Data Logging

Harjit Singh

Problem Statement

- When the mouse is running, and it does something unexpected, I've been left guessing "what just happened?"

- If you are really good, you can tell what happened.
- Or if you are really lucky, you have a reproducible situation.

- My strategy is to store what the mouse saw/did and then from that determine the issue.

Examples

- At practice, during APEC 2014, the maze had a 15 cell straight followed by a 180 degree turn.
  - The mouse would navigate down the straight but half way through the turn, do a “spin out”.
  - I had logging turned on and was able to see that the rotational position went from positive to negative!
  - The servo tried to track it but then couldn’t and shut down the mouse as “must be crashed”.
  - The sign change was caused by an overflow in the turn profiler.
  - The logging enabled me to see what happened without trying to reproduce it.
    - I was able to focus on implementing a fix, rather than spending time to instrument and reproduce the issue.

Traditional Approach

- Select a few variables and setup triggers to enable logging in certain situations.
  - The log data is stored in RAM.
  - This requires that each time logging is to be done, the user has to re-setup the logging code.
  - If the processor crashes, the log is lost.

- If something else of interest happens and you weren't logging it, you are out of luck.

Proposed Approach

- Log
  - **All** the time.
  - Log to flash – it is non-volatile.

Logging to RAM VS Flash

- RAM logging is “easy”:
  - Declare an array.
  - Write data to it.
  - Write speed is multi-megabytes a second – not that you have enough RAM to do this.
  - When it is time to transfer it to the host, read the array and write it out over the serial port.

- Flash logging requires:
  - That you copy the data from a buffer to the flash device.
  - With proper part selection, the write speed can be 500KB/s.
  - You implement a simple file system.

Flash Device Options

- Serial Flash
  - There are parts that can program 256 bytes in 0.3 ms or 512 bytes in 0.4 ms.
  - 16 MB and 32 MB parts available in 6 mm X 8 mm package.
  - Erasing full device takes 170 seconds.
- FRAM
  - Neat write feature – send the write command and address, **once**, and then stream data!
  - No Erase required.
  - Largest size I've seen is 256 KB.
- Internal Flash
  - Flash needs to be multi-bank so that you can execute code from part of it while you are writing to a different part.
  - Check that it is fast enough.
  - Check endurance - typically 10K write cycles, which is enough.
  - Size depends on the part you are using.

