Note to self: Each time you work on the project, you should enter the date, start time and end time. You should document the references you consult, the main points from your reading, important tables or circuits from your reading, the ideas you have, the circuits you design, the software you write, simulation and experimental test results, your successes and failures, etc. You will submit your current notebook each week along with your weekly report, see below.

Illa's 2804 Project Notebook

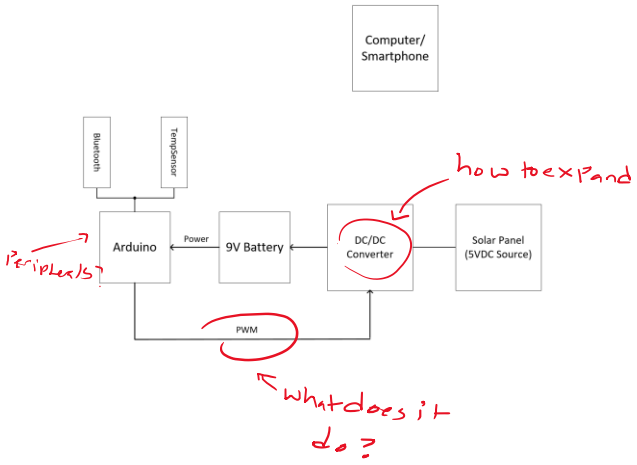
Click on the hyperlink to see the start of the notes taken during the referenced milestone

Table of contents

7/14/21
2:17pm – 3:00pm

To Do's milestone #1

- Intro
  - bkgrnd, proposed sol, significance
- overview
  - explain design + general plan
- Block diagram
  - key functions for hardware (blks)
  - key functions for software + digital
  - analog to digital interface
- Break into milestones
  - list goals for hardware + software
  - specify deliverables
  - specify testing (components + combined)



Software → only can use bluetooth lib

Preliminary Research (sites visited and articles read):

https://en.wikipedia.org/wiki/DC-to-DC_converter (Wikipedia page on DC/DC converter)

Converter_Steady_State_Analysis.pdf (Info for DC/DC converter circuit I think?)

http://ww1.microchip.com/downloads/en/devicedoc/20001942g.pdf (datasheet for temp sensor)

Converter chapter notes

Buck converter → reducing DC voltage using only non-dissipative switches, inductors + caps

↓

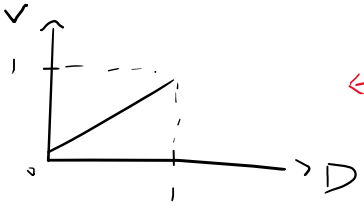
rect wave form

$$\langle V_s \rangle = \frac{1}{T_s} \int_0^{T_s} V_s(t)\,dt$$

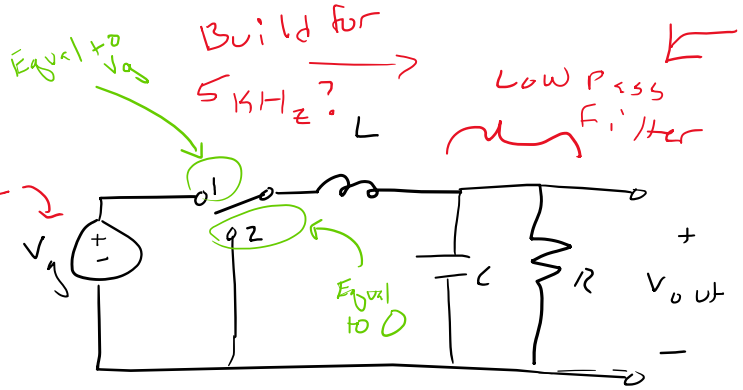
$$\langle V_s \rangle = \frac{1}{T_s} \left(D T_s \; V_s \right) = D V_g$$

DC input volt

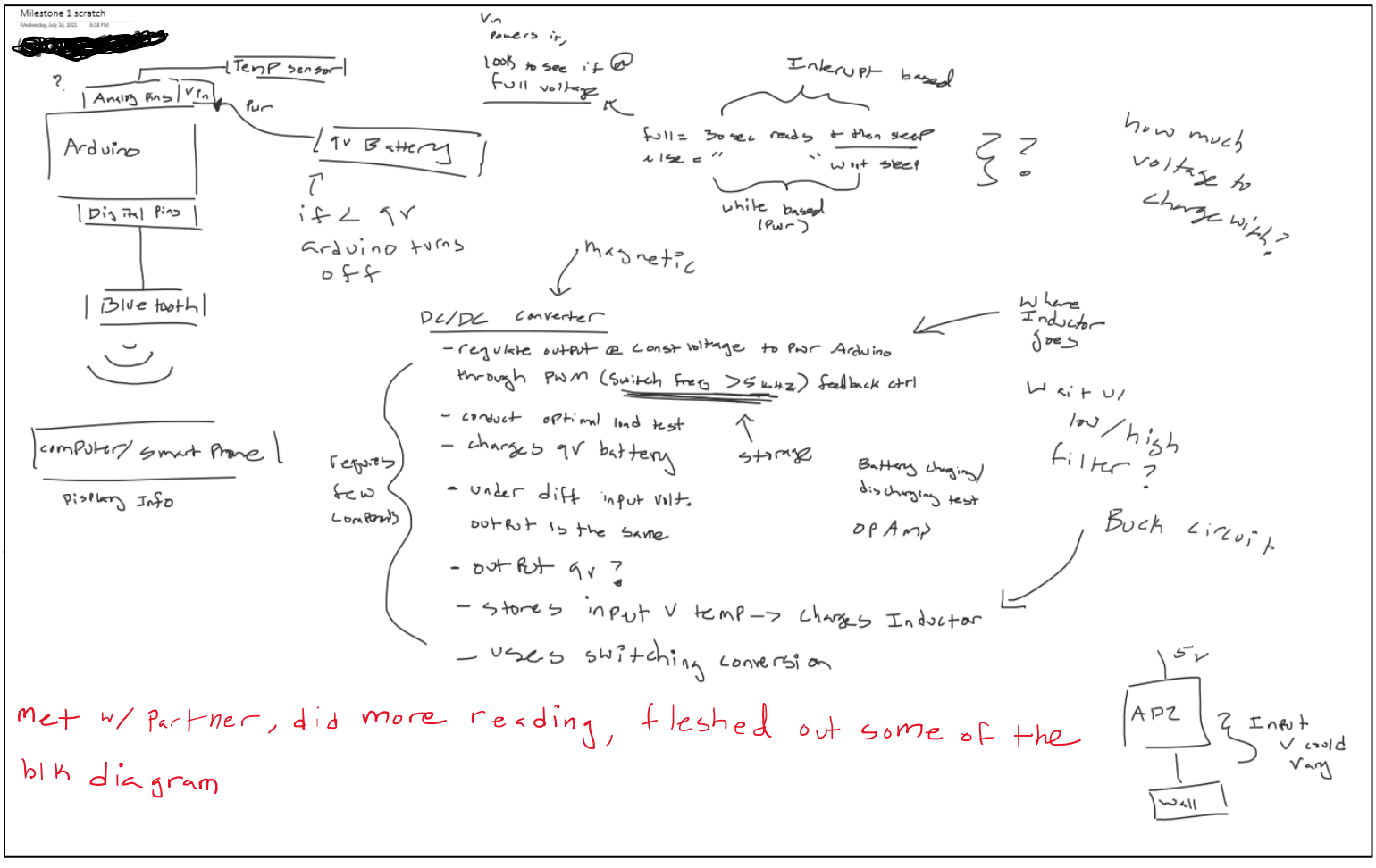
Duty cycle



Buck converter DC out V. vs Duty cycle



7/14/21
6:21pm – 7:40pm



met w/ partner, did more reading, fleshed out some of the blk diagram

7/15/21
1:00am – 1:55 am

Peripherals of Arduino?

- clock: for 30 sec intervals
- monitor/display: to show value
- Bluetooth: to communicate data
- Temp sensor: to read temp
- Input volt val: to pwr + to analyze

Something for PWM?

needed Programming

- clock module
- Interrupt based code
- while based code
- Read + interpret temp
- Display module
- Bluetooth module

Pwr setting

Arduino Research notes

uses simplified C++

major Arduino components

- usb connector → used to load code (can pwr)
- Pwr port → Pwr port (10ps @ 5 v)?
- Analog pins → (6 pins) temp sensor. only measure volt.
- Digital Pins → In or out (read comp. sig)
- Reset switch → run from start
- crystal osillator → tics, math stuff
- TX RX Leds → Transmit, recive data
- micro controller → Brain (has flash mem, ram 32kB 2kB)
- USB Interface chip → signal translator

} might need to reconsider pwr idea from earlier

can be reconfig as dig

~ (tilde) pins can be used for PWM (simulate analog out)

Arduino References (Read or watched to refresh memory / find info)

https://www.arduino.cc/en/Guide/ArduinoUno

https://www.youtube.com/watch?v=_ItSHuIJAJ8

7/15/21
11:42am –2:00pm

Converter reading notes cont.

converter options

- Buck -> dec V; $v \leq V_g$
- ★ boost -> Inc V; $v \geq V_g$
- buck-boost -> Inc or dec but Polarity reverses

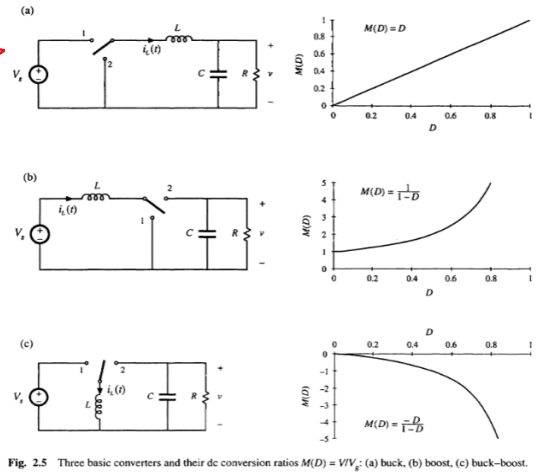


Fig. 2.5 Three basic converters and their dc conversion ratios $M(D) = V/V_g$: (a) buck, (b) boost, (c) buck–boost.

