EE/CprE/SE 491 - sdmay19-31:
Multi-Purpose Automated Robotic Mixer (mpARM)
Week 12 Report
January 26 – February 1
Client: Alexander Stoytchev/Brett Altena
Faculty Advisor: Alexander Stoytchev

Team Members

Drew Caneff — *3D Printer Specialist/CAD Designer/Accountant*
Amos Hunter — *Electromechanical Specialist/Meeting Scribe*
Brett Altena — *Meeting Facilitator/ Computer Vision Developer*
Kristian Wadolowski — *Report manager/Front-end programmer/Computer vision Developer*
Jase Grant — *Embedded Systems/ Assignment Manager*

Summary of Progress this Report

- **PCB-** I added some female connectors to integrate with the male headers of the stepper motor drivers, because from my experience these components tend to burnout the most on a control board. Instead of having to replace the whole PCB which would cost $60 and a couple weeks to order parts I figured we would just buy extra drivers which cost $5 and replace them as needed should the need arise. This method allows for drivers to be replaced without any soldering.
- **Thor arm (Main bearing) -** After determining that it might take a week for the last ordered component for the THOR arm I decided to find a work around. Fortunately, since the bearing is strictly made in China and can be difficult to obtain several Makers have made similar bearings which can be 3D printed. While this will not perform one to one as the steel bearing, I figured I'd print one just in case we can no longer afford to wait on the part to arrive. Keep in mind that 3D printed bearing will be made from plastic not metal so while it might be lighter it will most likely not perform as smooth of an action as the metal one. Keep in mind this is only being used as a backup at the moment.
- **FPGA -** Coded and i set up most of the pipeline for the FPGA for the cpu processes and also set up some of the GPIO ports for outputs to test the FPGA set-up
  Ran into issues when trying to compile the FPGA set up. Some of the clocks are messed up and are sending signals wrong.
  Trying to get a full version of the program for the FPGA set up so that i can use all the testing functions.
  Set up some of the interfaces for the interface for the FPGA.
  Some issues with the cameras version and I am trying to find a way around it because the version is newer and is harder to interface and hack
  Have some issues setting up the project for the FPGA code at first because of the new board.
  might need to borrow a J-Tag for the FPGA from the school.
  Haven't gotten to programming the FPGA yet trying to get the code to compile and to code it a bit.
- **Test circuit -** Based on input from other team members in a meeting, the need for a user interface and/or testing circuit was demonstrated. From this information, I volunteered to provide this. I therefore designed such a circuit. The purpose of which is to provide a voltage output from each of four switches. To implement this, I start with a DC power supply from a switched-mode power supply such as are used for charging smartphones. I then divide the electricity across four switches in parallel. Each switch is debounced with shunt capacitors to ground. An indicator LED is in series with each switch to verify to the user that the circuit is completed. The FPGA is the final element which completes the circuit
- **Motion detection -** The new approach is analyzing the pancake's readiness by using motion detection to

accurately count how many bubbles have popped/ formed. The number of bubbles needed for the recipe chosen is 35. Therefore, the program must be able to recognize that the surface of the pancake has been displaced 35 different times. The program will work by taking in the video feed and dissecting the stream frame by frame and determining whether the each pixel is similar enough to the last received frame. A counter will be kept track of to determine how many displacements have occurred, and once the set number has been reached the program will signal the robotic arm to flip the pancake. The conceptual logic of the program is thought through, but the logistics of developing the program is incomplete.

Pending Issues

- Batter issues (Splashing, feeding, mixing, type, etc...)
- Complete Computer vision code
- Assemble the arm

Individual Contributions

Team Member	Contribution	Weekly Hours	Total Hours
Drew Caneff	➢ Completed main PCB, and started backup PCB ➢ Send off a request for additional parts ➢ 3D printed backup bearing	13	169
Amos Hunter	➢ Evaluated need for frame materials ➢ Developed cutting and drilling plans for frame ➢ Designed a user interface circuit ➢ Began work on test board	11	128.5
Brett Altena	➢ Reviewed computer vision progress ➢	7	139
Kristian Wadolowski	➢ Compiled and edited reports ➢ Continued research on Arduino Code	7	95
Jase Grant	➢ Worked on FPGA code ➢ Researched ports and integration ➢ Looked into accelerating FPGA pipeline	30	67

Plans for Upcoming Reporting Period

Team Member	Plans
Drew Caneff	➢ Start Thor arm construction ➢ Make a platform for the PCB
Amos Hunter	➢ Purchase materials ➢ Complete test circuit board ➢ Begin building frame
Brett Altena	➢ Research motion detection program
Kristian Wadolowski	➢ Continue to analyze the arduino code ➢ Look into 3rd party Thor arm

	documentation
Jase Grant	➢ Work on code and pipeline ➢ Start integrating camera ➢ Compile code and start FPGA C code

Gitlab Activity Summary

Action: joined, Tue Sep 04 2018
Author: dvcaneff
