Buck-Boost Converter Data

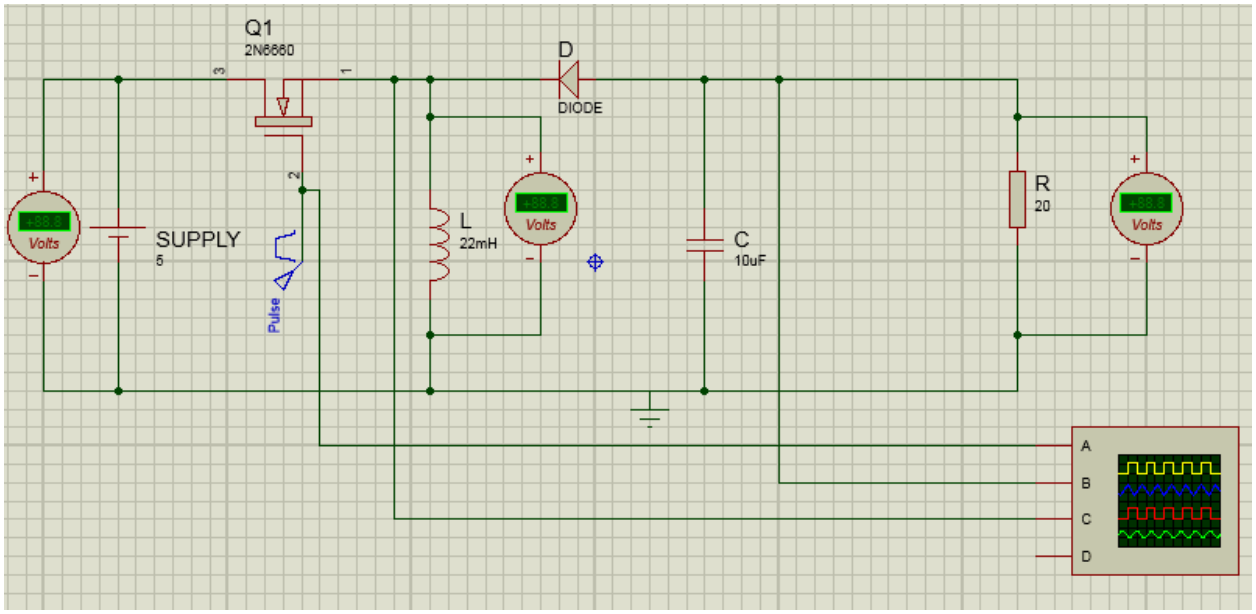
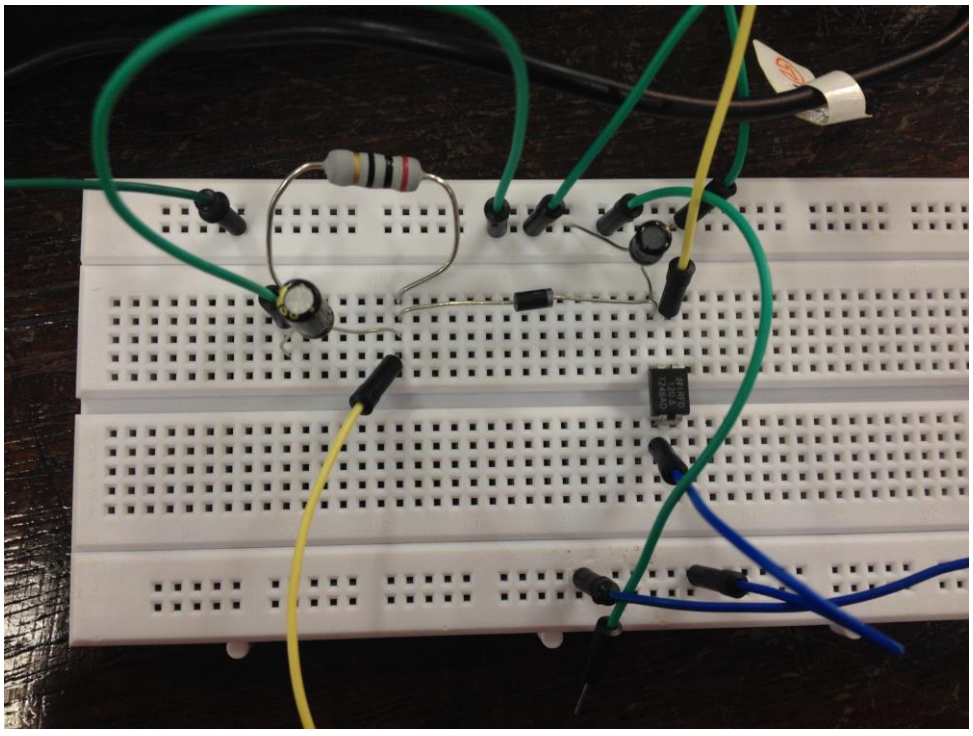


Figure 1: Proposed Buck-Boost converter circuit

The oscillator is connected to Pulse Generator (yellow), the voltages across inductor, V_L (red) and capacitor, V_C (blue).

