Brian Mmari & Christian Owusu
ECE 302
11/2/2023

Navigation Lab Report

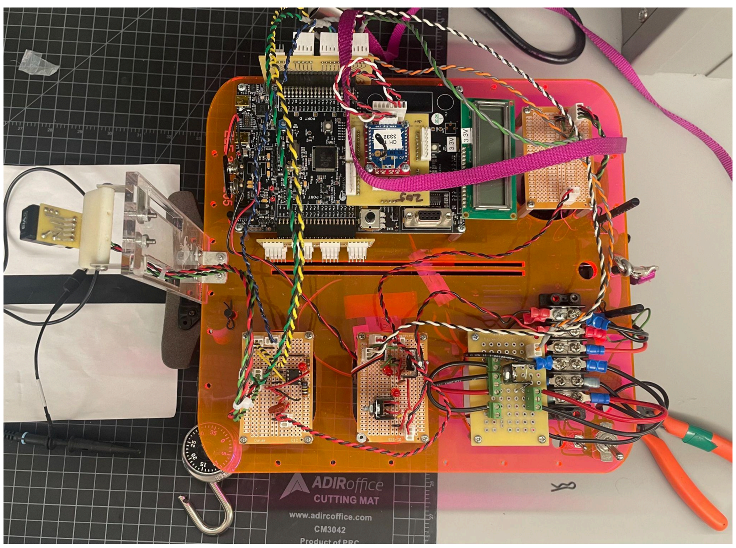
STATEMENT OF OBJECTIVE

The objective of this project was to design and build an autonomous car that can maintain a constant speed whilst navigating a black race track using a camera.

To implement this, we used a P controller where our parameter Kp was tuned on the basis of the direction of steering as well as errors made while following the black line.
We also tuned the PWM_Motor_CMPR on the basis of how fast/slow our car was moving on the track.

KEY SUB-SYSTEMS AND COMPONENTS

From the image above, the following are the key subsystems used:

F 

(1) Motor Board:

- PWM MOSFET: This is used for converting analog signals from the PSOC to digital signals to drive the motor. When the PWM signal from the PSOC is high, the voltage, VGS = +(logic signal is high), and this causes the MOSFET to allow current to pass through using electrons, because it's an n-channel MOSFET, to drive the motors. Otherwise when the logic signal is low, VGS = 0, then the MOSFET does not conduct and hence the motor is off.
- 100 ohms resistor: We need a small resistance at the gate terminal of the MOSFET so as to allow for higher switching time, because the base capacitor of the mOSFET gets charged quicker as well as prevent high power dissipation from our batteries.

- A diode: We connect a diode parallel to the motor so as to invert the digital signal from the PWM MOSFET as well as allow unidirectional movement of current. When the digital signal is high, then voltage flows across the path of least resistance, the diode path, leaving the motor OFF. When T. When the digital signal is low, then voltage flows across the motor and drives the motor. The diode also helps avoid any induced current back to the MOSFET.

(2) Power Board:
- Voltage regulator, 7805: This is a voltage regulator chip that helps output 5V no matter the input voltage. The output of this voltage regulator is fed into the Hall Effect board.
- Voltage regulator, 7806: This is a voltage regulator chip that helps output 6V no matter the input voltage. The output of this voltage regulator is fed into the servo motors.
- LED: It is good practice to always have LEDs so as to ensure that the power board is being fed power and also enable easy troubleshooting when something isn't working.

(3) Hall Effect Board:
- LED: It is good practice to always have LEDs so as to ensure that the power board is being fed power and also enable easy troubleshooting when something isn't working.
- Connection to a Hall Effect Sensor: The Hall Effect sensor, A110 which has 3 pins, gets connected to the Hall Effect Board. The purpose of the Hall Effect sensor is to fluctuate output voltage between high (5V) and low(0V) under the influence of magnetic fields. Thus when the hall effect sensor sees a magnet, it switches to 0V, and 5V vice versa

(4) PSOC(Programmable System on Chip)
- Programmed using a PSOC Creator
- XBee: Used for communication between the PSOC and the UART output, that is emitted into the computer.

