|  |  |  |  |
| --- | --- | --- | --- |
| **2주차 결과보고서** | | | |
| **과목명** | 전자회로실험2 | **분반** | 005 |
| **학과** | 전자공학과 | **이름** | **이진호** |
| **제목** | ATmega128 USART | | |

**1. 코드 작성**

(1)

|  |
| --- |
| #define F\_CPU 16000000UL //16MHz 설정  #define BAUD 9600 //9600bfs 설정  #define MYUBRR F\_CPU/16/BAUD-1 //UBRR 계산  #include <util/delay.h>  #include <avr/io.h>  // USART 초기화 함수  void USART\_Init(unsigned int ubrr){      UBRR0H = (unsigned char)(ubrr >> 8); //0으로 초기화      UBRR0L = (unsigned char)ubrr; //그대로 출력      UCSR0B =(1<<RXEN0) | (1<<TXEN0); //송수신 가능      //0000 0000 0000 0011 => 0000 0000 0000 1100  char size 12로 설정      UCSR0C = (3<< UCSZ0);    }  // USART 데이터 전송 함수  void USART\_Transmit(char data){      while(!((UCSR0A)&(1<<UDRE0))); //입력이 안들어오면 반복      UDR0 = data; //UDR0에 데이터 저장  }  // USART 데이터 수신 함수  char USART\_Receive(){      while(!(UCSR0A &(1<<RXC0))); //입력이 안들어오면 반복      return UDR0; //UDR0 반환  }  //메인 함수  int main(void)  {      DDRF=0x00; //PORTF를 INPUT으로 설정      DDRA=0xFF; //PORTA를 OUTPUT으로 설정      USART\_Init(MYUBRR); //UBRR 초기화        while (1)      {          char a = USART\_Receive(); //USART에 입력되는 값을 char 에 저장            int num=a-48; //아스키 코드 변환하여 저장            PORTA=num; //PORTA의 핀에 변환 된 값을 출력            \_delay\_ms(100); //0.1초 딜레이      }  } |
|  |
| 과제 2  #define F\_CPU 16000000UL //16MHz 설정  #define BAUD 9600 //9600bfs 설정  #define MYUBRR F\_CPU/16/BAUD-1 //UBRR 계산  #include <util/delay.h>  #include <avr/io.h>  // USART 초기화 함수  void USART\_Init(unsigned int ubrr){      UBRR0H = (unsigned char)(ubrr >> 8); //0으로 초기화      UBRR0L = (unsigned char)ubrr; //그대로 출력      UCSR0B =(1<<RXEN0) | (1<<TXEN0); //송수신 가능      //0000 0000 0000 0011 -> 0000 0000 0000 1100 char size 12로 설정      UCSR0C = (3<< UCSZ0);    }  // USART 데이터 전송 함수  void USART\_Transmit(char data){      while(!((UCSR0A)&(1<<UDRE0))); //입력이 안들어오면 반복      UDR0 = data; //UDR0에 데이터 저장  }  // USART 데이터 수신 함수  char USART\_Receive(){      while(!(UCSR0A &(1<<RXC0))); //입력이 안들어오면 반복      return UDR0; //UDR0 반환  }  //메인 함수  int main(void)  {      DDRF=0x00; //PORTF를 INPUT으로 설정      DDRA=0xFF; //PORTA를 OUTPUT으로 설정      USART\_Init(MYUBRR); //UBRR 초기화      char showPIN;        while (1)      {          //PINA에 입력이 들어왔을 때          if(PINA){              PORTF=PINA; //PINA의 입력 값을 포트F의 핀에 출력              //PINA의 입력 값에 맞는 값을 showPIN에 저장              switch(PINA){                  case 0x01:                      showPIN='1'; break;                  case 0x02:                      showPIN='2'; break;                  case 0x04:                      showPIN='3'; break;                  case 0x08:                      showPIN='4'; break;                  case 0x10:                      showPIN='5'; break;                  case 0x20:                      showPIN='6'; break;                  case 0x40:                      showPIN='7'; break;                  case 0x80:                      showPIN='8'; break;              }              USART\_Transmit(showPIN); //showPIN의 값을 시리얼 모니터에 출력          }          \_delay\_ms(100); //0.1초 딜레이      }  } |

**2. 고찰**

이번 실험에서는 ATMega에서 USART 시리얼 통신을 이용한 프로그래밍 실습을 진행하였다. 먼저 시리얼과 통신이 가능하도록 초기 설정을 𝑈𝐵𝑅𝑅 = 𝑓𝑂𝑆𝐶 16(𝐵𝐴𝑈𝐷) 𝑒𝑥) 𝑓𝑂𝑆𝐶 = 16𝑀𝐻𝑧,𝐵𝐴𝑈𝐷 = 9600𝑏𝑝𝑠, 𝐴𝑠𝑦𝑛𝑐 − 1 = 103 로 하였다.

그 후 USART 초기화 함수, 데이터 전송 수신 함수를 각각 만들어 메인에서 함수를 호출 하여 데이터 송수신을 가능하도록 구현하였다. 이를 기반으로 첫번째 과제는 터미널에서 숫자를 입력받아 2진수로 변환한 뒤, 변환된 숫자를 LED로 표현하는 것이였다. 이는 char값으로 들어오는 값을 int형으로 저장할때 숫자 0에 해당하는 아스키코드값을 빼주어 저장하고 이를 포트의 핀에 출력하는 식으로 해결하였다. 두번째 과제는 스위치를 누르면 어떤 스위치가 눌렸는지 확인하고, 터미널에 해당 스위치 번호를 띄우고, LED를 켜는 것이였다. 이는 PIN에 입력이 들어왔을 때, 이 값을 포트의 핀에 바로 출력 하여 스위치 번호를 보여주게 하고, 터미널에는 입력한 PIN의 값을 스위치형식으로 맞는 값을 찾아 char 형식으로 저장한 후 시리얼 모니터에 출력하는 식으로 구성하였다. 아무래도 첫번째 실험 시간에 두자리 수를 구현하지 못한것이 아쉽지만 아마도 두자리 수를 구현하기 위해서 문자열로 특정한 키를 누를때까지 값을 받은 후 저장하고, 이를 다시 아스키 코드로 변환하여 LED에 2진수로 출력하면 될것으로 예상된다.