V_D: 5V

V_G: 8V



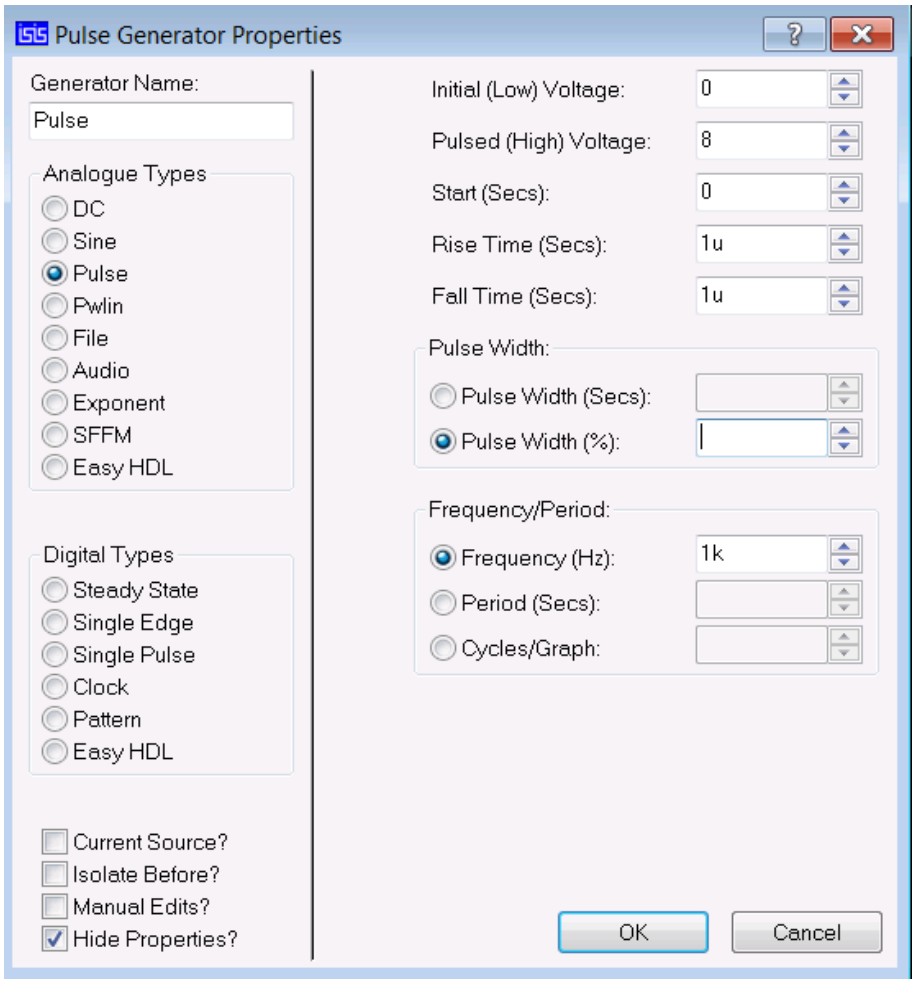
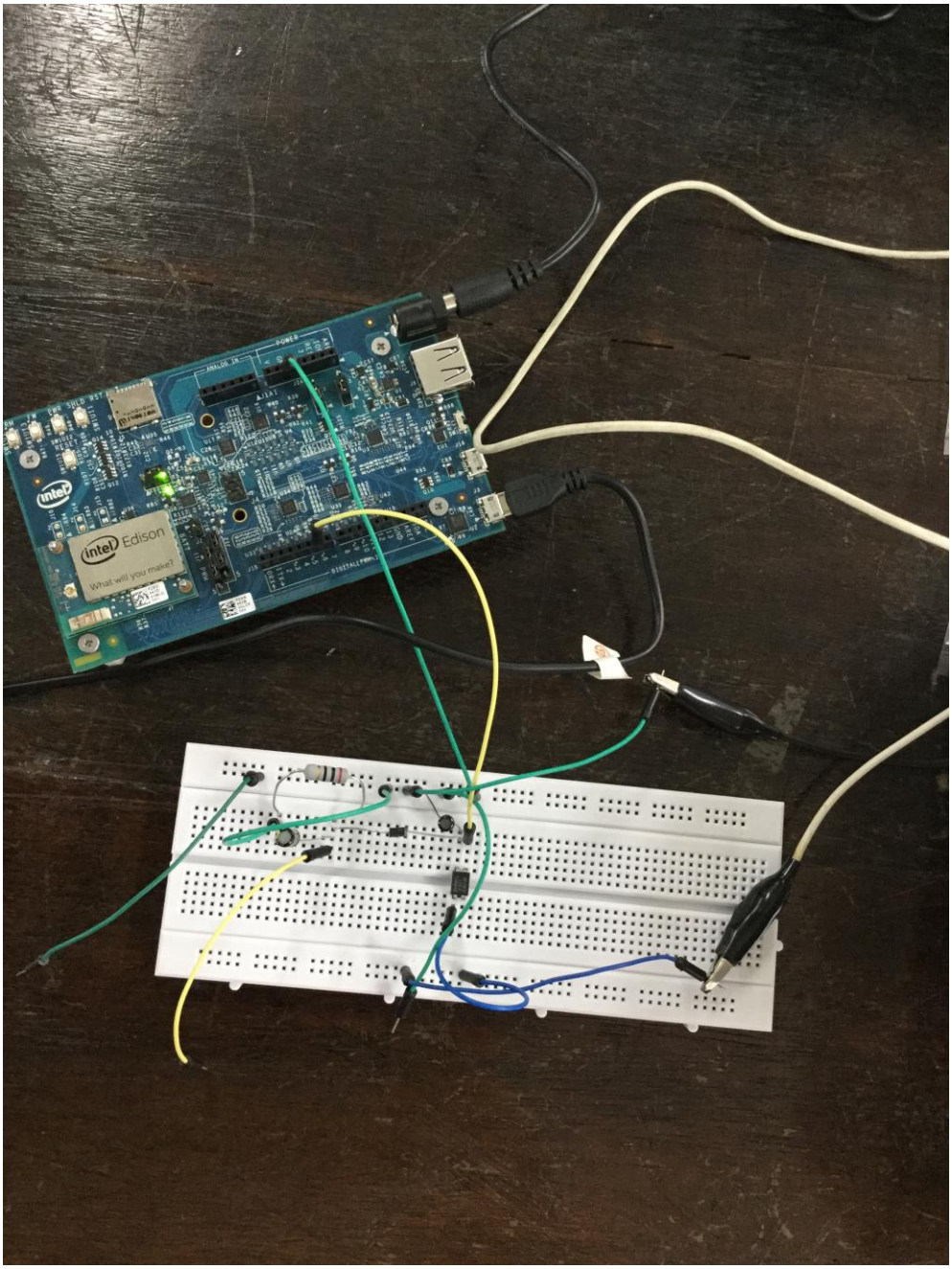


