MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Department of Electrical Engineering and Computer Science
6.S08
Introduction to EECS: Interconnected Embedded Systems

MIDTERM EXAM
Wednesday, April 12, 2017
7:30 PM – 9:30 PM

This examination contains **5** problems on pages 2-23.

Whenever a numerical value is requested, indicate the units clearly.

This examination is open book/open notes.

Calculators are allowed. Cell phones, laptops, music players, and other communication/entertainment devices are prohibited.

Please put your name where indicated, on the top of each page. Also enter your full name, Kerberos, and section number below:

Name:_____

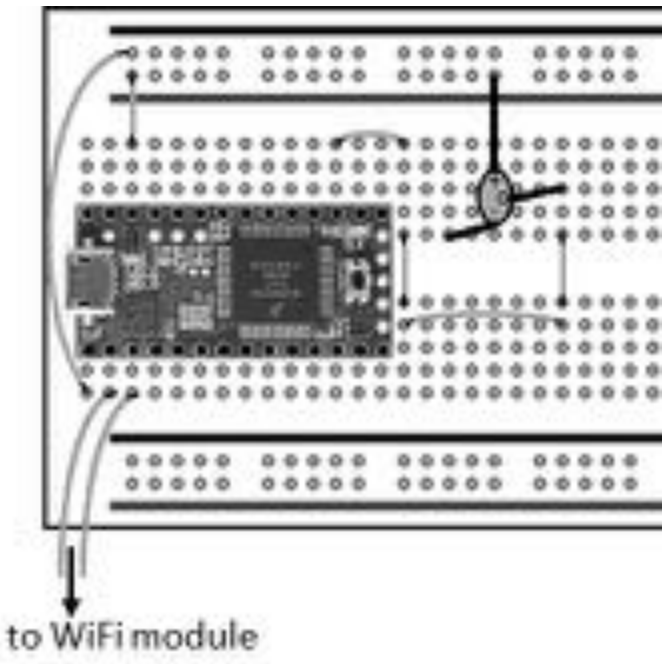
Kerberos:_____

Section (1, 2, or 3):_____

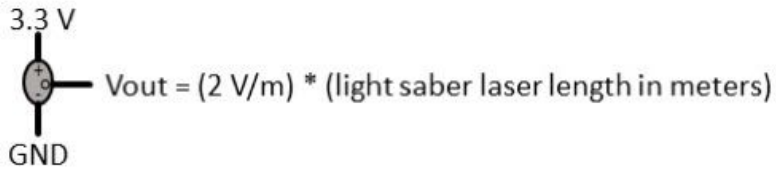
1	/15	
2	/20	
3	/12	
4	/22	
5	/14	
TOTAL	83	

Problem 1. A new hope (15 pts).

After narrowly escaping from an Imperial Destroyer, Luke and Leia relax in the Falcon and look over Luke's newly acquired light saber. Eager to show it off, Luke hits the ON button, and light saber appears but is only 10 cm long. To troubleshoot, he and Leia unscrew the casing of the light saber, and inside they find the following system:



Since both Luke and Leia previously took 6.S08, they immediately identify the Teensy-based system that runs the light saber. Besides the Teensy itself and a ESP8266 WiFi module (not shown), the system includes a laser module that is used to measure the length of the light saber's light. It is a 3-terminal device where two of the terminals are connected to power and ground, and where the third terminal outputs an analog voltage (in Volts) according to the equation below:



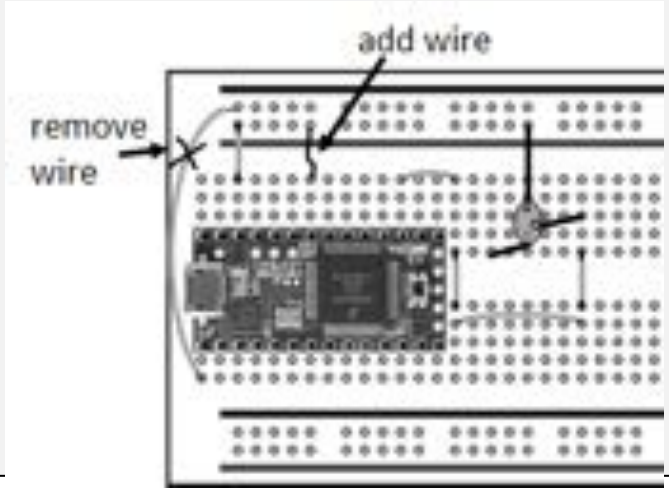
So a light saber length of 10 cm would result in an output voltage of 0.2 V.

Leia takes out her laptop, starts the Arduino IDE, and loads up the code that runs the light saber. Part of the code listing is shown below:

```
...some code...

10    const int IOT_UPDATE_INTERVAL = 3000;        // how often to get from cloud
11    const int LIGHT_MODULE = A3;                  // light module analog pin
12    const String IESC = "rebel_alliance.mit.edu";       // server address
13    const String KERBEROS = "skywalker";                // your kerberos
14    const String PATH = "/6S08dev/" + KERBEROS + "/light_saber.py";
16
17    void setup()
18    {
19      Serial.begin(115200);
20      Wire.begin();
21      SPI.setSCK(14);
22      SPI.begin();
23      analogReadResolution(14);
24
25      // Wifi setup
26      wifi.begin();
27      wifi.connectWifi("millennium_falcon", "chewie");
28      while (!wifi.isConnected()); //wait for connection
29    }
30
31    void loop() {
32          int saber_length = analogRead(LIGHT_MODULE);

...more code...
```