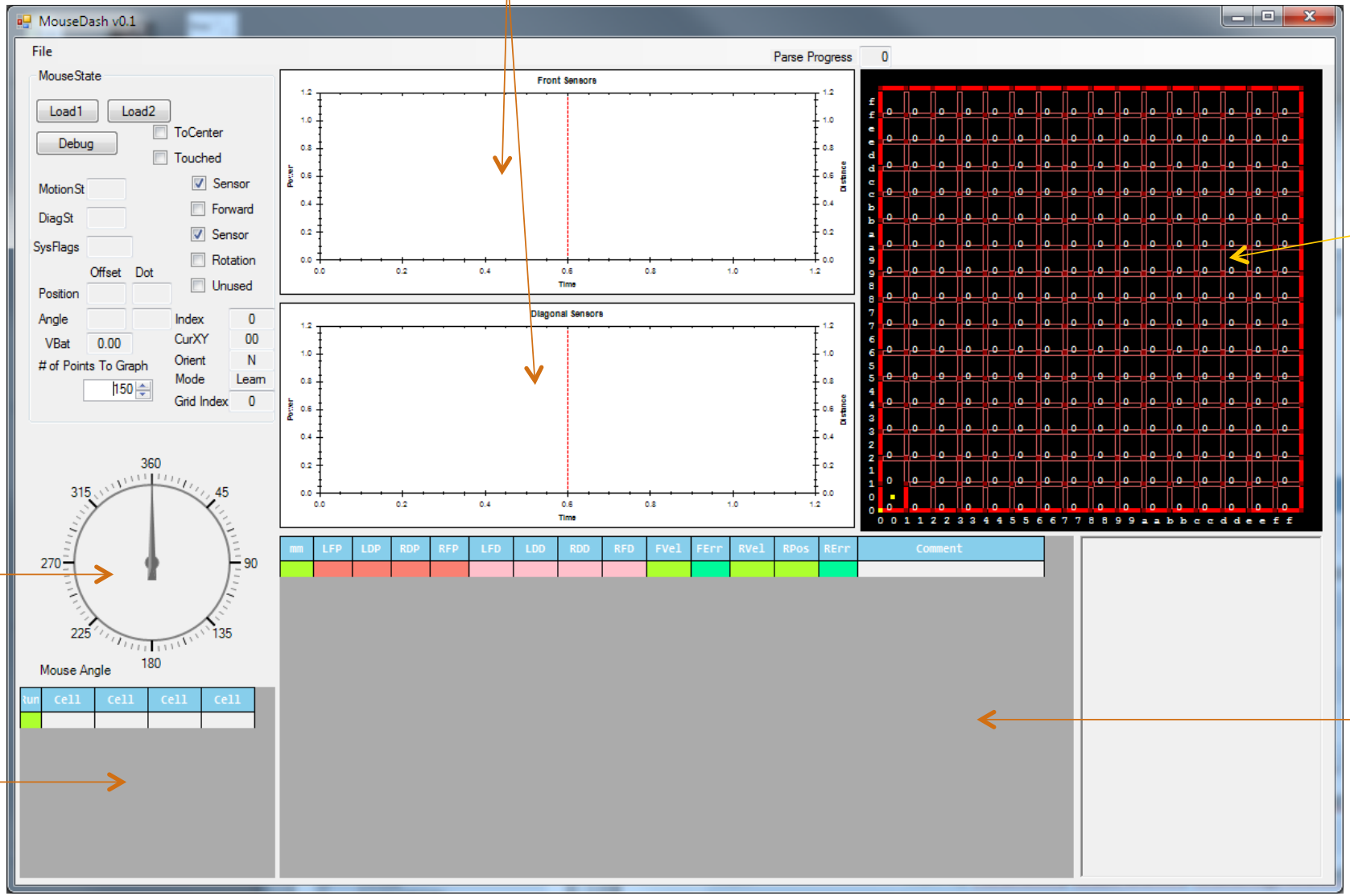
Flash Information

- For the APEC 2014 contest, the log file size was 2.7MB.
- I use Micron N25Q128 which is a 16MB flash which can program 256 bytes in 0.5 ms.
- In the future, I might use the Spansion S25FL256S device because it has a lower erase time.

What To Log?

Command	Description / Payload
PrintString	Log the NULL terminated string.
Error	Error code. For example: Battery voltage is too low.
MarkedWall	Which walls were marked.
CellLocation	Current cell coordinate.
Orient	Current orientation.
Path	Path the mouse is attempting to execute.
ProfServo	FwdPosErr, DesFwdVel, DesRotPos, RotPosErr, Battery Voltage.
Sensor	Left Front Sensor, Right Front Sensor, Left Diag Sensor, Right Diag Sensor.
Corrector	Lateral and longitudinal corrector values.
Run	Run #, FwdAccel, MaxFwdVel, MaxDiagFwdVel, Run mode.
RunInfo	Run success/fail and run time.

