Design Team 1

Arduino Friendly PSoC Shield

Design Issues

Cecilia Acosta

Brett Donlon

Matt Durak

Aaron Thompson

Nathan Ward

Abstract

In this project our team (Design team 1), was faced with using the PSoC to interface with the Arduino Ethernet shield. However, beyond the issue of merely creating a functional design, certain design issues must also be addressed. These issues affect the long-term viability of our product which, in-turn affects the approach of this project's development, therefore it is critical to our project to properly assess and compensate for these constraints. The core issues which affect our project include managing the project lifecycle, complying with standards and regulations, and addressing our project's intellectual property.

Product Lifecycle Management

Design team 1's project, in terms of logistics, could be absorbed by Cypress with relative ease. The current form of the project utilizes the PSoC Creator development environment, PSoC First Touch kit, various Arduino shields, a PCB to connect the PSoC kit with the Arduino shields, and modified Arduino libraries that were ported over for compatibility in PSoC Creator. Design team 1 provided the hardware solution (PCB) as well as the ported libraries for using Arduino with PSoC. The code, as specified by the project sponsor, will be made available on the internet through Cypress's existing website at no cost to the user and the PCB delivery could utilize the existing Cypress supply chain. The hardware, much like the existing product line of Cypress could be made available on the Cypress website, as well as by the many licensed vendors who provide Cypress development kits.

The product usage would require little extra documentation and effort on the part of Cypress as well. The Arduino community is widespread and all of the Arduino boards, shields, and projects are well documented. There are some minor differences in the ported Arduino code which could be documented for the end users in order to make existing Arduino code work. For any new Arduino shield or board that comes into the market, the new libraries could be ported over utilizing the application note provided by design team 1. The ability to program PSoC Creator also adds to ease of use and adaptability for users. Cypress also has a large amount of application notes and documentation on using PSoC on their website. The maintenance required for the Arduino Friendly PSoC Shield is minimal. The hardware would require little to no service and the little amount of service could be completed by the user. The software updates or any

additional Arduino libraries ported over to compile in PSoC Creator could simply be posted online and downloaded by the users through the existing PSoC Update software. In terms of power consumption, the project performs well. Discussed later in the discussion on standards, the project uses relatively low voltage and low power.

Due to the existing Cypress distribution and product support, as well as the ever-growing and thriving Arduino community, design team 1's project can be marketed and distributed by Cypress without extensive additions and re-working of the company's logistics. The need to create PSoC Creator reusable component libraries for all Arduino shields exists. Due to the fact that there is no standard for the use of the pins on the Arduino for different shields, schematic layout in PSoC Creator is different for each shield. Pre-defined component libraries for each shield could be developed which would greatly improve user experience with the Arduino Friendly PSoC shield. This is being done with both the Ethernet and Motor Shields by design team 1.

As technology evolves existing technology has to be updated to be compatible with newer technology and continue to be relevant. Obsolete technology or electronic waste can be recycled, refurbished or dismantled to be used in newer technology. Software can be simpler to maintain compared to hardware, because there are no physical components or constraints such as space and weight. Software can be updated and patched to fix bugs in order to continue to be relevant and secure. Recycling obsolete hardware presents issues that have to be considered such as toxic contents and data security.

This project is mostly software oriented and as technology evolves it will only have to be updated. Over time, the software symbol that we created in PSoC Creator will have to be updated in conjunction with updates that Arduino may make to its shields. Arduino upgrading and reconfiguring its shield will pose a unique problem seeing as the software developed in this project is a Cypress sponsored product. Cypress will need to be sure to follow updates of the Arduino code and patch its own libraries. Updates to PSoC Creator can have positive and negative effect on our software. Improvements could include the use of other programming languages to preform logic operations within the virtual component; this advantage is currently included in newer versions of PSoC Programmer. It is also possible that later versions will include better C++ support, which will improve the software used in this project. Disadvantages

would include changes to the complier which would ultimately change the way we integrate Arduino's C++ source code into PSoC's C libraries. Hardware changes to the Arduino boards will also force this team to change the PCB layout which is configured to fit with Arduino's 28-pin shields.

As technology evolves toward faster and smaller microprocessors, our design for the PSoC may become irrelevant. Once our current components become incompatible the software and the PCB will have to be disposed of efficiently and responsibly. The software can be recycled to be used as algorithms for newer systems. The hardware such as the PCB and PSoC 5 development board can be recycled into newer hardware components. The pins on the PSoC which are metal can be remade into a multitude of products. Solder used on the boards has to be taken into consideration when reusing the metals. It can be hazardous to the environment due to its lead content. The boards which are silicon can be disposed of without causing damage to the environment. The Arduino shield can pose a security vulnerability when disposing of it along with the SD card. The SD card could hold valuable information that some may feel would pose a risk to their security and personal information.

Design Standards

Following design standards is a crucially important constraint to any design. For any type of application there are certain hard limits on various aspects of electronic operation. These standards can range from specifying the amount of electromagnetic interference emitted, the materials used, even energy efficiency. Just as it is critically important for all engineering designs to follow certain standards, our project must follow standards as well, both for the sake of Cypress and for our customer base in general. Standards are in place in order to protect designers, manufacturers, and consumers. Some of the most important standards concern issues involving operator safety, such as IEC 60038, which quantitatively defines what constitutes high voltage. Although standards like these are important, not every standard is applicable to our project, in order to effectively utilize the guidelines set by organizations such as the IEC it is critical to assess the standards which would affect our design.