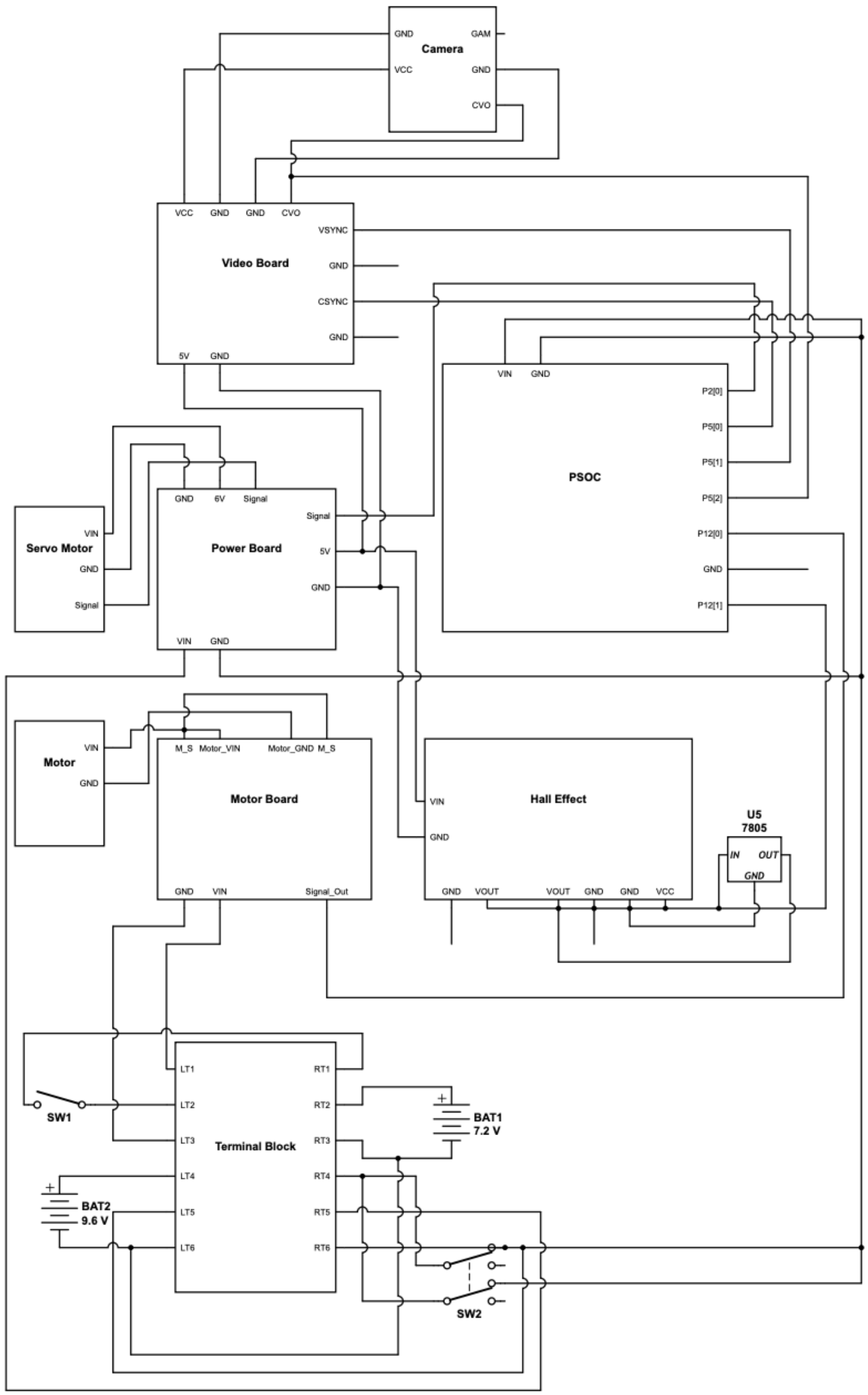
(5) Video Board:
- LED: It is good practice to always have LEDs so as to ensure that the video board is being fed power and also enable easy troubleshooting when something isn't working.
- Connection to a Voltage regulator, 7805: Because we needed to power our video board with 5V, we made a KK connection from the 7805 output to the video board to supply it with 5V.
- LM1881: A video sync operator that was mounted on an IC socket was used to extract timing information mainly composite and vertical sync data outputs.
    - Connections for the outputs and inputs of the LM1881 were done on the basis of what the datasheet required us to do. A few extra connections that we did included setting up an LED on the VCC to easily check power

and a 75 ohms resistor in parallel with the 0.1uF to make the video circuit impedance match the camera.

