Anthony Louis Rosenblum

Design Definition Progress Report

EGEN-310R, Team E7

Computer Engineering

2/12/2019

Define Progress Report

***Instructions**: Read all Milestone 2 information. Answer all questions. Digitally submit this document with your answers inserted. Name the document your-last-name. Any other documents requested should be uploaded as well, with a similar naming convention, eg your-last-name_ 1. <u>Realize that your design responsibilities do not end with the questions you answer</u>. The questions are simply there to guide you to pertinent information, relevant steps you might need to take, etc. One Milestone 2 document should be submitted for each discipline involved in the design. That means if there are multiple designers working within the same discipline then **while each must submit** a Define Milestone, it may be the **same** document. Note that Discovery and Define progress reports are graded as pass/fail, with a "passing" grade indicating that the instructor acknowledges that the student has attempted to answer all questions with some level of depth. DESIGNERS are responsible for ensuring the accuracy of ALL calculations.*

***Design process steps included in this Milestone:** The main purpose of the Define Progress Report is for the design team to work towards turning verbal and written requirements into engineering specifications and design metrics. Designers should continue to research areas of technical relevance and begin working towards creating initial purchase lists (AFTER specifications have been determined) and begin planning physical experiments and numerical modeling exercises such that they may better understand design tradeoffs.*

***Teaming Expectations**: If more than one designer on a team is working within a discipline, for this milestone they may work together. The entire team will have to work together to form a complete set of functions, objectives and constraints which will then define the entire multidisciplinary design problem.*

***Documentation Considerations**: The final report will have a chapter devoted to the "discovery" and "define" phases of the design project so keeping relevant, organized notes is imperative. At this stage, the majority of the discovery work should be completed. Consider writing a draft of this chapter of your report. The final report will need to include a correct functional diagram and objectives tree for the multi-disciplinary design project as a whole. The group should strive to have these diagrams finalized by the time the Define Progress Report is submitted.*

Design Theory

At this stage in the process, designers may have enough knowledge to understand what physical and technical concepts may be important to the design, but still lack clear definition of exactly what the design **must do** and what the **goals** are for the design. Each design team has compiled a list of *solution neutral* attributes, which were derived from asking "WHY?" Why might we want a particular feature or component in the design? Asking "why" enough times, eventually led us to a list of **design attributes**. Now the attributes must be organized into those that are functions and those that are objectives.

Functions refer to things that the design must DO. Functions are verb oriented, action words. Once identified they are organized into a "Black-Box" functional diagram (see Dym Chapter 4) which forces the designer to clearly identify **inputs** and **outputs** of the design. Design inputs are typically materials, energy and/or information. Design outputs are transformed materials, energy and information (including waste). Once inputs and outputs are very clear to the designer, the second stage functional diagram is created, in which the "top" of the box is removed and **subfunctions** are identified. Most, if not all, subfunctions should be assigned a **functional specification**. A specification answers, using specific numbers and engineering units "How much?", "How large?", "What size?", etc. All potential design solutions need to satisfy the functionality identified in the functional diagram.

Objectives are the goals for the design. Objectives are adverbs and adjectives, describing what the design should be or how or to what extent the design should do something. Objective should be organized into a tree that relates high level objectives to lower level ones (see Dym Chapter 3). Objectives at the lowest level of the tree should be assigned **metrics**. That is, designers should ask how could two competing designs, each satisfying minimal functionality, be judged against the objectives? Measurable metrics could have units, could be binary (yes/no), or could be some way of quantifying an essentially qualitative metric (think "aesthetically pleasing").

Constraints arise when there are clear limits on the design that don't necessarily relate to the functionality that it needs to achieve. Constraints might stem from safety-related objectives. They could be physical dimensions if the design must "fit" somewhere. They often relate to following standards or design codes. Constraints form go/no-go boundaries for the design and MUST be satisfied.

Looking forward, we will use functions for design **ideation**. Designers can brainstorm different ways of satisfying each function and then mix and match those methods together, resulting in a large number of potentially viable solutions to the design problem. Feasible design concepts (those that satisfy minimum functionality) will then be compared against design objectives and metrics will be applied to decide which design satisfies those objectives to the highest degree.

