รายการอ้างอิง

ภาษาไทย

วริทธิ์ อึ๊งภากรณ์ ศ.ดร. และ วิบูลย์ แสงวีระพันธุ์ศิริ ผศ.ดร. . รายงานผลการวิจัยเรื่องการออกแบบและควบคุมหุ่นยนต์อุตสาหกรรมชนิด 3 ข้อต่อที่มีการเคลื่อนที่แบบพิกัดฉาก ม.ป.ท. , 2535.

วิบูลย์ แสงวีระพันธุ์ศิริ รศ.ดร. . การควบคุมระบบพลศาสตร์ . พิมพ์ครั้งที่ 1. กรุงเทพมหานคร : สำนักพิมพ์จุฬาลงกรณ์มหาวิทยาลัย, 2538.

ภาษาอังกฤษ

John G. Bollinger ,Neil A. Duffie . Computer Control of Machines and Processes . Addison-Wesley ,1988.

Katsuhiko Ogata . Modern control engineering . Prentice-Hall International ,1990.

Tomokazu Hirabayashi ,Kazuo Yamafuji . Control of the Variable-Structure-Type Locomotive Robot(Walking Forms and Controlling Methods of the Leg-Leg Type) (JSME Series3, Vol.35) . n.p ,1992.

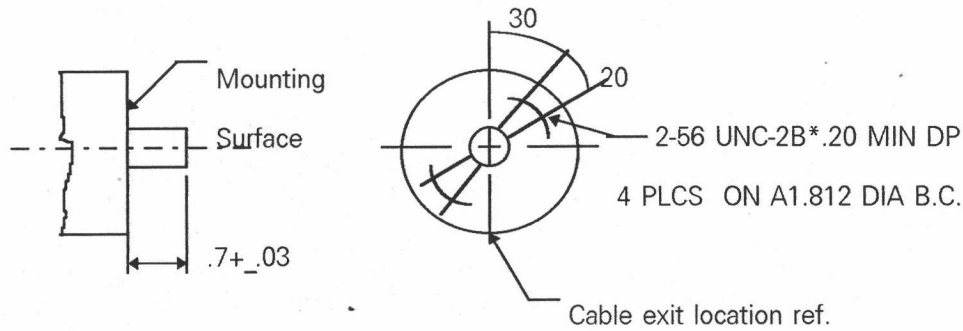
William J.Palm . Control System Engineerings . John Wiley & Sons ,1986.

ภาคผนวก ก.

แสดงอุปกรณ์และส่วนต่างๆ ของหุ่นยนต์เดินด้วยกลไกสี่ลิงค์

เอ็นโคดเดอร์ที่ใช้ในการวัดหรือตรวจจับตำแหน่ง

เป็นอุปกรณ์ที่ใช้ในการควบคุมตำแหน่งแบบป้อนกลับในระบบ ซึ่งเป็นเอ็นโคดเดอร์ของเร็นโค(Renco Encoders) โดยมีลักษณะการยึดติดหรือติดตั้งดังรูปที่ ก.1



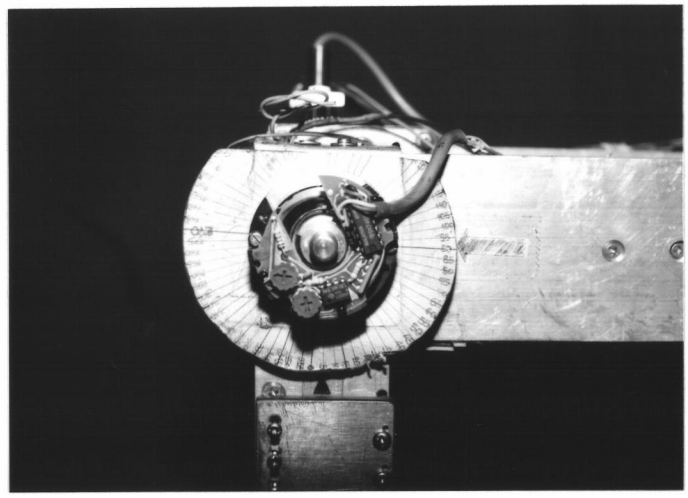
รูปที่ ก.1 แสดงวิธีการยึดติดเอ็นโคดเดอร์

และภายใต้การติดตั้งสิ่งสำคัญคือระยะห่างระหว่างจานหมุนกับตัวรับส่งสัญญาณโดยระยะดังกล่าวที่เอ็นโคดเดอร์ละเอียดไม่เท่ากัน การติดตั้งระยะดังกล่าวเป็นไปตามตารางที่ ก.2