Sensor or Motion Velocity & Error graphs

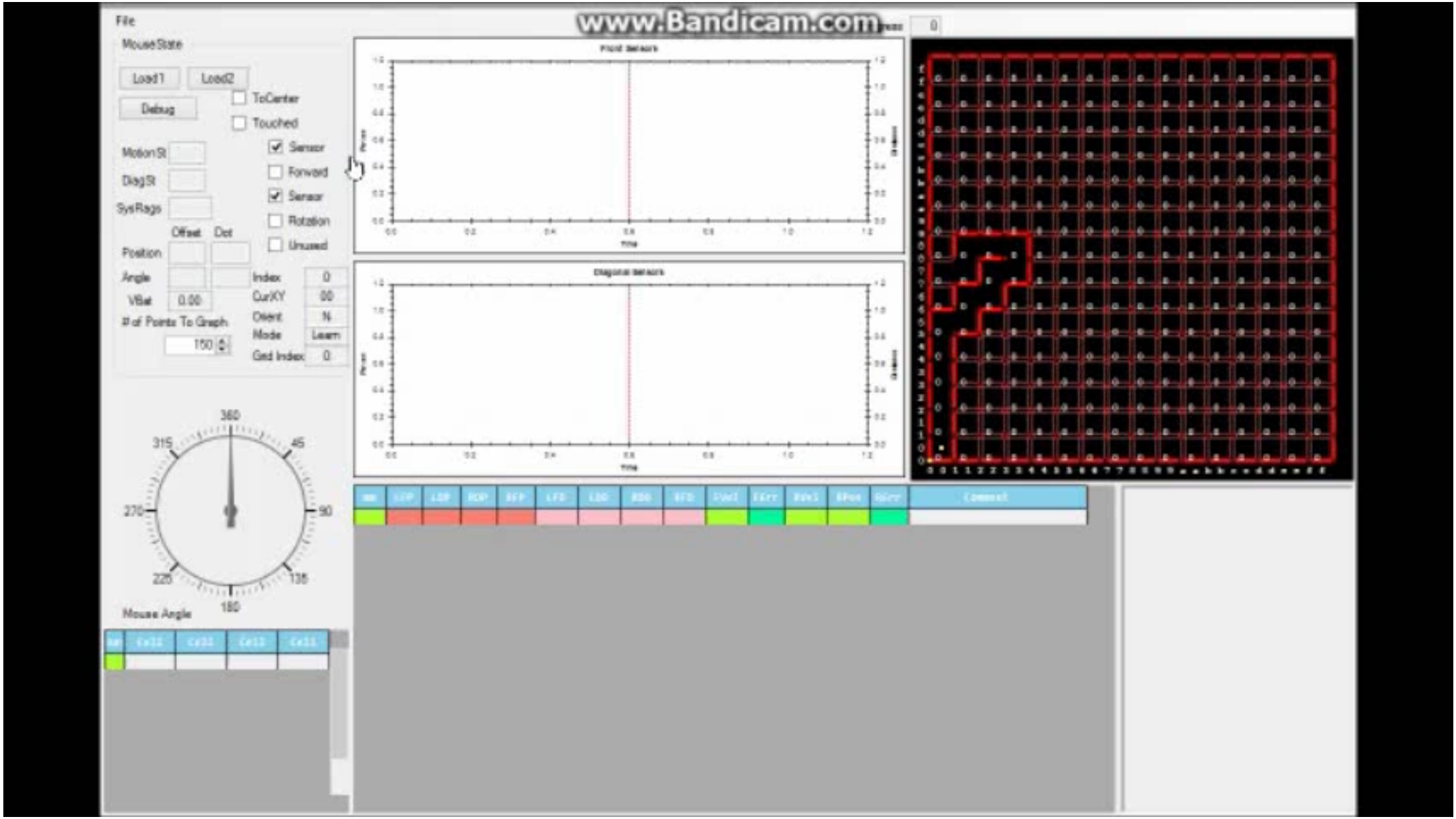


Current maz
information

rrent Heading

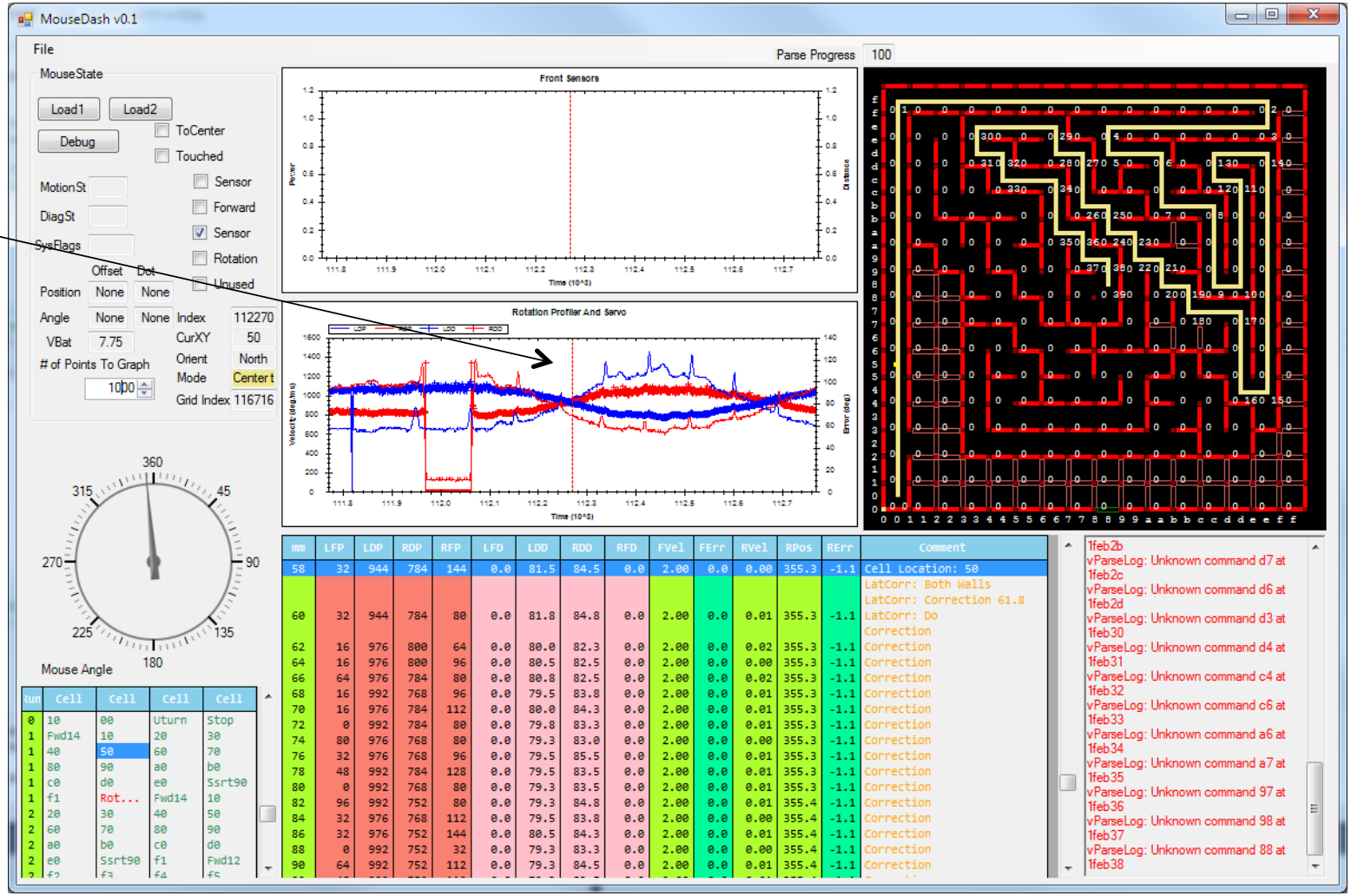
Motion & Se
numerical
information

Cell selector


File
www.Bandicam.com
MouseState
Load1
Load2
Debug
ToCenter
Touched
Sensor
Forward
Sensor
Rotation
Unused
MotionSt
DiagSt