↑ P.5 of pdf

How to make switch?

Is switch related to PWM?

Buck

Impossible for filter to remove all stuff, so there is small V_{ripple}

$|V_{ripple}| \ll V$; $v(t) \approx V$

$V_L = V_g - v(t)$

$V_L(t) = L \dfrac{di_L(t)}{dt}$

$\dfrac{di_L(t)}{dt} \approx \dfrac{V}{L}$

Knowing Peak $i(t)$ is important for specifying ratings

use to select buck L

$$L = \frac{V_g - V}{2\,\Delta i_L} DT_s$$

↑ current ripple
10% - 20% full load
Val of DC component I

stopped on pg 9

Site used: https://www.analogictips.com/pulse-width-modulation-pwm/

Research on PWM

- Is the switch for converter, releasing + stoping the charge?

how to implement?

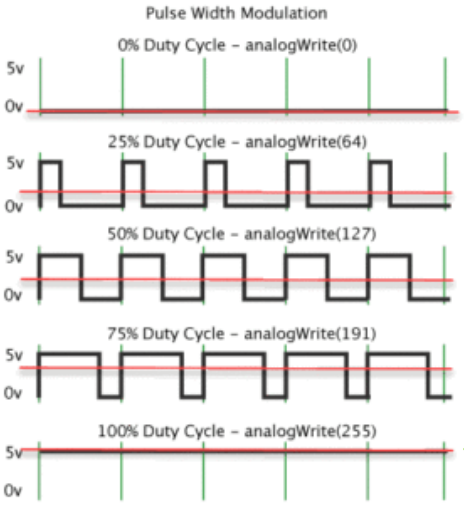


Figure 1: An example of a PWM signal shown at several duty cycles and a high voltage level of 5 volts. The red line is the average voltage that the driven device (e.g., a motor) is experiencing. (Source: **Timothy Hirzel**,)

Pulse is on 50% of the time

calculating duty cycle could help w/ building converter

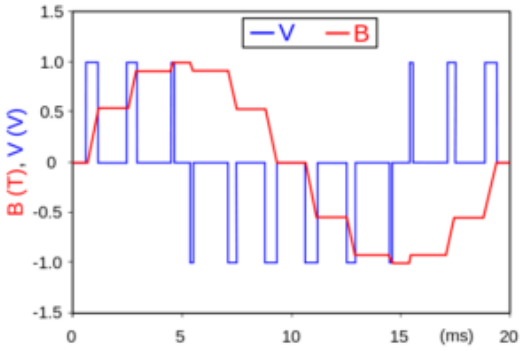


Figure 2: The blue lines are PWM output from an MCU, and the red line is the average voltage. In this case, the pulse width (and corresponding duty cycle) change so that the average voltage looks more like an analog output that is not in a steady state such as shown in Figure 1. (Source: **Zureks – Own work, CC BY-SA 3.0**,)

$$V_{Avg} = V_{high} \times D$$

V_{Avg}: Instantaneous V

D: Instantaneous D

"we use duty cycle + frequency to describe PWM"

ECE Embedded syst notes

$$D = \frac{\text{on Time}}{\text{Period}}$$

$T = \frac{1}{f}$

need timer for PWM

Prescalar P

Load val N

$$N \times P = \frac{f_s}{f_{PWM}}$$

f_s ← syst

f_{PWM} ← PWM

PWM for syst counts to when indext is @ value then switches counts till discharge

Interupts set so when timer exp, triggers?

Proj 3 had PWM I think

Toggle/reset or Toggleset?

Inductors Review

$$V(t) = L\frac{di(t)}{dt}$$

$$L = \frac{\Phi_B}{I}$$

Φ_B ← magnetic flux

I ← current

We are making a Toroidal Inductor I think?

$$L = \frac{N^2 \mu A}{l}$$

# of turns

$\mu_r \mu_0$ ← relative perm

l ← Avg length of coil

A ← Area of coil in m^2 (πr^2)

$$L = 0.01595N^2\left(D - \sqrt{D^2 - d^2}\right)$$

- L = inductance (μH)
- d = diameter of coil winding (in)
- N = number of turns
- D = 2 * radius of revolution (in)

$$L \approx 0.007975\frac{d^2 N^2}{D}$$

- L = inductance (μH)
- d = diameter of coil winding (in)
- N = number of turns
- D = 2 * radius of revolution (in)