GAP SETTING			
RESOLUTION	GAP TOLERANCE	GAP	
		TYPE 1,3,&4	TYPE 2
0 to 250	± 0.005	0.020	NA
251 to 500	+ 0.002	.008	NA
501 to 1024	+. 002 - .000	.005	.005
1025 to 1500	+.001 - .000	.004	.004
1501 to 2500	- .000	.003	.003

ตารางที่ ก.1 แสดงการติดตั้งระยะแก๊พของเอ็นโคดเดอร์

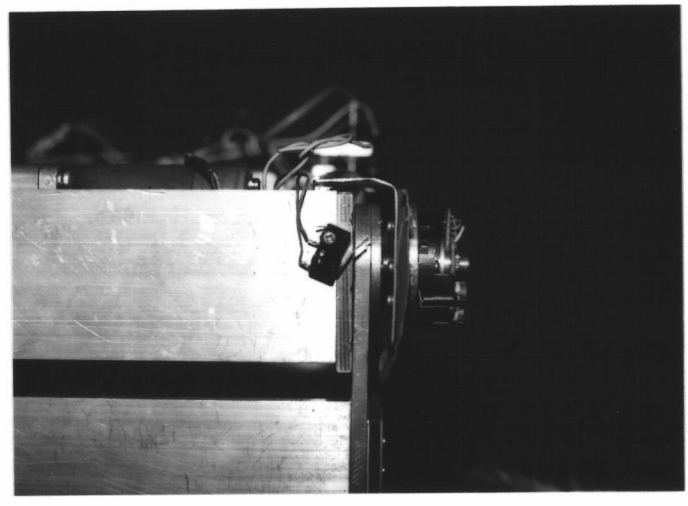
โดยที่ระบบของเรามีความละเอียดหรือรีโซลูชั่นเท่ากับ 800 พัลส์ต่อรอบ ซึ่งแสดงดังรูปที่ ก.2 แสดงตำแหน่งที่ติดตั้งของเอ็นโคดเดอร์ จากรูปก็คือส่วนที่เป็นฝาครอบดำกลมโดยติดตั้งอยู่ที่ด้านของเอาท์พุทมอเตอร์ทั้งขาหน้าและขาหลัง



รูปที่ ก.2 แสดงตำแหน่งที่ติดตั้งของเอ็นโคดเดอร์

ไมโครสวิทช์กับการควบคุม

ไมโครสวิทช์ที่ใช้ในที่นี้มี 2 หน้าที่ด้วยกันคือ ส่วนที่ติตอยู่ที่ด้านข้างลำตัวทั้งสองข้างและสองตัวที่ติดด้านหลัง เป็นส่วนที่ใช้ในการป้องกันการเคลื่อนที่เกินขีดจำกัดของระบบที่ป้องกันการทำให้ระบบเสียหาย มีลักษณะและตำแหน่งการยึดติดดังรูป

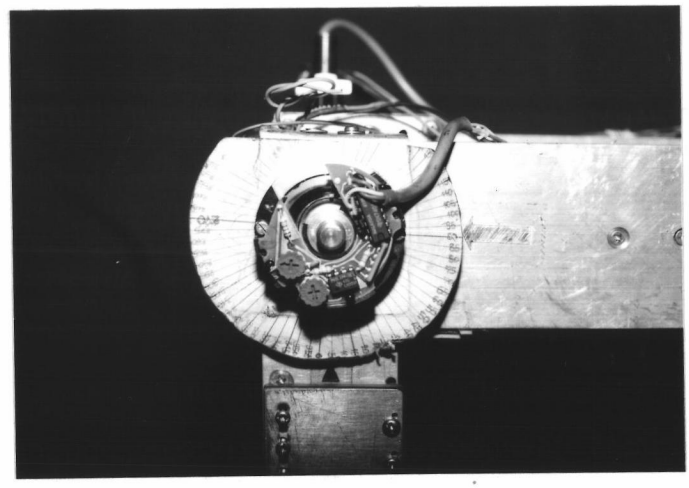


รูปที่ ก.3 แสดงตำแหน่งที่ติดตั้งของไมโครสวิทช์ป้องกันการเคลื่อนที่เกินขีดจำกัด

จากรูปที่ ก.3 ส่วนของไมโครสวิทช์ส่วนที่2 ที่ใช้สำหรับในการตรวจสอบสถานะภาพเพื่อกำหนดทิศทางการหมุนของมอเตอร์เพื่อเข้าสู่ตำแหน่งเริ่มต้นหรือโฮมของระบบ(Home position) ซึ่งติดตั้งอยู่บริเวณตำแหน่งกึ่งกลางด้านบนของขาแต่ละคู่ซึ่งมีการปิดเปิดโดยการแตะของลูกเบี้ยว