Functional Specifications and Measurable Metrics

Include a copy of your team's functional diagram (both with the black box top "on" and "removed") and objectives tree with this document submission. NOTE: These diagrams should be digitized and report ready; NOT a snapshot of a hand drawn diagram. They should be saved as a separate pdf document.

1. **Complete the table below to the best of your ability for sub-functions that fall within your discipline purview (add rows as necessary). You will need to determine as many specifications for those functions as you can – some will come from numerical models, some from background research, some may be educated guesses. Indicate where your specifications came from in the appropriate column.**

Function	Specification	Reference/Resource
Sends electrical signals to motors	**See attached microcontroller specification**	**See attached microcontroller specification**
Distribute power between motors	**See attached motor controller specification**	**See attached microcontroller specification**
Sends digital signals to microcontroller	**Bluetooth/Wifi**	**Raspberry Pi 3**
Supplies current to motor driver	**LiPo/NIMH battery**	**Venom Power**

2. **Complete the table below to the best of your ability for all low-level objectives that fall within your discipline purview (add rows as necessary). You will need to provide a measurable metric for each low-level objective.**

Low-level objective	Metric
Supply power	Watts
Supply current	Amps/ Amp-hours
Remote communication	Type
Reverse current direction	Yes or no
Support multiple motors	Number

3. **List any constraints relevant to your portion of the design problem. Be sure to check all relevant Cat's Conundrum Documentation.**

- Commercially available parts designed for the RC market are NOT allowed (electronic speed control, transmitter, or receiver).
- Vehicle must be remotely controlled. Touching the vehicle during the course demo will result in lost points
- The vehicle must be able to complete the course without changing power source (battery, other)
- Central microcontroller must be programmable
- Vehicle must fit under tunnel

Microcontroller Specification Sheet

Yellow highlighted column is required. "Actual Product Specifications" recommended as you consider different products, but not required for this Progress Report. You will be required to demonstrate that chosen product meets specifications in the final report.

	Ideal Specifications	**Actual Product Specifications**	**Actual Product Specification**
	Values be based on calculations with reasonable assumptions	**Product Name:** Arduino Uno	**Product Name:** Raspberry Pi 3 Model A+
Cost ($)	$20.00	$18.33	$25.00
Programing Language	Python (preferable language for our EE)	C/C#	Python, and many more
Method of wireless com if applicable	Bluetooth		Bluetooth
# of GPIO pins	40	14	40
# of analog input pins	10	6	N/A
Flash memory (KB)	16	32	512
Min/Max Operating Voltage (Volts)	-9 volts to 9 volts	0 volts to 5 volts	0 volts to 5 volts
Logic voltage (Volts)	9 volts		
Maximum output current (amps)	3 amps		
Dimensions (Length x width x height)	50 mm x 35 mm x 15 mm		

Weight (grams)	4.20 grams		
Mounting method	Wiring holes and zip ties		
List safety features (voltage protection, thermal shutdown, etc):	Waterproofing,		
Control Interface (USB, TTL Serial, analog voltage, I2C, etc.)	USB to serial communication		
Is CAD drawing available (yes/no)	Yes, in addition to wiring schematic and pin-out		
Level of internet support: 0 to 5 (with 5 = incredible support)	5, if goal is to communicate over shared wireless network		
List Additional Features Below	Internal current reversal and variable resistor network		