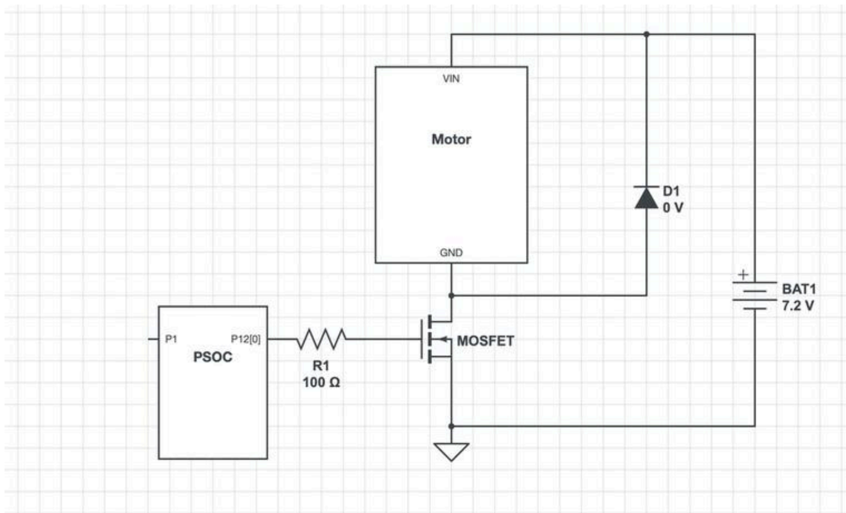
-

SCHEMATIC DIAGRAM OF CIRCUITS

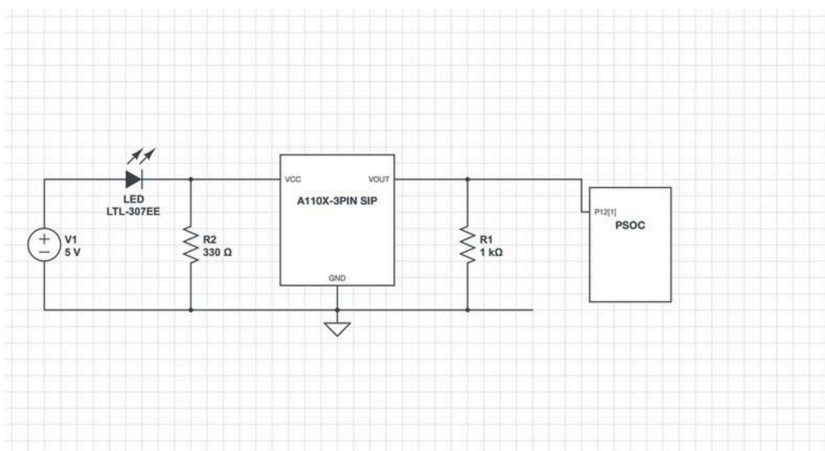
OVERVIEW SCHEMATIC:



