|  |  |  |  |
| --- | --- | --- | --- |
| **1주차 결과보고서** | | | |
| **과목명** | 전자회로실험2 | **분반** | 005 |
| **학과** | 전자공학과 | **이름** | **이진호** |
| **제목** | ATmega128 Development Environment & GPIO | | |

**1. 코드 작성**

(1)

|  |
| --- |
| 실습 1  #include <avr/io.h>  #include <util/delay.h>  int main(void)  {      DDRF = 0xFF; //PORTF를 OUTPUT으로 설정      while (1)      {          실습 1          PORTF = 0xFF; //PORTF의 모든 핀을 High로 전환          \_delay\_ms(1000); //1초 딜레이          PORTF = 0x00; //PORTF의 모든 핀을 LOW로 전환          \_delay\_ms(1000); //1초 딜레이  }  } |
|  |
| 과제 1  #include <avr/io.h>  #include <util/delay.h>  int main(void)  {      DDRF = 0xFF; //PORTF를 OUTPUT으로 설정      DDRC = 0x00; //PORTC를 INPUT으로 설정        while (1)      {              if((PINC & 0x01) == 0x01) //1번 PIN을 누를 경우                  PORTF = 0xFF; //PORTF의 모든 핀 HIGH로 전환              else                  PORTF = 0x00; //PORTF의 모든 핀 LOW로 전환              \_delay\_ms(500); //0.5초 딜레이  }  } |
| 과제 2  #include <avr/io.h>  #include <util/delay.h>  int main(void)  {      DDRF = 0xFF; //PORTF를 OUTPUT으로 설정      DDRC = 0x00; //PORTC를 INPUT으로 설정  int flag = 0;        while (1)      {  //1번 핀을 누를 경우, flag 가 0일때          if(((PINC & 0x01) == 0x01) && flag == 0)              PORTF = 0xFF; //PORTF의 모든 핀 HIGH로 전환  flag = 1; //flag 1로 변경  //1번 핀을 누를 경우, flag가 1일때          else if(((PINC & 0x01) == 0x01) && flag == 1){              PORTF = 0x00; //PORTF의 모든 핀 LOW로 전환  flag = 0; //flag 0로 변경          }          \_delay\_ms(500); //delay 0.5초  }  } |
| 과제 3  #include <avr/io.h>  #include <util/delay.h>  int main(void)  {      DDRF = 0xFF; //PORTF를 OUTPUT으로 설정      DDRC = 0x00; //PORTC를 INPUT으로 설정       //1번 스위치를 누를 경우          if((PINC & 0x01)== 0x01){              while(1){  //PORTF의 1번핀 HIGH                  PORTF = 0x01;                  if((PINC & 0x02) == 0x02){                       PORTF = 0x01; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 2번핀 HIGH                  PORTF = 0x02;                  if((PINC & 0x02) == 0x02){                       PORTF = 0x02; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 3번핀 HIGH                  PORTF = 0x04;                  if((PINC & 0x02) == 0x02){                       PORTF = 0x04; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 4번핀 HIGH                  PORTF = 0x08;                  if((PINC & 0x02) == 0x02){                       PORTF = 0x08; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 5번핀 HIGH                  PORTF = 0x10;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x10; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 6번핀 HIGH                  PORTF = 0x20;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x20; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 7번핀 HIGH                  PORTF = 0x40;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x40; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 8번핀 HIGH                  PORTF = 0x80;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x80; break;                  }  //1초 딜레이         \_delay\_ms(1000);  //PORTF의 7번핀 HIGH                  PORTF = 0x40;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x40; break;                  }  //1초 딜레이  \_delay\_ms(1000);  //PORTF의 6번핀 HIGH                  PORTF = 0x20;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x20; break;                  }  //1초 딜레이  \_delay\_ms(1000);  //PORTF의 5번핀 HIGH                  PORTF = 0x10;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x10; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 4번핀 HIGH                  PORTF = 0x08;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x08; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 3번핀 HIGH                  PORTF = 0x04;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x04; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 2번핀 HIGH                  PORTF = 0x02;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x02; break;                  }  //1초 딜레이                  \_delay\_ms(1000);  //PORTF의 1번핀 HIGH                  PORTF = 0x01;  //2번 PIN을 누를 경우 해당 위치 정지                  if((PINC & 0x02) == 0x02){                       PORTF = 0x01; break;                  }              }          }  } |
| 3번 수정  #include <avr/io.h>  #include <util/delay.h>  int main(void)  {      //포지션 1번 핀으로 초기화      int position = 0x01;      //방향 초기화      int direction = 0;      //포트 OUTPUT 설정      DDRF = 0xFF;      //포트 INPUT 설정      DDRC = 0x00;        while (1)      {          //1번 스위치를 눌렀을때          if((PINC & 0x01)== 0x01){              //position 핀 HIGH              PORTF = position;              while(1){                  //방향에 따라 케이스 나눈다.                  if(direction==0){                      //0번 방향                      case 0:                          //8번핀에 도달했을 때                          if(position == 0x80){                              //방향 1로 바꿈                              direction = 1;                          }                          //position 1씩 좌로 이동                          else{                              position = position << 1;                          }                          break;                  }                  //1번 방향                  else if(direction==1){                          //1번핀에 도달했을 때                          if(position == 0x01){                              //방향 0으로 바꿈                              direction = 0;                          }                          else{                              //position 1씩 우로 이동                              position = position >> 1;                          }                          break;                  }                  //2번 스위치를 눌렀을 때 정지                  if((PINC & 0x02) == 0x02) break;                  //1초 딜레이                  \_delay\_ms(1000);              }          }      }  } |

**2. 고찰**

ATmega를 처음 사용해보았기 때문에 아무래도 기본적인 ATmega 소프트웨어에 적응 하는 것이 힘들었다. 그러나 기본적으로 C언어를 이용하여 코드를 짜기 때문에 코드를 짜는 것은 익숙했다. 1번 과제에서는 포트C에서 pin 1번을 누를 경우 포트 F에서 모든 핀을 High로 전환 하고, 그 이외의 경우에서는 포트 F의 모든 핀을 low로 전환하는 코드를 만들었고 동작하였다.

2번 과제에서는 스위치를 누르면 모든 LED가 껏다켜졌다 하는 것을 구현 하는 것이 였는데, 간단하게 flag를 이용해서 스위치를 눌렀을 때 동작을 달리 하도록 구분하였다. 3번 과제에서는 처음에는 단지 나열 하여서 동작을 하도록 구성하였지만, 계속 고민하다 16진수의 비트 연산자를 이용하여 핀의 위치를 position에 저장하고 핀을 이동하도록 하였다. 또한 왕복을 하기 때문에 핀의 위치가 마지막이나 처음에 도달하였을 때 direction을 바꿔 비트 연산을 반대로 전환하도록 하였다. 실습 환경이 완벽하지 못하여 도중에 컴퓨터를 바꿔서 실험을 하였기에 시간이 오래걸렸던 것 같다. 이에 대한 대비를 실험 전에 미리 확인하고 다음실험에 임해야겠다.