

**전자회로실험 보고서**

학과 전자공학과

학번 1218165 /12161509

이름 김재욱/ 최장호

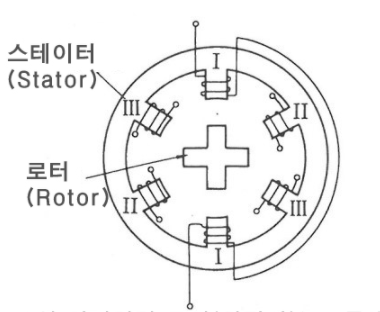
제출일 2020-11-24

stepping moter

**실습 이론**

한 바퀴의 회전을 여러 스텝으로 나누어 구동하며 한 펄스가 입력될때 마다 일정한 각도로 모터가 회전하도록 제어한다.

stepping moter 의 원리



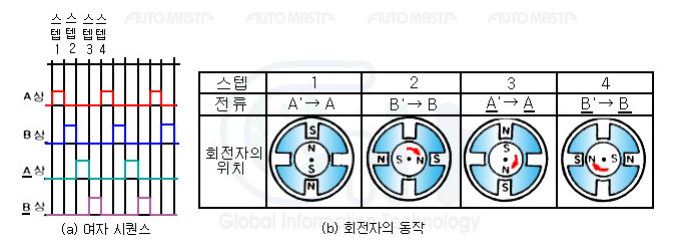
strpping moter는 회전축의 로터라는 자석과 바깥쪽에 코일로 감겨진 스테이터 라는 전자석으로 이뤄져 있더. 스테이터에 감겨진 코일에 펄스 전류를 흘리면 자기력이 발생해 N극은 S극을 끌어당기고 S극은 N극을 끌어당기는 과정을 반복해 모터가 회전한다.

로터가 한바퀴를 도는 것을 한 주기라 한다면 한 주기의 회전은 모터의 각을 1.8도 움직인다.

stepping moter 의 종류

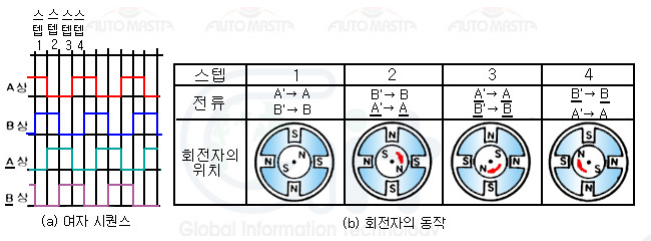
1상 여자 STEPPING MOTER

항상 하나의 상(스테이터))에만 전류를 흐르게 하는 방식이며 한 스텝당 90도씩 움직인다



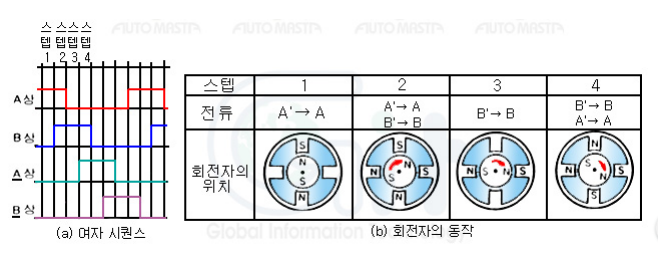
2상 여자 STEPPING MOTER

항상 두개의 상(스테이터)에만 전류를 흐르게 하는 방식이며 한 스텝당 90도씩 움직인다



1-2상 여자 STEPPING MOTER

하나의 상(스테이터)과 두개의 상(스테이터)을 번갈아 전류를 흐르게 하는 방식이며 한 스텝당 45도씩 움직인다



**실습내용 및 결과**

실습1

|  |
| --- |
|  |
| /\*  \* Week11-1.c  \*  \* Created: 2020-11-20 오전 11:14:25  \* Author : lobgd  \*/  #define F\_CPU 16000000  #include <avr/io.h>  #include <util/delay.h>  int main(){  DDRE = 0xff ;  PORTE = 0x33;  while(1){  PORTE = 0x66;  \_delay\_us(50);  PORTE = 0xcc;  \_delay\_us(50);  PORTE = 0x99;  \_delay\_us(50);  PORTE = 0x33;  \_delay\_us(50);  }  return 0;  } |
| 오프라인 수업에서 조교님께 결과를 확인 받음. |