Figure 2: Values are set for of t_{rise}, t_{fall} and Frequency

Duty Cycle	V_L
30%	(Buck)
50%	8V
80%	(Boost)

Table 1: Output V_L for 30% of duty cycle at the gate

Circuit setup with Intel Edison



Activity 1: set 30% duty cycle Edison PWM

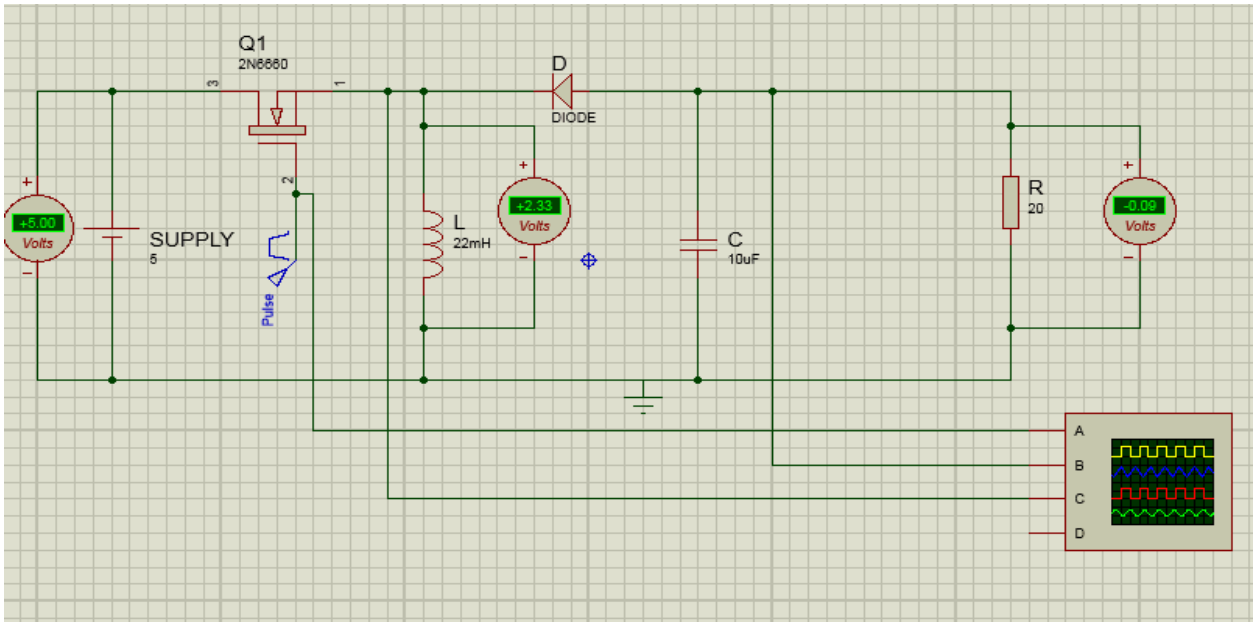


Figure 3: Output voltage across the inductor, V_L for 30% duty cycle

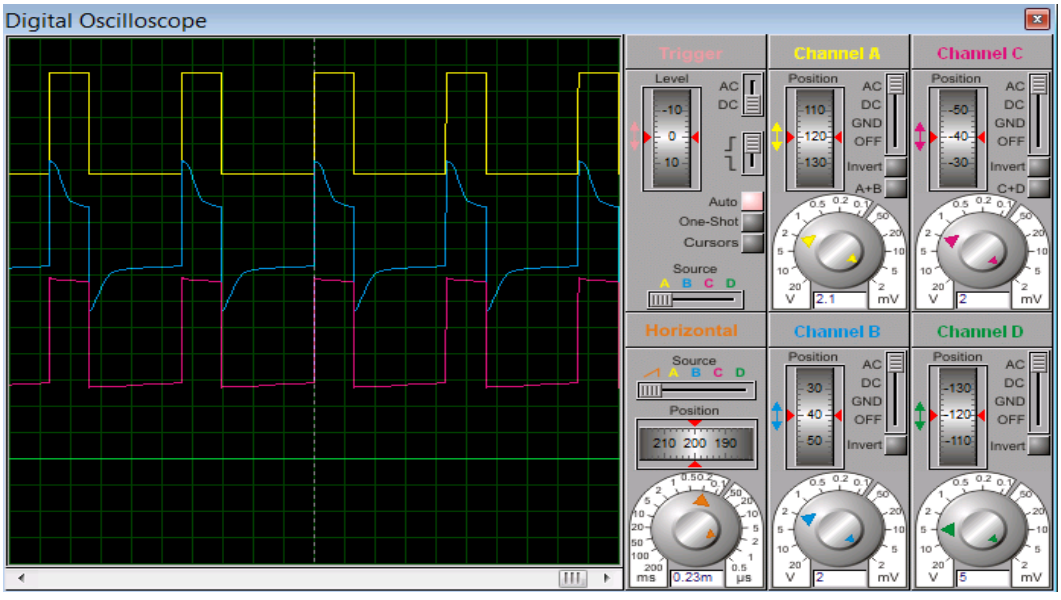
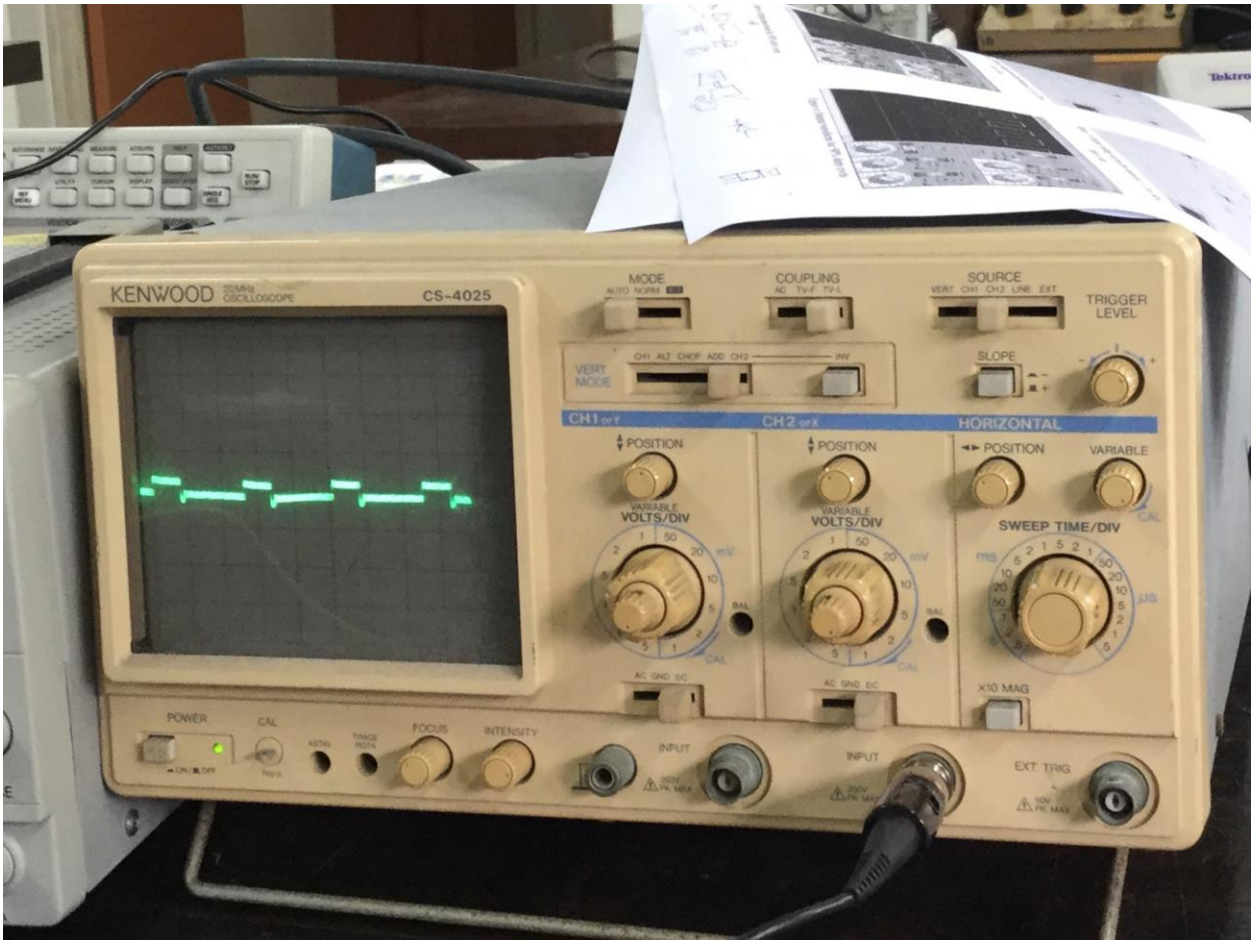