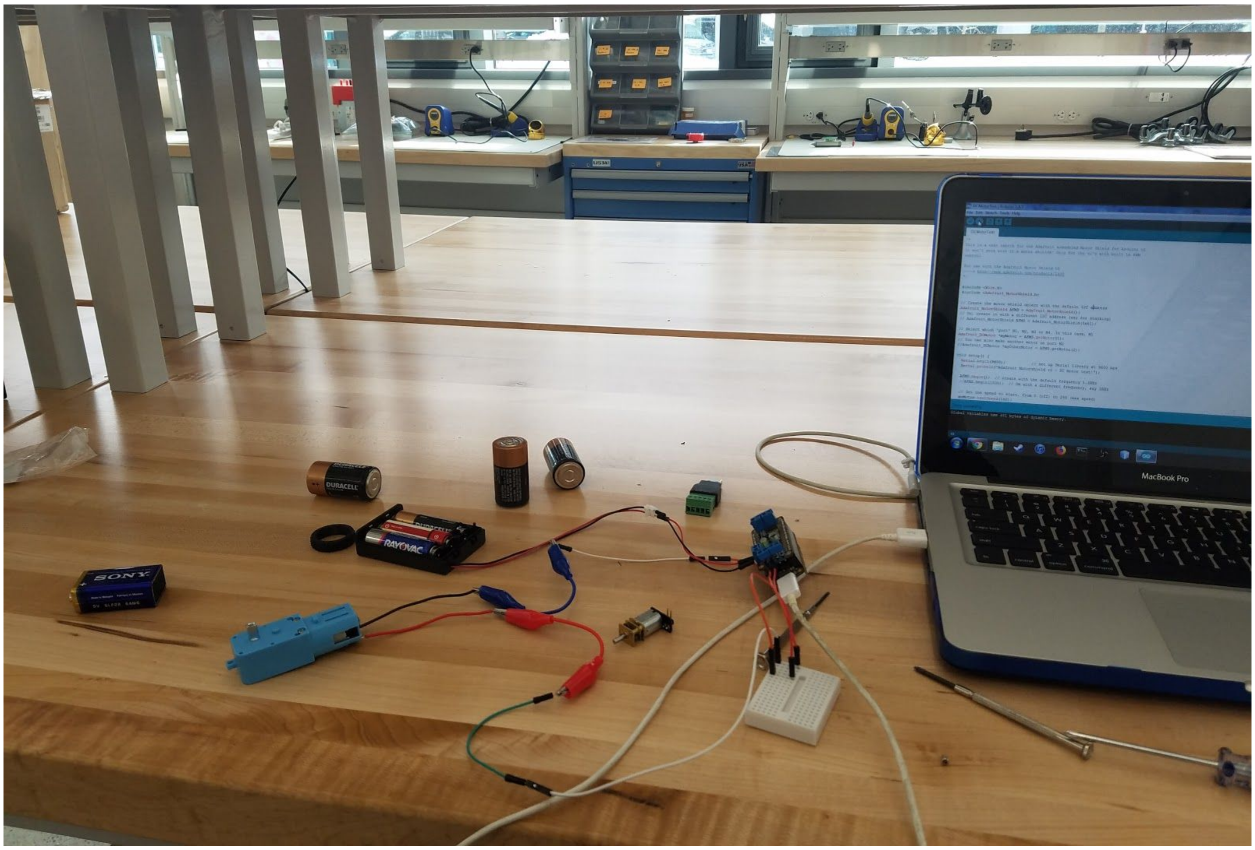
Motor Driver Specification Sheet

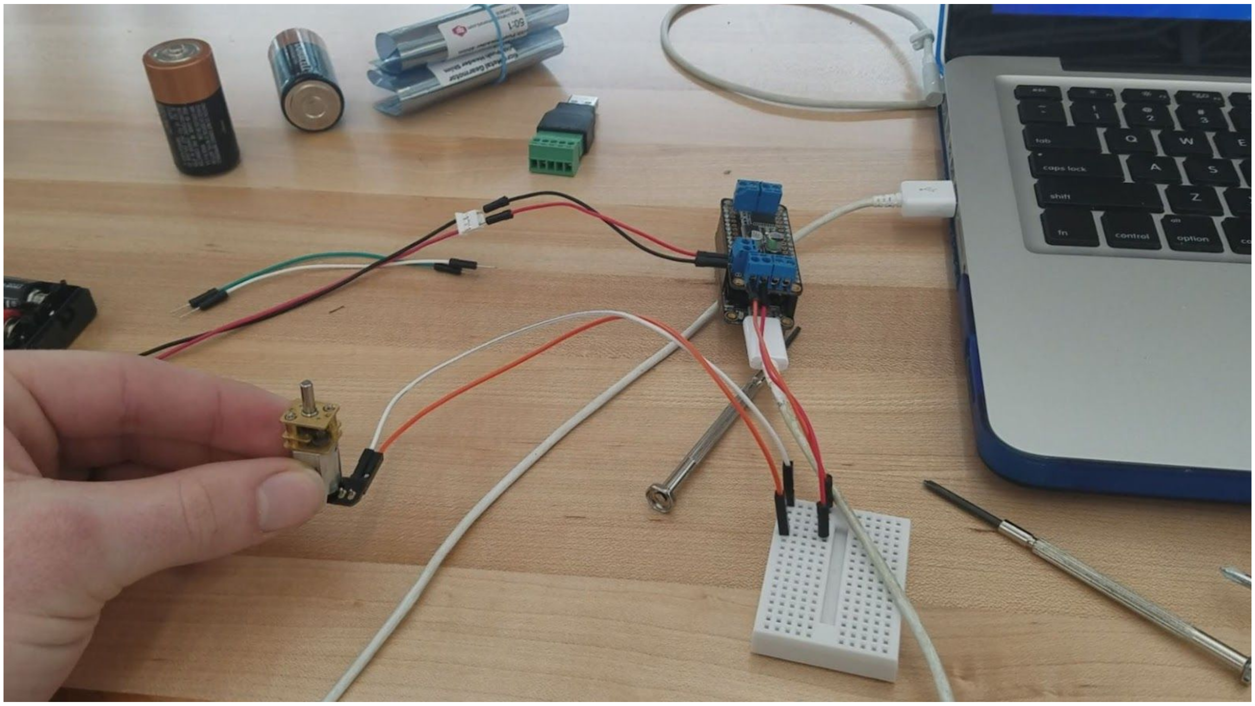
Yellow highlighted column is required. "Actual Product Specifications" recommended as you consider different products, but not required for this Progress Report. You will be required to demonstrate that chosen product meets specifications in the final report.

	Ideal Specifications	**Actual Product Specifications**	**Actual Product Specification**
	Values be based on calculations with reasonable assumptions	**Product Name:____________________**	**Product Name:____________________**
Cost ($)	$7.00		
# of Motor Supported	Four motors		
# of Servos Supported	Four servos		
Stepper Motor Compatible (yes/no)	Yes		
Min/Max Operating Voltage (Volts)	5 volts to 35 volts		
Continuous/Peak Output Current per channel (Amps)	2 amps		
Dimensions (Length x width x height)	43 mm x 43 mm x 27 mm		
Mounting method	Wiring holes and zip ties		
Input pin min/max voltage	0 volts to 5 volts		
List safety features (voltage protection, thermal shutdown, etc):	Small heating, strong anti-reference		
Specific microcontroller compatibility (no or yes with board name)	Arduino – Yes, Raspberry Pi – Yes		
Control Interface (USB, TTL Serial, RS-232, CAN, analog voltage, I2C, etc.)	USB to serial communication		
Is CAD drawing available (yes/no)	Yes		
List Additional Features Below	Internal current reversal and variable resistor network		

Planning for physical experiments and/or numerical modeling

1. **As you consider which microcontroller to choose, it would be logical to try controlling a motor and/or servo with each microcontroller under consideration. The Makerspace has microcontrollers and motors and servos which you can borrow for such a comparative test.**





