Aaron Aox
Andrew Wilhelm
Chenjie Wu
Siwei Zhang
ENGR 1182
Dr. Herak 8:00am

A17 -Verification Plan

1.Detailed Verification Plan

In the verification, we define full mark to "success". That means, the system requirement, which means that the related to user need, is satisfied.

i. Chip processing speed.

The processing speed of our arduino chip indicates how many instructions the processor can execute in every second. If the processing speed is too low, it may not be able to execute the binary codes on time (the compiling of source code is finished on PC), and there may be some problems in the running of our model system.

Directly measures the flops value is difficult, so we can measure the time it complete a chunk of code. There is a software available on github: https://github.com/PaulStoffregen/CoreMark that can "Measures the number for times per second your processor can perform a variety of common tasks: linked list management, matrix multiply, and executing state machines.", according to its readme.md file. For a reference, the Arduino Due chip has a core mark of 94.95. Therefore, we would set 94.95 to full mark. We would load this code into our arduino chip and receive the benchmark score, and compare it with Arduino Due's score. We would test 3 times with interval of 5 minutes to prevent overheating, and take the average.

ii.Power supply efficiency

The power supply efficiency means how much energy could be used by us appropriately . When the ratio of the efficiency goes higher, it indicates that more energy could be used to drive the train. Of course,the lower power supply efficiency means the waste of resources.

We can directly measure the the power supply efficiency by using a wattmeter and a resistor. Simply connect resistor with a large resistance to power socket and use watts meter to measure its power W1. Then, connect a small-resistance resistor behind the power supply and measure its power W2. The efficiency can be gotten by W1/W2. The goal is 80 %, which is the ideal requirement and is set to full mark. The zero score is set at 72%.

iii. regenerative brake energy recovering efficiency.
It measures how many percents of kinetic energy can be recovered back to form of electricity by the braking system.

We would buy a power meter and a small bulb. Firstly, we would calculate model train's speed at full power by measuring the time of finishing 1 loop. Then, its kinetic energy can be calculated by $Ek = (1/2)m*v^2$. Then, we would stop the power supply but do not apply braking, and measure how long it takes for the friction to stop the train. The acceleration can be known by $a = v/t$. Then, the frictional force is known by $f = ma$. Then, we would let the train to run in full power, and apply full braking after stopping power supply, while recording the distance s used for braking. Then, we connect our bulb to the battery in the cabin, and measure the power of our bulb. We would record the time and wait until power run out, and the energy is $w*t$. Lithium-ion battery's discharging efficiency is surprising high so we only need to take into consideration of friction. $Ek-fs$ is input of our regenerative braking system, and $w*t$ is energy recovered. output divided by input is our convert efficiency. The goal is 83%, which is set to full mark. 55% of efficiency is set to zero. We would test 3 times and taking average.

iv. backup electricity system reaction time
It indicate the time between main power supply fails and backup power supply take over the responsibility.

Simply measure the time between unplugging main power supply and when backup power supply take over. The ideal requirement is 1s, which is set to full mark. Every second more than 1s results in -25% score. We would test for 3 times and take the average.

v.Train's cabin's weight
It's literally the factor shows us the weight of each cabin of the train. As we all know, the lighter cabin help as to save more resources and the appropriate material makes the cabin become harder. So how to choose the material and control the weight of the train is a big deal.

Aluminium alloys are widely used in engineering structures and components where light weight or corrosion resistance is required. So we choose aluminium alloys to be the material of the cabin's weight.Then we just need to prepare a weight and compare the volume of our train and outdated trains. To be more detailed, we can keep the volume of our train as the outdated version, and calculate the weight of our model.

The full mark is set at 0.8kg. Every 0.1 kg exceeds results in -25% points.

vi. Braking distance.
It refers to the braking distance of our train from full speed. It relates to user need of safety. The full mark is set at 2.2 inch. The range of score linearly spans over 2.2-3 inch.