พรอกซิมิตี้สวิทช์

พรอกซิมิตี้สวิทช์ที่ใช้นี้เป็นส่วนสำคัญในการบอกระบบควบคุมว่าตรงจุดดังกล่าวคือตำแหน่งเริ่มต้นของระบบหรือศูนย์ๆ ของระบบซึ่งอยู่ในลักษณะท่ายืนเหยียดตรง โ ดยตำแหน่งที่ติดตั้งอยู่ตรงกึ่งกลางด้านบนของเท้าคู่หน้าและหลังแสดงดังรูปที่ ก.4



รูปที่ ก.4 แสดงตำแหน่งการติดตั้งพรอกซิมิตี้สวทช์สำหรับการเข้าตำแหน่งโฮม

เซอร์โวแอมปลิฟายแบบลิเนียร์

จากรูปที่ ก.5 แสดงคอนเน็คเตอร์ต่างของแอมปลิฟายที่ใช้ในการควบคุมซึ่งมีลักษณะเป็นแบบเชิงเส้นโดยมีเกนการขยายเท่ากับ 3.3 จากรูปคอนเน็คเตอร์คู่สีฟ้าเป็นทางเข้าของสัญญาณจากไมโครสวิทช์ที่ป้องกันการเคลื่อนที่เกินขีดจำกัดในทิศทางทวนเข็มและคู่สีเหลืองในทิศทางตามเข็มนาฬิกาโดยด้านบนของแต่ละช่องของคู่เสียบจะมีไฟแสดงการวิ่งเกินขีดจำกัด และถัดมาเป็นส่วนที่เป็นรูเสียบสีเขียวและสีดำเป็นสัญญาณที่รับเข้ามา ภายใต้การควบคุมในโหมดของคอมพิวเตอร์จากลูปการควบคุมผ่านคาร์ดดิจิตอลเป็นอนาลอก และที่รูเสียบคู่ด้านขวามือสุดของรูปคือสีดำกับสีแดงเป็นส่วนของการส่งสัญญาณออกไปควบคุมหรือขับมอเตอร์ ด้วยสวิทช์ ที่เป็น ก้านสีเหลืองซึ่งเป็นสวิทช์เลือกการควบคุมซึ่งมี 2 โหมดคือ โหมดคอมพิวเตอร์และ

โหมดของการควบคุมด้วยมือซึ่งปรับสัญญาณได้จากตัวโวลุมหรือโพเทนทิโอมิเตอร์สีฟ้า และเมื่อมอเตอร์มีการเคลื่อนที่เกินขีดจำกัดก็จะเกิดการปิดสวิทช์ในวงจรที่จะส่งสัญญาณออกไปควบคุม ซึ่งเราจะสามารถกลับมาควบคุมอีกครั้งโดยการใช้สวิทช์รีเซ็ทซึ่งเป็นก้านสีแดง ส่วนของการปิดเปิดเครื่องขยายหรือแอมปลิฟายอยู่ที่สวิทช์ปิดเปิดสีเหลืองดังรูปที่ ก.5

รูปที่ ก.5 แสดงเซอร์โวแอมปลิฟายที่ใช้ในการควบคุม

พีซีเล็ปคาร์ดที่ใช้ในการควบคุม (PCL812-PG)

เป็นคาร์ดซึ่งประกอบด้วยส่วนต่างๆ ดังนี้คือ

1. ส่วนของการแปลงสัญญาณจากอนาลอกเป็นดิจิตอล

ช่องสัญญาณมีทั้งหมด 16 ช่อง

ความละเอียด 12 บิท

ช่วงของการอินพุท เป็นแบบไบโพลาร์: +/-5V,+/-2.5V,+/-1.25V,+/-0.625V,+/-0.3125V.
ทุกๆ ช่วงของการอินพุทสามารถโปรแกรมได้ด้วยซอฟท์แวร์

โวลต์เตจสูงสุด +/-30V

อัตราเร็วในการแปลงสัญญาณ 30khz

ความแม่นยำ 0.015 % ของการอ่าน +/-1 บิท

ความเป็นเชิงเส้น +/-1 bit

2. ส่วนของการแปลงสัญญาณจากดิจิตอลเป็นอนาลอก

ช่องสัญญาณ	2 ช่อง
ความละเอียด	12 บิท
ช่วงการเอาท์พุท	0 ถึง +5V หรือ 0 ถึง+10 V โดยการจำกัดโวลต์อ้างอิงที่ -5Vหรือ -10V
โวลต์เตจอ้างอิง	ภายใน: -5 V ภายนอก ดี.ซี หรือ เอ.ซี, +/-10V
ความเป็นเชิงเส้น	+/-1/2 บิท
ขับเอาท์พุทได้	+/-5 mA
เซ็ทติ้งไทม์	30 microseconds

