second PCB was started immediately after this, as the child side boards were complete until packaging began. Numerous revisions of this board were made, as errors were discovered in the layouts of some of the components. Additionally, optimization of the routing was performed as some of the traces were very noise sensitive or required accurate timing. After

receiving the PCB, I nearly destroyed half of the components in a soldering disaster. Many hours of later, the board was at least allowing us to flash the micro and run code.

Next, flashing of the board test and debugging code was the top priority. This allowed us to individually trigger each peripheral and test complete functionality as I soldered them to the PCB one-by-one. When each individual piece was completed, the initial prototyping software was loaded on to the board and software development continued. At this point, my primary contributions became testing the software on all of the different vehicles we were trying to support and updating the CAN transceiver code to control the SWCAN network. Other areas of programming include developing the audio support, decoding the RF transceiver messages, and correctly structuring the interrupts such that all features of the system operated in real-time.

At this point my contributions were complete until the packaging team required me to modify some of the boards to fit their cases.

A.2 Contributions of Andrew Sydelko:

The project started out with Mike having some concrete ideas in his head about how the various parts should work. From early on, I had to learn what his ideas were so that we could translate them into a working project. Using his ideas, I was able to do the Packaging Constraints homework and come up with what the final packaging might look like. Other than a few minor changes, the end product looked very much like what I had come up with for the preliminary packaging. The only changes were a few buttons, no DB-9 port for the OBD-II cable and no external interface on the child-side device for detecting child state.

When we started doing the software part of the project, I had to come up to speed on using CodeWarrior. The last time I had programmed a microcontroller, it was straight assembly. CodeWarrior is very picky about how you do things, but once I learned it, setting the appropriate register values and writing code turned out to be pretty easy. One of the first software features that we attempted to get working was interfacing to the CAN emulator that we had. Craig and I

used some example code found to get basic CAN messages sent. It took us a while to get the emulator to see the messages, but once the correct initialization of the baud rate was done, we could finally see the messages on the CAN bus.

Other software work that I helped with was audio playback through a PWM output on the micro. Once the necessary hardware was in place on the PCB, Mike and I worked on figuring out why the audio wasn't playing. Initially, all we could hear was static. Through troubleshooting and debugging we found that the flash area that we thought was being read for the audio was wrong, which was why it was just static. Once we figured that out I suggested that we temporarily house the test audio sample in the main part of the code. This worked and proved that the hardware was functioning correctly. The next step was to move the audio data to flash. This ended up being harder than expected due to the way CodeWarrior wanted the source setup to correctly program the flash. Once we figured that out, CodeWarrior through up another barrier, a 32kB C code limit on the basic version. Converting the audio source files directly to assembly solved this problem.

After the main PCB was fully populated, I started to think about actually packaging everything to look nice. At this year's Fort Wayne HamFest, I found perfect enclosures for both the car-side PCB and for the child-side PCB. Probably the hardest challenge in packaging was getting the OBD-II cable to have a connector that would connect to the header that we had on the car-side PCB. Using a Radio Shack copper grid board, I soldered a female 10x2 header to the board and then drilled two holes for the cable wires to run through for strain relief. Once the whole assembly was soldered together, I cut down the board to fit, while attached to the PCB, within the project enclosure. The other parts of the enclosure was just drilling the appropriate holes for the speaker grill, the LEDs and the buttons. The child-side board was even easier, with just three holes being needed.

After packaging, Mike and I tested everything again inside their final enclosure. It all worked and we demoed it to the course staff. I also worked on the User Manual with Craig and finished putting together the submission for the Final Report.

A.3 Contributions of Craig Pilcher:

During the early stages, Mike had already laid out much of the hardware for both devices, so I worked with his early progress on the design constraint homework. I considered the main design constraint to be reliability, but we chose components based on several factors. The 9S12 was chosen for ease of use and expansive memory, the linear voltage regulator for its low cost and RFI/EMI, and the class D amplifier because it was suggested by the course staff (and switched out in the end product). The RF transceivers chosen were a little expensive, but contained lots of built-in functionality needed for the project, like pairing ability. Some components chosen (such as the original CAN transceivers and 256k microcontroller) were not available at any vendors, and changes had to be made accordingly.