Figure 4: Example Output waveform for 30% duty cycle

Expectation observation: Input, 30% duty cycle



Code testing

30.js 50.js 80.js

```
1  var m = require('mraa');
2  console.log("Libmraa version: " + m.getVersion());
3
4  var generate = new m.Pwm(6); // hook up at PWM pin 6 at Edison
5  var dc = 0.3; // set duty cycle: 30%
6
7  generate.period_us(200); // set period as 200 ms
8  generate.enable(true);
9
10 setInterval(function() {
11     generate.write(dc);
12     console.log("duty cycle (%): " + dc * 100);
13 }, 1000);
```

Activity 2: set 50% duty cycle Edison PWM

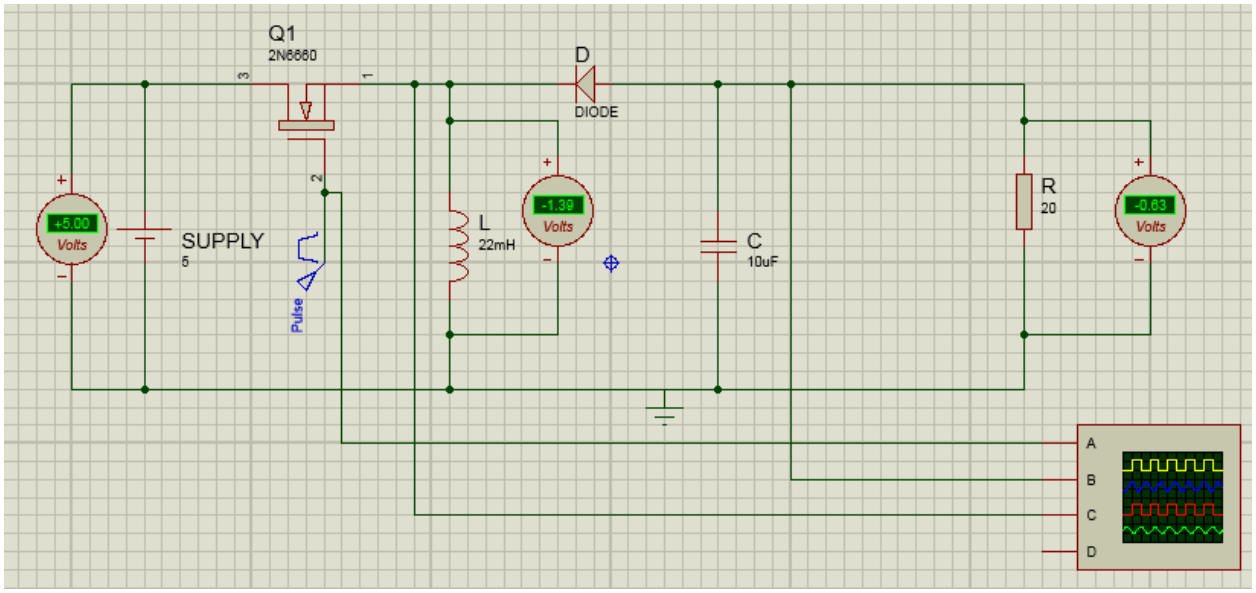


Figure 5: Output voltage across the inductor, V_L for 50% duty cycle

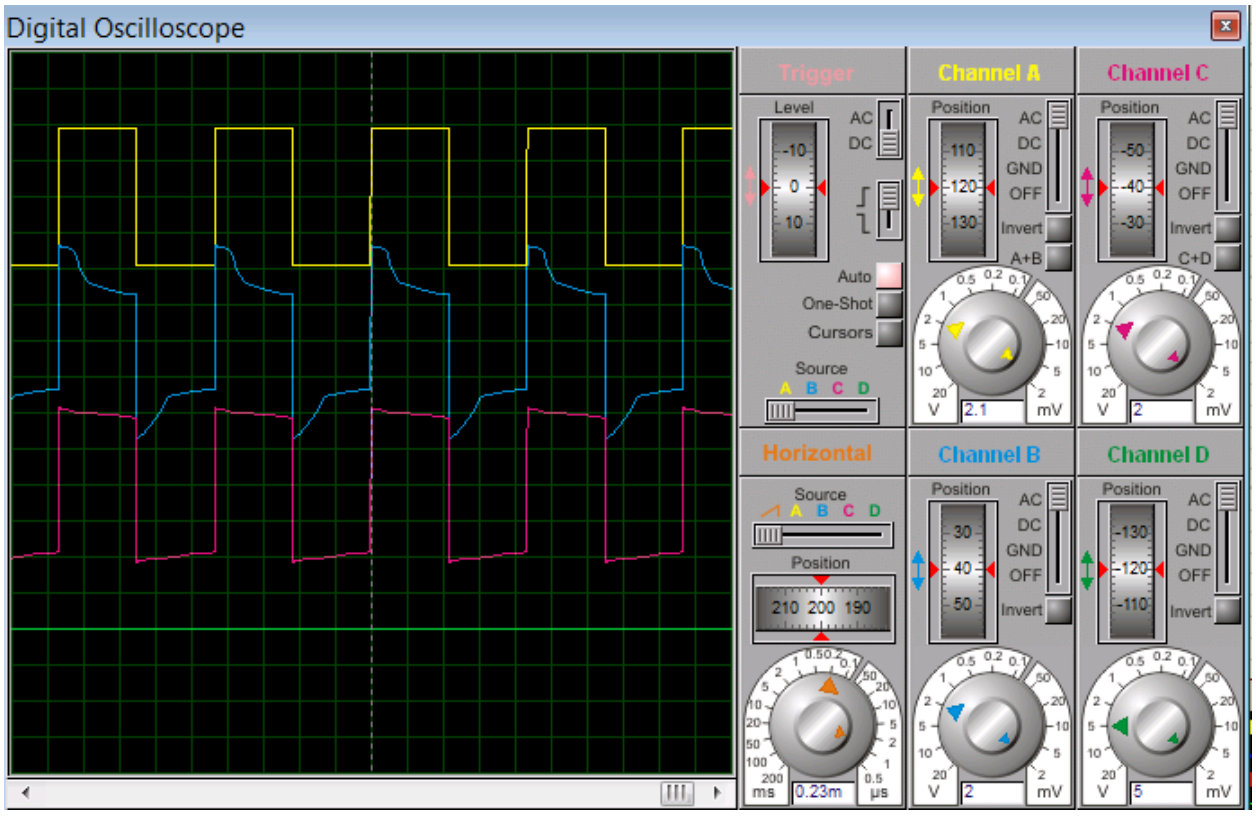
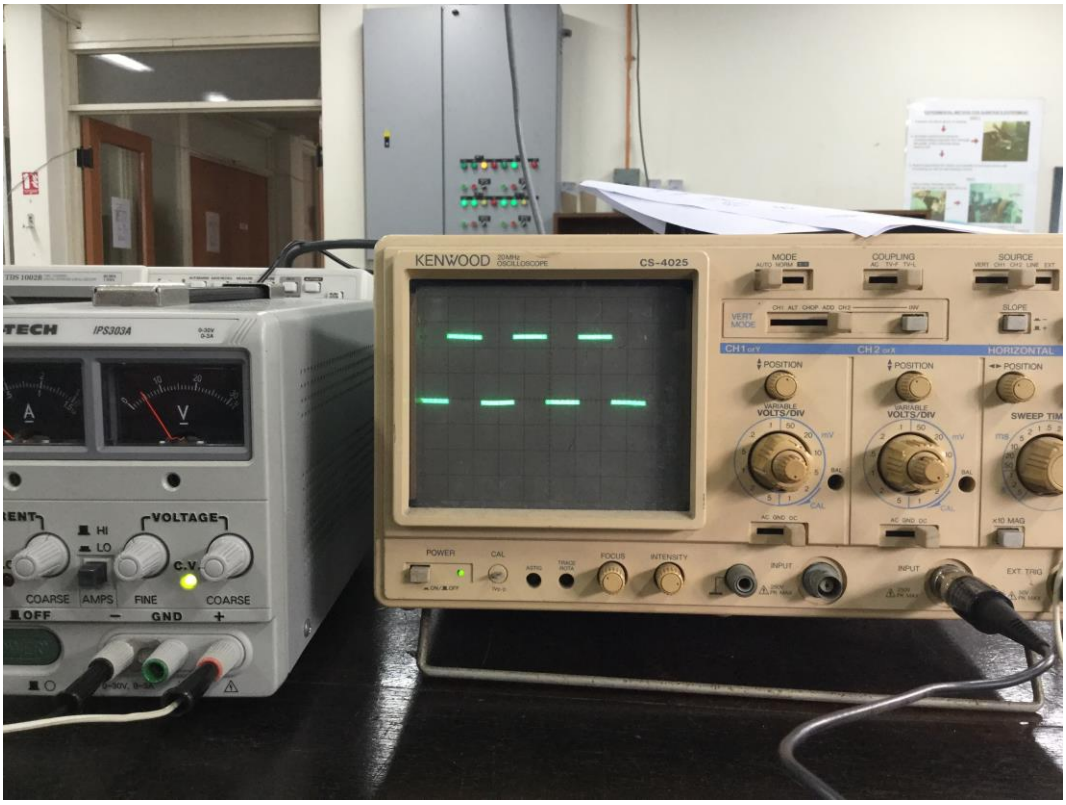