approx when $d < 0.1D$

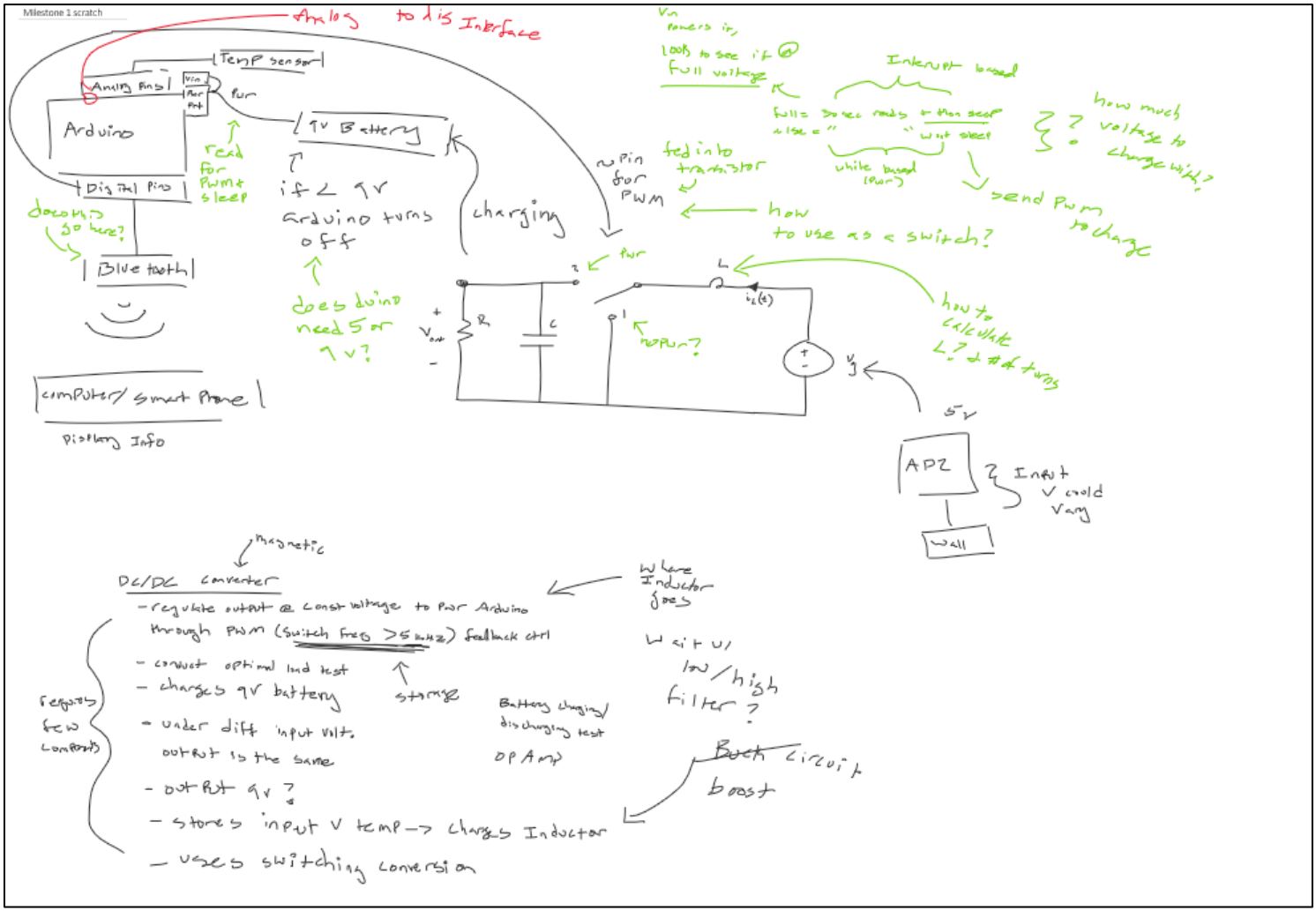
Img from wikipedia

$$L \approx \frac{\mu N^2 A}{2\pi r}$$

A = cross-sectional area

r = toroid radius to centerline

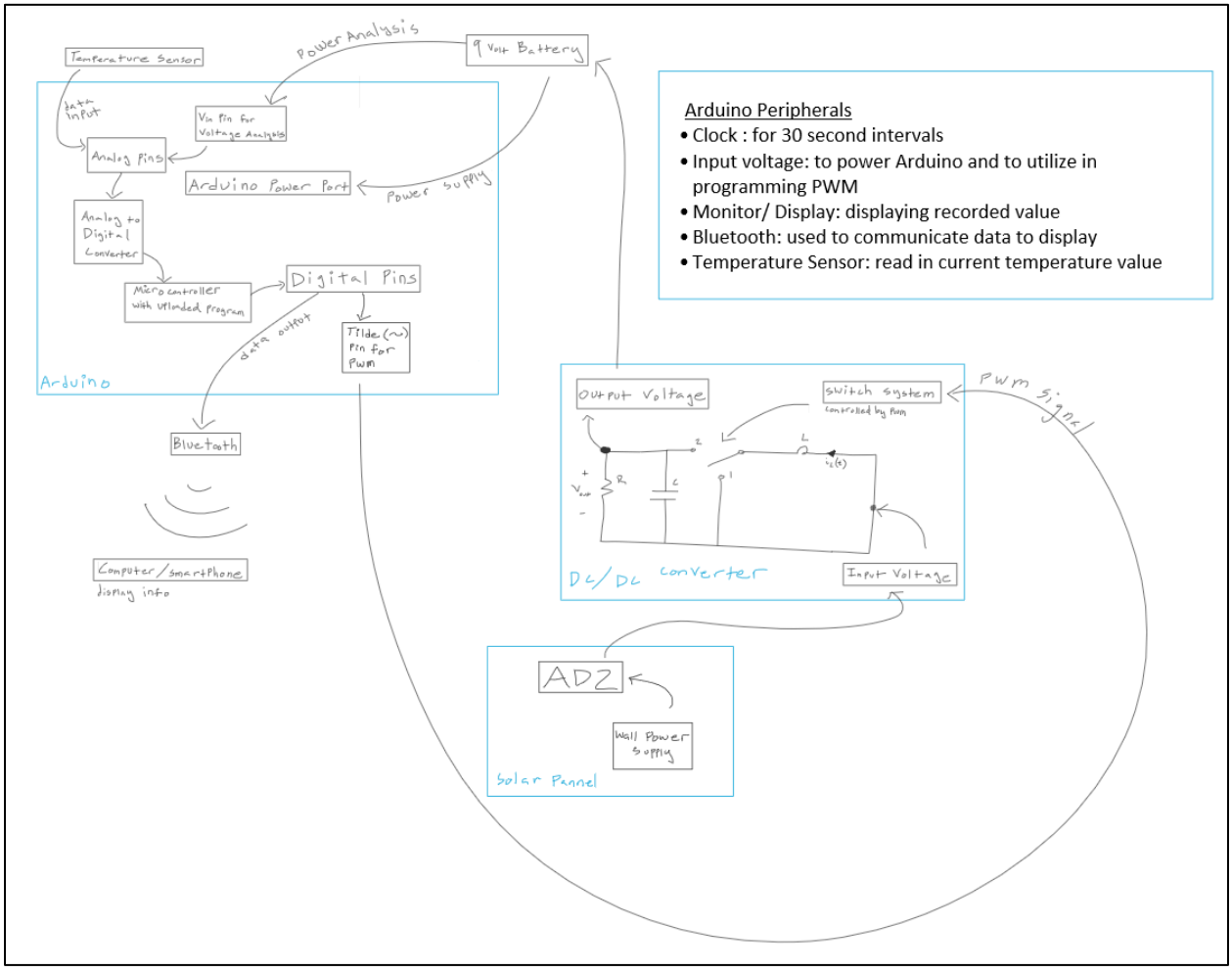
491 × 94



7/15/21
5:20pm- 7:20

Useful for programming Bluetooth: https://www.geeksforgeeks.org/all-about-hc-05-bluetooth-module-connection-with-android/

Talked with partner and finalized report. Made clean version of block diagram and wrote part of the outline



7/20/21

5:10pm – 7:00pm

Today's Goals

- Calculate Values for DC/DC converter
  - Inductor Specifics
  - Resistor + cap values
  - Figure out switch
- Simulate in LT spice
  - Test behavior
  - optimal Load test

$V_L = V_g - V$

$V_L = 5 - 9 = -4$ ← shouldn't be −?

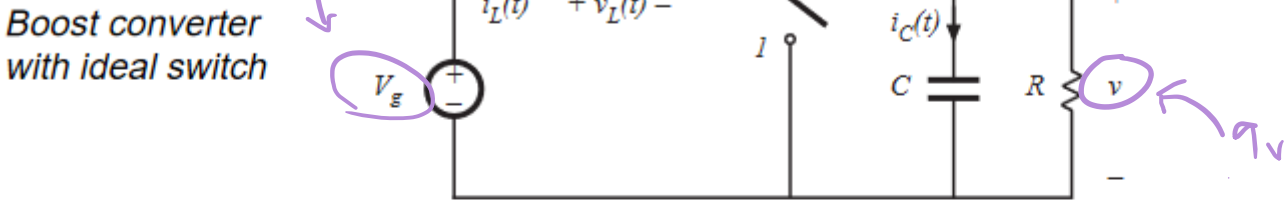
$L_{min} = \frac{D(1-D)^2 R}{2f}$ (mH; f ← freq; D ← duty cycle)

$V = V_g D$ ← duty cycle?

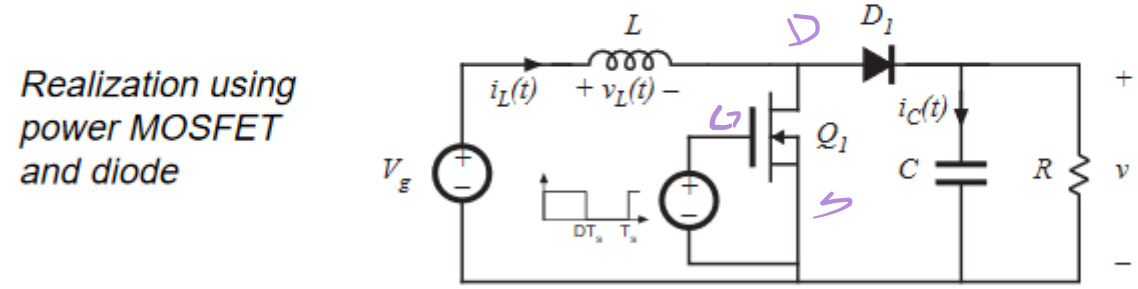
do we need a filter?

switching 5 kH

4 to 5 V



9 V



Average Current in the Boost Inductor

$$L \approx \frac{\mu N^2 A}{2\pi r}$$

491 × 94

A = cross-sectional area

r = toroid radius to centerline

- Average input current and average inductor current are equal
- Two basic ways to calculate I_{IN} and I_L:

$$I_L = I_{IN} = \frac{I_O}{(1-D)\times\eta_{EST}}$$

$$I_L = I_{IN} = \frac{I_O \times V_O}{V_{IN} \times \eta_{EST}}$$

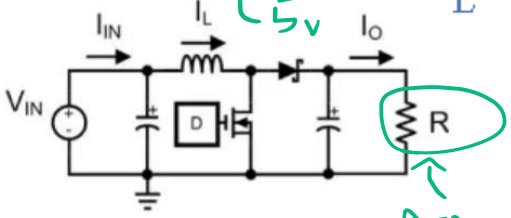
9V

5V

$$\Phi = BA\cos\theta$$

$$V_L = N\frac{d\Phi}{dt} = \frac{\mu N^2 A}{\ell}\frac{di}{dt}$$

- Use 85-90% for η_{EST}
  - Non-sync
- Use 85-95% for η_{EST}
  - Synchronous



Any size?

$$V(t) = L\frac{di(t)}{dt}$$

$$L = \frac{\phi_B}{I}$$

magnetic flux

current

$$-\frac{d\Phi_B}{dt} = v_L = L\frac{di_L}{dt}$$

ask later

V_{max} L?

assume $V_L = -4$

$$I_L = \frac{V_L}{X_L} = \frac{V}{2\pi f L} = \frac{V}{2\pi\left(\frac{\mu N^2 A}{2\pi r}\right)} = \frac{I_O \times V_O}{V_{in}\times\eta_{est}}$$

?

Inductance

How do we calculate L, C, + R?

$$V_o = \frac{V_{in}(1-\alpha)}{\frac{r}{R} + (1-\alpha)^2}$$

$$\frac{V_o}{V_{in}} = \frac{1}{1-\alpha}$$

$$\frac{1}{5} \Rightarrow \alpha \approx 0.04$$

$$T_{on} = \alpha T$$

$$\alpha_{max} = 1 - \sqrt{\frac{r}{R}}$$

$$1-\alpha = \sqrt{\frac{r}{R}}$$

$$I = \frac{V}{X_L}$$

where V is the rms voltage ac

$$X_L = 2\pi f L$$

Self Inductance –the Maths

- When self inductance occurs the flux is proportional to the current in the inductor

$$\Phi = L \times I$$

where:

Φ = magnetic flux (Wb)
L = self-inductance (H)
I = current (a)

- Substituted into Faraday's law;

$$V = \frac{-L\Delta I}{\Delta t}$$

- Units for self-inductance is the Henry (symbol H)

https://www.powerelectronicsnews.com/the-dc-dc-boost-converter-power-supply-design-tutorial-section-5-1/

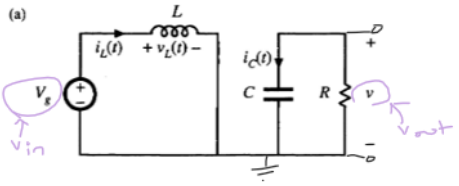
7/21/21
2:00pm – 3:06pm

Reading PDF so I can try a bit more to figure out converter.

OP1:

switch in pos 1:

charges Inductor, cap discharges

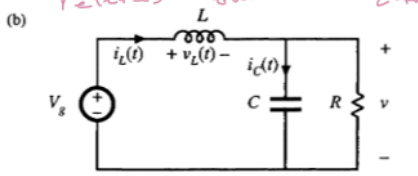


$V_L = V_g$ $\quad i_C = -\frac{v}{R}$

ripple about $v \approx V$

switch in pos 2:

releases Inductor charge, cap charges



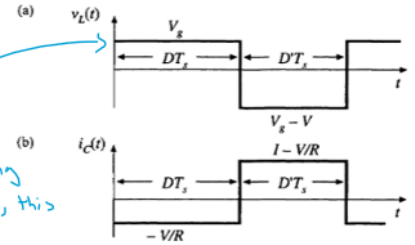
cap volt $= V_g + V_L$?