My biggest contribution came on the software side. In the beginning of the semester, I researched audio applications on Freescale microcontrollers, finding how much sample space would be available and the bitrate to which the audio would be encoded. I researched the software design constraints for our project and re-familiarized myself with the S12 architecture for the software design narrative homework. When it came time for the software design, I generated the software flow and outlined all the modules that needed to be written. I chose an interrupt-driven software flow because it lent itself to the structure of the product's application. Flags are set when events happen like buckling and car ignition, and if certain flags are set when a certain event happens, actions are taken (in this case, setting off an alarm state).

Of my time devoted to the project, most of it was spent on debugging code. I assisted in writing the main loop for changing vehicle state, as well as deciding which functions should be called in which states. Andy and I found code for sending CAN messages and debugged the software to send the messages at the correct baud rate, and fixed the auto-generated UART code which caused framing errors when sending debug messages. I found an example of how to create 16kb arrays of audio data in Code Warrior and helped find the correct commands to store the samples in flash.

Towards the end of the project, I assisted Andy with the packaging of the device. This involved determining which pins of the OBD-II port to use and connecting them to headers. We also drilled holes for buttons and LEDs and made labels for each. Because there was no design work left, I did some brainstorming about what could have been done better on the project. The major deficiencies of the project that I found included buttons for the buckling mechanism instead of a buckle sensor or weight sensor, as well as the fact that rolling down the car windows will not save a child if the outside temperature especially cold.

A.4 Contributions of Chris Cadwallader:

My contributions were primarily related to software and interfacing. With the dev boards in our hands early in the semester we immediately began writing driver routines for seriel communication and debugging. These routines allowed us to prototype our audio and ATD circuits and verify that they work.

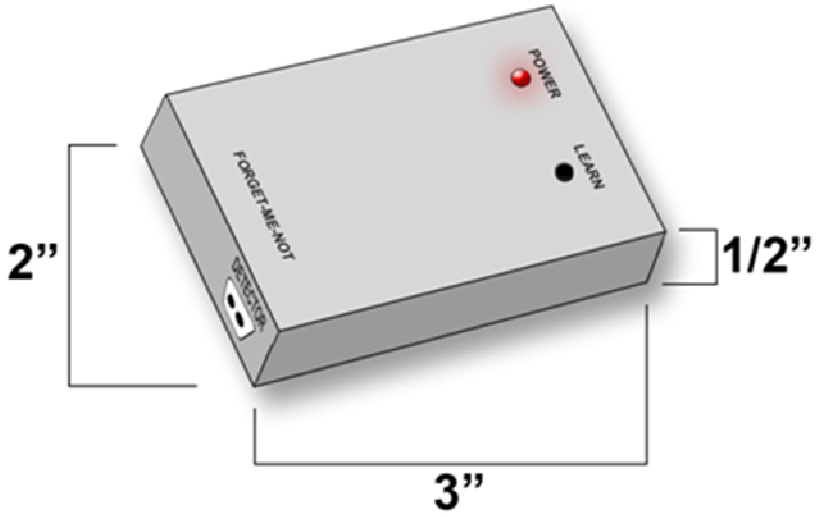
We knew early into the semester that CAN communication was one of our ultimate goals so after the basic debugging code was set up I worked with my group to configure the CAN module correctly. My experience working with CAN and LIN from an internship was very helpful in giving my team an understanding of how CAN works. After the CAN module was correctly configured we used our debugging interface to verify we could send and receive messages over a CAN bus. The SAINT-2 network emulator was very helpful in our testing process. By being able to send and receive over a CAN bus we were then able to control any aspect of a vehicle that the vehicle would allow. It was only a matter of sending the right message. Once the CAN driver was written, Andy, Craig and I all worked on the main state machine to track the vehicle and child states and triggering audio and CAN messages accordingly.

In addition to my contributions on the software side I also assisted the hardware team in verifying and debugging their design. When it came time to do a design review we sat down

and reevaulated every design choice we could. All major component datasheets were reexamined paying close attention to the hardware interface for correctness. All pinouts were looked at again especially the microcontrollers and the schematic was cross-referenced with the PCB layout to ensure consistency between the two. This was a very time consuming process but through it we managed to find several errors in the design and incorrect component values.