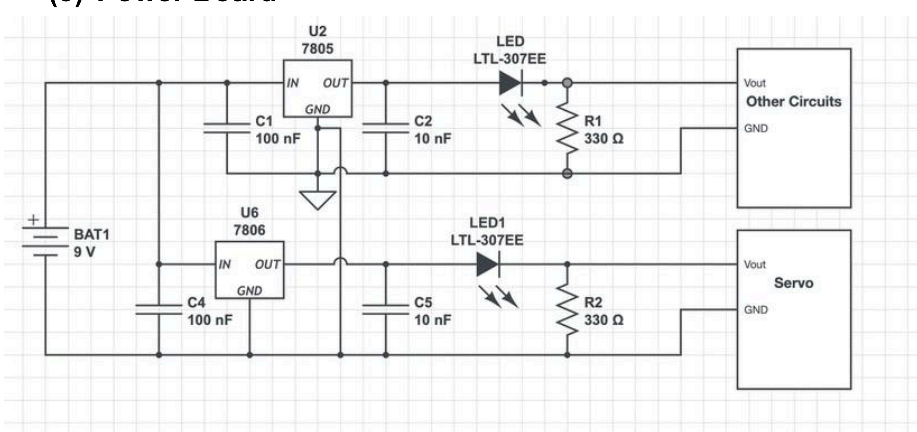
(1) Motor Board



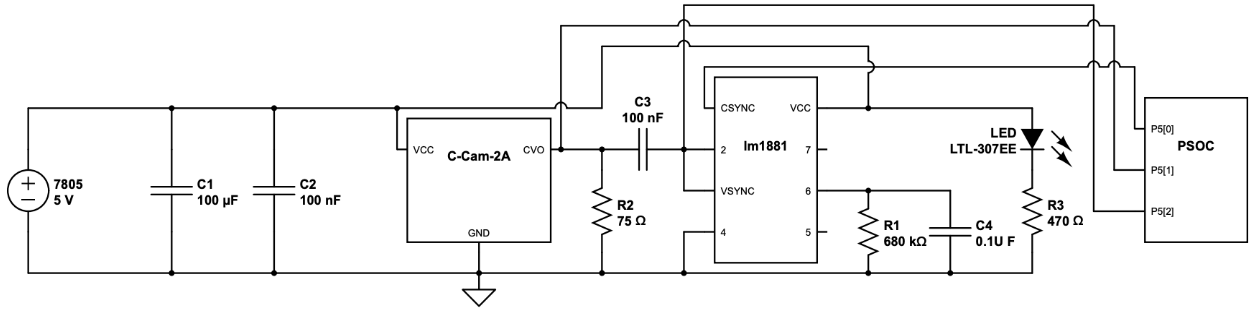
(2) Hall Effect Board



(3) Power Board



(4) Video Board

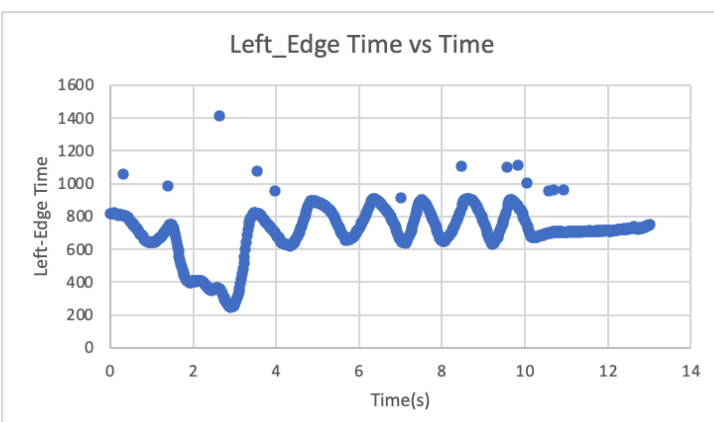


Data

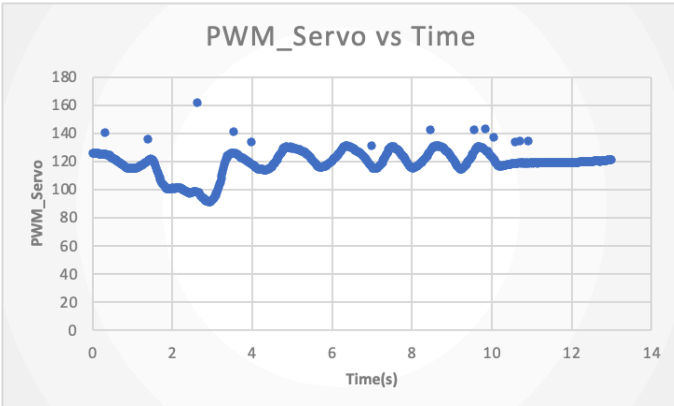
Telemetry Data

1. Graph indicating error vs time(s) on a small portion of the race track. Data extracted from PUTTy.

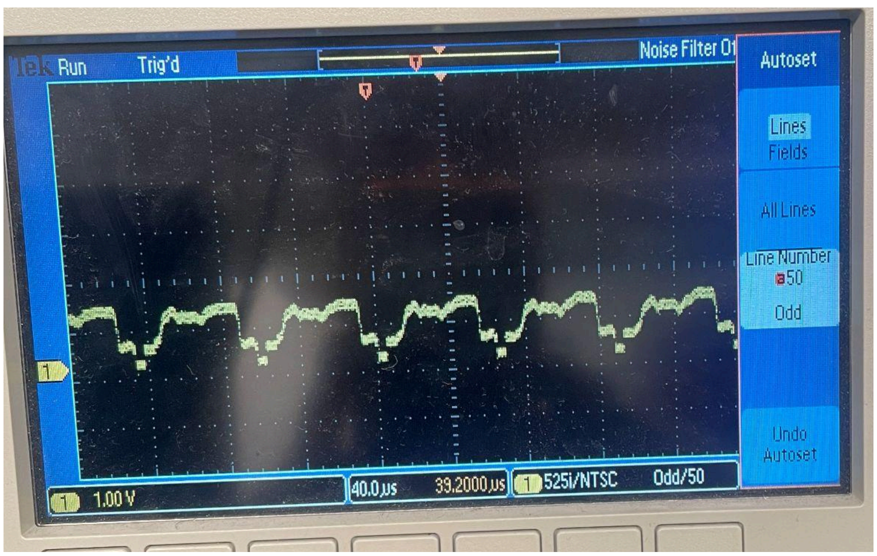
2. Graph indicating the time for the camera to detect the Left_Edge of the black line vs time(s) on a small portion of the race track. Data extracted from PUTTy.



