|  |  |
| --- | --- |
| MECHATRONICS INTL-PROGRAM | ASSIGNMENT COVERSHEET |

|  |  |  |
| --- | --- | --- |
| STUDENT DETAILS | | |
| STUDENT NAME | 于航 | |
| STUDENT NAME (PINYIN) | YU HANG | |
| STUDENT ID | 17222054 | |
| CLASS NUMBER | 1702 | |
| ASSIGNMENT DETAILS | | |
| COURSE NAME |  | |
| COURSE NUMBER |  | |
| ASSIGNMENT NAME |  | |
| LECTURER/TUTOR |  | |
| SUBMISSION DETAILS | | |
| DATE DUE | 05/26/2020 | |
| SUBMITTION DATE | 05/25/2020 | |
| STAFF ONLY | | |
| RECEIVED BY |  | DATE/TIME |
| MARKER |  | GRADE |

Declaration and Acknowledgement

By submitting this, I declare that:

1. This assignment meets all the requirements of the subject as student in the relevant subject outline, which I have read.
2. (a) This assessment item is entirely my own work, except where I have included fully-documented references to the work of others.

(b) The material contained in this assessment item has not previously been submitted for assessment.

1. I acknowledge that:

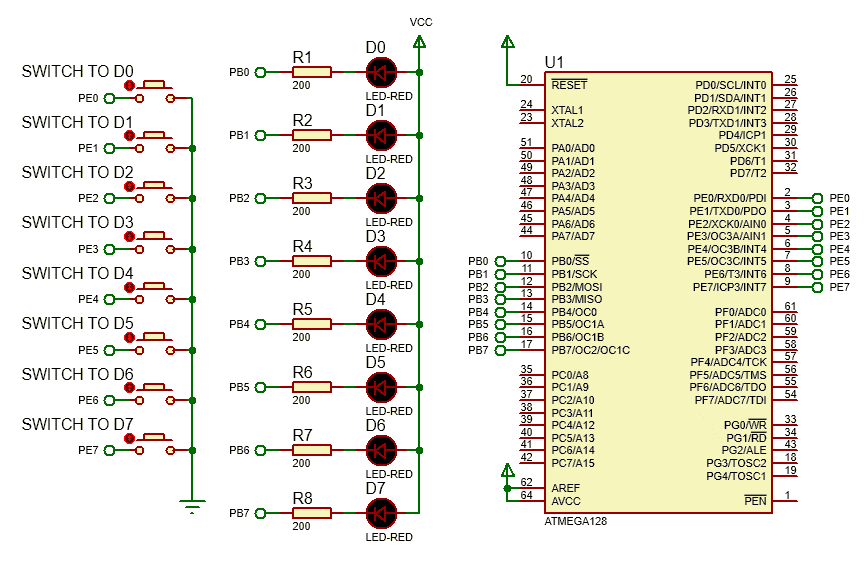
(a) The marker of this assessment item may, for the purpose of assessing this assignment, reproduce this assignment and provide a copy to another member of academic staff.

(b) If required to do so, I will provide an electronic copy of this assessment item to the marker.

1. I am aware that late submission without an authorized extension from the subject coordinator may incur a penalty.

Please note: Assignments are not to be submitted by fax and must be submitted during Lectures/tutorials/laboratories or directly to the academic. Only under special circumstances will the Administrative Staff collect assignments

Experiment 1:LED display

1. Object of experiment
   1. To master the method of writing and debug programs by using Atmeg128 development board and AVR Studio, and operate and be able to use the instrument correctly at the same time.
   2. Be able to explain the function of Atmega128 single chip I/O port, identify the circuit structure of the collar drive.
   3. Be able to write programs to display LED lights, write Atmega128 I/O port input and output program design, write and debug Atmeg128 single-chip microcomputer I/O port control LED lights through experiments.
2. Experiment content
   1. basic experiment
      1. Using the ATmega128 development board, write a program to successively light LED lights with port B as output and display in a cycle.
      2. Write a program to collect the status of the switch from port E, and control the LED lights connected by port B to turn on or off
   2. extended experiment
      1. Write a program to make port B output, need to make LED lights D7 ~ D0 left four lights flash three times, then the right four lights flash three times, finally in the middle four lights flash three times, and loop display.
3. Experimental schematic diagram
4. Code and flow diagram
   1. basic experiment I
      1. flow diagram
      2. code

|  |
| --- |
| 1. #include <iom128v.h> 2. #define LED PORTB 4. unsigned **char** i = 0; 5. unsigned **char** tab[16] = {0XFE,0XFC,0XF8,0XF0,0XE0,0XC0,0X80,   0X00,0X01,0X03,0X07,0X0F,0X1F,0X3F,0X7F,0XFF};   1. //LOGIC TO CONTROL LEDs 3. //to get 1 us delay in 8MHz crystal 4. **void** delay\_us(unsigned **int** microsecond){ 5. **do**{ 6. microsecond--; 7. }**while** (microsecond>1); 8. } 10. // to get 1 ms delay 11. **void** delay\_ms(unsigned **int** millisecond){ 12. **while** (millisecond--){ 13. delay\_us(999); 14. } 15. } 17. **void** port\_init(**void**){ 18. DDRB=0xff; //Set B port as output with pull-up 19. PORTB = 0xFF; 20. DDRE = 0x00; //Set E port as input with pull-up 21. PORTE = 0xFF; 22. } 24. **void** main(**void**){ 25. port\_init(); 26. delay\_ms(100); 27. **while**(1){ 28. **for**(i = 0;i < 16;i ++){ 29. LED = tab[i]; 30. delay\_ms(1000); 31. } 32. } 33. } |