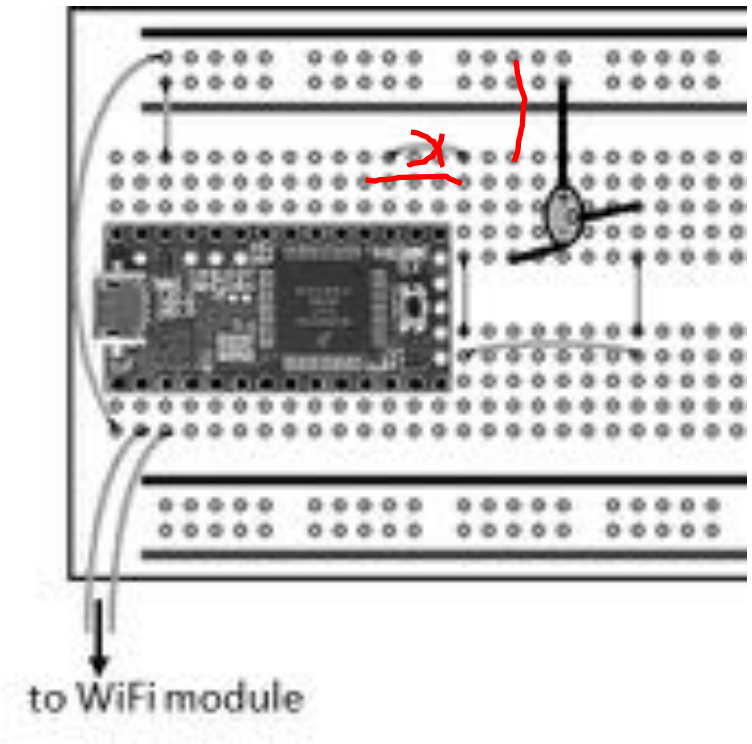
A. Looking over the code and the system, Leia immediately identifies two wiring errors with the Teensy and sensor (the Wifi module is wired correctly). Fix the wiring by (1) breaking wires by drawing an "X" through them, and/or (2) adding new wires by drawing them in. The Teensy pinout is provided at the end of the exam. Here's an example:

EXAMPLE:



The one to fix is on the next page…

FIX THIS ONE:



B. After fixing the wiring, Luke still can't get the light saber to work. Leia adds a `Serial.println(saber_length)` statement to the code after line 32 and compiles/uploads the code to the microcontroller and opens up the Serial monitor to see the readout. She then adjusts a screw on the light saber, which sets the laser light to be **0.2 m** long (measured by a tape measure).

What value is printed to the Serial Monitor? _____ **1985 (or 1986 for partial)** _____

C. Write one line of C code to properly convert this value to meters and store it in a variable called `light_saber_length` of type `float`.

```
float light_saber_length = saber_length*3.3/16384/2;
```

D. Later on in the code listing, Leia comes across the section of code that sends the light saber length to the IoT server. This piece of code is replicated below:

```
300    String post_data = "kerb=" + KERBEROS + "&lsl=" +
              String(light_saber_length);
301
302    wifi.sendRequest(POST, IESC , 80, PATH, post_data, false);
```

Assuming that `light_saber_length` has a value of 1.03, what value is stored in the *Teensy's memory* in variable `lsl` after line 300 executes? Circle one:

A. 1.03

B. The ASCII codes for 1.03

C. The hex values for 1.03

D. `lsl` does not exist.

E. Turning now to the server-side Python code, Leia opens up `light_saber.py`. Part of the code listing is shown here:

```
100    def web_handler(request):
101
102        method = request['REQUEST_METHOD']
103
104        if method == "POST":
105            parameters = request['POST']
106
107            saber_length = parameters['lsl']
108            kerbs = parameters['kerb']
109
110            db = records.Database('mysql://student:rebels@localhost/rebel_data')
111
112            db.query("INSERT into light_saber (kerberos, saber_data) VALUES (:k, :l)",
               k=kerbs, l=saber_length)
```

Assume that the Teensy code line 302 has executed and a `wifi.sendRequest()` has been sent with the `post_data` string from Part D.

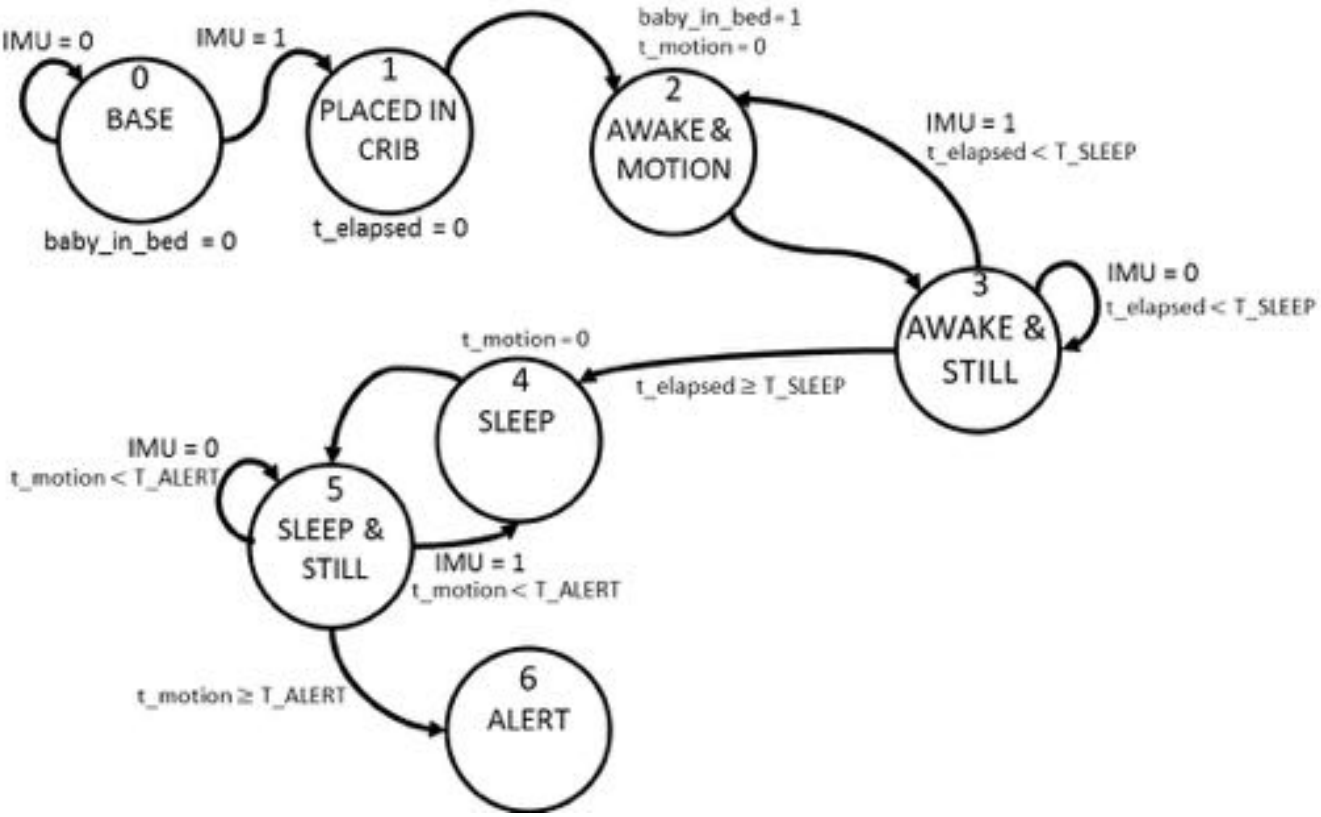
Below, enter the values for following variables as they exist in *Python memory* after `light_saber.py` line 112 executes. If a variable does not exist in Python memory at that point or its value is unspecified, write N/A at the prompt. Be sure to indicate the data type.

```
saber_length = __”1.03”___

kerb = _N/A__

lsl = __N/A__

kerbs = ___”skywalker”_

kerberos = ___N/A____

saber_data = ___N/A__
```