$V_L = V_g - V$

$i_C = i_L - \frac{V}{R}$

$v \approx V$, $i_L \approx I$ $\Rightarrow$ $V_L = V_g - V$, $i_C = I - \frac{V}{R}$

Eqn used to sketch Inductor + cap wave forms



according to text, this implies $V_{out} > V_g$

output / input ie duty cycle

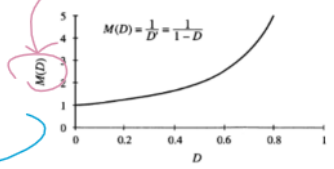


Fig. 2.16 Dc conversion ratio $M(D)$ of the boost converter.

The total volt-seconds applied to the inductor over one switching period are:

$$\int_0^{T_s} v_L(t)dt = (V_g)DT_s + (V_g - V)DT_s$$

T_s is the Period, ie $\frac{1}{f}$

How to use? are important?

Instructor Yu Conversation

(0.1)(700mA)

similar eqn for cap

$\Delta i_L = \frac{V_g}{2L} D T_s$

V_{in}, switching freq, Inductor, Duty cycle

Inductor current ripple

See text for details but basically pick a value smaller than that

$T_s = \frac{1}{sf}$ Proj description: $f_s > 5KHz$

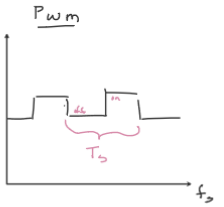
$\frac{T_{on}}{T_s} = D$

duty cycle varies b c $V = \frac{V_g}{D'} = \frac{V_g}{1-D}$ $\leftarrow D' = 1 - D$

Proj description: $4_V \leq V_g \leq 5_V$

If input voltage varies $\rightarrow$ D varies s.t. output voltage is constant

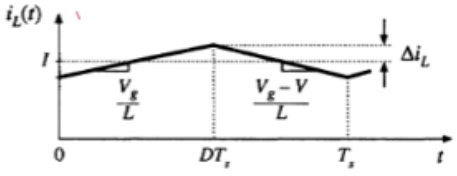
Pwm



If we assume V_g is constant then D doesn't vary

7/21/21
5:00pm – 7:00pm

Fig. 2.18 Boost converter inductor current waveform $i_L(t)$.



$$\Delta V = \frac{V}{2RC} D T_s$$

Voltage ripple peak magnitude

capacitor

R_{load}

References for finding Δi_L

https://ridleyengineering.com/design-center-ridley-engineering/39-magnetics/66-031-choosing-the-inductor-for-a-buck-converter.html

https://www.instructables.com/Using-the-Power-Supplies-With-the-Analog-Discovery/

Δi_L % range from 2.5% to 50% → 10% Pretty commonly used

no more than 3%

ADZ has:

Current max of 700mA

Voltage max of 5v

found in waveforms

Pick a decent duty cycle

Calculations

Inductor

$$\Delta i_L = \frac{V_{in}}{2L} DT_s$$

$$L = \frac{V_{in}}{2\Delta i_L} DT_s$$

$$L = \frac{(5v)}{2(0.03)(700mA)} (0.5)\left(\frac{1}{5.5kHz}\right)$$

$$L = 0.0108\ H$$

make sure ADZ can handle

Capacitor

close to 0.

$$\Delta V = \frac{V_{out}}{2RC} D T_s$$

$$C = \frac{V_{out}}{2R\Delta V} D T_s$$

$$C = \frac{(9v)}{2(1.5k\Omega)(0.03)(5)} (0.5)\left(\frac{1}{5.5kHz}\right)$$

$$C = 1.8 \times 10^{-6} = 1.8\mu F$$

choosing 2.2μF from kit

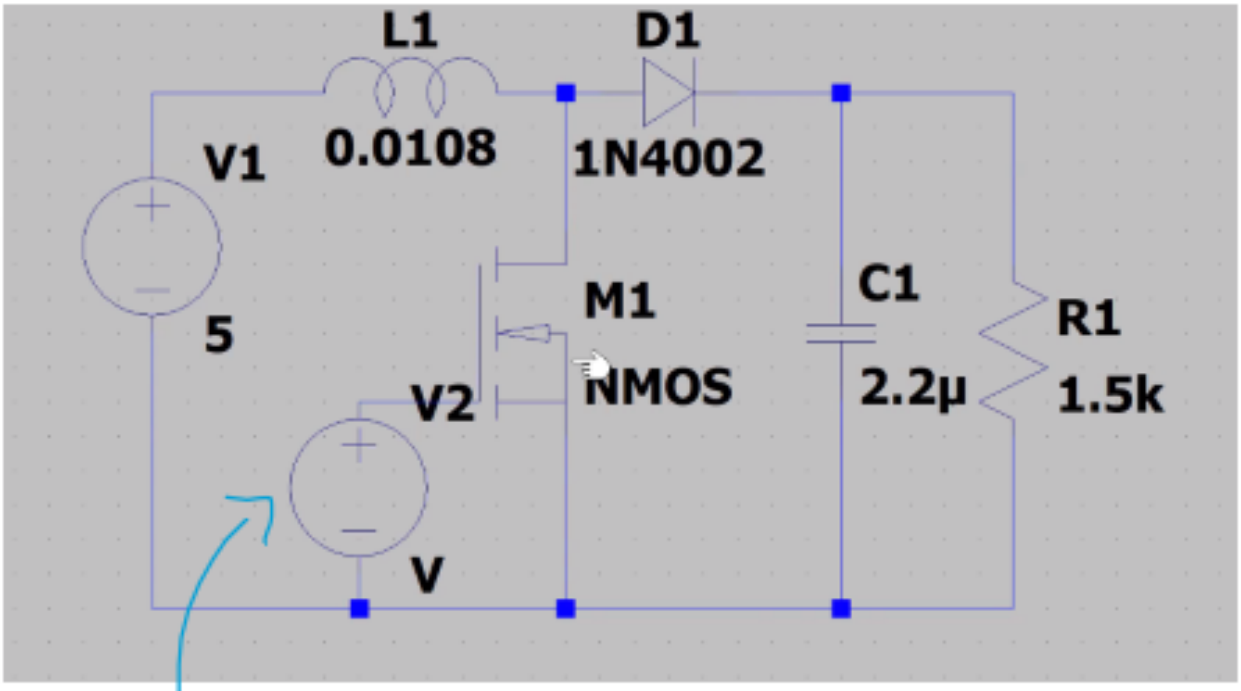
choosing $R = 1.5k\Omega$

LT spice

1N4001 ← diode

ZN7000A ← N-mosfet

How to make work



is PWM

7/23/21
3:30pm – 11:00pm (some breaks taken)

Worked by self to make changes to LT Spice recommended. Also worked on refining block diagram for report and creating the Optimal Load Test graph. Uncertain if calculations are correct. Will check later.

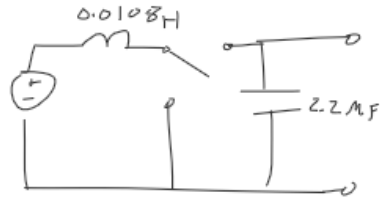
Efficency curve based on load (R_{LMS})

"Sweep" across different R_L

Plot Ω vs. Pwr

$R_L = R_S$ is where max Power transfer is usually

need to find R_S for our circuit



$$X_c = \frac{1}{j\omega C}$$

$$X_L = j\omega L$$

$$\omega = 2\pi f$$

$$f = 5.5\,kHz$$

$$\omega = 2\pi(5.5\times10^3) = 34557.5$$

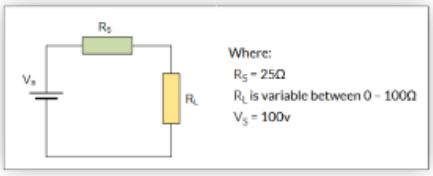
$$X_c = \frac{1}{j(34557.5)(2.2\,MF)} = -13.153j \quad \text{note } \frac{1}{i} = -i$$