Figure 6: Example Output waveform for 50% duty cycle

Expectation observation



Code testing

30.js 50.js 80.js

```
var m = require('mraa');
console.log("Libmraa version: " + m.getVersion());

var generate = new m.Pwm(6); // hook up at PWM pin 6 at Edison
var dc = 0.5; // set duty cycle: 50%

generate.period_us(200); // set period as 200 ms
generate.enable(true);

setInterval(function() {
    generate.write(dc);
    console.log("duty cycle (%): " + dc * 100);
}, 1000);
```

Activity 3: set 80% duty cycle Edison PWM

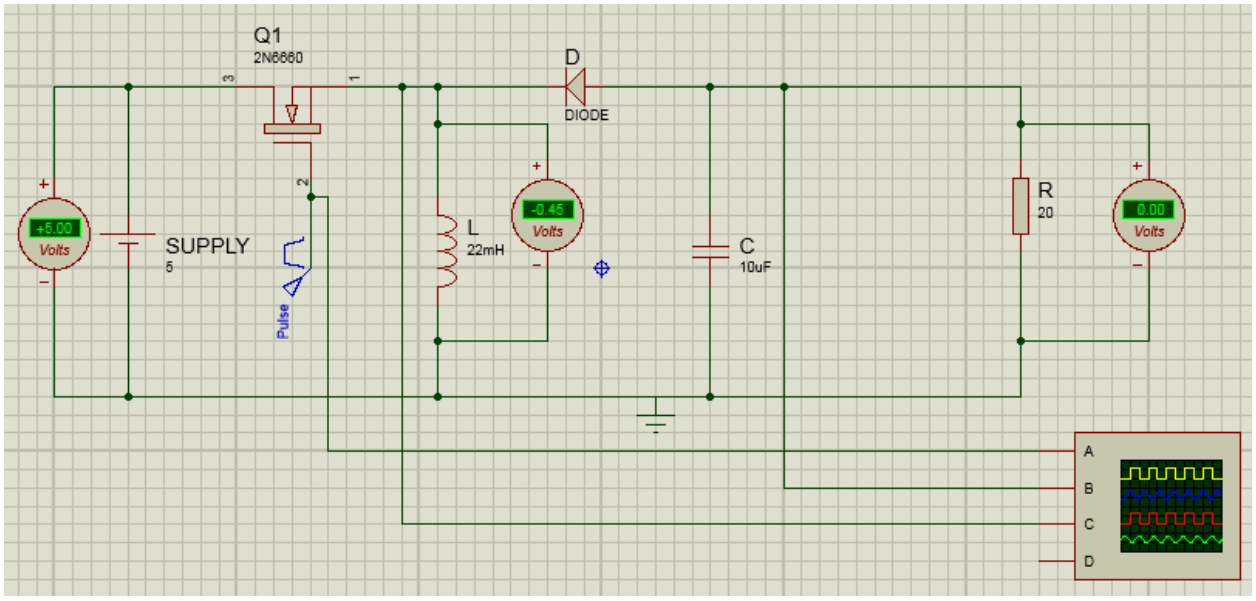


Figure 7: Output voltage across the inductor, V_L for 80% duty cycle

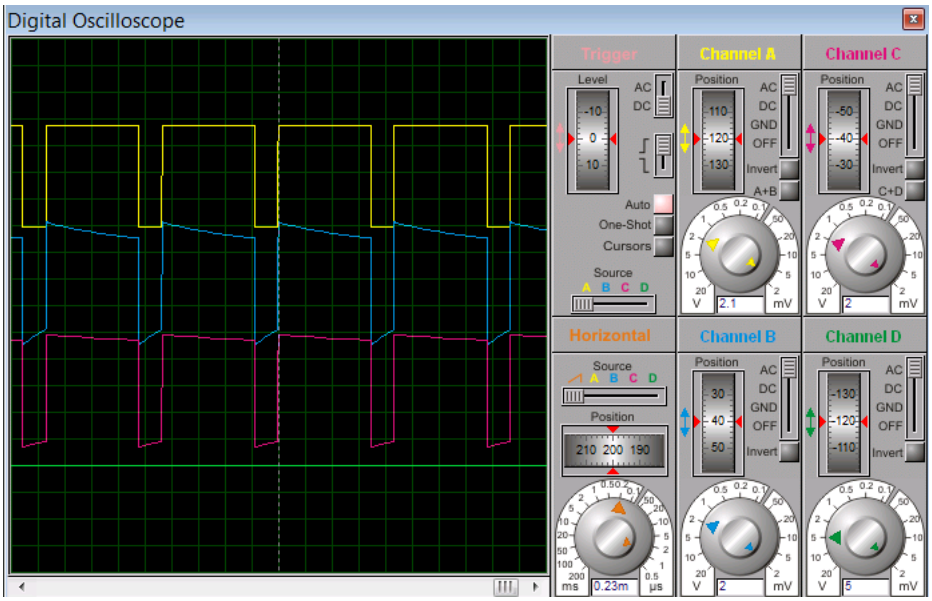
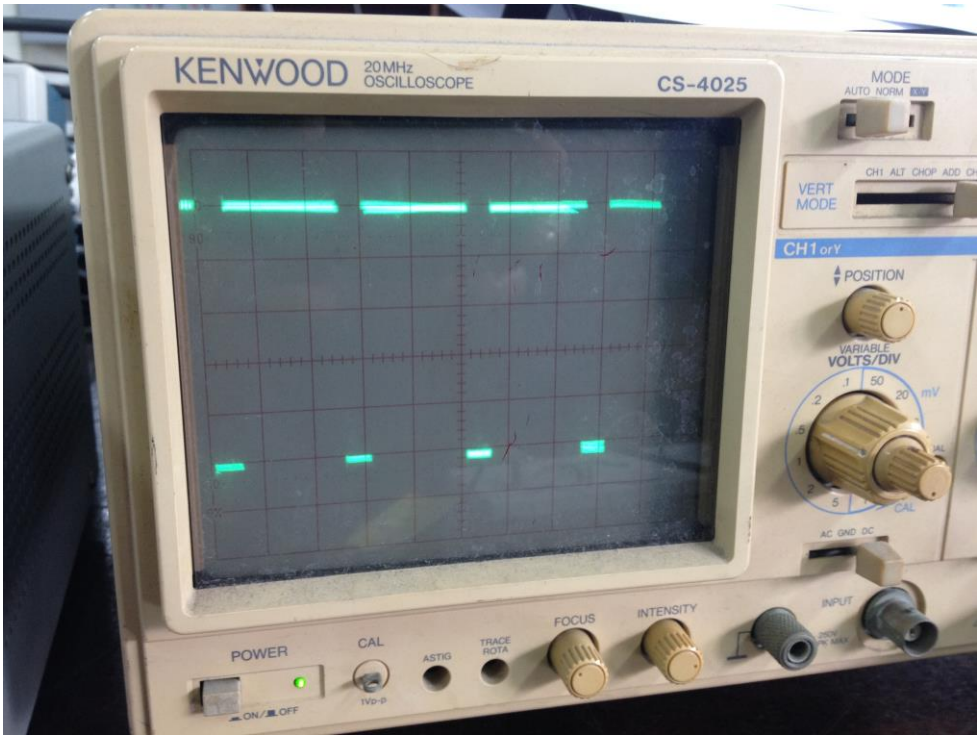