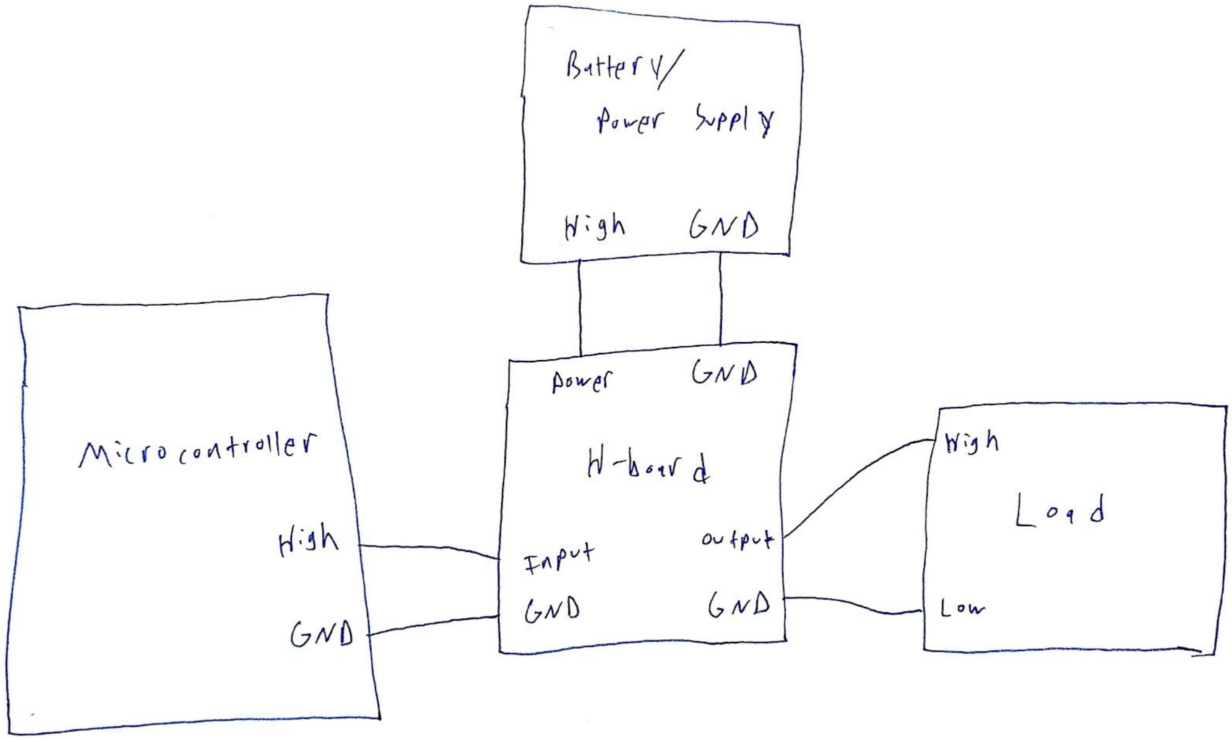
a. **Choose at least two different microcontrollers to test. Identify them here.**

- Adafruit Huzzah Feather ESP8266 wifi
- Adafruit Feather 32u4 Basic Proto

b. **Choose at least two different loads to test. Identify them here.**

- 50:1 DC metal gearmotor
- 90:1 DC metal gearmotor

c. **Sketch the circuit that you create with any other required components.**



d. Refer to your stated design objectives. What metrics are important in this test?

- Power consumption and efficiency
- Voltage load and current draw
- Remote control capability
- Ease of use

e. What do you conclude from the test, based on those identified important metrics?

The ESP8266 and 32u4 microcontrollers were fairly similar. They were each easily programmable and powered by a 5V power supply. Each interfaced directly with a motor driver designed to be used with arduino. The ESP8266 was capable of receiving signals over wifi while the 32u4 comes with a bluetooth chip.

Both the 50:1 and 90:1 DC motors functioned appropriately when provided a power supply of 4.5V from 3 double A batteries. However the 50:1 motor was able to complete more rotation cycles when provided the same amount power supply.

f. **Will addition of the motor driver to the circuit make a difference to the choice of microcontroller? Why?**

It would be best to choose a motor driver designed specifically for the microcontroller being in used. Because there are motor drivers designed for many different microcontrollers it makes the most sense to select a microcontroller based on other objectives and constraints and then select the appropriate motor driver.

2. **Identify *at least* one other experiment or numerical model that you will create to better understand design parameters. One idea might be to add the complication wireless control on the motor control circuit (work with CS if you have one). Plan out this/these experiment(s)/model(s). Keep in mind this can/will be used in the final report.**

a. **Experiment/model seeks to determine:**

Most efficient and error proof way to route commands to motors in design requested by group MEs/METs.

b. **Design parameters to be investigated:**

Programming minimalism and error/erroneous command prevention.

c. **What constitutes a "good" result?**

A program that successfully routes desired commands to their intended operations through the motors while preventing errors from arising.

d. **Supplies required:**

MATLAB for pseudocode enumeration, comments, and analysis.

Language

specific IDE/compiler for evaluation.

e. Data/information to be recorded:

How many function and/or loops are needed for the controls to operate properly?

