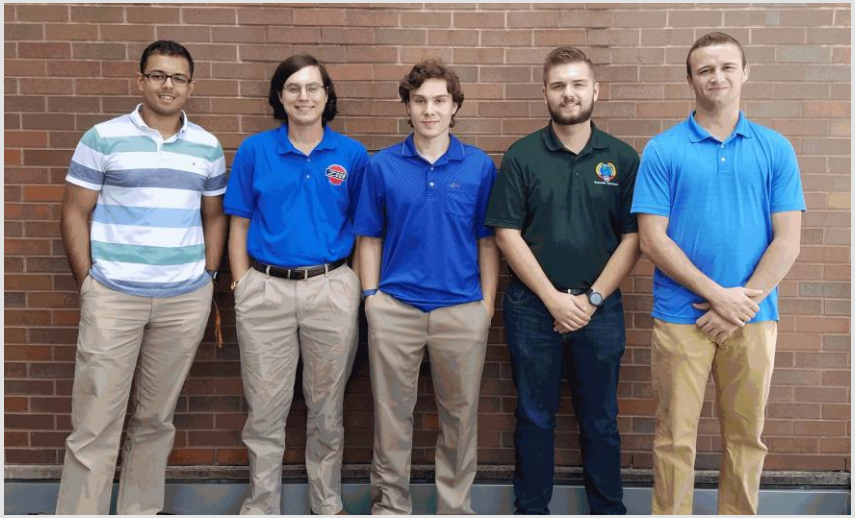
MSD Team 20225: The Doc B Racing Real-Time Telemetry System

The Telemetry Bois

Adam Bork, Sahil Gogna, Josh Henderson,
Michael Schroeder, Nick Washco

Introductions:



Member	Role	Contact
Nick Washco	EE Lead	njw6366@rit.edu
Josh Henderson	Facilitator	jh2290@rit.edu
Sahil Gogna	Project Manager	sxg4060@rit.edu
Michael Schroeder	CE Lead	mjs5127@g.rit.edu
Adam Bork	Team Liason	acb6075@rit.edu

Agenda:

- Team Vision for Subsystem Build and Test Phase
- SD Write
- Queue Management
- Board Changes
- Test Plans
- Risk Assessments Review
- Plans for the Next Phase
- Q&A

Team Vision for Subsystem Build and Test Phase

- Continue debugging of existing SD card code .
- Work towards a solution that allows data to be saved and read from the SD card.
- Complete installation of accelerometer with SM resistors and capacitors as well as USB port.
- Start hardware testing of accelerometer installed on PCB board. (work still needs to be done)
- Complete purchasing of necessary parts for project including portable DC power supply so work can be continued outside the lab.
- Continue updating Risk Assessment with changes that have taken place in the project.
- Develop a working 3D design for new enclosure.
- Continue to update edge page with current test plans and results.