Appendix B: Packaging






POWER
LEARN
PROJECT
FORGET-ME-NOT


TRANSMIT
BUCKLE
UNBUCKLE
PROJECT
FORGET-ME-NOT

Appendix C: Schematic

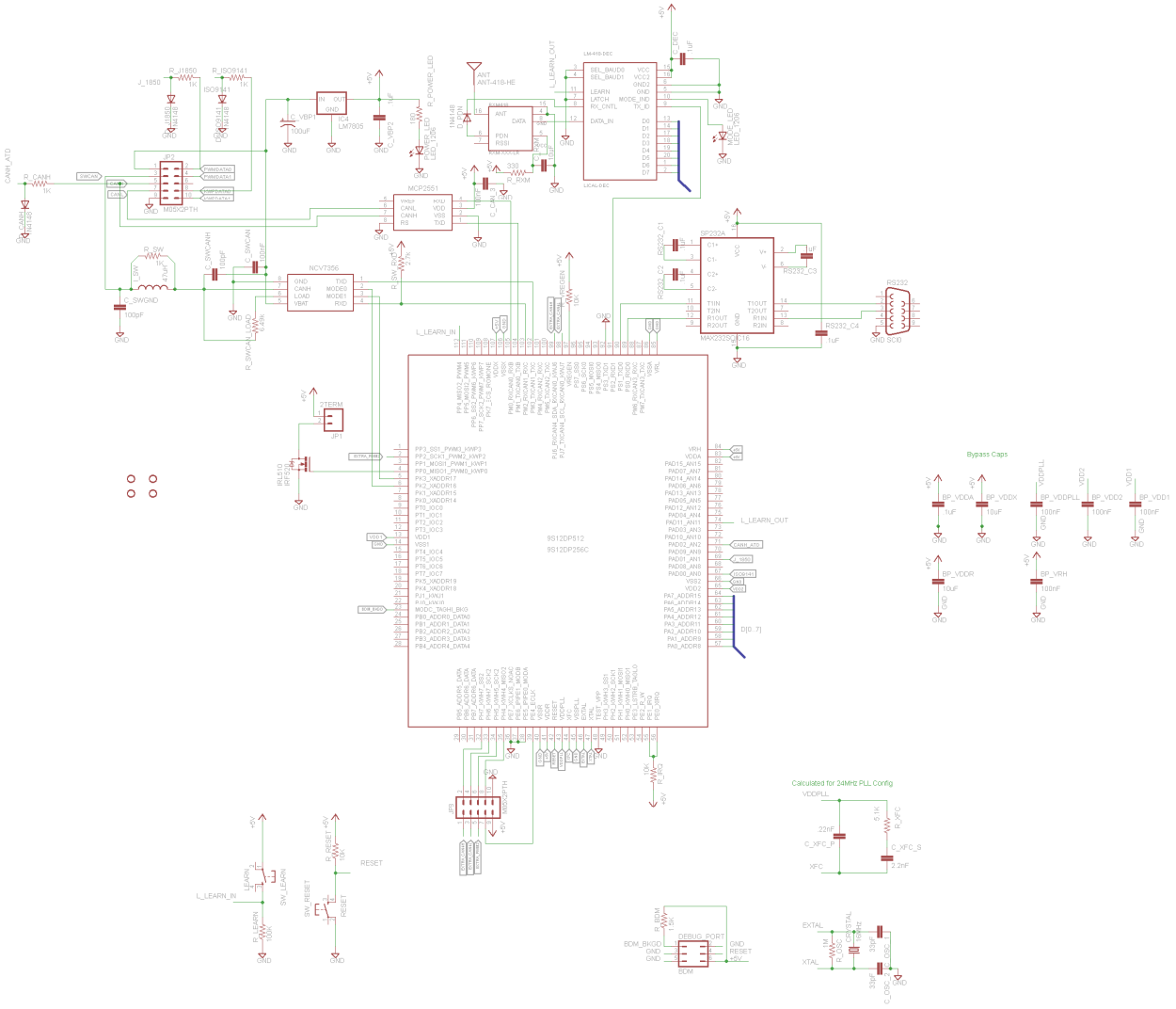


Figure C.1 – Main Board Schematic

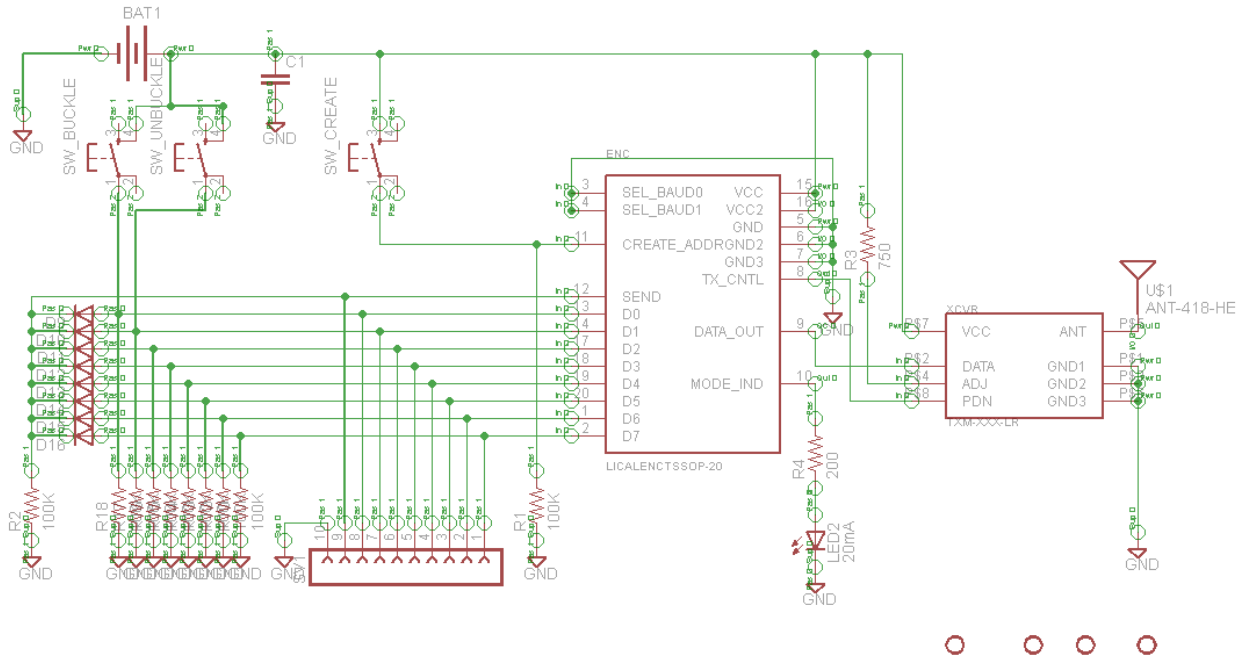


Figure C.2 – Child Board Schematic

Appendix D: PCB Layout Top and Bottom Copper

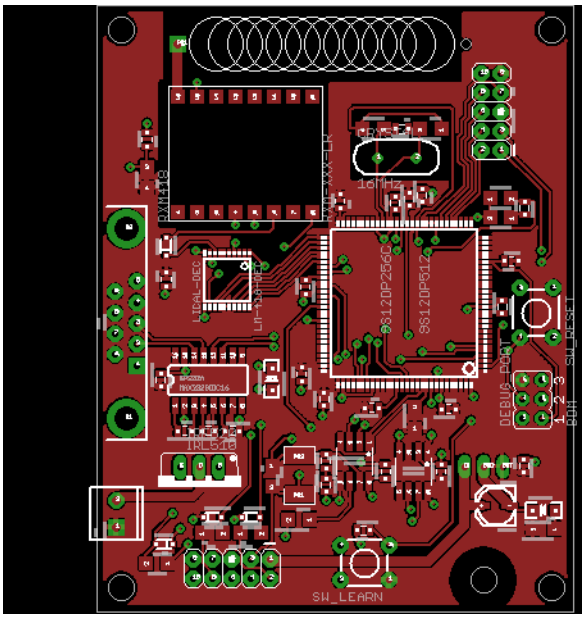


Figure D.1 –Car Side Top Copper with Silkscreen

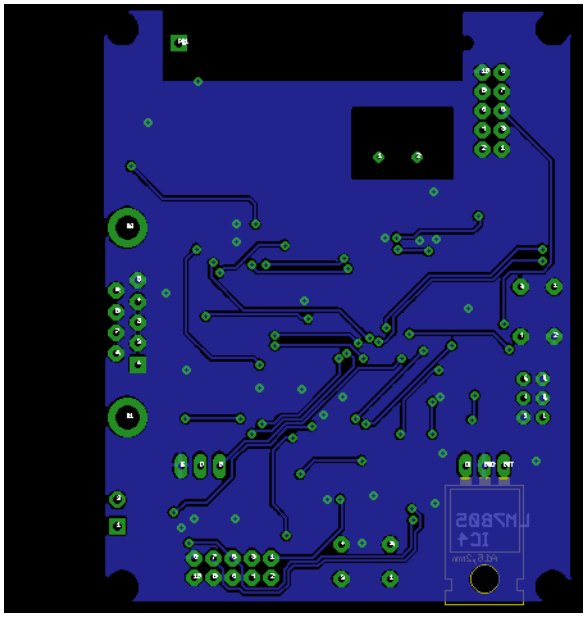


Figure D.2 – Bottom Copper with Silkscreen

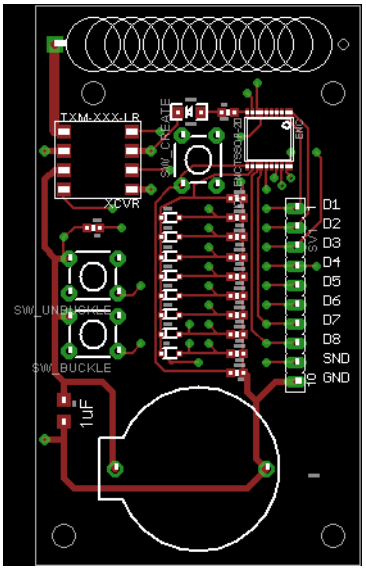


Figure D.3 – Child Side Top Copper

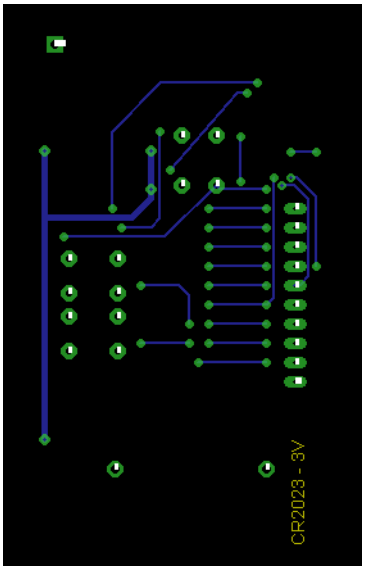