SysFlags
Offset
Dist
Position
Angle
VBat
0.00
# of Points To Graph
150
Index
CurXY
Orient
Mode
Learn
Grid Index
Front Sensors
Diagonal Sensors
Time
Comment
360
45
90
135
180
225
270
315
Mouse Angle

APEC 2014: First Speed Run

Mouse weaving down the maze.



APEC 2014: Second Speed Run

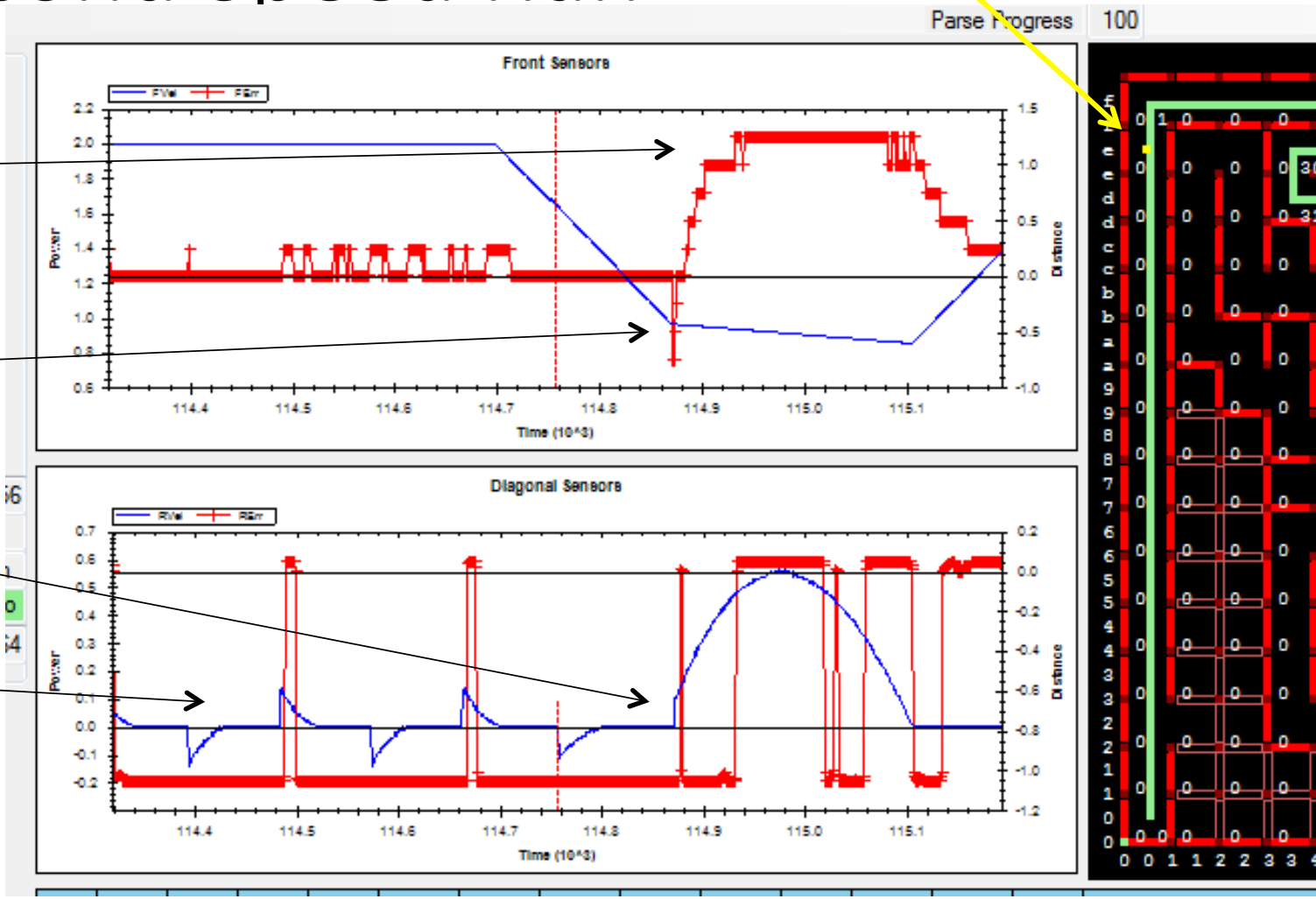
Mouse is here.