$$X_L = j(34557.5)(0.0108_H) = 374.00j$$

$$R_S = 374.00j + 13.153j = 387.15j$$

$$|R_S| = \sqrt{(0)^2 + (387.15)^2} = 387.15$$

Maximum Power Transfer Example No1



$$I = \frac{V_S}{R_S + R_L} \quad \text{and} \quad P = I^2 R_L$$

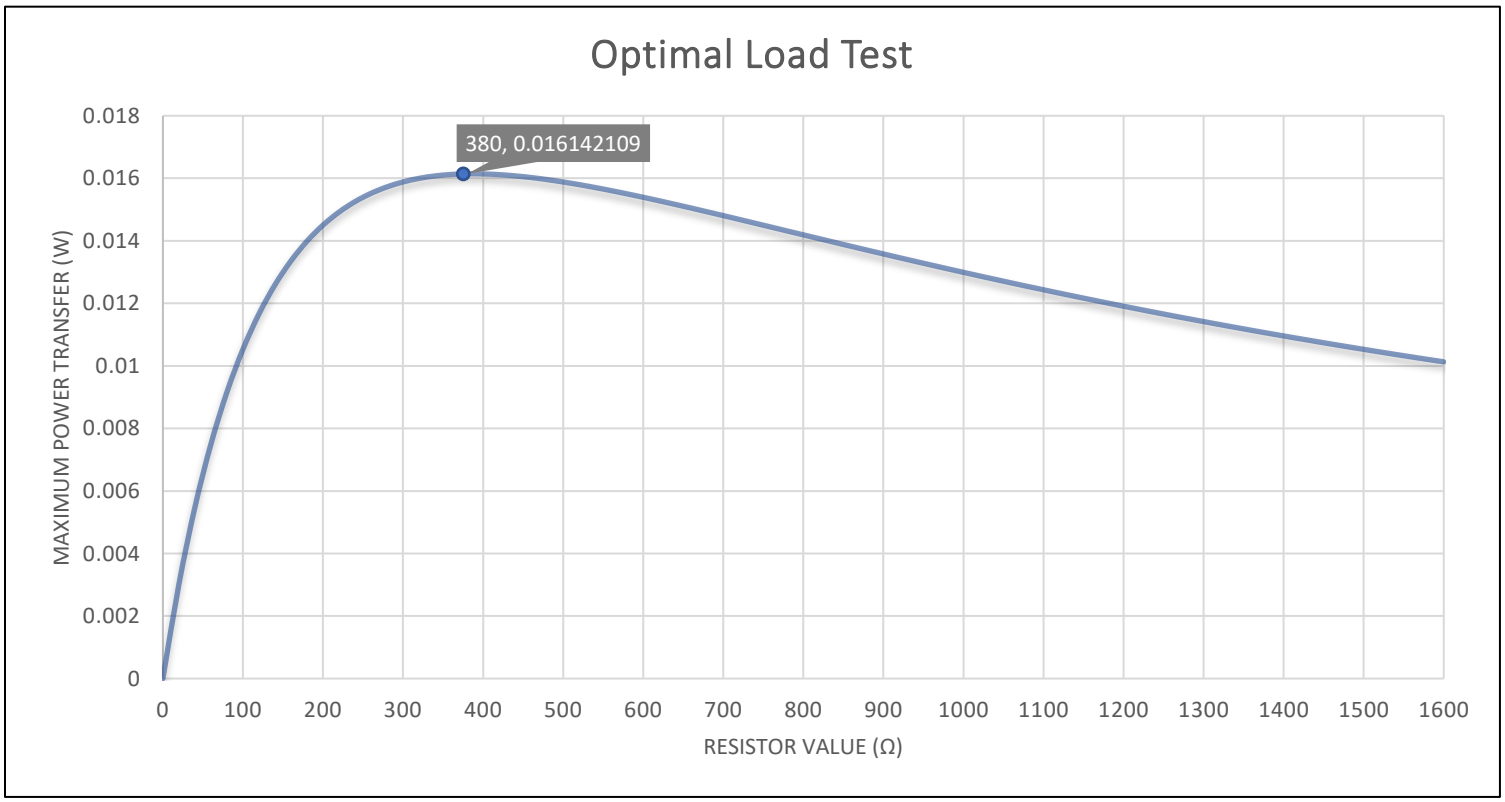
https://www.electronics-tutorials.ws/dccircuits/dcp_9.html

For better understanding

AD2 has max bandwidth of 12 MHz so 5.5 kHz is fine

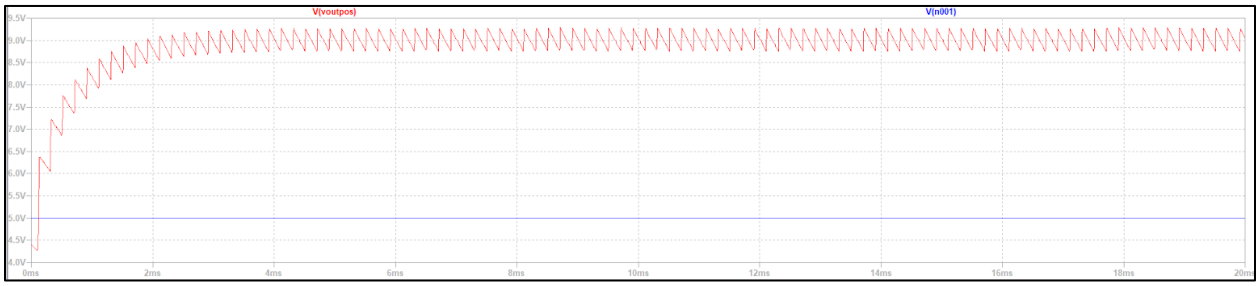
$$|Z_e|^2 = R^2 + X^2$$

real, Imag, imp. magnitude



7/24/21
3:00pm – 7:00pm

Just refined math, redid calculations and wrote final report



Calculations

Inductor

$\Delta i_L = \frac{V_{in}}{2L} DT_s$

$L = \frac{V_{in}}{2\Delta i_L} DT_s$

$= \frac{(5V)}{(2)(100mA)} (0.5)\left(\frac{1}{20\ kHz}\right)$

$L = 1.79 \times 10^{-4}\ H_Z = 179\ \mu H$

Choosing 162 uH based on LT spice

Capacitor

$\Delta V = \frac{V_{out}}{2RC} DT_s$

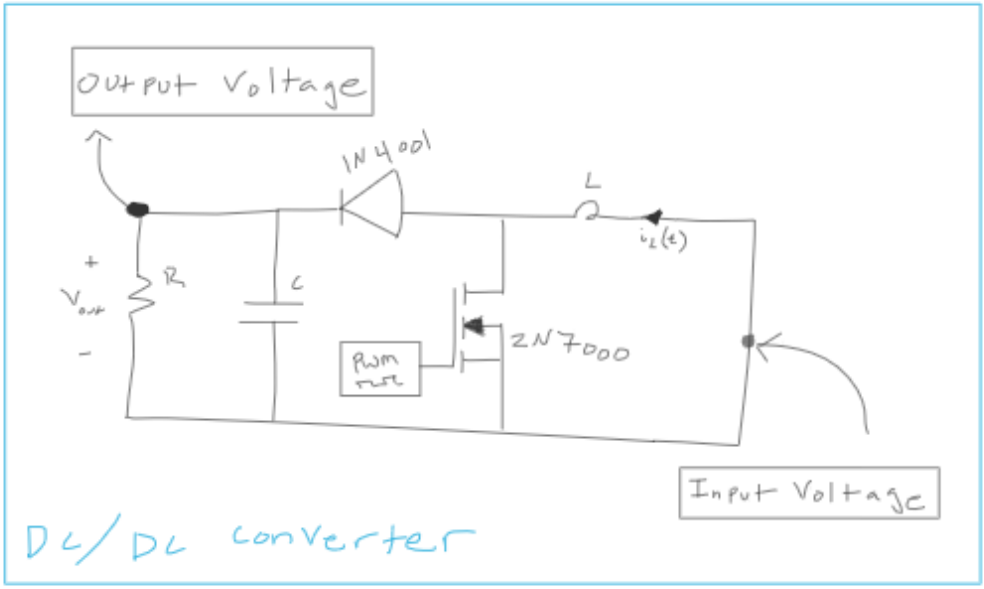
$C = \frac{V_{out}}{2R\Delta V} DT_s$

$C = \frac{(9v)}{2(1.5_{k\Omega})(0.03)(5)} (0.5)\left(\frac{1}{5.5kHz}\right)$

$C = 1.8 \times 10^{-6} = 1.8\ \mu F$

chosing 2.2uF from kit

chosing R = 1.5 kΩ



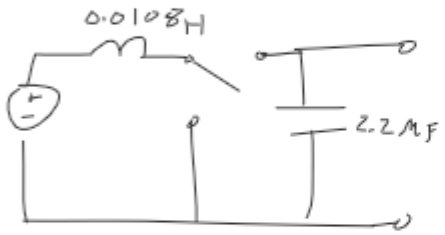
Efficency curve based on load (R_{load})

"Sweep" across different R_L

Plot Ω vs. Pwr

$R_L = R_s$ is where max Power transfer is usually

need to find R_s for our circuit



$$X_c = \frac{1}{j\omega C}$$

$$X_L = j\omega L$$

$$\omega = 2\pi f$$

$$f = 5.5_{kHz}$$

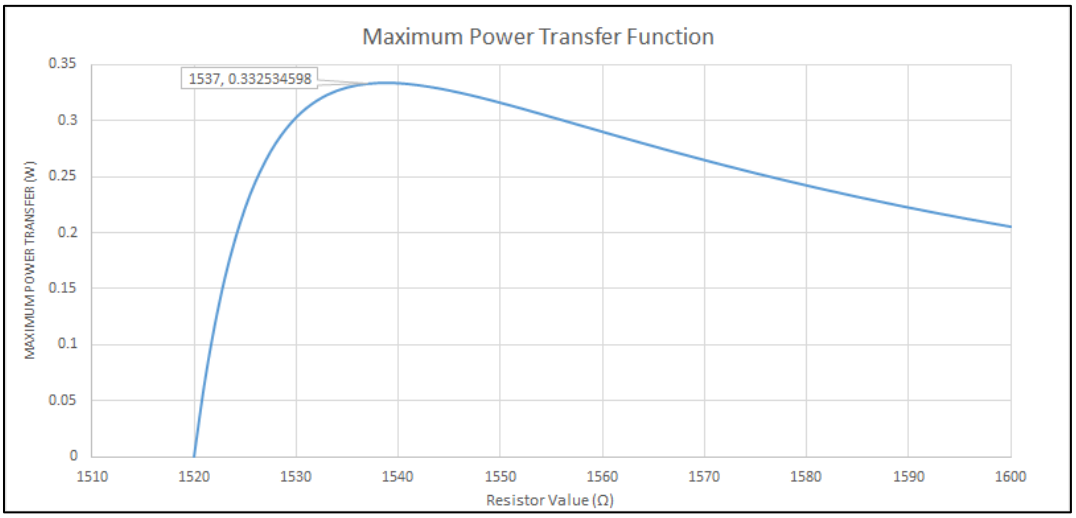
$$\omega = 2\pi(5.5 \times 10^3) = 34557.5$$

$$X_c = \frac{1}{j(34557.5)(2.2\,MF)} = -13.153j$$ note $\frac{1}{j} = -j$

$$X_L = j(34557.5)(162\,mH) = 5.5\,j$$

$$R_s = 5.5\,j + 13.153j = 18.75j$$

$$|R_s| = \sqrt{(0)^2 + (18.75)^2} = |18.75j|$$



7/28/21
5:30pm – 9:00pm

Arduino Research notes

uses simplified C++

major Arduino components

- usb connector → used to load code (can pwr)
- Pwr port → Pwr port (ops @ 5 v)?