A large portion of this project involves not the creation of physical hardware, but rather the utilization of software-based solutions. The physical components which have been designed in

this project include a PCB which connects the PSoC to the Arduino Ethernet shield and a simple packaging solution. Beyond this, any hardware component that goes beyond the scope of a traditional microcontroller has been programmed as a virtual component in the PSoC environment. By keeping our custom-made physical components to a minimum, we avoid facing issues with electronic standardization as the crucial design issues facing software relate more to addressing intellectual property, which are discussed in the following section. Concerning the physical components, the PCB and the packaging, many standards are already known to have been fulfilled. For instance, factors such as the materials being used in these components have already been analyzed for standard compliance by their respective manufacturers (such as DipTrace). Additionally many specifications can be evaluated by the IPC 2221, which lists a multitude of standards relating to topics as varied as through-hole spacing, adhesive material specification, and on-board dielectric characterization. Many of the standards of the IPC 2221 which relate to board layout are automatically verified by DFM (Design for Manufacturability) checks, such as conductor or ground plate spacing. Beyond this, many issues such as standards in voltage application are not applicable since our project utilized relatively low voltages. Additional standards and regulations, related to electromagnetic interference and environmental issues are addressed later in this paper.

Environmental Issues

One primary concern for this project is FCC (Federal Communication Commission) compliance issues. Testing this product to find out whether it is FCC compliant or not was outside of the scope of this senior design course. This is due to needing special facilities or hundreds of dollars to properly do so. However, it is known that the Arduino Ethernet Shield and the PSoC starter kit by themselves are FCC compliant; the products could not be in the market if they were not. The problem is that FCC requires that the final design has to be tested with all its components together. For this testing to be done properly a certified FCC facility testing site is required.

This brings about a second issue. As this is a project sponsored by Cypress to include the PSoC in designs that Arduino products could accomplish on their own, there could be a legal issue if the final product fails to pass FCC regulations. As mentioned above, if it fails the test it would be due to the final design being tested including the PSoC and the Arduino Shield together.

Arduino as a company could perceive this as Cypress creating a negative image of Arduino and could bring about a lawsuit against Cypress.

Also, in a way related to what is mentioned above, there could be compatibility issues like EMI (Electromagnetic interference). The final product was tested using the PSoC starter kit with the Arduino Ethernet shield, which, on its own functions correctly. The problems could come about if other hardware is added. The project is designed as a base project with the possibility of adding complexity by adding other components. The components to be added would include other daughter boards or shields from Arduino, other Cypress hardware or additional external components. This would make the design more complex. Therefore, it could bring about unforeseen interference and higher radiation sources making it not FCC compliant. These designs would not only prohibit putting it on the market, due to not complying with federal regulations, it could also make the project not work as expected. This is especially true when wireless components or motor components are added to the design.

Another related design issue to point out is concerning the power source of the project. The final product had the Arduino Ethernet shield draw power from the PSoC. If more components are added, and no additional power sources are included, the project would probably not work as expected if it worked at all. Although adding a power source would fix the initial problem, this will also increase the probability of EMI and decrease the chances that the project would pass FCC regulations.

To solve these issues there should be an agreement between Cypress and Arduino, if the two companies worked together, the products can properly be tested. Although, this arrangement is unlikely, since Cypress goal is to replace some of the Arduino components with their own. So as of now the issues listed above are very much a concern.

Intellectual Property

The design issue of intellectual property is primarily related to the software aspect of this design project. As mentioned in this paper, the hardware components are primarily those designed and developed by Arduino and Cypress for use by other electronics designers to develop products. The hardware for this project is a printed circuit board with the proper pin spacing, placement, and connection to interface the Cypress PSoC with the Arduino shields.

Software intellectual property and licensing has many different forms. It can generally be divided into commercial, or closed-source, and open source. Commercial software which is sold to end users typically operates under a closed source model. The source code for this software is not made available to the public. In other cases, some source code is made available as needed, such as header files which need to be included in user code, or example source code for projects. This source code is still restricted by an end user license agreement (EULA). The other end of the spectrum includes completely free and open source software. This code is available for free and may be modified or redistributed. There are several popular licenses which state how redistribution will be allowed, and whether new derivative works may be used commercially or if they must enforce the same license. Some popular licenses include the GNU General Public License (GPL), the BSD License, the MIT License, and the Apache License.

The source code used in this project is both closed and open source. The PSoC Creator application itself is a closed source integrated development environment (IDE). End users may freely download the software so long as they abide by the EULA. PSoC Creator includes configurable virtual components which each generate C source code. This source code is made available for users to look at and interface with for their application code. It is not, however, open source so it is subject to the EULA and may not be freely modified and distributed.

The Arduino libraries themselves are open source. The parts used in this project are released under the GNU Lesser General Public License. Fortunately, this is a permissive open source license in terms of interaction with proprietary software. The license allows the free copying, modification, and distribution of the source code, provided that the LGPL notice remains in the code and the derivative work is also released under LGPL. The LGPL improves upon the GPL by allowing the LGPL code to link against non-LGPL code, such as proprietary software. In this project, the Arduino libraries are modified to interface with PSoC and to use certain Cypress function calls in their implementation. This is allowable under the LGPL license so long as the LGPL license text is included and the software developed under this project is made freely available for modification and redistribution under the LGPL. It will be the responsibility of Cypress in its distribution of this code to adhere to the license. This open source software model will hopefully be more attractive to Arduino users who may wish to modify the source code and redistribute it for their own projects.