Embedded Student Development System

Jay Hechter (hechtej@wwu.edu)

WESTERN WASHINGTON UNIVERSITY
MAKE WAVES.

Abstract

The Student Development System (SDS) is a development kit designed for electrical engineering students to gain experience with embedded systems. The SDS aims to provide an approachable and reusable framework for students to complete projects in classes such as Embedded Systems (EECE444) and Digital Signal Processing (EECE433). It could even be used as the foundation for future capstone projects.

It consists of a microcontroller development board, a graphical LCD touchscreen, and a digital microphone. Students working with the kit will have access to a library written in the C programming language, allowing their code to respond to buttons being pressed on the touchscreen, output text and images to the screen, and process audio recorded by the microphone. Students will have the ability to customize the screen's layout by adding, removing, or repositioning buttons as they see fit.

Motivation

In the past, the EECE curriculum used Kinetis' K65 Tower boards for the same purposes that the SDS is intended to be used for. However, the Tower boards are no longer available for purchase and the number of students taking relevant classes already exceeds the number of Tower boards available.

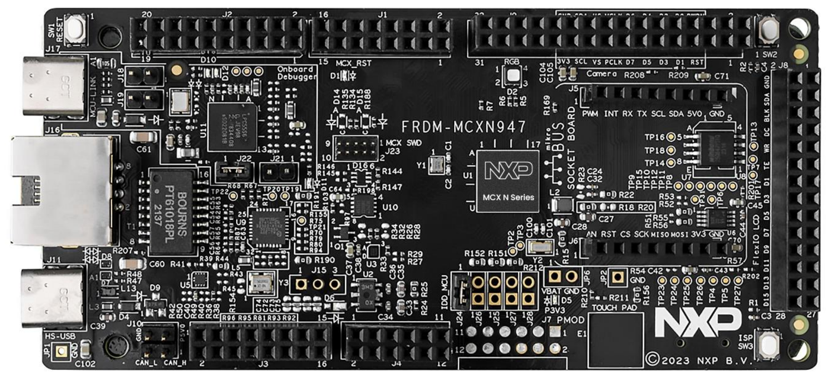
This project hopes to fulfill the need for a robust replacement to the Tower board, with many of the restraints and requirements of the project being chosen specifically to create a system that would be familiar to those who have developed on the Tower board before, and backwards compatible with many of the software libraries that have been used in the past.

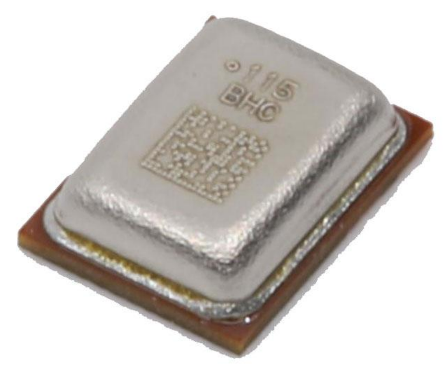
Limitations

As the SDS is intended to be a framework for future projects to be built upon, it is important that this framework be lightweight. In specific, the SDS should not use too much of the FRDM board's flash storage, RAM or runtime CPU cycles, and multiple pins of the FRDM board should be left unused for students to attach their own peripherals alongside the LCD and microphone.

Methods and Materials

The SDS has three main hardware components. The first is the FRDM-MCXN947 Development Board, which will serve as the core of the system. The second is the LCD-PAR-S035, a 3.5" 480 by 320 pixel LCD touch display. The LCD serves as the primary means of input and output for all programs created by users of the system, providing the ability to display text, graphics, and pressable virtual buttons. One of the FRDM board's modules, known as SmartDMA, is used to copy graphics to the LCD with minimal CPU usage, while I2C is used to receive touch information.





Third is the Knowles SPH0141LM4H-1 digital microphone, allowing for programs which are able to record and process audio. Another FRDM board module, known as MICFIL, will be used to record samples of audio from this microphone, and eDMA will be used to copy this audio into a user-accessible buffer, where any manner of audio processing can be performed.

As for software, all of these processes, as well as any other processes that users of the system wish to add, will be managed by the Cesium real-time operating system, also known as Cs/OS.

A library known as the Light and Versatile Graphics Library, or LVGL, will be used to simplify the process of creating the graphics to be drawn to the LCD. LVGL is able to be used to create complex graphical user interfaces, but can be quite complicated to work with. Because of this, the SDS provides users with a small subset of LVGL's capabilities, emphasizing ease of use.