Forward Servo Error is 1.25 mm.

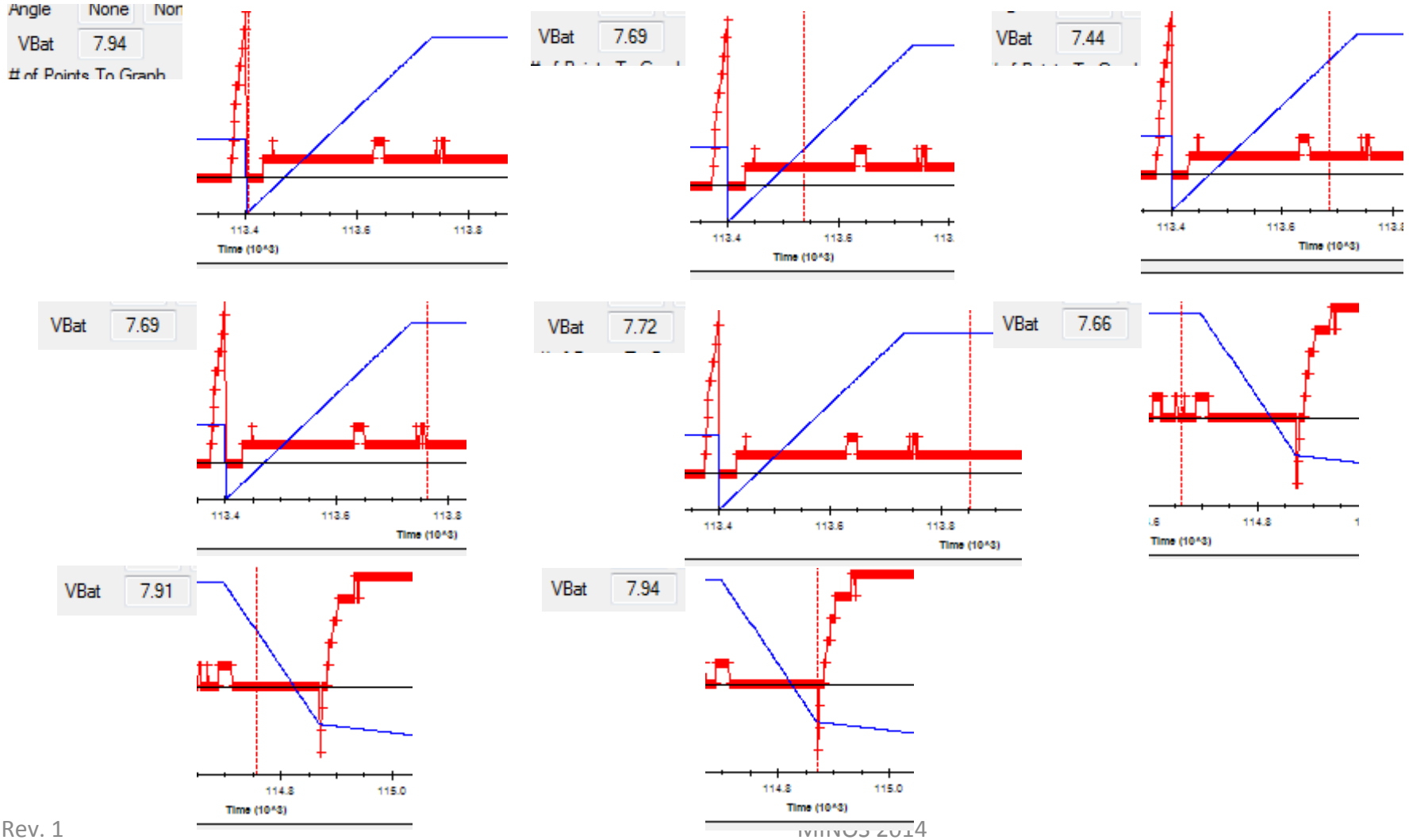
Decelerating through the turn.