Is the code superfluous? Can it be more easily organized and understood?

Is the code ambiguous?

What does the car do when it does not receive a signal?

What does the car do when it receives conflicting signals?

Mathematical Model

Update your spreadsheet model with any parameters which have been determined since Milestone 1. Be sure to note how such parameters were determined (experimentally, through research, or estimation with sound basis).

Resubmit your excel spreadsheet(s) and summarize major changes made HERE:

```
%% Section 2: Calculations for Definition Report
% Last updated 2/6/2019

% Using design requests from ME/METs

% Assume 4 motors, each controlling one wheel, no servos in system

% Determine necessary functions and how they could be nested

% System Design 1

wheels = [[L1,R1],[L2,R2]];

% Forward

L1.fwd
R1.fwd
L2.fwd
R2.fwd
```

```
% Reverse

L1.rvr
R1.rvr
L2.rvr
R2.rvr

% Turn left in place

L1.rvr
R1.fwd
L2.rvr
R2.fwd

% Turn right in place

L1.fwd
R1.rvr
L2.fwd
R2.rvr

% System 1 requires 4 functions, all at top level. No ambiguity, large number of
lines needed, low number of loops, car will stop while there is no signal.

% System 2

left_wheels = [L1,L2]
right_wheels = [R1,R2]

left_wheels.fwd = [x.fwd,y.fwd]
left_wheels.rvr = [x.rvr,y.rvr]

right_wheels.fwd = [x.fwd,y.fwd]
right_wheels.rvr = [x.rvr,y.rvr]

% Forward
left_wheels.fwd
right_wheels.fwd

% Reverse
left_wheels.rvr
right_wheels.rvr

% Turn left in place
left_wheels.rvr
right_wheels.fwd
```

```
% Turn right in place
left_wheels.fwd
right_wheels.rvr

% System 2 requires 8 functions, 4 at top level, 4 nested underneath. More neatly
organized but requires more moving pieces in code. Car stops when no signal is
detected. No ambiguity.

% System 3

while (go = 1):
    left_wheels.fwd
    right_wheels.fwd
    BRA down
end

while (turn = 1):
    left_wheels.rvr
    right_wheels.fwd
    BRA down
end

while (go = -1):
    left_wheels.rvr
    right_wheels.rvr
    BRA down
end

while (turn = -1):
    left_wheels.fwd
    right_wheels.rvr
    BRA top
end

% System 3 requires 4 while loops, and 4 functions nested underneath. Car continues
moving until a new signal is received. Does not stop. High ambiguity. Multiple
variables need to be assessed.
```

Purchase List

Begin to identify any other items that *may* need to be purchased for your portion of the design. Begin to compile a list of those items here and their estimated costs.

Part name/part #	Vendor	Quantity	Item cost	Shipping cost	Ship Time

Raspberry Pi 3 Model A+	**Adafruit**	**1**	**$25.00**	**$11.31**	**8 days**
L298N H-bridge Motor Controller	**Amazon**	**2**	**$7.01**	**$0.00**	**2 days**
Venom 20C 2S 3200 mAh 7.4V LiPo Battery	**Amazon**	**1**	**$31.99**	**$0.00**	**2 days**

References

Include those that you used to complete this assignment, use standard formatting so that these references can be copied into your final report.

Adafruit Industries. “Raspberry Pi Model 3 A .” *Adafruit Industries Blog RSS*

“Arduino Uno Rev3 (DIP) by Arduino Corporation | Embedded System Development Boards and Kits.” *Arrow.com*, Arrow.com

Arduino Uno Rev3, store.arduino.cc/usa/arduino-uno-rev3.

Boxall, John. “Tutorial - L298N Dual Motor Controller Module 2A and Arduino.” *Tronixlabs Australia*

"MC9S08QG8 HCS08 MCU 8-Bit HCS08 Embedded Evaluation Board." Digi-Key Electronics. Digi-Key Electronics, Web. 4 Feb 2019.

“Raspberry Pi 3 Model A .” *Rotate Display 90º? - Raspberry Pi Forums*

“Venom 20C 2S 3200mAh 7.4V Hard Case LiPo Battery with Universal Plug System.” Venom Power