

```
/*
```

```
FILE: STEPPER.C Adapted from Stepper.c by P.Oh  
DESC: Speed and Positioning options for Stepper Motor
```

```
Port A has 5 lines attached and described as follows:
```

```
OP = PHASE    Line A.0  1  enable / 0 disable
```

```
HS = HALF STEP  Line A.1  2  enable / 0 disable
```

```
SI = STEP INPUT  Line A.2  4  enable / 0 disable
```

```
DR = DIRECTION  Line A.3  8  enable / 0 disable
```

```
OE : OUTPUT ENABLE Line A.4 16 enable / 0 disable
```

```
*/
```

```
#include<stdio.h>
```

```
#include<stdlib.h>
```

```
#include<dos.h>    /* outportb, inportb defined here    */
```

```
#include<conio.h>  /* formatted text functions defined here */
```

```
void StepTheMotor(int, int, int);
```

```
void Selection(int, int, int);
```

```
void MoveToFixedAngle(int, int, int);
```

```
void Continuous(int, int, int);
```

```
void main(void) {
```

```
int Sgnal;

int BASEADDR;

int PORTA, PORTB, PORTC;

int CNTRL;

int Choice;

int StepsPerRev;

int DR = 0;

clrscr();      /* clear screen */

window(5,5,75,30); /* set up text window */

gotoxy(1,1);

printf("Enter Base Address (decimal) e.g. 608\n");

gotoxy(1,2); scanf("%d", &BASEADDR);

PORTA = BASEADDR;

PORTB = BASEADDR + 1;

PORTC = BASEADDR + 2;

CNTRL = BASEADDR + 3;

outportb(CNTRL, 128); /* configure all ports for output */

outportb(PORTB, 0); /* Ports B and C not used, so just set to 0 */
```

```
outportb(PORTC, 0);
```

```
StepsPerRev = 96;
```

```
Selection(PORTA, StepsPerRev, DR);
```

```
} /* end of main */
```

```
void Selection(int PORTA, int StepsPerRev, int DR) {
```

```
    /* Motor speed and position options */
```

```
    int DegPerStep;
```

```
    int Choice;
```

```
    clrscr();
```

```
    gotoxy(1,4); printf("Motor requires %d steps per rev\n", StepsPerRev);
```

```
    DegPerStep = (int)(360 / StepsPerRev);
```

```
    gotoxy(40,4); printf("=> thus 1 step = %d degrees\n", DegPerStep);
```

```
    do {
```

```
        gotoxy(1,6); printf("(1) To rotate to a certain angle clockwise\n");
```

```

gotoxy(1,7); printf("(2) To rotate to a certain angle counter-clockwise\n");
gotoxy(1,8); printf("(3) To rotate continuously - with speed options\n");
gotoxy(1,9); printf("(4) To quit\n");
gotoxy(1,10); printf("Selection =>\n");
gotoxy(14,10); scanf("%d", &Choice);

switch(Choice) {
    case 1 : DR = 0;
        MoveToFixedAngle(PORTA, DR, DegPerStep);
        break;
    case 2 : DR = 8;
        MoveToFixedAngle(PORTA, DR, DegPerStep);
        break;
    case 3 : Continuous(PORTA, DR, StepsPerRev);
        break;
    case 4 : outportb(PORTA, 16); /* shutdown all signals */
        gotoxy(1,22); printf("Quitting\n");
        exit(0);
    default : gotoxy(1,14); printf("Choose 1, 2, 3, or 4\n");
        break;
}; /* end select */
} while (1); /* can only exit Selection if user selects 4 */