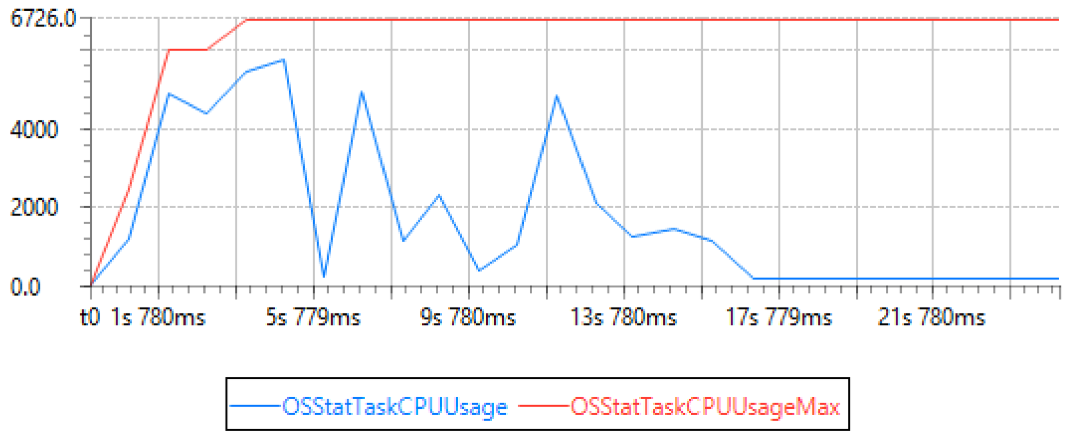
Results

The SDS has four widgets in total:

1) Buttons, which will call user-written code when pressed.
2) Labels, which user programs will be able to write to with 7-bit ASCII characters.
3) Images, with users able to import any small image of their choosing.
4) Charts, which can be used to create dynamic graphs that can be updated as the program runs.

The example program shown to the left uses 31% of the SDS' flash storage and 43% of its RAM, both of which leave plenty of resources for students to create their own projects.

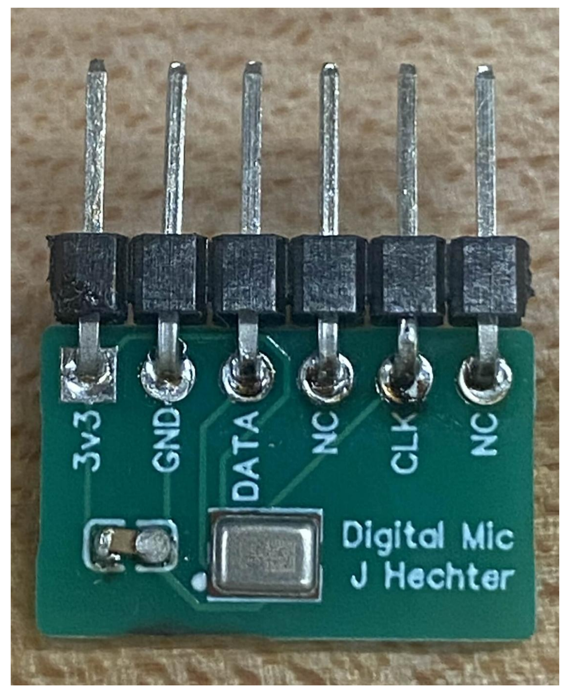
This program's CPU load over time was also analyzed. Text was redrawn multiple times per second for about 15 seconds, then the screen was left static. The highest CPU load reached was 67.26%, but the typical CPU load fluctuated between 10% and 45% during high activity, and remained at 2% when idle.



While 67% CPU load is rather high, the average CPU load throughout this process was closer to 20%. This should leave students with plenty of computational resources to implement their projects, with the caveat that they should not expect high frame rates on the LCD.

Future Work

The board containing the digital microphone has been designed and assembled, but no software has yet been written to allow for students to record audio. This could be implemented in the way described above.



In addition, the FRDM board's processor has a second CPU core which is currently unused. Offsetting some of the more intensive graphical operations of the system onto a second core would both free up flash storage and drastically reduce CPU load on the primary core, allowing for students to create more complex projects.

Acknowledgements

Todd Morton
WWU's EECE Department

WWU

Capstone Project
Electrical & Computer Engineering Program