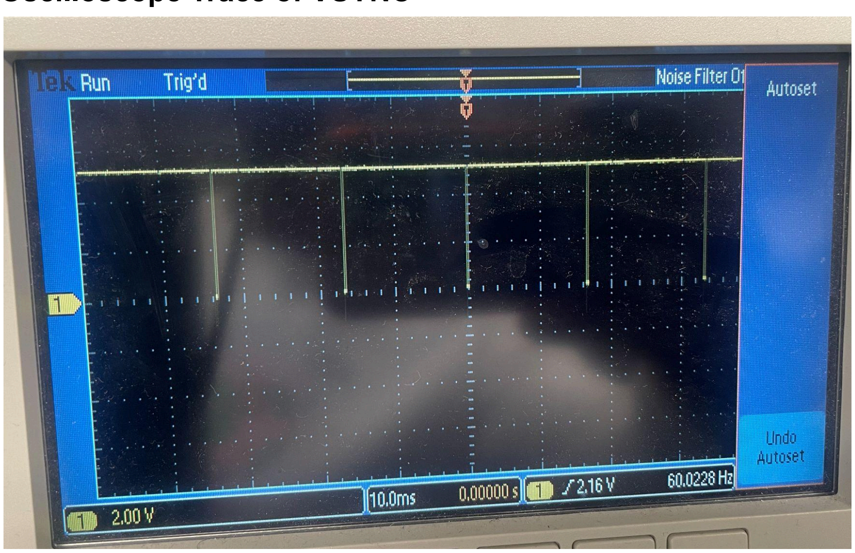
3. Graph indicating the PWM signal of the servo motor vs time(s) on a small portion of the race track. Data extracted from PUTTy. The shape of this graph is very similar to graph 2 as expected.



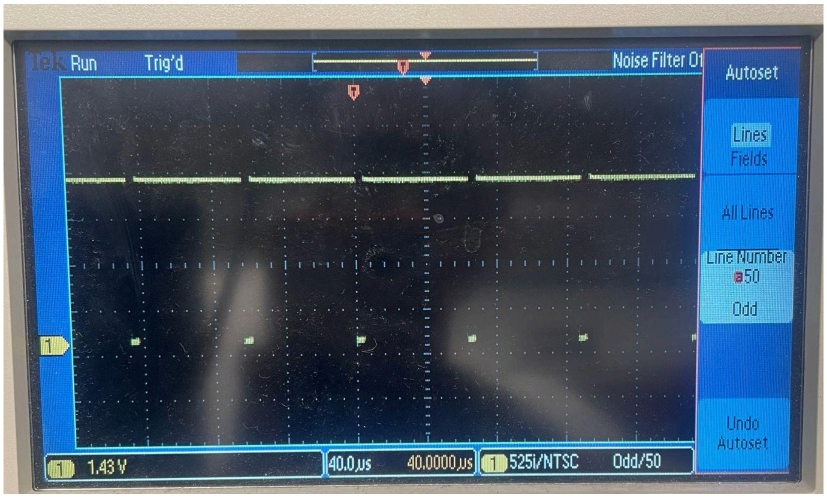
Oscilloscope Trace of Raw Camera Data (CVO)



