question | Like **0** views

lab10 still not working - tests 1-3 pass, test 4 fails - not sure what is wrong?

Here is a code snippet and the command window output of the tests that both pass and fail.
It appears that the code works fine for test 1 where all 3 sensors are set, test 2 where the the
walk light should come on, test 3 where the west rd green light should come on, but it fails (times out)
for test 4 where the south rd green light should come on. I've double checked my FSM and the code
but not seeing my error. Any help appreciated! I'm continuing on for now to the week c11 videos - trying
to catch up this week a bit! Thanks.

```
Running with Code Size Limit: 32K
Load "C:\\Keil\\Labware\\Lab10_TrafficLight\\Lab10.axf"


*** Restricted Version with 32768 Byte Code Size Limit
*** Currently used: 18844 Bytes (57%)


Start grading
Clock rate appears to be : 80 MHz
Running 5 tests


0) Initialization tests :


- (PE2-0) have been selected as the input pins
- Verifying input configuration...
Pass: PORTE DEN bits (2-0) are high
Pass: PORTE DIR bits (2-0) are low
Pass: PORTE AFSEL bits (2-0) are low
Pass: PORTE AMSEL bits (2-0) are low
Pass: PORTE PCTL bits (11-0) are low


- (PB5-0) have been selected as the output pins
- Verifying output configuration...
Pass: PORTB DEN bits (5-0) are high
Pass: PORTB DIR bits (5-0) are high
Pass: PORTB AFSEL bits (5-0) are low
Pass: PORTB AMSEL bits (5-0) are low
Pass: PORTB PCTL bits (23-0) are low


1) Servicing all 3 requests, lights should make a complete cycle
FSM: Transition: 2 to 3
FSM: Transition: 3 to 4
FSM: Transition: 4 to 5
FSM: Transition: 5 to 1
FSM: Transition: 1 to 6
FSM: Transition: 6 to 1
FSM: Transition: 1 to 7
FSM: Transition: 7 to 1
FSM: Transition: 1 to 7
FSM: Transition: 7 to 9
FSM: Transition: 9 to 2
Pass: All requests were serviced
Pass: South request was serviced
Pass: West request was serviced
Pass: Walk request was serviced
- Test PASSED


2) Servicing walk button, walk light should come on
FSM: Bad Transition: 2 to 1
FSM: Transition: 1 to 6
Pass: All requests were serviced
Pass: Walk request was serviced
```

- Test PASSED

3) Servicing west button, west green light should come on
FSM: Transition: 6 to 1
FSM: Transition: 1 to 7
FSM: Transition: 7 to 1
FSM: Transition: 1 to 7
FSM: Transition: 7 to 9
FSM: Transition: 9 to 2
Pass: All requests were serviced
Pass: West request was serviced
- Test PASSED

4) Servicing south button, south green light should come on
* FAIL: Did not service all requests before the timeout
* FAIL: South request was not serviced
- Test FAILED

Done grading. Score is 65

Code snippet: (Note: plan to rename the states according to what was in the lab description - somehow I missed the suggestions on what they should be called when I first wrote it!

```
STyp FSM[9]={
 {0x31,50,{goWest, waitWest, goWest, waitWest, goWalk, waitWest, goWalk, waitWest }},
 {0x51,50,{goSouth, goSouth, goSouth, goSouth, goSouth, goSouth, goSouth, goSouth }},
 {0x85,50,{goSouth, goSouth, waitSouth, waitSouth, goWalk, goWalk, waitSouth, waitSouth }},
 {0x89,50,{goWest, goWest, goWest, goWest, goWest, goWest, goWest, goWalk }},
 {0x92,50,{goWest, hurryWalk, hurryWalk, hurryWalk, goWalk, hurryWalk, hurryWalk, hurryWalk }},
 {0x91,50,{noWalkNoCars1, noWalkNoCars1, noWalkNoCars1, noWalkNoCars1, noWalkNoCars1, noWalkNoCars1, noWalkNoCars1, noWalkNoCars1  }},
 {0x90,50,{dontWalkNoCars, dontWalkNoCars, dontWalkNoCars, dontWalkNoCars, dontWalkNoCars, dontWalkNoCars, dontWalkNoCars, dontWalkNoCars }},

 {0x91,50,{noWalkNoCars2, noWalkNoCars2, noWalkNoCars2, noWalkNoCars2, noWalkNoCars2, noWalkNoCars2, noWalkNoCars2, noWalkNoCars2 }},
 {0x90,50,{goWest, goWest, goWest, goWest, goWest, goWest, goWest, goWest }}};
#define SENSOR                  (*((volatile unsigned long *)0x4002401C)) //bits 2-0 port E
#define LIGHT                (*((volatile unsigned long *)0x400050FC)) //bits 5-0 port B
  while(1){
       //Moore machine - output based on current state
    LIGHT = FSM[S].Out >> 2;  // set west and south road traffic LED lights (PB5-0)
       GPIO_PORTF_DATA_R = ((FSM[S].Out & 0x2) << 2) | ((FSM[S].Out & 0x1) << 1); // set walk/don't walk leds (PF3 and PF1)
     //wait for time relevant to state
    SysTick_Wait10ms(FSM[S].Time);
       //get input sensors for cars (one for west rd, one for south rd) and one for pedestrian
    Input = SENSOR;     // read sensors (SENSOR defines to read bits PE2-0 ---no need to shift right 2 bits defined this way)
       //Moore machine - next state based on Input and current state
       S = FSM[S].Next[Input];
  }
```

lab10

Just now by Karen West

the students' answer, *where students collectively construct a single answer*

Click to start off the wiki answer

followup discussions *for lingering questions and comments*

Home

Click or ask a question to get started.