To marsure the mean braking distance, we need to set a mark in a specific position. And when the train arrive the point with full speed, we should press the stop button. Then we need to mark the parking place. After repeating 10 times. we can calculate the mean barking distance by the 10 results.

2. Verification Score Card

A. Verification score card (first pass)

System Requirements	Vehicle Reqs	Score Rubric	Score
chip processing speed	1.2GFlops-2.4GFlops	-1% for each benchmark score less than 94.95	10
power supply efficiency	72~80%	-12.5% for each 1% less than 80%	10
regenerative brake energy recovering efficiency.	55~83%	-3% for rach 1% less than 83%	10
backup electricity system reaction time	1s - 5s	-25% for each second longer than 1s.	10
Train cabin’s weight	0.8 - 1.2kg	-25% for each kilogram more than 0.8 kg	10
Mean braking distance	2.2~3 inch	-12.5% for each 0.1 inch longer than 2.2 inch.	10

B.

	chip processing speed	power supply efficiency	regenerative brake energy recovering efficiency.	backup electricity system reaction time	Train cabin's weight	Mean braking distance	**weight**
Ease of Use	1			3	3		4
Fast Commute							4
Safe	3			9		9	5
private					3		3
Always Ready	9			3			4
Reliable	1		3	9		3	4
Internet Connectivity	3						1
Low cost		3	9		3		4
Customizable					3		2
Environment friendly		9	9		3		2
Importance	17	12	21	24	15	12	

Determine scaled points:

	Unscaled	Scaled	Rounded	Points (100)
chip processing speed	10	16.8%	17%	17
power supply efficiency	10	11.9%	12%	12
regenerative brake energy recovering efficiency.	10	20.8%	21%	21
backup electricity system reaction time	10	23.4%	23%	23
Train cabin's weight	10	14.9%	15%	15
Mean braking distance	10	11.9%	12%	12
		100%	100%	

Verification Score Card (Updated)

System Requirements	Vehicle Reqs	Score Rubric	Score
chip processing speed	1.2GFlops-2.4GFlops	-1% for each benchmark score less than 94.95	17
power supply efficiency	72~80%	-12.5% for each 1% less than 80%	12
regenerative brake energy recovering efficiency.	55~83%	-3% for rach 1% less than 83%	21
backup electricity system reaction time	1s - 5s	-25% for each second longer than 1s.	23
Train cabin's weight	0.8 - 1.2kg	-25% for each kilogram more than 0.8 kg	15
Mean braking distance	2.2~3 inch	-12.5% for each 0.1 inch longer than 2.2 inch.	12

3. Realistic Test

Based on our determined scaled point matrix, we determined 3 most important requirements: backup electricity system reaction time, regenerative brake energy recovering efficiency and chip processing speed. Of course,We have designed a verification plan and show the process to three users we invited. We setup the devices we need and began to verify the first requirement: backup electricity system reaction time. Siwei compiled a chunk of code that instruct the model train to run at 100% power. It takes a few seconds for the PC and arduino chip to process the code, then the train started. After it reached full speed, Chenjie unplugged the power supply. The train was dragged fiercely by regenerative brake. However, within 1 second, backup power took over the control and the train started to accelerate.

Then, we demonstrated high energy efficiency of regenerative braking system. We started the train and made it stop without brake. Aaron measured the stopping distance, and calculated that the frictional force was 0.2N. Then, another stop was performed with full break. Andrew connected a bulb with battery, and connected a power meter onto it. We waited until the battery ran out, and the time was 136s. We performed calculations and demonstrated that the efficiency was 82.7%, which is almost ideal. .

At last, Chenjie pulled the benchmark code from github and sent it to Siwei, who sent it into Arduino chip. The chip began to automatically perform intense calculations. After nearly 5 minutes, the software showed a 92.11 score, which is high enough.

The users appreciated our achievements of our model system.