} might need to reconsider pwr idea from earlier

- Analog Pins → (6 pins) temp sensor. only measure volt.
- Digital Pins → In or out (read comp. sig)

can be reconfig as dig

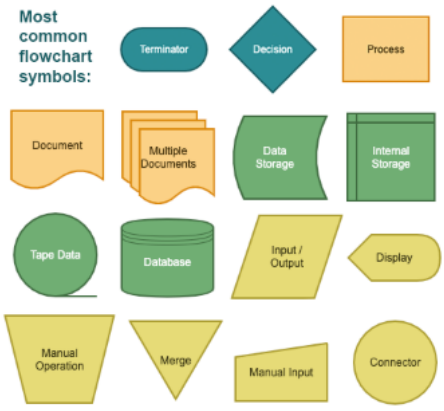
- Reset switch → run from start
- crystal osillator → ticks, math stuff
- TX RX Leds → Transmit, recive data
- micro controller → Brain (has flash mem, ram 32kB 2kB)
- USB Interface chip → signal translator

~ (tilde) pins can be used for PWM (simulate analog out)

Goals for hardware: ***DC/DC converter should be able to recharge the 9V battery when the battery is less than fully charged. The DC/DC converter should also have PWM feedback control.***
Goals for software: ***The Arduino should be able to communicate with the DC/DC converter through PWM, and the Arduino should operate based on the charge of the 9V battery.***
Deliverables:
Deliverable 4: Show battery charging and discharging waveforms
Testing outline:
Checking the waveforms as well as if the Arduino is working properly to test out the charging and discharging of the battery

Flow chart

- Pwm
- 2 states
- how to switch between states



https://www.gliffy.com/blog/guide-to-flowchart-symbols

https://create.arduino.cc/projecthub/muhammad-aqib/arduino-pwm-tutorial-ae9d71

https://www.arduino.cc/en/Tutorial/SecretsOfArduinoPWM

Pwm values (from ltspice)

$f = 32\ kHz$
$T_s = 31.25\ \mu$
$T_{on} = 13.8$
$D = 44.16\ \%$

Question

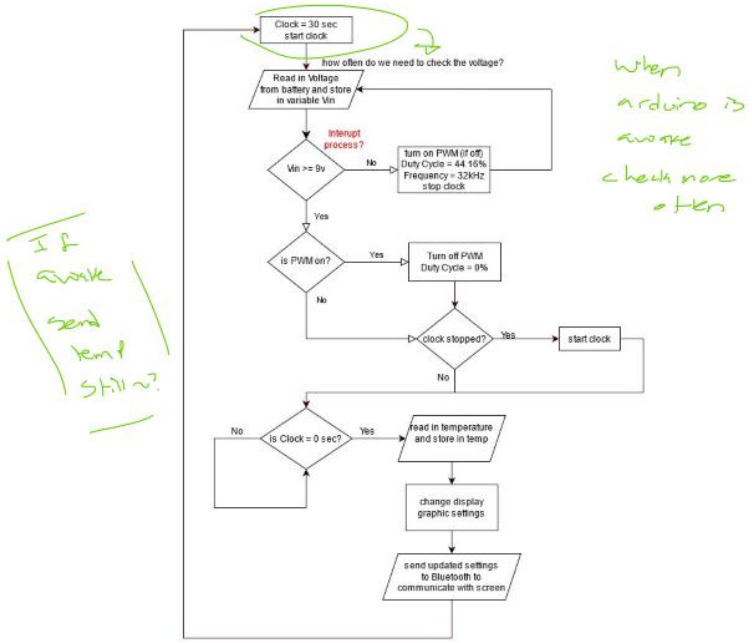
If in charging state does it still need temp?

To dos

- flowchart
- Physical circuit

7/29/21
12:00pm – 2:00pm

Symbol	Symbol Name	Description
⟶ ⟵	Flow Lines	Used to connect symbols
⬭	Terminal	Used to start, pause or halt in the program logic
▱	Input/output	Represents the information entering or leaving the system
▭	Processing	Represents arithmetic and logical instructions
◇	Decision	Represents a decision to be made
○	Connector	Used to join different flow lines
▭	Sub function	used to call function



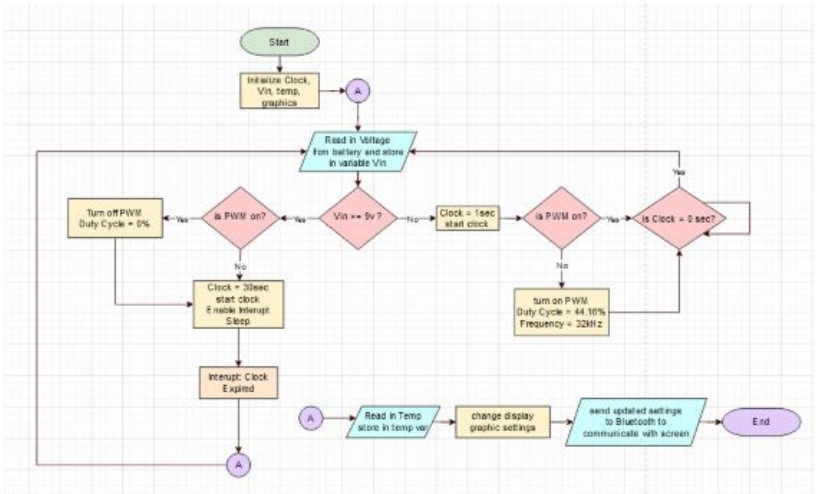
If awake send temp still ~?

when arduino is awake check more often

Researching PWM + Battery charging

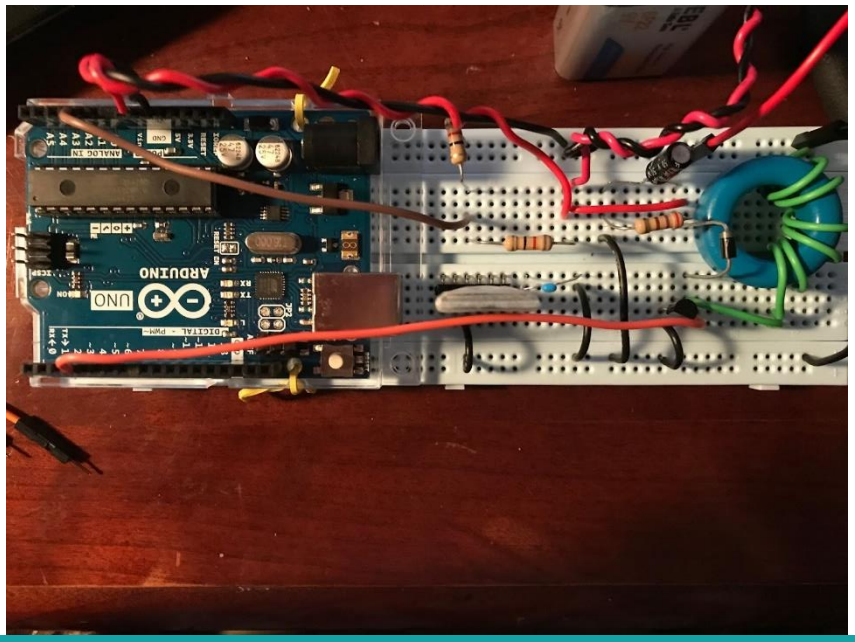
- need an interupt system — for sleepmode,

does it still checks voltage when in sleep mode?



7/30/21
6:00pm – 9:00pm

Built physical circuit while Saashi wrote the program



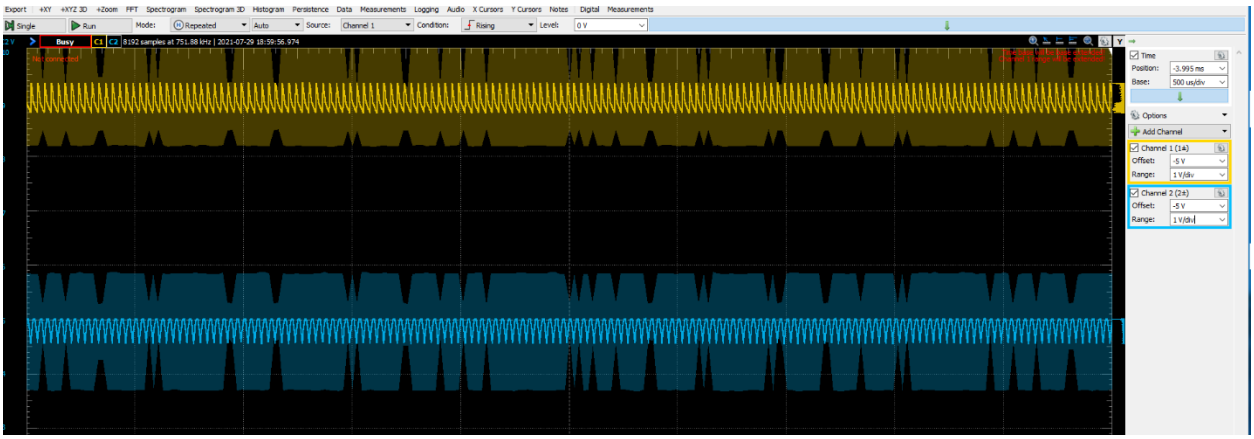
PWM_code

```
int analogPin = A2; //analog pin that will be connected to voltage divider with Vin
int charge = 0;
bool pwmOn = false;
void setup() {
  // put your setup code here, to run once:
  TCCR2B = TCCR2B & B11111000 | B00000001; //for PWM frequency of 31372.55 Hz
  pinMode(3, OUTPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  charge = analogRead(analogPin);
  if (charge >= 4.5) {
    if (pwmOn) {
      analogWrite(3, 0);
      pwmOn = false;
    }
    //start clock
    //send to sleep every thirty seconds
    //read in temp, change display graphic settings, send updated settings to bluetooth
  }
  else {
    //start clock
    if (!pwmOn) {
      analogWrite(3, 113); //duty cycle * 255 = .4416 * 255
      pwmOn = true;
    }
    //while clock has not run out of time don't read voltage
  }
}
```