Problem 2. BAD Company (20 pts).

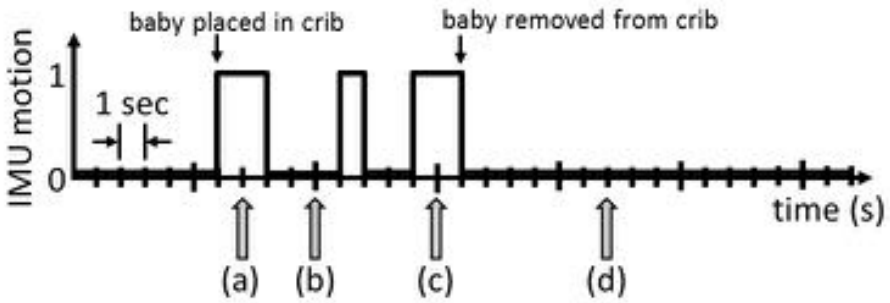
After taking 6.S08, you decide to drop out of MIT and join a promising new startup: Baby Automation Devices (BAD). This stealth-mode startup is developing a smart crib that alerts parents if their baby stops moving during sleep, which is associated with sudden infant death syndrome (SIDS).

Here is the engineering team's current state machine diagram, which is still in development:



The `IMU` variable is 1 when the baby is moving and 0 otherwise. You can assume that this state machine is a clocked synchronous state machine, meaning that inputs are read and state changes occur once every time step. Any unlabeled state transitions mean that all inputs lead to that transition. Only one state change may occur in each time step. You can assume that this microcontroller can perform a state transition one thousand times each second. Finally, the two timers `t_motion` and `t_elapsed` operate as `elapsedMillis` timers that start at 0 upon power-up (when the system is initially in state 0).

A. To test the state diagram functionality, the company runs an experiment (Experiment 1) and obtains actual data from a baby in a crib, shown below. Via the use of cameras, the company can also tell what the baby is actually doing.



Assuming T_SLEEP is set to correspond to 15 seconds, and T_ALERT is set to correspond to 10 seconds, and that the system starts in state 0 at time 0, determine the possible state(s) of the system and the possible values of the two timers given the above time course of motion input, at each of the timepoints below.

Timepoint (a)

STATE (CIRCLE ALL POSSIBLE): 0 1 2 3 4 5 6

t_motion = _____

t_elapsed = _____

Possible answers:
States 2 & 3, t_motion = 0 or 1 ms, t_elapsed = 1000 ms
States 2 & t_motion=0 ms, t_elapsed = 999 ms
States 3 & t_motion=1 ms, t_elapsed = 1000 ms
Answer must be denominated in ms or explicitly marked with units
Off-by-one millisecond errors are accepted

Timepoint (b)

STATE (CIRCLE ALL POSSIBLE): 0 1 2 3 4 5 6

t_motion = _____

t_elapsed = _____

State 3, t_motion = 2000 ms, t_elapsed = 4000 ms
Answer must be denominated in ms or explicitly marked with units
Off-by-one millisecond errors are accepted

Timepoint (c)

STATE (CIRCLE ALL POSSIBLE): 0 1 2 3 4 5 6

t_motion = _____

t_elapsed = _____

Possible answers:
State 2 & 3, t_motion = 0 or 1 ms, t_elapsed = 9000 ms
State 2 & t_motion=0 ms, t_elapsed = 9999 ms
State 3 & t_motion=1 ms, t_elapsed = 10000 ms
Answer must be denominated in ms or explicitly marked with units
Off-by-one millisecond errors are accepted

Timepoint (d)

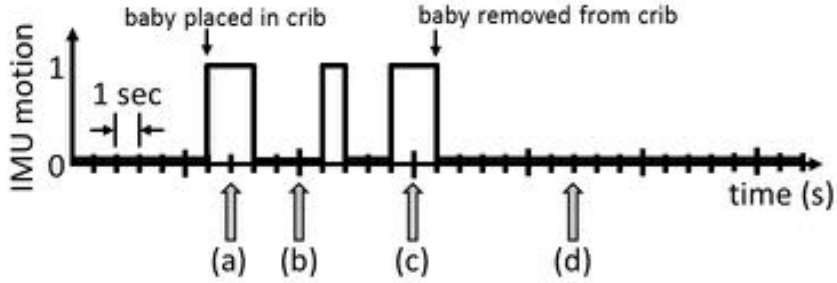
STATE (CIRCLE ALL POSSIBLE): 0 1 2 3 4 5 6