* 1. basic experiment II
     1. Flow diagram
     2. code

|  |
| --- |
| 1. #include <iom128v.h> 2. #define LED PORTB 4. //to get 1 us delay in 8MHz crystal 5. **void** delay\_us(unsigned **int** microsecond){ 6. **do**{ 7. microsecond--; 8. }**while** (microsecond>1); 9. } 11. // to get 1 ms delay 12. **void** delay\_ms(unsigned **int** millisecond){ 13. **while** (millisecond--){ 14. delay\_us(999); 15. } 16. } 18. **void** port\_init(**void**){ 19. DDRB=0xff; //Set  as output 20. PORTB = 0xFF; 21. DDRE = 0x00; //Set E port as input with pull-up 22. PORTE = 0xFF; 23. } 25. **void** main(**void**){ 26. port\_init(); 27. delay\_ms(100); 28. **while**(1) 29. { 31. **if**(PINE!=0XFF){ 32. delay\_ms(100); 33. **if**(PINE!=0XFF){ 34. LED=PINE; 35. delay\_ms(1000); 36. LED=0xff; 37. } 38. } 39. } 40. } |

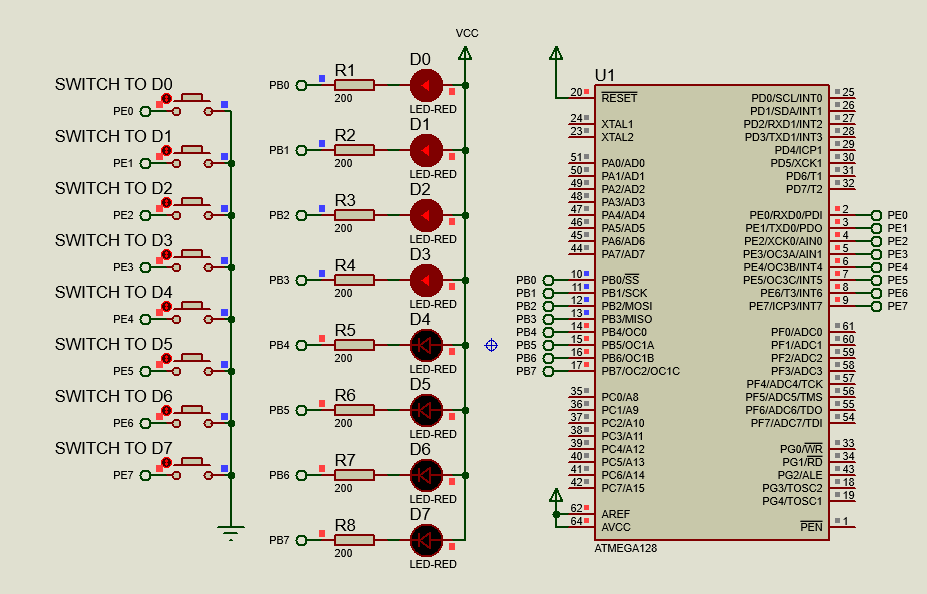
* 1. extended experiment
     1. flow diagram



