PERCOBAAN 8 MOTOR STEPPER

TUJUAN

- 1. Memahami rangkaian interface mikrokontroller dengan motor stepper
- 2. Memahami rangkaian driver motor stepper ULN2003
- 3. Memahami bahasa assembly untuk mengatur arah putaran motor stepper
- 4. Memahami bahasa assembly untuk mengatur arah putaran motor stepper dengan menggunakan saklar.



Gambar 8.1. Rangkaian interface motor stepper dengan driver ULN2003

DASAR TEORI

Pada Motor DC biasa, akan berputar dan berputar terus selama power supply ada. Tidak ada rangkaian cerdas tertentu yang diperlukan untuk mengendalikan motor tersebut, kecuali hanya memperlambat putaran atau membalik putaran, dengan menerapkan polaritas balik. Motor stepper adalah sangat berbeda. Jika anda memberikan power pada motor ini, maka motor ini akan berada dalam keadaan diam, agar motor dapat berputar, anda harus merubah sinyal yang masuk ke motor. Sebagai ilustrasi, dapat dibayangkan sebuah kompas dengan elektromagnet disekitarnya. Sebagaimana digambarkan pada gambar 8.2., apabila power yang diberikan pada elektromagnet diganti, maka akan merubah posisi jarum dari kompas.



Gambar 8.2. Ilustrasi sebuah kompas dengan elektromagnet

Dengan empat buah elektromagnet maka gerakan akan melompat secara kasar. Sekarang bayangkan susunan yang sama dengan 100 elektromagnet yang mengitari kompas. Dengan mangatur energi yang mengalir pada setiap elektromagnet dalam berurutan, maka jarum akan memerlukan sebanyak 100 langkah untuk melakukan satu kali putaran. Tetapi dengan pengaturan 100 elektromagnet secara individu, akan memerlukan elektronika yang kompleks.



Gambar 8.3. Ilustrasi motor stepper dengan jarum kompas dengan elektromagnet

Pada ilustrasi tersebut, huruf-huruf yang melingkar mewakili elektromagnet. Semua magnet dengan huruf yang sama berada dalam keadaan koneksi. Ketika anda memberi arus pada rangkaian tersebut, maka semua elektromagnet dengan huruf yang sama akan on pada saat itu, untuk menggerakkan kompas, maka elektromaget berikutnya harus dialiri arus, sehingga akan menimbulkan gerakan.

Α	В	С	D	KOMENTAR
1	0	0	0	Take a step clock wise
0	1	0	0	another step clock wise
0	0	1	0	another step clock wise
0	0	0	1	another step clock wise
0	0	0	1	No step take
0	0	1	0	Take a step back

Table 8.1. Full Step Mode

Mode Half Steps

Dengan menghidupkan dua koil pada waktu yang bersamaan maka motor akan berada dalam posisi diantaranya.



Gambar 8.4. Half step mode

Table 2.7.2. Half Step Mode

Α	В	С	D	KOMENTAR
1	0	0	0	Take a step clock wise
1	1	0	0	Half a step clock wise
0	1	0	0	The complete full step clock wise
0	1	1	0	another half step clock wise
0	0	1	0	The complete full step clock wise
0	0	1	1	Another half step clock wise
0	0	0	1	The complete full step clock wise
1	0	0	1	another half step clock wise
1	0	0	0	Start position



Gambar 8.3. Bentuk fisik motor stepper disk drive 1,2"

Percobaan 8.1.