Turn started LATE!

Still Weaving

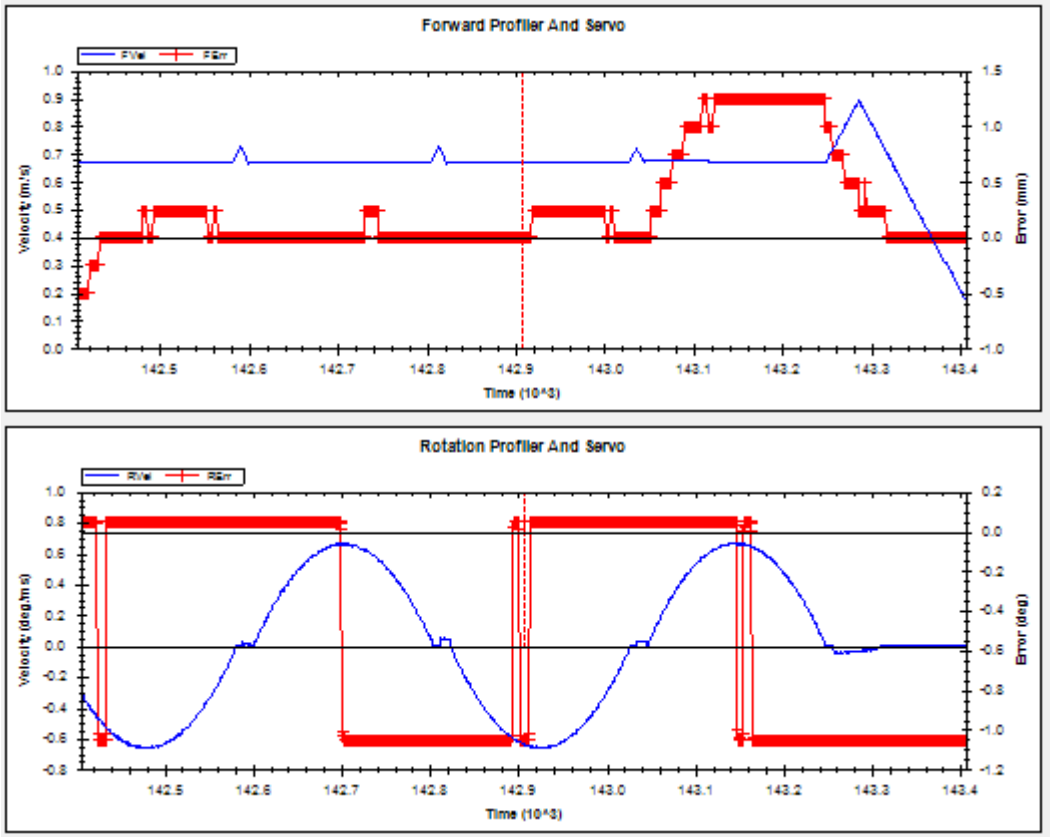


Battery Voltage During Motion



APEC 2014: Oddities

Notice acceleration blips between turns.



Mouse was decelerating, then it decided to turn and for some reason the deceleration is insane. And a little bit later, during the turn, it accelerates!