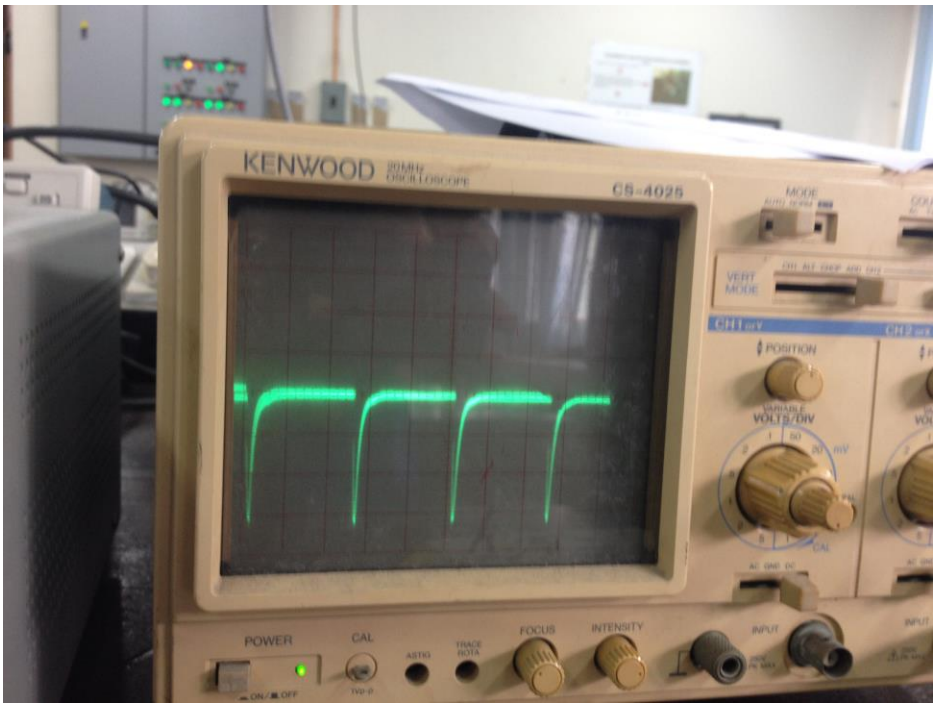


Figure 8: Sample Output waveform for 80% duty cycle

Expectation: Input, 80% duty cycle



Expectation: Output, 80% duty cycle



Code testing

30.js 50.js 80.js

```
1  var m = require('mraa');
2  console.log("Libmraa version: " + m.getVersion());
3
4  var generate = new m.Pwm(6); // hook up at PWM pin 6 at Edison
5  var dc = 0.8; // set duty cycle: 80%
6
7  generate.period_us(200); // set period as 200 ms
8  generate.enable(true);
9
10 setInterval(function() {
11     generate.write(dc);
12     console.log("duty cycle (%): " + dc * 100);
13 }, 1000);
```

Prepared by : Prof. DR. Ishak Aris and Syed Khairil Syed Kharuddin

Modified by: Hadrihl