실습2

|  |
| --- |
|  |
| /\*  \* Week11-2.c  \*  \* Created: 2020-11-20 오전 11:39:18  \* Author : lobgd  \*/  #define F\_CPU 16000000  #define \_\_DELAY\_BACKWARD\_COMPATIBLE\_\_  #include <avr/io.h>  #include <util/delay.h>  #include <avr/interrupt.h>  unsigned char motor;  unsigned char speed;  unsigned char a;  ISR(INT0\_vect){  if (motor==1) {  motor = 0;  }  else {  motor = 1;  }  \_delay\_ms(50);  }  ISR(INT1\_vect){  if(speed != 1){speed -=1;}  \_delay\_ms(500);  }  ISR(INT2\_vect){speed +=1;  \_delay\_ms(50);  }  void motor\_on(){  if(motor){switch(a){  case 1:  PORTE = 0x66;a+=1;\_delay\_us(100\*speed);break;  case 2:  PORTE = 0xcc;a+=1;\_delay\_us(100\*speed);break;  case 3:  PORTE = 0x99;a+=1;\_delay\_us(100\*speed);break;  case 4:  PORTE = 0x33;a=1;\_delay\_us(100\*speed);break;  }  }  else{switch(a){  case 1:  PORTE = 0x33;a=4;\_delay\_us(100\*speed);break;  case 2:  PORTE = 0x66;a-=1;\_delay\_us(100\*speed);break;  case 3:  PORTE = 0xcc;a-=1;\_delay\_us(100\*speed);break;  case 4:  PORTE = 0x99;a-=1;\_delay\_us(100\*speed);break;}  }  }  int main(){  DDRE = 0xff ;  PORTE = 0x33;  DDRD = 0b00000000;  EICRA = 0b11111111;  EIMSK = 0b00001111;  SREG |= 0x80;  speed = 5;  a = 1;  while(1){  motor\_on();  }  } |
| 오프라인 수업에서 조교님께 결과를 확인 받음. |

실습 3

|  |
| --- |
|  |
| #define F\_CPU 16000000  #define \_\_DELAY\_BACKWARD\_COMPATIBLE\_\_  #include <avr/io.h>  #include <util/delay.h>  #include <avr/interrupt.h>  unsigned char motor;  unsigned char speed;  unsigned char a;  unsigned count = 0;  ISR(INT0\_vect){  forward2();  count=0;  \_delay\_ms(50);  }  ISR(INT1\_vect){  reverse2();  count=0;  \_delay\_ms(50);  }  ISR(INT2\_vect){  forward12();  count=0;  \_delay\_ms(50);  }  ISR(INT3\_vect){  reverse12();  count=0;  \_delay\_ms(50);  }  void forward2(){  while(count<1650){  switch(a){  case 1:  PORTE = 0x66;a+=1;\_delay\_ms(100\*speed);break;  case 2:  PORTE = 0xcc;a+=1;\_delay\_ms(100\*speed);break;  case 3:  PORTE = 0x99;a+=1;\_delay\_ms(100\*speed);break;  case 4:  PORTE = 0x33;a=1;\_delay\_ms(100\*speed);break;  }  count++;  }  }  void reverse2(){  while(count<6600){  switch(a){  case 1:  PORTE = 0x33;a=4;\_delay\_ms(100\*speed);break;  case 2:  PORTE = 0x66;a-=1;\_delay\_ms(100\*speed);break;  case 3:  PORTE = 0xcc;a-=1;\_delay\_ms(100\*speed);break;  case 4:  PORTE = 0x99;a-=1;\_delay\_ms(100\*speed);break;  }  count++;  }    }  void forward12(){  switch(a){  case 1:  PORTE = 0b00000001;a+=1;\_delay\_ms(100\*speed);break;  case 2:  PORTE = 0b00000011;a+=1;\_delay\_ms(100\*speed);break;  case 3:  PORTE = 0b00000010;a+=1;\_delay\_ms(100\*speed);break;  case 4:  PORTE = 0b00000110;a+=1;\_delay\_ms(100\*speed);break;  case 5:  PORTE = 0b00000100;a+=1;\_delay\_ms(100\*speed);break;  case 6:  PORTE = 0b00001100;a+=1;\_delay\_ms(100\*speed);break;  case 7:  PORTE = 0b00001000;a+=1;\_delay\_ms(100\*speed);break;  case 8:  PORTE = 0b00001001;a=1;\_delay\_ms(100\*speed);break;  }  }  void reverse12(){  while(count<300){  switch(a){  case 1:  PORTE = 0b00000001;a=8;\_delay\_us(100\*speed);break;  case 2:  PORTE = 0b00000011;a-=1;\_delay\_us(100\*speed);break;  case 3:  PORTE = 0b00000010;a-=1;\_delay\_us(100\*speed);break;  case 4:  PORTE = 0b00000110;a-=1;\_delay\_us(100\*speed);break;  case 5:  PORTE = 0b00000100;a-=1;\_delay\_us(100\*speed);break;  case 6:  PORTE = 0b00001100;a-=1;\_delay\_us(100\*speed);break;  case 7:  PORTE = 0b00001000;a-=1;\_delay\_us(100\*speed);break;  case 8:  PORTE = 0b00001001;a-=1;\_delay\_us(100\*speed);break;  }  count++;  }  }  int main(){  DDRE = 0xff ;  PORTE = 0x33;  DDRD = 0b00000000;  PORTD = 0b11111111;  EICRA = 0b11111111;  EIMSK = 0b00001111;  SREG |= 0x80;  speed = 5;  a = 1;  while(1){  }  } |
| 2상 여자방식으로만 조교님께 확인 받음 |