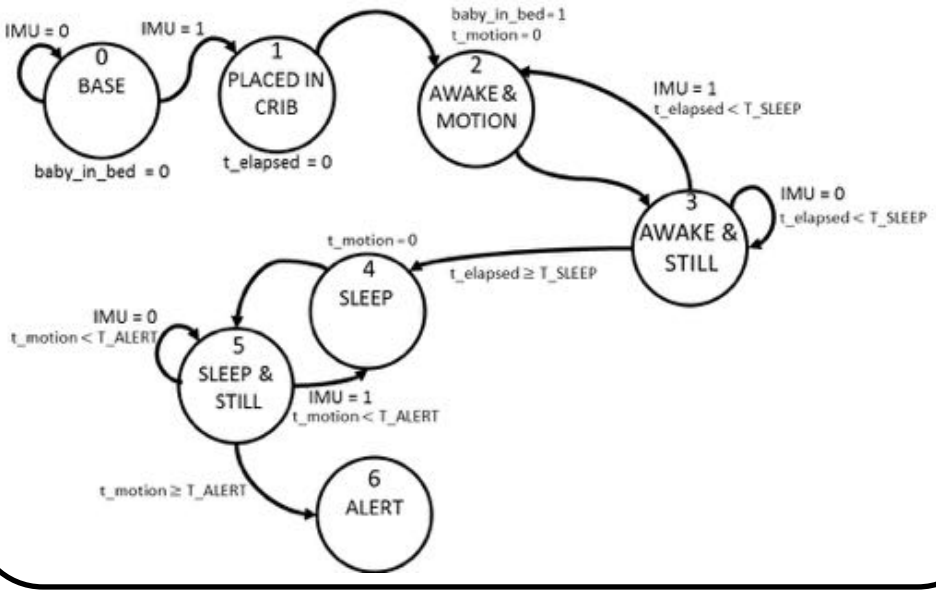
t_motion = _____

t_elapsed = _____

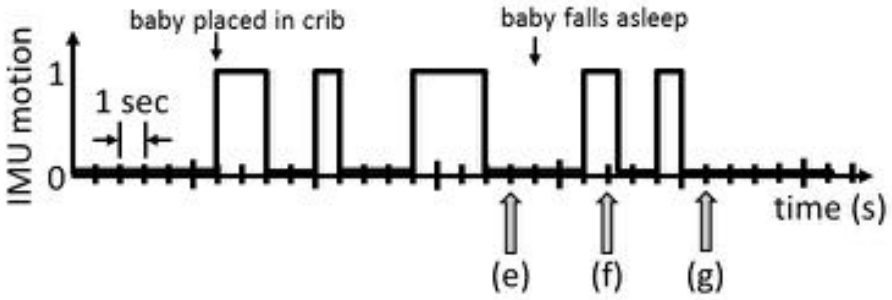
State 5, t_motion = 1000 ms, t_elapsed = 16000 ms
Answer must be denominated in ms or explicitly marked with units
Off-by-one millisecond errors are accepted

State Diagram and Experiment 1 data repeated for convenience

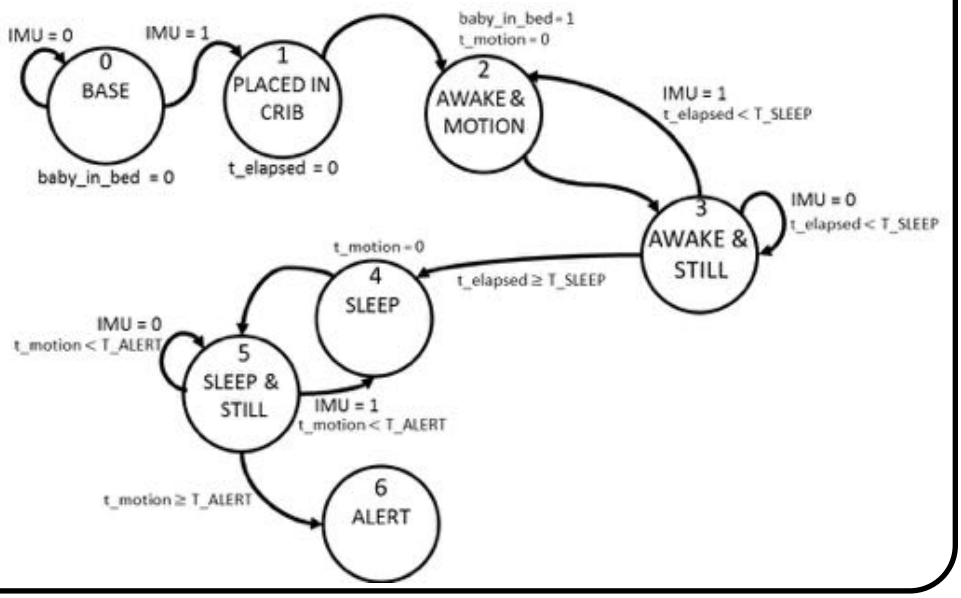




B. Another experiment is run (Experiment 2). Assuming T_SLEEP is set to correspond to 15 seconds, and T_ALERT is set to correspond to 10 seconds, and that the system starts in state 0 at time 0, determine the possible state(s) of the system and the possible values of the two timers given the above time course of motion input, at each of the timepoints below.



State Diagram repeated for convenience



Timepoint (e)

STATE (CIRCLE ALL POSSIBLE): 0 1 2 3 4 5 6

t_motion = _____

t_elapsed = _____

State 3, t_motion = 1000 ms, t_elapsed = 12000 ms
Answer must be denominated in ms or explicitly marked with units
Off-by-one millisecond errors are accepted

Timepoint (f)

STATE (CIRCLE ALL POSSIBLE): 0 1 2 3 4 5 6

t_motion = _____

t_elapsed = _____

State 4 & 5, t_motion = 0 or 1 ms, t_elapsed = 16000 ms
Answer must be denominated in ms or explicitly marked with units
Off-by-one millisecond errors are accepted

Timepoint (g)

STATE (CIRCLE ALL POSSIBLE): 0 1 2 3 4 5 6

t_motion = _____

t_elapsed = _____

State 5, t_motion = 1000 ms, t_elapsed = 20000 ms
Answer must be denominated in ms or explicitly marked with units
Off-by-one millisecond errors are accepted