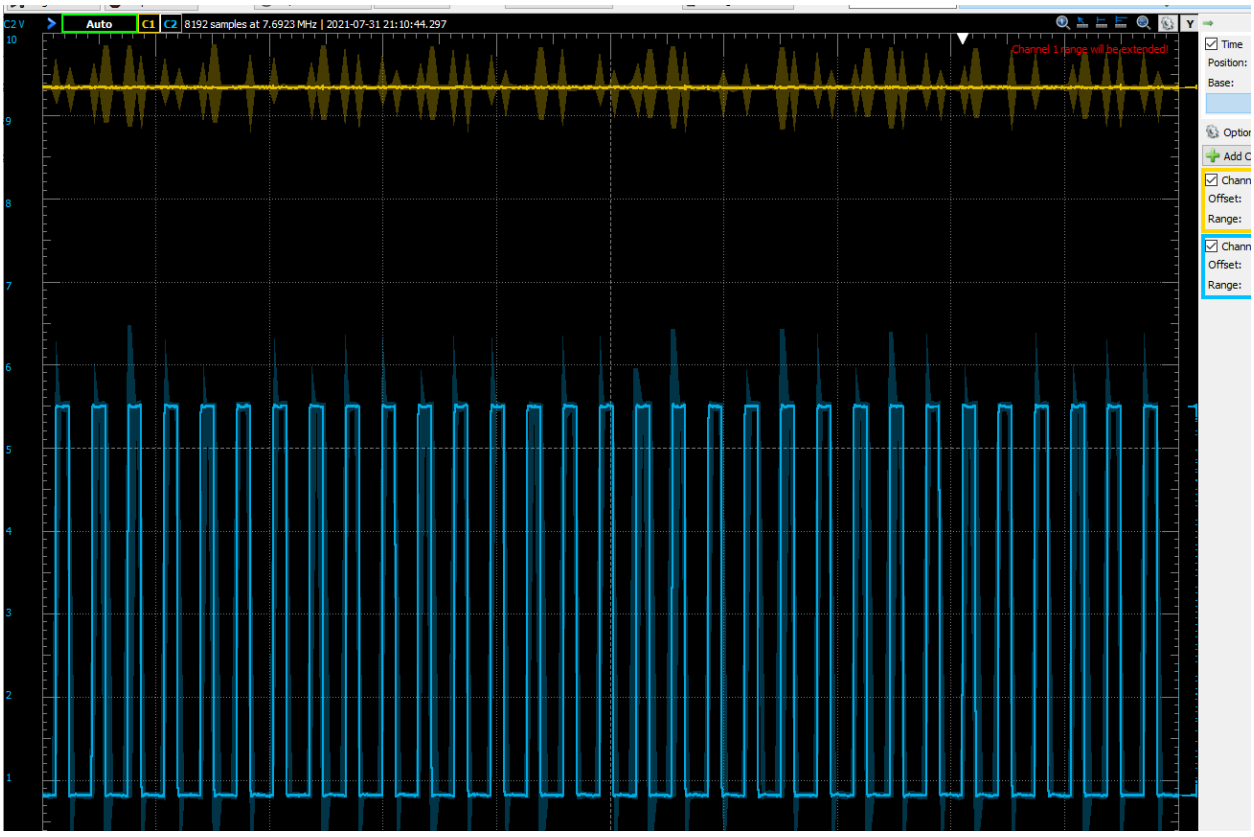
Output of circuit with a waveform generator wave being used as a substitute PWM

(blue is Vin, Yellow is Vout)



7/31/21
6:00pm – 11:00pm

Final testing of physical circuit done in waveforms below blue is the Arduino's PWM, yellow is the output voltage charging the battery. Milestone report was written and submitted. This was submitted



later

```
3.76 is >= 9V
3.75 is >= 9V
3.76 is >= 9V
3.75 is < 9V
3.75 is >= 9V
3.75 is >= 9V
3.76 is >= 9V
3.75 is >= 9V
3.75 is < 9V
3.75 is < 9V
3.75 is >= 9V
3.75 is >= 9V
3.76 is >= 9V
3.76 is >= 9V
3.75 is < 9V
3.75 is
```

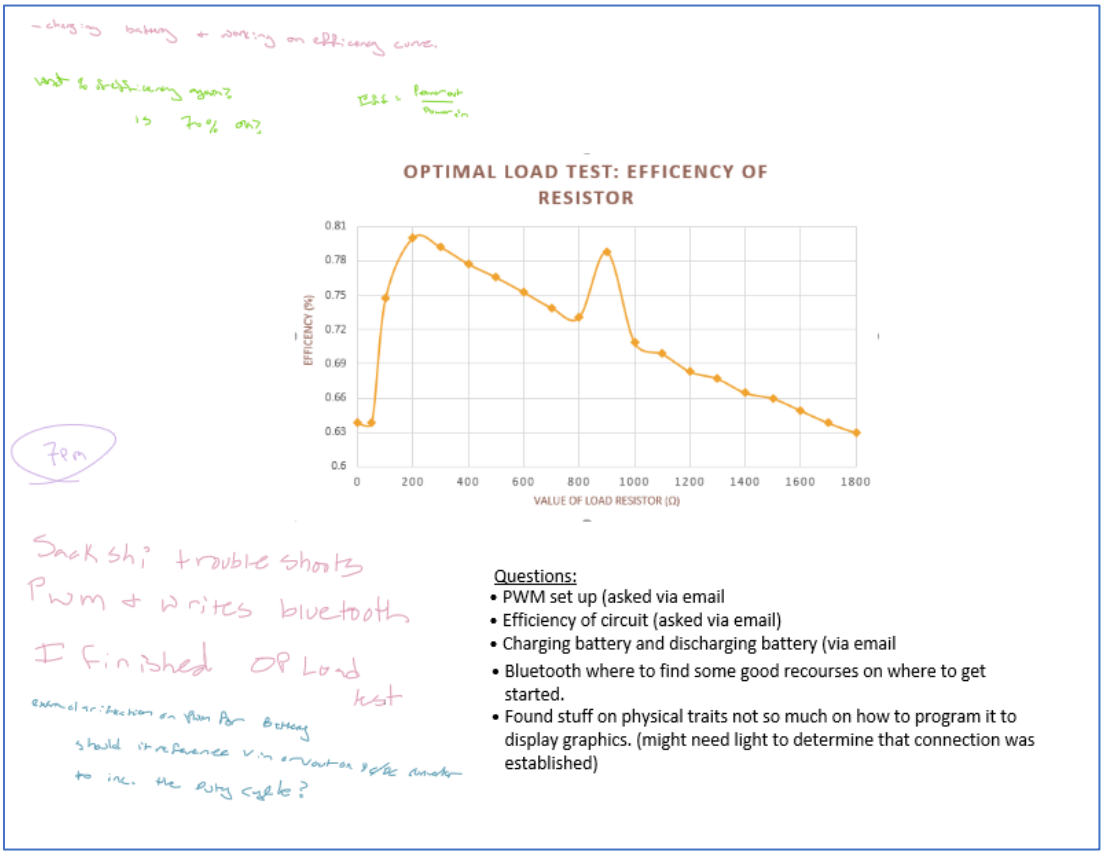
8/2/21
12:00pm – 2:00pm

12:00pm 8/2/21

To Do's
- Last steps
- fix curve
- feedback? / charging curve
- code changes?

Started setting up excel for Optimal Load

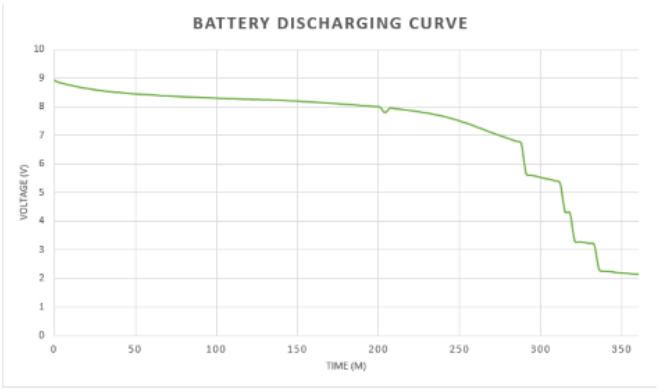
8/3/21
2:00pm – 8:30pm (some breaks taken)



Questions:
- PWM set up (asked via email)
- Efficiency of circuit (asked via email)
- Charging battery and discharging battery (via email)
- Bluetooth where to find some good recourses on where to get started.
- Found stuff on physical traits not so much on how to program it to display graphics. (might need light to determine that connection was established)