**찰**

**고찰**

atmega128 PORTE에 DC모터 컨트롤러를 연결하였다.

2상여자 방식으로 실습 1과 2를 하였는데 실습을 하면서 여러 문제들이 발생하였다.

먼저 실습 1은 스테핑 모터를 2상여자 기반으로 360도 회전 하는 것 이었다.

여기서 문제는 delay가 ms단위로 되어있을 때 모터의 회전이 거의 보이지 않는다는 것 이었다.

이 문제는 delay 단위를 us단위로 바꿔주어 해결을 하였다.

두 번째 실습은 버튼에 따라 회전속도 증가, 회전속도 감소, 회전 방향변화를 하게 하는 것 이었다.

이번에도 ms를 us로 바꿔주면 문제는 해결이 되었다.

마지막 실습은 버튼에 따라 2상여자 기반 시계방향 90도 회전, 2상여자기반 반시계 180도 회전, 1-2상 여자 기반 시계방향 90도, 1-2상 여자기반 반시계 방향 180도 회전이었다.

여기서 생각보다 많은 문제를 직면했었다. 먼저 버튼 1,2 에 따른 2여상 기반으로 회전하는 과정을 구현하기 위해,

이론대로 원하는 각도만큼 원하는 방향으로 회전을 하도록 구현을 하였다.

예를들어 PORTE = 0x66; \_delay\_us(200); PORTE = 0xcc; \_delay\_us(200); 이런식으로 하여 시계방향으로 90도 회전하게 하였으나 모터는 반응을 하지 않았다.

그러던 도중 프로테우스와는 다르게 실제 스테핑 모터는 한싸이클을 도는 코드를 구현했을 때 약 1.8도 회전한다는 걸 알게되었고, 50번의 싸이클을 반복하도록 하였으나, 원하는 만큼 회전하지 않아서 임의로 반복해야하는 횟수를 추정한 결과1650번을 반복하면 약90도를 시계방향으로 회전하였다.

그리고 이 데이터를 기반으로 반시계방향은 180도를 회전하여야하기 때문에 3300번의 반복을 하였으나 실제로 반복해야하는 값은 6600회다.

그 후 3번 4번 버튼의 과정을 수행하기 위해, 이론상 1-2상 여자 방식으로 코드를 구현하여 실행해 보았으나 모터는 반응을 하지 않았다.

그 이유는 실험에서 사용한 모터는 2상여자 기반이기 때문이었다.

그리하여 프로테우스로 3번 4번 버튼에 대한 실험만 실행한 결과 90도회전을 하도록 하여도, 180도 회전을 하도록 하여도 -90도에서 시작하여 -45도에서 멈추는 과정만을 반복하였다.