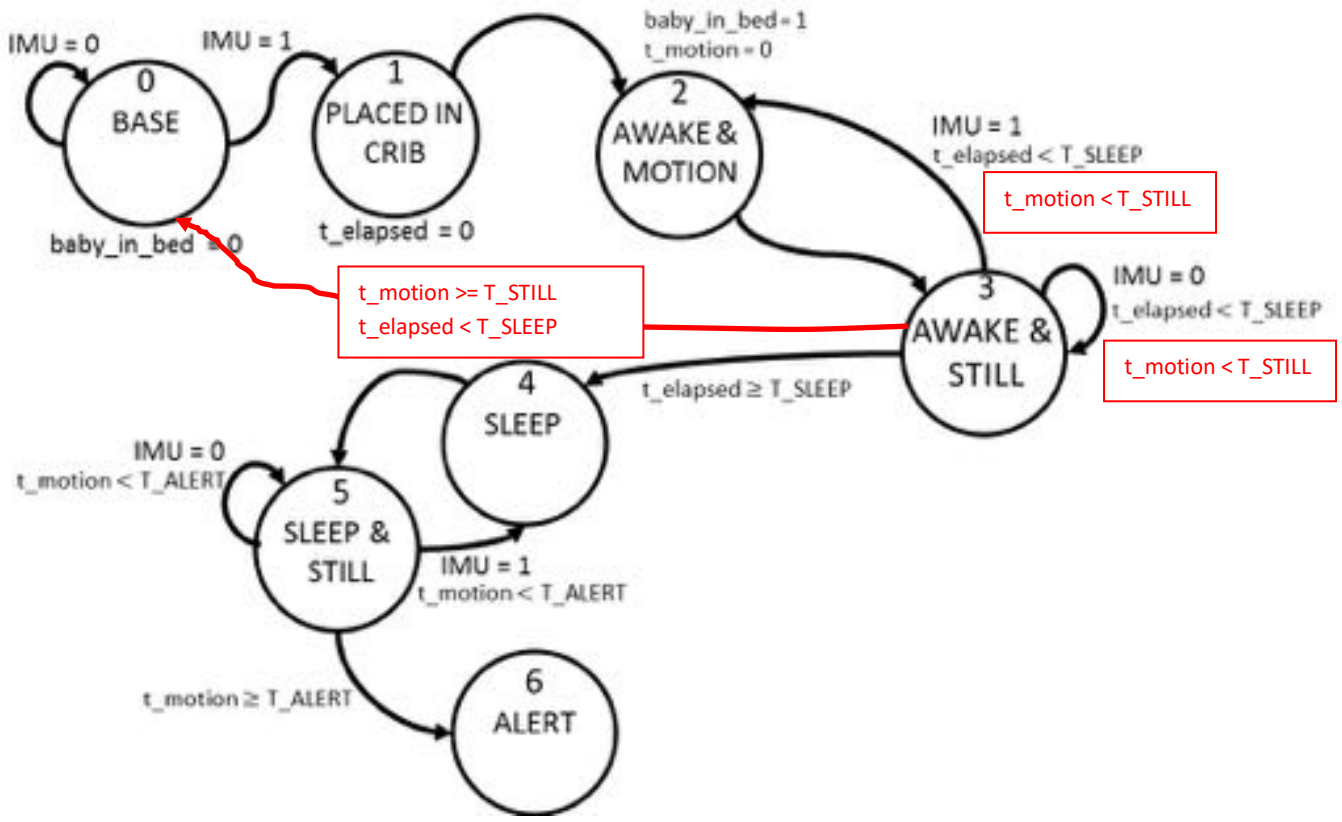
C. Assuming the system continues onward from timepoint (g) above, at what time will the system enter state 6, assuming IMU=0 for all time after timepoint (g)?

time = __35 sec___

D. The team does more experiments and realizes this is not a very good sleep monitor. The desired proper behavior, which is not correctly implemented in the state machine diagram, is as follows:

1. It should first detect whether a baby has been put into the crib. When the baby is first placed in the crib, the baby is awake, so the system keeps track of motion.
2. If the baby stops moving for at least 5 seconds while `t_elapsed < T_SLEEP`, we should assume that the baby has been removed from the crib and return to state 0.
3. After some time, the baby falls asleep.
4. Once the baby is asleep, if the baby does not move sufficiently often, an alert is sent to the parent's phone.
5. The system should be able to reset after an alert, and, when the baby wakes up after a long sleep, the system should not alert.

Fix the state diagram so that it properly accounts for Step 2 above, by adding **ONE state transition** and using **ONE of the existing timer variables (though you can add a new timer constant to it)**. You may need to edit other transitions to avoid ambiguity in state transitions.



Name and value of new timer constant (if any): ___ T_STILL = 5000_____

E. Even after the fix in Part D, there are still at least two events missing from the desired behavior. In 1 or 2 sentences, describe one of the missing events.

Examples:

- Not possible to reset after alert
- Moves to sleep automatically at t_elapsed>=T_SLEEP even if baby is moving
- No states for baby waking up after sleeping

Problem 3. The Key (12 pts).

A. To avoid detection by the Glorificus, Dawn and Buffy decide to communicate *in English* via a symmetric key Caesar cipher using the ASCII table in their set of symbols. An ASCII table is provided at the end of the exam.

They decide to only use the symbols between ASCII codes 65 and 90, inclusive. Unfortunately, since they haven't taken 6.S08, they decide to ignore spaces and punctuation. They send the following message:

```
UIABMZ, Q LMTQDMZ BPM ATIGMZ. APM EPW GWC UWAB LMAQZM. AWZZG -- EPWU
```

a) Determine the key used for this code. The key is the positive shift used when going from plaintext to ciphertext.