SD Test Read/Write Code

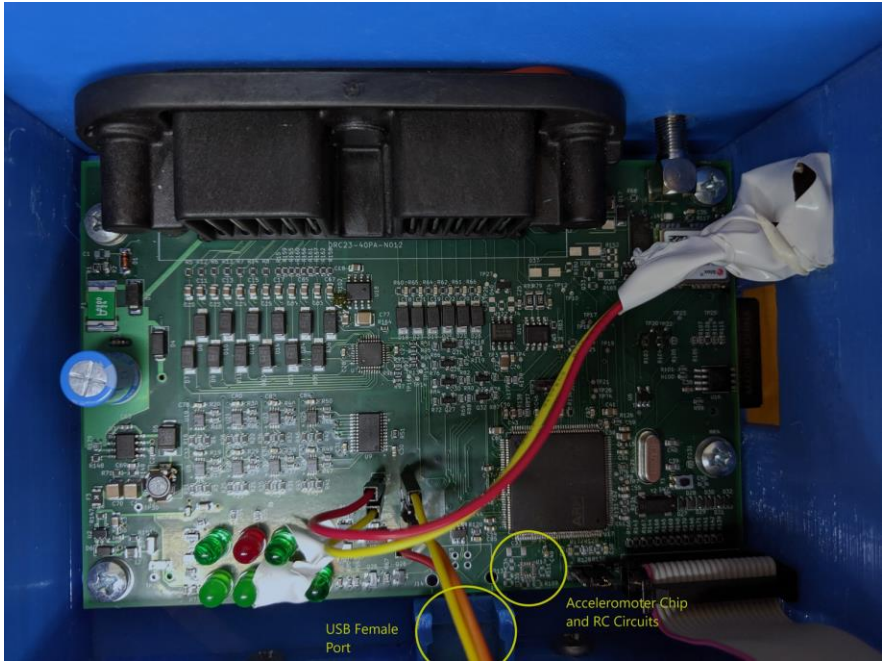
```
void SDFileTest(void)
{
  FIL MyFile;
  FRESULT res;
  /* FatFs function common result code */
  uint32_t byteswritten, bytesread;   /* File write/read counts */
  uint8_t wtext[] = "Hello, STM32F4 FatFs! Hey it works!!!"; /* File write buffer */
  uint8_t rtext[100];  /* File read buffer */
  //printf('SD File create\n');
  // Create file test
  if (f_open(&MyFile, "STMTXT", FA_CREATE_ALWAYS | FA_WRITE) != FR_OK)
    Error_Handler();
  res = f_write(&MyFile, wtext, sizeof(wtext), (void *)&byteswritten);
  if ((byteswritten == 0) || (res != FR_OK))
    Error_Handler();   f_close(&MyFile);
  //printf('done!\n');
  //printf('SD File open\n');
  // Read file test
  if (f_open(&MyFile, "STMTXT", FA_READ) != FR_OK)
    Error_Handler();
  res = f_read(&MyFile, rtext, sizeof(rtext), (UINT*)&bytesread);
  if ((bytesread == 0) || (res != FR_OK))
    Error_Handler();
  f_close(&MyFile);
  if ((bytesread != byteswritten))
    Error_Handler();
  //printf('done!\n');
}
```

Testing: Queue Management

```
void WriteQueues(uint8_t* buffer1, uint8_t* buffer2, unit8_t* sensorBuff)
{
    int count = 0;
    switch (buf_choice)
    {
        case 1:

            for (count = 0; count < SD_CARD_BUFFER_SIZE; count++)
            {
                // Copy byte-by-byte
                memcpy(buffer1[count], sensorBuff[count], sizeof(uint8_t));
            }

            // DMA Write to SD Card
            SDWrite(buffer1);
            buf_choice = 2;

            break;

        case 2:

            for (count = 0; count < SD_CARD_BUFFER_SIZE; count++)
            {
                // Copy byte-by-byte
                memcpy(buffer2[count], sensorBuff[count], sizeof(uint8_t));
            }

            // DMA Write to SD Card
            SDWrite(buffer2);
            buf_choice = 1;

            break;

        default:
                break;
    }
}
```

Board Changes with SMT Lab

- Accelerometer and USB Port Added to PCB



Before:

After:

Test Plans

- **Purpose:**
  - The purpose of the testing plan this phase was to implement and test viable SD Card Read/Write methods as well as debug logical analyzer ports on the microcontroller.
- **Instructions:**
  - A meeting with Dr. Carlos Barrios from the Electrical Engineering Department assisted us in coming up with more ways of implementing the SD Card Read/Write method via analyzing logical analyzers on the board.
  - Hardware schematics were analyzed to discover where the logical analyzers were
  - Understand more of the code-base implementation