Figure D.4 – Child Side Bottom Copper

Appendix E: Parts List Spreadsheet

Child Board BOM (Bill-of-Materials)

Part	Value	Device	Package	Description
BAT1		BATTERY20PTH	BATTCOM_20MM_PTH	Battery Holders
C1		CAP1206	1206	Capacitor
D9		DIODESOD	SOD-323	Diode
D10		DIODESOD	SOD-323	Diode
D11		DIODESOD	SOD-323	Diode
D12		DIODESOD	SOD-323	Diode
D13		DIODESOD	SOD-323	Diode
D14		DIODESOD	SOD-323	Diode
D15		DIODESOD	SOD-323	Diode
D16		DIODESOD	SOD-323	Diode
ENC	LICALENCTSSOP-20	LICALENCTSSOP-20	TSSOP-20	Linx Techologies' MS Series Encod
JP1	STAND-OFFTIGHT	STAND-OFFTIGHT	STAND-OFF-TIGHT	Stand Off
JP2	STAND-OFFTIGHT	STAND-OFFTIGHT	STAND-OFF-TIGHT	Stand Off
JP3	STAND-OFFTIGHT	STAND-OFFTIGHT	STAND-OFF-TIGHT	Stand Off
JP4	STAND-OFFTIGHT	STAND-OFFTIGHT	STAND-OFF-TIGHT	Stand Off
LED2	20mA	LED1206	LED-1206	LEDs
R1	100K	RESISTOR0603-RES	0603-RES	Resistor
R2	100K	RESISTOR0603-RES	0603-RES	Resistor
R3	750	RESISTOR0603-RES	0603-RES	Resistor
R4	200	RESISTOR0603-RES	0603-RES	Resistor
R10	100K	RESISTOR0603-RES	0603-RES	Resistor
R12	100K	RESISTOR0603-RES	0603-RES	Resistor
R13	100K	RESISTOR0603-RES	0603-RES	Resistor
R14	100K	RESISTOR0603-RES	0603-RES	Resistor
R15	100K	RESISTOR0603-RES	0603-RES	Resistor
R16	100K	RESISTOR0603-RES	0603-RES	Resistor
R17	100K	RESISTOR0603-RES	0603-RES	Resistor
R18	100K	RESISTOR0603-	0603-RES	Resistor