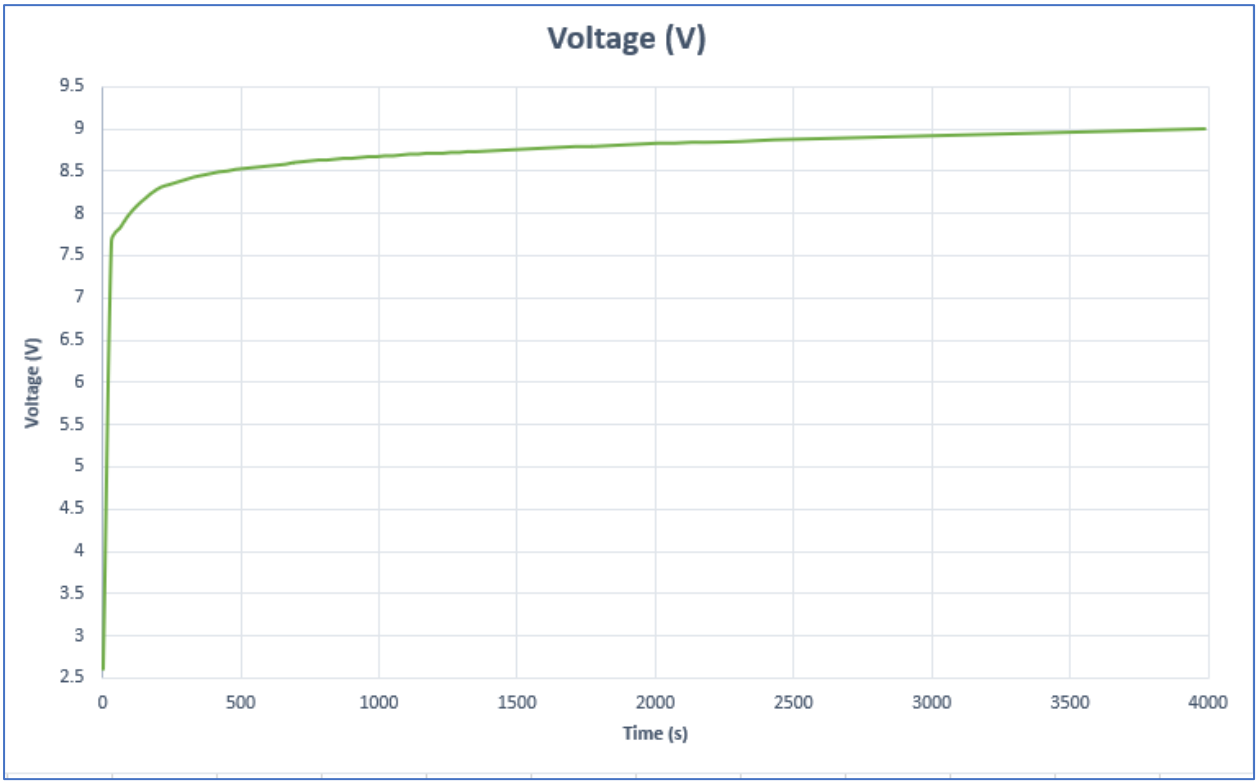
8/4/21
5:00pm – 10:30pm

Partner worked on PWM + Bluetooth code
while I built circuit + discharged
the battery



8/5/21
1:00pm – 2:30pm

just worked on battery charging curve



8/5/21
5:00pm – 9:30pm

started working on interrupt system. Partner was supposed to help, but she just kinda threw the work on me.

clock / interrupt search

https://donalmorrissey.blogspot.com/2010/04/putting-arduino-diecimila-to-sleep-part.html

https://donalmorrissey.blogspot.com/2011/11/arduinoatmega168-timers-for-sleeping.html

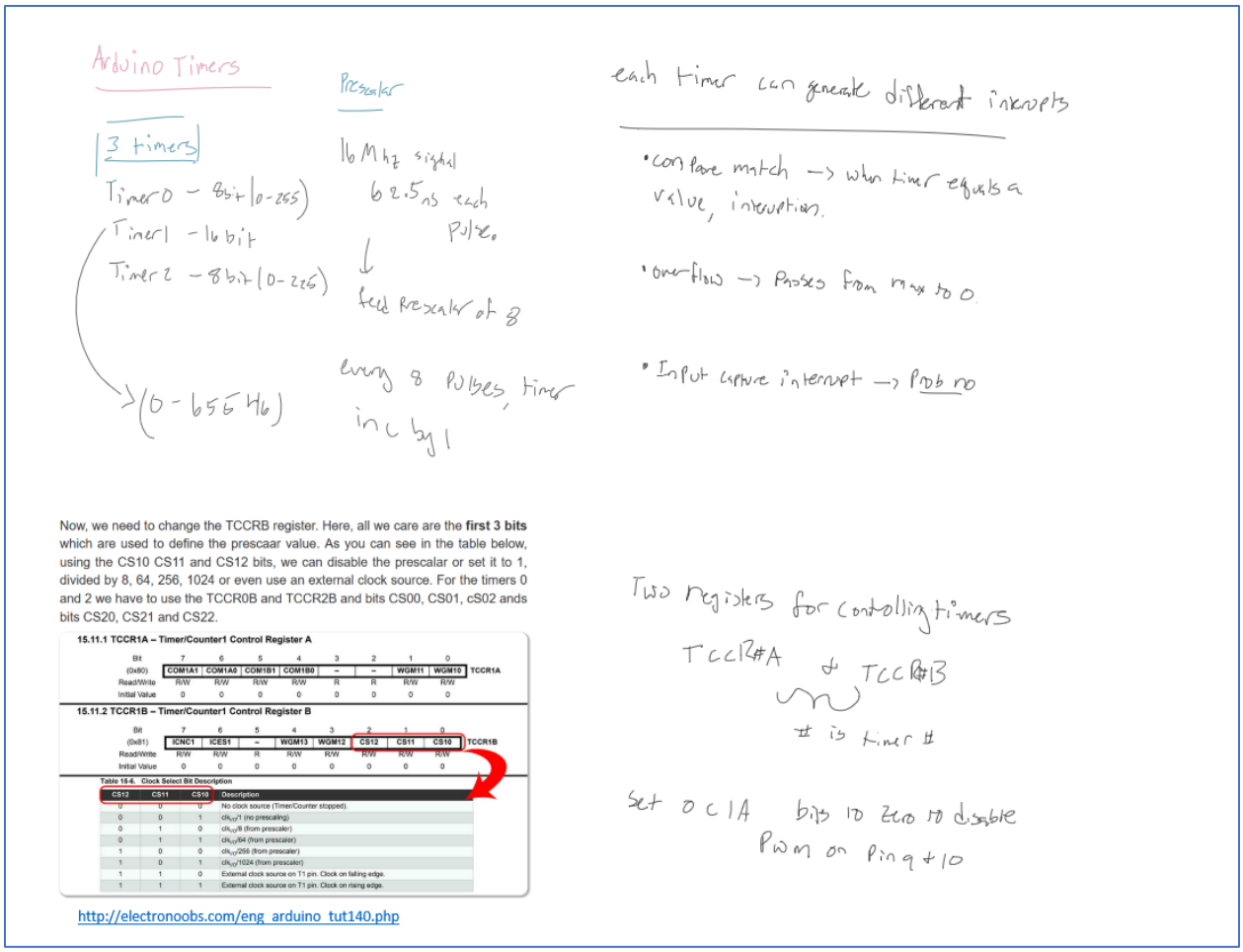
http://www.gammon.com.au/forum/?id=11497

https://thekurks.net/blog/2018/1/24/guide-to-arduino-sleep-mode


```
27 void Going_To_Sleep(){
28     sleep_enable();//Enabling sleep mode
29     attachInterrupt(0, wakeUp, LOW);//attaching a interrupt to pin d2
30     set_sleep_mode(SLEEP_MODE_PWR_DOWN);//Setting the sleep mode, in our case full sleep
31     digitalWrite(LED_BUILTIN,LOW);//turning LED off
32     delay(1000); //wait a second to allow the led to be turned off before going to sleep
33     sleep_cpu();//activating sleep mode
34     Serial.println("just woke up!");//next line of code executed after the interrupt
35     digitalWrite(LED_BUILTIN,HIGH);//turning LED on
36   }
```

```
38 void wakeUp(){
39   Serial.println("Interrrupt Fired");//Print message to serial monitor
40    sleep_disable();//Disable sleep mode
41   detachInterrupt(0); //Removes the interrupt from pin 2;
42 }
```

Image_3



8/6/21
5:00pm – 9:30pm

worked on code for interrupt and timer. Sent a few emails to ask questions while building it. I forgot to screen shot the code. I worked on a smaller file instead of the final code so I could test with an LED

8/7/21
6:00pm – 11:30pm

finished the program finally. Also wrote parts of the report and edited the entire thing. Added what is seen below to final code.

```
//figuring out interupts and timers

#include <avr/sleep.h> //do I need this?

#include <SoftwareSerial.h>
```

```
int LED = 13; //pin of LED being used

int timerDone = 0;

const uint16_t t1_load = 0; //counter value

const uint16_t t1_compareVal = 31250; //comparison val got from calulating prescalar of 1024 for timer of 2 sec



void setup() {

  Serial.begin(9600); //send and recive at 9600 BaudRate


  pinMode(LED, OUTPUT);//Set the pin to be OUTPUT for LED


  TCCR1A = 0; //resetting Timer1 register A to zero


  //setting prescalar to 1024

  TCCR1B |= (1 << CS12);

  TCCR1B &= ~(1<< CS11);

  TCCR1B |= (1 << CS10);


 //Reset Timer1 and set the compare value

  TCNT1 = t1_load;

  OCR1A = t1_compareVal;


  TIMSK1 = (1 << OCIE1A); //enable Timer1 compare interrupt


  sei(); //enable global interupts


}
```

```
//a method to put the arduino to sleep for 8 seconds
void SleepyTimeMode(){
  set_sleep_mode(SLEEP_MODE_IDLE); //set mode to full sleep mode
  sleep_enable(); //enabling sleep mode

  digitalWrite(LED, LOW);
  delay(1000); //short delay to turn off LED.

  sleep_mode(); //activate sleep mode
  sleep_disable();
  Serial.print("timer Expired \n"); //wake up here

}

void loop() {
  // put your main code here, to run repeatedly:

  // Serial.print("test\n");
  if (timerDone == 1){
    timerDone = 0;
    digitalWrite(LED, HIGH);
    delay (1000);
    for (int x = 0; x < 30; x++){
      SleepyTimeMode();
    }

  }



   //digitalWrite(LED,HIGH);  //Write new state to LED
   //delay(500);
```

```
}


//Interupt service for timer1
ISR(TIMER1_COMPA_vect) {


  TCNT1 = t1_load; //reset pin to zero
  //PORTB ^= ~(1 << LED);
 // digitalWrite(LED, HIGH);


 if(timerDone == 0){
    timerDone = 1;
 }


}
```

Last Week

There weren't any specific new developments beyond developing the power point presentation and trying to debug the project further. The final report was written and submitted. Definitely frustrated with partner who left me to implement at least 2/3 of the project.