Test Plans Documentation

P20225 Test Plan				
IO Tests				
Name	Description	Purpose	Target Completion	Personnel
SD Test	Save text file to SD Card	Test SD save function	2/13	Mike, Sahil
Serial Test	Communicate with PC via terminal	Test USB functionality	2/13	Mike, Sahil
Config File	Create Config and pass to board	Test system adaptabilty	2/13	Mike, Sahil
Hardware Tests				
Name	Description	Purpose	Target Completion	Personnel
Accelerometer	Pass voltages to accel to trace output	Test function and validity of accelerometer	2/13	Josh, Nick, Adam

Risk Assessment

Risk Assessment						
Risk Type	**Risk Item**	**Effect**	**Likelihood**	**Severity**	**Importance**	**Action to Minimize Risk**
Electrical	Short board in testing/use	Board is unusable	2	3	6	Practice safe testing, be careful of exposure
Electrical	Sensor damaged/ does not work	Data from sensor not able to be read	1	2	2	Ensure sensors are set up correctly and test functionality
Electrical	Sensor not calibrated correctly	Invalid/incorrect data	2	1	2	Verify calibration procedure is closely followed
Electrical	Wiring Harness/Connections fail	Sensor data lost	1	2	2	Practice careful wiring procedures
Electrical	Electric Shock	Harm to team member, component damage	1	2	2	Install properly and verify team members take precaution
~~Electrical~~	~~Adding new Accelorometer and USB port~~	~~Possibly board damage, further rework~~	~~1~~	~~2~~	~~2~~	~~Verify understand circuit before installing, work with SMT lab personel~~
Electrical	Digital and Analog Sensors don't integrate to embedded system	Full data set not able to be recorded	1	2	2	Test all sensors on system
Mechanical	Enclosure not Secure	Potential Damage to Enclosure/Car	1	3	3	Follow instalation instruction
Mechanical	Enclosure too close to Engine Bay	Components on board damaged due to heat	1	2	2	Instalation of enclousre in car far enough away from engine bay
Mechanical	3D print not finished completley	Time and Monetary loss, not able to implement new enclosure design	2	2	4	Verify 3D model sent is correctly formatted
Mechanical	Enclosure exceeds temperature value designed to withstand	Sensor Failure/Board damage	2	3	6	Add heat wrap to ensure enclosure is able to withstand most extreme temperatures
Planning	Not enough time to fix code base and to save data	Product not complete	1	2	2	Work on schedule, don't fall behind
Planning	No access to track to test device	Unsure of how system behaves in a real test environment	2	2	4	Look for oppurtunities to test, test product in own cars prior to at the track
Planning	Not having Demo Ready for Imagine RIT	Not able to meet requirment for MSD 2	1	3	3	Stay on task and work according to schedule
Software	Bugs in code	Code doesn't perform as expected	2	2	4	Verify functionality in simulation
~~Software~~	~~SD Card funtionality not working~~	~~SD Card data not able to be saved/read~~	~~2~~	~~3~~	~~6~~	~~Finish code and work out bugs/ test functionality~~
Software	SD card not reading correct data	Incorrect sensor data being read, corrupts data	2	2	4	Verify Correct code is being added to SD card functionality
Software	Embedded data not compatible	incorrect data being anaylzed	2	1	2	Verify proper integrations bewtween system componets
Software	Not able to fix bottlenecking in code	Unable to read/write data	2	3	6	Work with Subject Matter experts, ensure testing time/environment is available for code

- Removed risks for SM parts installed on board
- Removed risk for SD card functionality not complete
- Added risk for SD reading corrupted data

Plans for Next Phase: Integrated System Build & Test Phase

- Link SD Card Read and Write Functionality with Sensor Information, to collect meaningful data.
- Complete the code for the accelerometer self test and complete the test to check for preciseness.
- Continue working on redesign of enclosure.
- Continue to work on modification of the wire harness.
- Once sensor data can be saved to SD card with meaningful data, test with multiple sensors.
- Continue to update Test Plans and Risk Assessments.
- Create a program to generate config files and map sensor data to a map.

Thank You from The Telemetry Bois

Questions?