		RES		
SV1		FE10-1	FE10	FEMALE HEADER
SW_BUCKLE		TAC_SWITCHPTH	TACTILE-PTH	Momentary Switch
SW_CREATE		TAC_SWITCHPTH	TACTILE-PTH	Momentary Switch
SW_UNBUCKLE		TAC_SWITCHPTH	TACTILE-PTH	Momentary Switch
U$1	ANT-418-HE	ANT-418-HE	TH	Antenna
XCVR	TXM-XXX-LR	TXM-XXX-LR	TXM-XXX-LR	Linx Techologies' LR Series Transmitter

Car Board BOM (Bill-of-Materials)

Part	Value	Device	Package	Description
9S12DP512	9S12DP256C	9S12DP256C	SQFP-S-20X20-112	Freescale Microprocessor
ANT	ANT-418-HE	ANT-418-HE	TH	Linx 418MHz Antenna
BP_VDD1	100nF	CAP0805	805	Capacitor
BP_VDD2	100nF	CAP0805	805	Capacitor
BP_VDDA	.1uF	CAP0805	805	Capacitor
BP_VDDPLL	100nF	CAP0805	805	Capacitor
BP_VDDR	10uF	CAP0805	805	Capacitor
BP_VDDX	10uF	CAP0805	805	Capacitor
BP_VRH	100nF	CAP0805	805	Capacitor
CRYSTAL	16MHz	CRYSTALHC49US	HC49US	Crystals
C_CAN_3	100nF	CAP0805	805	Capacitor
C_DEC	.1uF	CAP0805	805	Capacitor
C_OSC_1	33pF	CAP1206	1206	Capacitor
C_OSC_2	33pF	CAP1206	1206	Capacitor
C_RXM	10uF	CAP0805	805	Capacitor
C_SWCAN	100nF	CAP0805	805	Capacitor
C_SWCANH	100pF	CAP0805	805	Capacitor
C_SWGND	100pF	CAP0805	805	Capacitor
C_VBP1	100uF	CAP_POLC	PANASONIC_C	Capacitor Polarized
C_VBP2	.1uF	CAP0805	805	Capacitor
C_XFC_P	.22nF	CAP1206	1206	Capacitor
C_XFC_S	2.2nF	CAP1206	1206	Capacitor
DEBUG_PORT	BDM	JP3Q	JP3Q	JUMPER
D_CANH	1N4148	DIODESOD	SOD-323	Diode
D_ISO9141	1N4148	DIODESOD	SOD-323	Diode
D_J1850	1N4148	DIODESOD	SOD-323	Diode
D_PDN	1N4148	DIODESOD	SOD-323	Diode

IC4	LM7805	V_REG_78XXSIDE	78XXL	Voltage Regulator
IRL510	IRF520	IRF520	TO220BV	HEXFET Power MOSFET
I_SW	47uH	INDUCTOR1206H*	INDUCTOR-1206	Inductors
JP1	2TERM	M023.5MM	SCREWTERMINAL-3.5MM-2	Header 2
JP2	M05X2PTH	M05X2PTH	AVR_ICSP	Header 5x2
JP3	STAND-OFF	STAND-OFF	STAND-OFF	Stand Off
JP4	STAND-OFF	STAND-OFF	STAND-OFF	Stand Off
JP5	STAND-OFF	STAND-OFF	STAND-OFF	Stand Off
JP6	STAND-OFF	STAND-OFF	STAND-OFF	Stand Off
JP8	M05X2PTH	M05X2PTH	AVR_ICSP	Header 5x2
LM-418-DEC	LICAL-DEC	LICAL-DEC	TSSOP-20	Linx MS Series Decoder
MCP2551	MCP2551	MCP2551	SO08	High-Speed CAN XCVR
MODE_LED	LED_1206	LED1206	LED-1206	LEDs
NCV7356	NCV7356-SOIC8	NCV7356-SOIC8	SOO8	Low Speed CAN XCVR
POWER_LED	LED_1206	LED1206	LED-1206	LEDs
RS232	SCI0	DB9	DB9	DB9 Connector
RS232_C1	.1uF	CAP0805	805	Capacitor
RS232_C2	.1uF	CAP0805	805	Capacitor
RS232_C3	.1uF	CAP0805	805	Capacitor
RS232_C4	.1uF	CAP0805	805	Capacitor
RXM418	RXM-XXX-LR	RXM-XXX-LR	RXM-XXX-LR	
R_BDM	1.5K	RESISTOR	805	Resistor
R_CANH	1K	RESISTOR1206	1206	Resistor
R_IRQ	10K	RESISTOR	805	Resistor
R_ISO9141	1K	RESISTOR1206	1206	Resistor
R_J1850	1K	RESISTOR1206	1206	Resistor
R_LEARN	100K	RESISTOR	805	Resistor
R_OSC	1M	RESISTOR	805	Resistor
R_POWER_LED	180	RESISTOR1206	1206	Resistor
R_RESET	10K	RESISTOR	805	Resistor
R_RXM	330	RESISTOR1206	1206	Resistor
R_SW	1K	RESISTOR1206	1206	Resistor
R_SWCAN_LOAD	6.49k	RESISTOR1206	1206	Resistor
R_SW_RXD	2.7k	RESISTOR1206	1206	Resistor
R_VREGEN	10K	RESISTOR	805	Resistor
R_XFC	5.1K	RESISTOR	805	Resistor
SP232A	MAX232SOIC16	MAX232SOIC16	SO016	RS232 Transceiver
SW_LEARN	LEARN	TAC_SWITCHPTH	TACTILE-PTH	Momentary Switch
SW_RESET	RESET	TAC_SWITCHPTH	TACTILE-PTH	Momentary Switch