Penggerak Motor Putar Searah Jarum Jam

Pada percobaan ini, motor stepper akan berputar searah jarum jam, terus menerus Untuk melakukan percobaan ini lakukan beberapa langkah sebagai berikut:

- 1. Hubungkan kabel parallel antara P0 dengan motor stepper
- 2. Hubungkan modul Microcontroller Trainer dengan power supply +5V
- 3. Hubungkan modul Microcontroller Trainer dengan rangkaian programmer
- 4. Buka Program M-IDE Studio for MCS-51, sebagai editor dan compiler program
- 5. Ketik program berikut ini:

```
org Oh
start: call StepCW
                   sjmp start
StepCW:mov P0,#11101111b ; Turn on driver 1
                  call delay ; call delay time
mov P0,#11011111b ; Turn on driver 2
call delay ; call delay time
mov P0,#10111111b ; Turn on driver 3
call delay ; call delay time
mov P0,#01111111b ; Turn on driver 4
call delay ; call delay time
                   ret
                   ;
StepCCW:
                  mov P0,#0111111b ; Turn on driver 1
call delay ; call delay time
mov P0,#1011111b ; Turn on driver 2
call delay ; call delay time
mov P0,#1101111b ; Turn on driver 3
call delay ; call delay time
mov P0,#11101111b ; Turn on driver 4
call delay ; call delay time
mot pate
                  ret
delay: mov R0,#255
delay1:mov R2,#255
                   djnz R2,$
                   djnz R0,delay1
                   ret
                   end
```

- 6. Simpanlah program yang anda ketik dan beri nama : prog81a.asm
- 7. Pada program MIDE tersebut pilih Build /F9 atau untuk melakukan kompilasi program dari *.asm ke *.hex.
- 8. Lakukan pemrograman mikrokontroller dengan menggunakan Program ISP Software (Lihat Petunjuk Penggunaan)
- 9. Lakukan pengamatan arah putaran motor stepper apakah sudah sesuai ?
- 10. Lakukan modifikasi program tersebut diatas, untuk putaran motor yang berlawanan arah jarum jam CCW.

Percobaan 8.2 Pengaturan Arah Putaran Motor Melalui Saklar

Pada percobaan ini, motor stepper akan berputar searah jarum jam, atau berlawanan arah jarum jam bergantung pada posisi push button yang ditekan (P2.0 atau P2.1) secara terus menerus.



Untuk melakukan percobaan ini lakukan beberapa langkah sebagai berikut:

- 1. Hubungkan kabel parallel antara P0 dengan motor stepper
- 2. Hubungkan modul Microcontroller Trainer dengan power supply +5V
- 3. Hubungkan modul Microcontroller Trainer dengan rangkaian programmer
- 4. Buka Program M-IDE Studio for MCS-51, sebagai editor dan compiler program
- 5. Ketik program berikut ini:

```
org Oh
start:
CW:
      JB P2.0,CCW
                     ; SW1
      call stepCW
      sjmp start
       ;
CCW:
      JB P2.1,CW
                     ;SW2
      Call stepCCW
      Sjmp start
       ;
StepCW:
      mov P0,#11101111b
                           ; Turn on driver 1
      call delay
                           ; call delay time
      mov P0,#11011111b
                           ; Turn on driver 2
      call delay
                           ; call delay time
      mov P0,#10111111b
                          ; Turn on driver 3
```

```
call delay ; call delay time
mov P0,#0111111b ; Turn on driver 4
call delay ; call delay time
ret
;
StepCCW:
  mov P0,#0111111b ; Turn on driver 1
call delay ; call delay time
mov P0,#1011111b ; Turn on driver 2
call delay ; call delay time
mov P0,#1101111b ; Turn on driver 3
call delay ; call delay time
mov P0,#11101111b ; Turn on driver 4
call delay ; call delay time
ret
;
delay: mov R0,#255
delay1:mov R2,#255
djnz R2,$
djnz R2,$
djnz R0,delay1
ret
end
```

- 6. Simpanlah program yang anda ketik dan beri nama : prog82a.asm
- 7. Pada program MIDE tersebut pilih Build /F9 atau untuk melakukan kompilasi program dari *.asm ke *.hex.
- Lakukan pemrograman mikrokontroller dengan menggunakan Program ISP Software (Lihat Petunjuk Penggunaan)
- 9. Lakukan pengamatan arah putaran motor stepper apakah sudah sesuai ?
- 10. Lakukan modifikasi program tersebut diatas, saklar push button yang lain.