Key: ____8__________

b) Determine the first three words of the plaintext message:

MASTER, I DELIVER

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

B. Willow and Anya decide to communicate using a Vigenère cipher using the same ASCII table and symbols between ASCII codes 65 and 90, inclusive (where an `A` corresponds to a shift of 0, `B` to 1, etc…). They send a ciphertext to each other, which starts with:

```
TFFDCSMWZJF
```

Unbeknownst to them, Spike is able to intercept the first few characters of the plaintext prior to encryption, which are:

```
SLAYERSRULE
```

In addition, he notices that those exact same 11 ciphertext characters repeat starting at character position 35 (where the first character in the message is character number 0).

a) Circle possible keyword lengths:

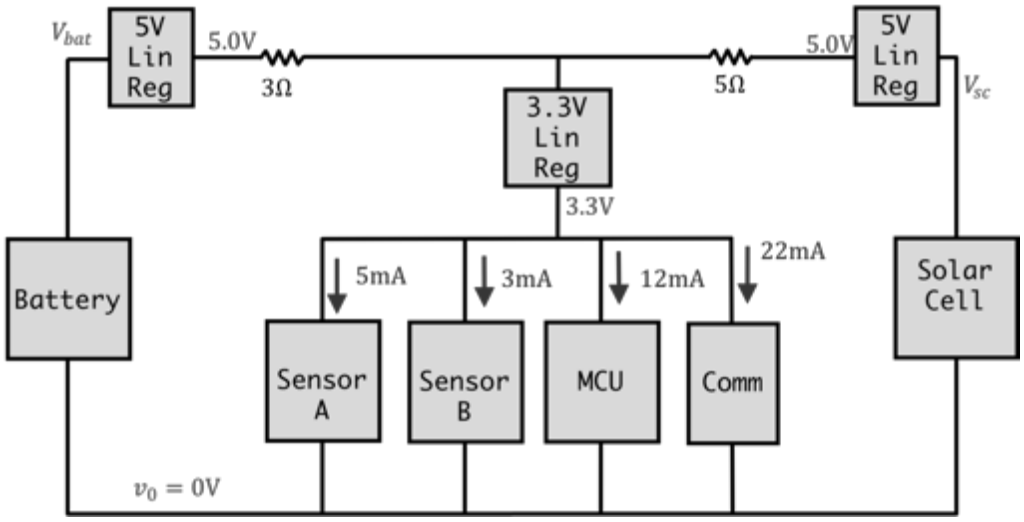
1 **2** **3** **4** **5** **6** **7** **8** **9** **10**

Possible answers:
5
5,7
5,10
5,7,10

b) Determine the shortest possible keyword:

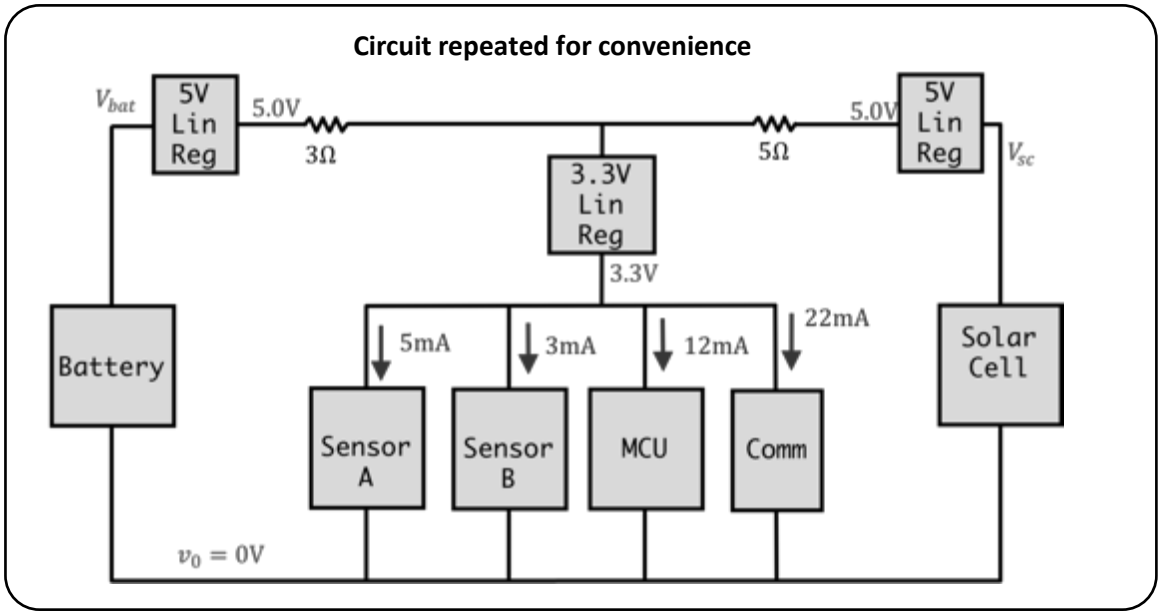
keyword: ___BUFFY_____

Problem 4. Power to the People (22 points). You have a hybrid system that runs off of two sources. One is a battery with 1200 mA-h capacity and voltage V_{bat}, which is regulated down to 5.0 V through a linear regulator. The second source is a solar cell with voltage V_{sc}, also regulated down to 5.0 V through a linear regulator, and which can produce anywhere from 0 mA to 50 mA of current depending on the amount of light on the cell. If there is no light impinging on the solar cell, there will be no power generated. There is also a 3.3 V linear regulator between those two sources and the rest of the system as shown in the circuit below, as well as two resistors that are used for current sensing. For parts A and B below, you may assume the components of the system possess the indicated currents.



A. Determine the lifetime of the system with all four devices (Sensor A, Sensor B, MCU, Comm) running using the currents specified and with no illumination on the system (specify units)