* + 1. code

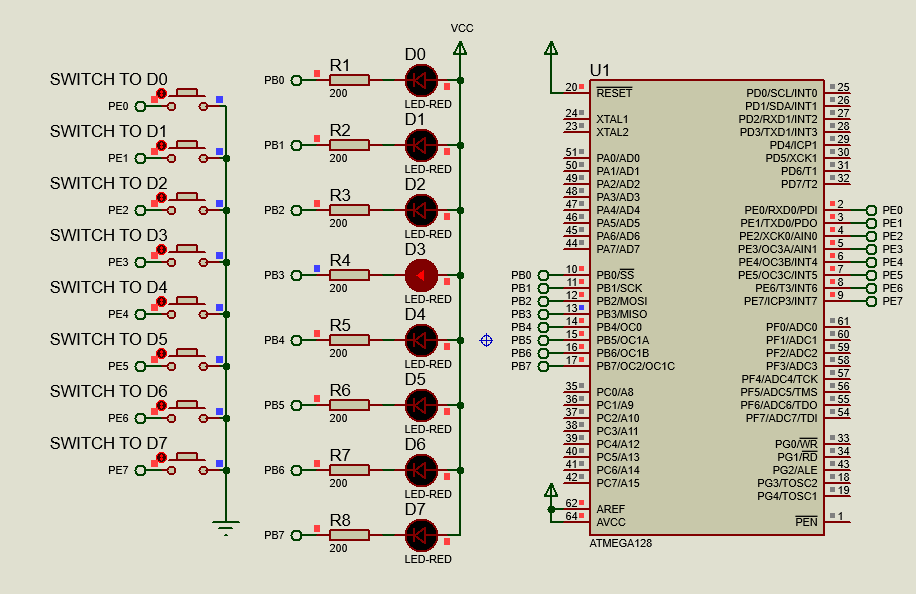
|  |
| --- |
| 1. #include <iom128v.h> 2. #define LED PORTB 4. unsigned **char** i = 0; 6. //to get 1 us delay in 8MHz crystal 7. **void** delay\_us(unsigned **int** microsecond){ 8. **do**{ 9. microsecond--; 10. }**while** (microsecond>1); 11. } 13. // to get 1 ms delay 14. **void** delay\_ms(unsigned **int** millisecond){ 15. **while** (millisecond--){ 16. delay\_us(999); 17. } 18. } 20. **void** port\_init(**void**){ 21. DDRB=0xff; //Set  as output 22. PORTB = 0xFF; 23. } 25. **void** main(**void**){ 26. port\_init(); 27. delay\_ms(100); 28. **while**(1){ 29. **for**(i=0;i<3;i++){ 30. LED=0xf0; 31. delay\_ms(500); 32. LED=0xff; 33. delay\_ms(500); 34. }//D0~D3 flash 3 times 35. **for**(i=0;i<3;i++){ 36. LED=0x0f; 37. delay\_ms(500); 38. LED=0xff; 39. delay\_ms(500); 40. }//D4~D7 flash 3 times 41. **for**(i=0;i<3;i++){ 42. LED=0xc3; 43. delay\_ms(500); 44. LED=0xff; 45. delay\_ms(500); 46. }//D2~D5 flash 3 times 47. } 48. } |

1. Experiment Result
   1. Basic experiment I



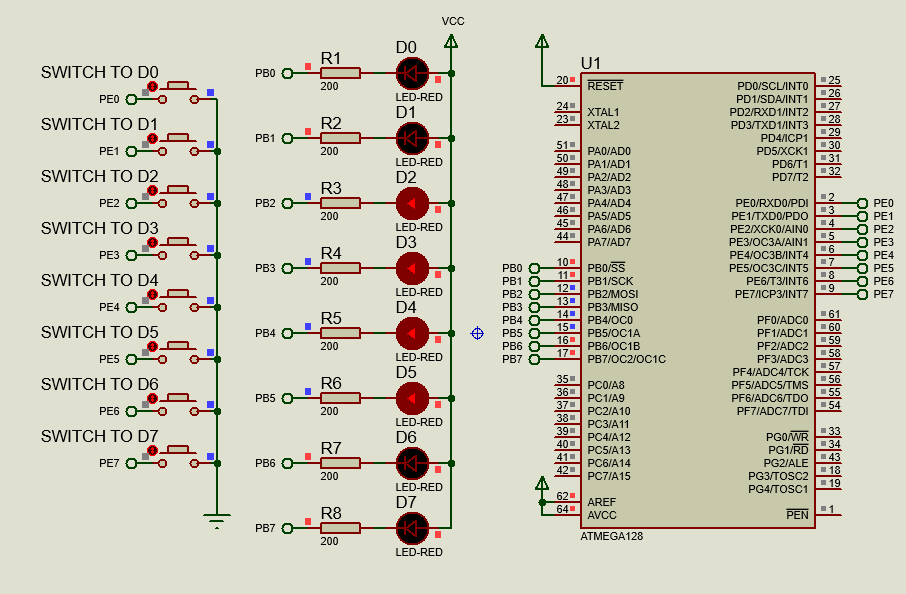
The LED lights are lit from D0 to D7 in turn. The time interval between the two LED lights is 1 second. When all the lights are lit, all the lights will be turned off from D0 to D7. The program is cycled through.

* 1. Basic experiment II



After pressing a button, the corresponding LED light will be lit for 1 second. If no button is pressed, no LED light will be lit.

* 1. Extended experiment



When the power switch is turned on, the LED light D0~D3 will flash three times, then D4~D7 will flash three times, and finally D2~D5 will flash three times. The program is cycled through.

1. Reflection

To initialize the I/O port of Atmega128 microcontroller to realize input and output, we should set registers of the ports. For each port in ATmega128 there are three relevant 8 bit registers:

* 1. Data Direction Register DDRx

DDRx is used to set to configure a specific port pin as output (1) or input (0).

* 1. Input Pin Address PINx

PINx is used to read the input data from port.

* 1. Data Register PORTx

PORTx is used to write the output data to port

To realize input function, we should set DDRx to 0,and if we want a pull-up, PORTx will be set to 1;

To realize output function, we should set DDRx to 1,and PORTx will be set to what we want to output.

1. Experience

In this experiment, by designing the circuit and controlling the display of LED light by the program, I mastered the basic circuit drawing of the single chip microcomputer and the basic I/O port control of the single chip microcomputer. Because there is no MCU development board, so cannot carry out physical operation, only through software simulation results. But the simulation is ideal, not as impressive as the actual operation of the hardware.