Oscilloscope Trace of VSYNC



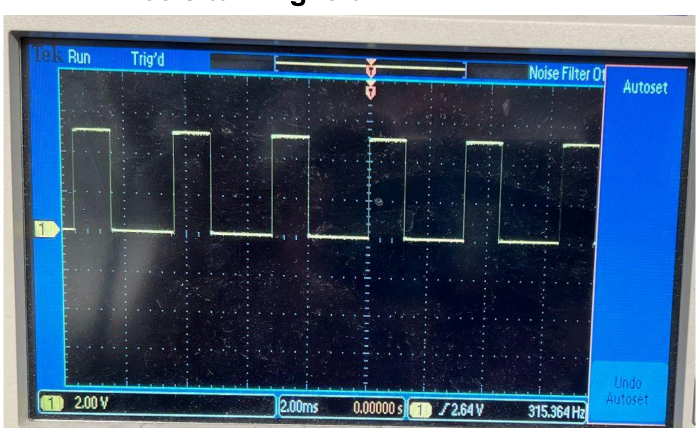
Oscilloscope Trace of CSYNC



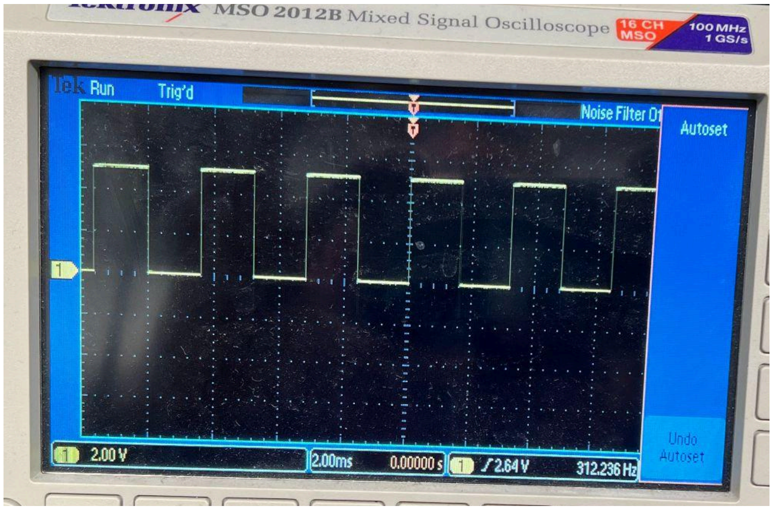
PWM Signal to Motor

PWM Signal to Servo Motor:

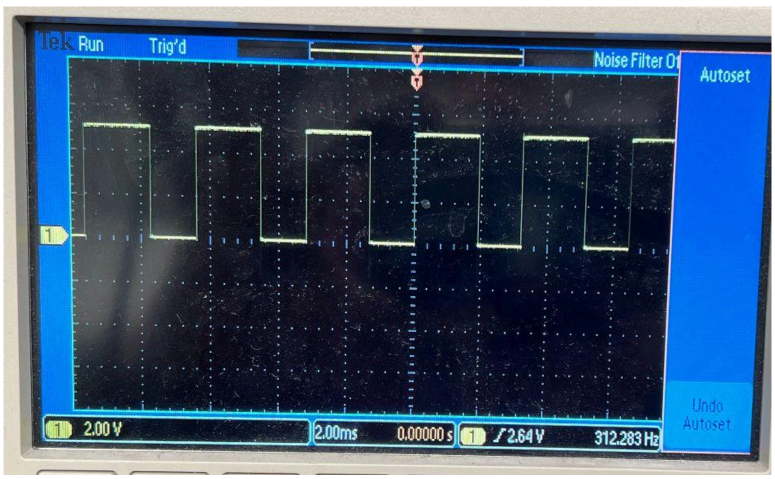
- **Wheels turning left**



- **Wheels straight**



- **Wheels turning right**



<u>CHALLENGES & DESIGN CHOICES MADE</u>

1) To supply the Video Board with 5V, we created another output using KK connectors from the 7805 in the Power Board
2) We made sure to include an LED on the Video Board to ensure that we can troubleshoot whenever we have issues with our boards because we could always cross out the problem of ‘no power’ when our LED is turned on.
3) Mounted the LM1881 on the IC socket instead of directly soldering it on the Video Board. This was done to ensure that the heat from the soldering action didn’t damage the chip.
4) Encountered a problem where our time values were being printed as constant values even with changing the position of the ‘paper race track’. We realized that our problem was that we were printing our Timer_ReadCapture function, instead of saving it to a variable and then printing the variable output.
5) Tuning the PWM_Motor_CMPR when tracking the timing of our car on the race track proved to be difficult. A faster car meant harder to take corners while a slower car meant not finishing the lap in time. Finding the ideal time required tuning and testing.

6) We also encountered a problem where our car was experiencing oscillations along the race track. To correct this problem we introduced a glitch filter between the analog comparator and the timer to eliminate noise.

WHAT COULD BE IMPROVED?

1) Printing a much longer mast. Because our mast was a bit short, we encountered 2 major problems, one being a bigger field of vision making the black line a bit thicker than it usually is. Another problem was the fact that because the camera is too close to the ground, it was also spotting the tiny dark spots outside the race track.
2) With an increase in our duty cycle, PWM_Motor_CMPR, we could have been able to finish the lap in a much faster time than the 50s time-lapse we did.