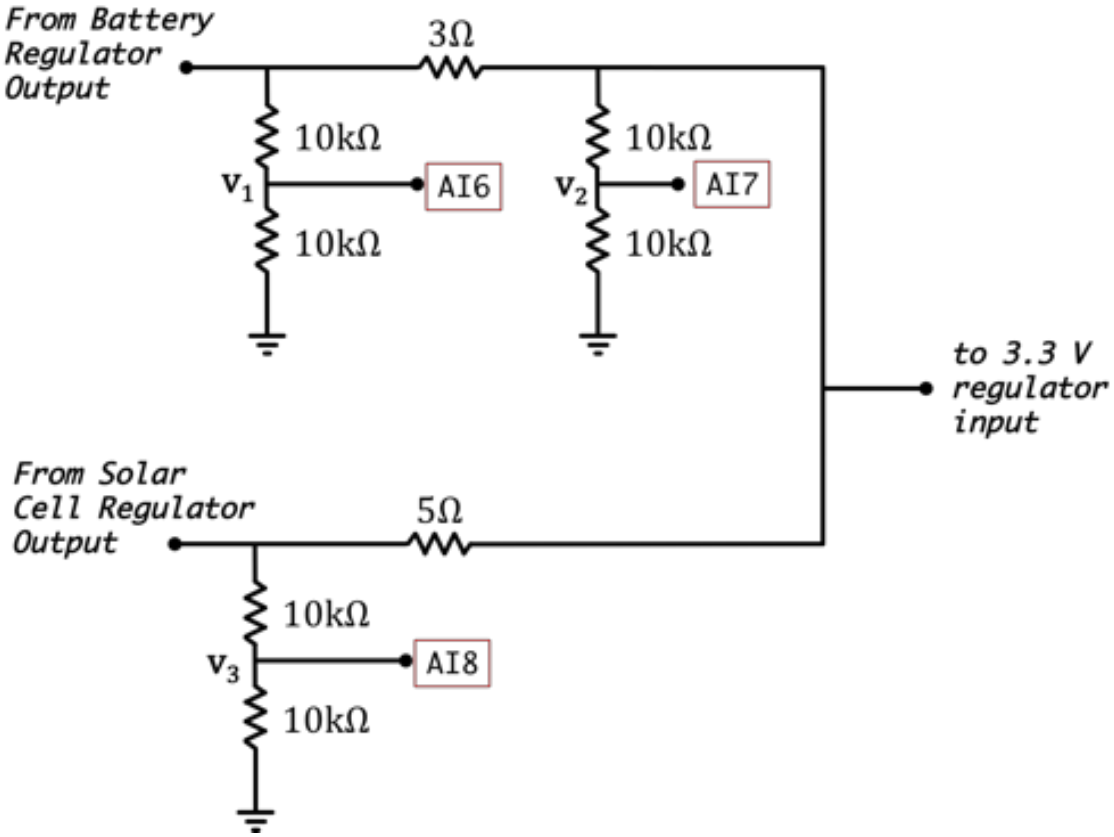
Lifetime: 1200 mAh/42 mA = 28.57 h



B. Assume the illumination on the solar cell is set to be such that the solar supply provides 10 mA. Determine the lifetime of the system (specify units)

Lifetime: 1200 mAh/32 mA = 37.5 h

C. The component currents used in Part A and B are rough estimates of the currents through the components. As we've seen in class the actual currents can vary significantly over time. To be able to measure real-time variations in the current and consumed power, you and your partner add in a system like we did in Lab 06/07. You reuse/modify circuits from lab to implement the following circuit:



Here, `AI*` refer to the one of the analog inputs on the Teensy (specified by the numbers):

You build the entire system and the wiring is good. Your partner made a first pass at some code that will analyze several aspects of power consumption of your system. In particular, the code will measure, calculate, and then report via appropriate Print statements:

- The total power being consumed by the microcontroller (MCU), two sensors, and communication module (Comm), to be reported in Watts
- The total power being provided by the battery, reported in Watts
- The total power provided by the solar cell, reported in Watts
- The total efficiency of the circuit, where efficiency is defined as:

$$\text{efficiency} = \frac{\text{power to (MCU+Sensors+Comm)}}{\text{power delivered by two sources}}$$

For the remaining parts of this problem, you made some measurements and found that V_{bat}=7.2V and V_{sc} = 6.1V. In addition, you may no longer assume currents through the microcontroller, sensors, or communication module are the values indicated/used Parts A and B.

<u>Repeated from bottom of previous page:</u> For the remaining parts of this problem, you made some measurements and found that V_{bat}=7.2V and V_{sc} = 6.1V. In addition, you may no longer assume currents through the microcontroller, sensors, or communication module are the values indicated/used Parts A and B.

```
float total_power=0; //power consumed by two sensors, mcu, and comm
float battery_prov_power=0; //power provided by battery;
float sc_prov_power=0; //power provided by solar cell
float system_efficiency=0; //system efficiency
//efficiency = (total power consumed by two sensors, mcu, comm)/(total power
provided)

void setup() {
  analogReadResolution(10);
  pinMode(13,OUTPUT);
  Serial.begin(115200);
}

void loop() {
  do_calculations();
  Serial.print("Total Power consumed: ");
  Serial.println(total_power);
  Serial.print("Battery Provided: ");
  Serial.println(battery_prov_power);
  Serial.print("Solar Cell Provided: ");
  Serial.println(sc_prov_power);
  Serial.print("Efficiency: ");
  Serial.println(system_efficiency);
  Serial.print("Life Time Remaining (in hours): ");
  Serial.println(calculate_remaining_life());
}

void do_calculations(){
  // Your code here
}

float calculate_remaining_life(){
  float energy = 3.7*1200;
  return energy/battery_prov_power;
}
```

Write the `do_calculations()` function to make it perform the proper calculations for `total_power`, `battery_prov_power`, `sc_prov_power`, and `system_efficiency`. You should not change the function header.

```
void do_calculations(){

  float v1 = analogRead(AI6) * 3.3/1023; // or assume v1 = 5;

  float v2 = analogRead(AI7) * 3.3/1023;

  float v3 = analogRead(AI8) * 3.3/1023; // or assume v3 = 5;

  float battery_current = (v1 - v2) * 2 / 3 ;

  float sc_current = (v3 - v2) * 2 / 5;

  battery_prov_power = battery_current * 7.2;

  sc_prov_power = sc_current * 6.1;

  total_power = (battery_current + sc_current) * 3.3;

  efficiency = total_power / (sc_prov_power + battery_prov_power);
```

D. Change **one** Line of code from the `calculate_remaining_life` function to have it return the properly calculated remaining system lifetime. You can only modify one line. Modifying two will result in a 0 for this section.

```
float calculate_remaining_life(){
  float energy = 3.7*1200;
  return energy/battery_prov_power;
}
```

Original Line:	Modified Line:
`float energy = 3.7*1200;`	float energy = 7.2 * 1200 * 0.001;
`return energy/battery_prov_power;`	NO CHANGE

Problem 5.