```

```
}; /* end of Selection */
```

```
void MoveToFixedAngle(int PORTA, int DR, int DegPerStep) {
```

```
    int Degrees;
```

```
    int NumberOfSteps;
```

```
    char *GetKey[2];
```

```
    int WaitSomeTime = 0;
```

```
    int i;
```

```
    clrscr();
```

```
    if(DR == 0) {
```

```
        gotoxy(1,4); printf("Will step CW fixed number of degrees\n");
```

```
    } else { /* DR is 8 */
```

```
        gotoxy(1,4); printf("Will step CCW fixed number of degrees\n");
```

```
    };
```

```
    gotoxy(1,5); printf("Enter number of degrees => ");
```

```
    gotoxy(28,5); scanf("%d", &Degrees);
```

```
    gotoxy(1,7); printf("Will step %d degrees", Degrees);
```

```
    gotoxy(1,8); printf("Hit a key to start");
```

```
    while (!kbhit());
```

```

NumberOfSteps = (int)(Degrees / DegPerStep);

for(i = 1; i <= NumberOfSteps; i++) {
    StepTheMotor(PORTA, DR, WaitSomeTime);
};

gotoxy(1,10); cprintf("Finished!");

gotoxy(1,11); cprintf("Hit a key then <ENTER> to do another angle");

gotoxy(42,11); cprintf("or q then <ENTER> to quit to main menu");

scanf("%s", &GetKey);

if(GetKey[0] == 'q' || GetKey[0] == 'Q') {
    clrscr();
    return;
} else {
    MoveToFixedAngle(PORTA, DR, DegPerStep);
};

}; /* end of MoveToFixedAngle */

void Continuous(int PORTA, int DR, int StepsPerRev) {
    int WaitSomeTime;

    int TapKey;

    int SpeedGrade;

    int OE;

```

```
clrscr();

gotoxy(1,2); cprintf("Tap a key:");

gotoxy(1,3); cprintf("(f)aster");

gotoxy(1,4); cprintf("(s)lower");

gotoxy(1,5); cprintf("(r)everse");

gotoxy(1,6); cprintf("(q)uit to main menu");

gotoxy(1,7);

WaitSomeTime = 0;

SpeedGrade = 1;

do {

    do {

        StepTheMotor(PORTA, DR, WaitSomeTime);

    } while(!kbhit());

    TapKey = getch();

    switch(TapKey) {

        case 102 : /* hit f key */

            WaitSomeTime = WaitSomeTime + 5;

            if(WaitSomeTime >= 100) {

                WaitSomeTime = 100;

                gotoxy(1,10); cprintf("Can't go faster");

                SpeedGrade = 20;

            }

        }

    }
```

```

        } else {

            gotoxy(1,10); cprintf("Going faster...");

            SpeedGrade = SpeedGrade + 1;

        };

        gotoxy(1,12); cprintf("speed grade = %2d", SpeedGrade);

        StepTheMotor(PORTA, DR, WaitSomeTime);

        break;

case 115 : /* hit s key */

    WaitSomeTime = WaitSomeTime - 5;

    if(WaitSomeTime <= 0) {

        WaitSomeTime = 0;

        gotoxy(1,10); cprintf("Won't go slower");

        SpeedGrade = 0;

    } else {

        gotoxy(1,10); cprintf("Going slower...");

        SpeedGrade = SpeedGrade - 1;

    };

    gotoxy(1,12); cprintf("speed grade = %2d", SpeedGrade);

    StepTheMotor(PORTA, DR, WaitSomeTime);

    break;

case 114 : /* hit the r key */

    gotoxy(1,14); cprintf("Reversing direction");

    delay(500);

```

```

        gotoxy(1,14); cprintf("          ");

        gotoxy(1,7);

        if(DR == 0) {
            DR = 8;

        } else {
            DR = 0;

        };

        StepTheMotor(PORTA, DR, WaitSomeTime);

        break;

    case 113 : /* hit the q key */

        OE = 16;

        outportb(PORTA, OE);

        clrscr();

        Selection(PORTA, StepsPerRev, DR);

        break;

    default : break;

};

} while (1);

}; /* end of Continuous */

void StepTheMotor(int PORTA, int DR, int WaitSomeTime) {

```

```
int i;

int SI = 4;

int OE = 0;

int FixedTime = 100; /* 100 msec */

outportb(PORTA, SI + DR + OE);

SI = 0;

OE = 0;

delay(FixedTime-WaitSomeTime); /* 100 msec */

outportb(PORTA, SI + DR + OE);

return;

}; /* --- end of StepTheMotor --- */
```