Appendix F: FMECA Worksheet

Figure 5: Power block

Failure No.	Failure Mode	Possible Causes	Failure Effects	Method of Detection	Criticality	Remarks
1A	Vcc > 5.0V	Failure of the 7805 linear regulator.	12V battery not stepped down to 5V. Damage to 5V components likely.	Observational: Smoke, excessive heat, no response from device	Moderate	Device unable to prevent harm or death but shows outward signs of malfunction.
2A	Vcc = 0V	Shorted bypass capacitor, failure of the 7805 by short to ground, r_power fried	No functionality	Observational: No response from device, Power_led dark	Moderate	
3A	Vcc = 5V Power_LED dark	Failure of the power led	Functionality unaffected	Observational: Device still responds with no other noticeable failures	Low	Device no longer provides visual indication of operation

Figure 6: CAN communications block

Failure No.	Failure Mode	Possible Causes	Failure Effects	Method of Detection	Criticality	Remarks
1B	CAN/SWCAN bus = 12V	Shorted leveling capacitor. Failed CAN/SWCAN transceiver	Communication on the CAN/SWCAN bus lost	Measurable: Zero activity on the bus for certain time duration	Low	Device loses ability to interact with host vehicle
2B	Slow transitions of bus signals	Opened inductor on the SWCAN bus	Communication on the SWCAN bus lost	Measureable: Activity on the CAN bus but not the SWCAN bus	Low	Functionality lost is dependent upon what is exclusive to the SWCAN bus for that particular vehicle

Figure 7: Wireless block

Failure No.	Failure Mode	Possible Causes	Failure Effects	Method of Detection	Criticality	Remarks
1C	Child side event not detected	Failure of the wireless transceiver or decoder module	False negatives	None	High	No known detection method without modification of the child side design

Figure 8: Audio block

Failure No.	Failure Mode	Possible Causes	Failure Effects	Method of Detection	Criticality	Remarks
1D	V_{DS} = 0V always	Failure to short of the IRL510 power MOSFET	Speaker constantly on.	Observational: User dissatisfaction. Drain of car battery	Low	
2D	VDS = 12V	Failure to open of the IRL510 power MOSFET	Lost ability to audibly warn the user	Observational: No audible interaction or warning from device	Low	Low criticality so long as CAN functionality still exists

Figure 9: Microcontroller block

Failure No.	Failure Mode	Possible Causes	Failure Effects	Method of Detection	Criticality	Remarks
1E	No Audio output/ constant audio output	Microcontroller port pin is suck high or low	Speaker constantly on or off.	Observational: User dissatisfaction. Drain of car battery	Low	
2E	Intermittent resetting	Noisy signal incoming on VCC/failure of the reset pin capacitor/resistor	Unknown behavior. May appear to be functioning correctly.	Observational: child side events missed under scrutiny	High	User may be unaware of the failure.