Consider the following set of difference equations and C++ code implementations:

A: $y[n] = 0.9\,y[n-1] + 0.1x[n]$

B: $y[n] = 0.9y[n-1] + 0.1x[n-1]$

C: $y[n] = x[n] - x[n-1]$

D: $y[n] = x[n] + x[n-1] + x[n-2]$

E: $y[n] = y[n-2] + 4x[n-3]$

F: $y[n] = 0.9x[n]$

G: $y[n] = 0.1y[n-1] + 0.9x[n]$

```
float a = 0;
float b = 0;
float c = 0;
float d = 0;
float e = 0;
float f = 0;

void setup() {
  Serial.begin(115200);
}

void loop() {
  float x = 0.000312*analogRead(A3);
  Serial.println(f(x));
}
```

Match the difference equations A-G with the C++ code for functions `f()` on the next page. If a function does not exist that satisfies the difference equation, enter N/A in the table below. If more than one function satisfies the difference equation, you need to enter only one. Functions may be used more than once.

Difference Equation	Function
A	7
B	10 or 11
C	12
D	2
E	N/A
F	5
G	3

*Note as shown, 7 through 12 are actually incorrect as answers because the datatype declaration is missing in the function. If students refused to use to use 7, 10, 11, and 12 for A, B, and C above, credit was given assuming that was the reason. If any of 7 through 12 were used in an answer, we assumed the student was not using that as reason.

1:

```
float f(float x){
  a = b;
  a = -x;
  return x+b;
}
```

2:

```
float f(float x){
  c = b;
  b = a;
  a = x;
  return a + b + c;
}
```

3:

```
float f(float x){
  a = 0.1*a + 0.9*x;
  return a;
}
```

4:

```
float f(float x){
  a = x;
  b = a;
  c = b;
  return 4*c + b;
}
```

5:

```
float f(float x){
  return 0.9*x;
}
```

6:

```
float f(float x){
  c = b;
  b = a;
  a = x;
  d = 4*c + f;
}
```

7:

```
float f(x){
  a = x;
  b = a;
  c = 0.9*c + 0.1*b;
  return c;
}
```

8:

```
float f(x){
  a = x;
  return x+a;
}
```

9:

```
float f(x){
  a = x;
  return x-a;
}
```

10:

```
float f(x){
  c = b;
  b = x;
  a = 0.9*a + 0.1*c;
  return a;
}
```

11:

```
float f(x){
  b = a;
  a = x;
  c = 0.9*c + 0.1*b;
  return c;
}
```

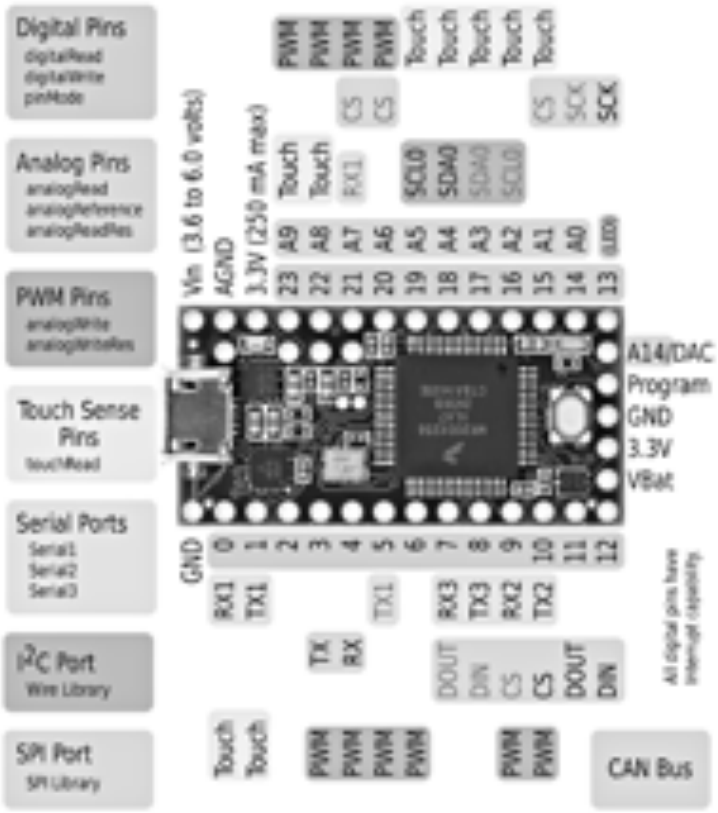
12:

```
float f(x){
  b = a;
  a = x;
  return x-b;
}
```

Reference Materials

Work on this page will not be graded.

Teensy pinout



`elapsedMillis` This special variable type automatically increases as time elapses, incrementing every millisecond. This makes it easy to check if a certain time has elapsed, while your program performs other work or checks for user input.

ASCII TABLE

Decimal	Char	Decimal	Char	Decimal	Char	Decimal	Char	Decimal	Char
32	space	51	3	70	F	89	Y	108	l
33	!	52	4	71	G	90	Z	109	m
34	"	53	5	72	H	91	[	110	n
35	#	54	6	73	I	92	\	111	o
36	$	55	7	74	J	93	]	112	p
37	%	56	8	75	K	94	^	113	q
38	&	57	9	76	L	95	_	114	r
39	'	58	:	77	M	96	`	115	s
40	(	59	;	78	N	97	a	116	t
41	)	60	<	79	O	98	b	117	u
42	*	61	=	80	P	99	c	118	v
43	+	62	>	81	Q	100	d	119	w
44	,	63	?	82	R	101	e	120	x
45	-	64	@	83	S	102	f	121	y
46	.	65	A	84	T	103	g	122	z
47	/	66	B	85	U	104	h	123	{
48	0	67	C	86	V	105	i	124	|
49	1	68	D	87	W	106	j	125	}
50	2	69	E	88	X	107	k	126	~

Extra page. Will not be graded.

Extra page. Will not be graded.

Extra page. Will not be graded.