3. ดิจิตอลอินพุท

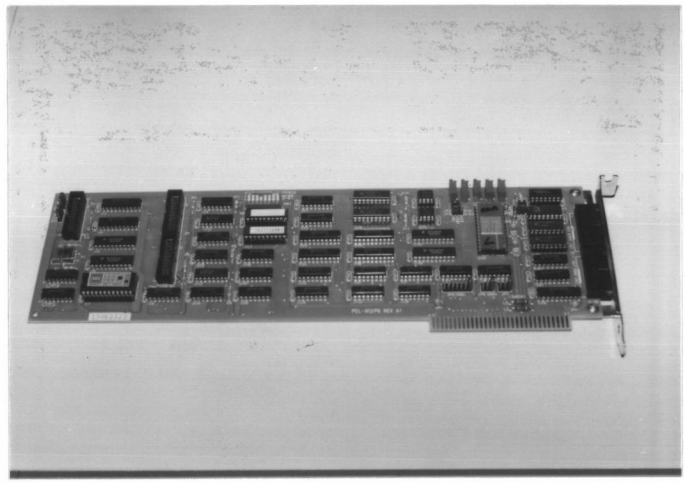
ช่องสัญญาณ	16 ช่อง
โวลต์อินพุท	ที่ค่าด้านต่ำเท่ากับ 0.8 โวลต์ และด้านสูง 2.0 โวลต์
ภาระที่อินพุท	ที่ค่าต่ำ 0.4 mA ที่ 0.5 V. ที่ค่าด้านสูง 0.05 mA ที่ 2.7 V.

4. ดิจิตอลเอาท์พุท

ช่องสัญญาณ	มีทั้งหมด 16 บิท
โวลต์เอาท์พุท	ด้านต่ำ 8 mA ที่ 0.5 V. ด้านสูง -0.4mA ที่ 2.4 V

คาร์ดดังกล่าวถูกใช้ในส่วนของการอ่านค่าและส่งค่าในการควบคุมกล่าวคือในส่วนของอนาลอกเป็นดิจิตอลใช้ในการอ่านค่าสัญญาณจากไมโครสวิทช์เพื่อบอกทิศทางการเคลื่อนที่ของมอเตอร์เข้าสู่ตำแหน่งเริ่มต้นและส่วนของอนาลอกเป็นดิจิตอลใช้ในการส่งค่าสัญญาณควบคุมออกไปควบคุมมอเตอร์ของขาหน้าและขาหลัง

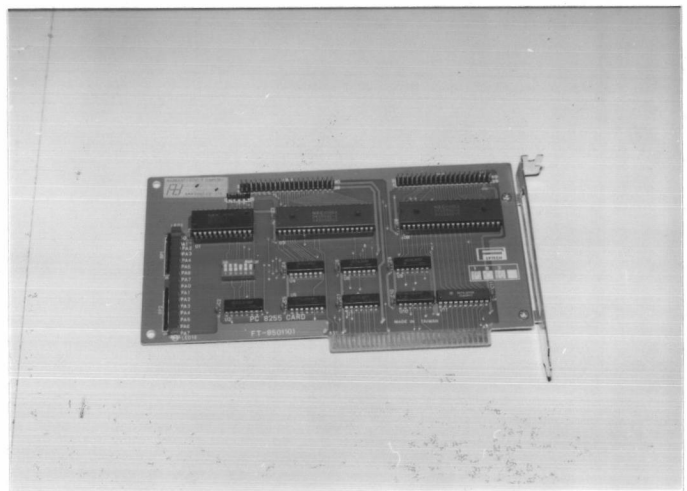
ส่วนของดิจิตอลอินพุทและเอาท์พุทไม่เพียงพอต่อการใช้งานในที่นี้เนื่องจากมีเพียง 16 บิทซึ่งรูปร่างลักษณะของคาร์ดถูกแสดงได้ดังรูปที่ ก.6



รูปที่ ก.6 แสดงพีซีแล็ปคาร์ดรุ่น 812PG ที่ใช้ในการควบคุม

คาร์ดดิจิตอลอินพุทและเอาท์พุท

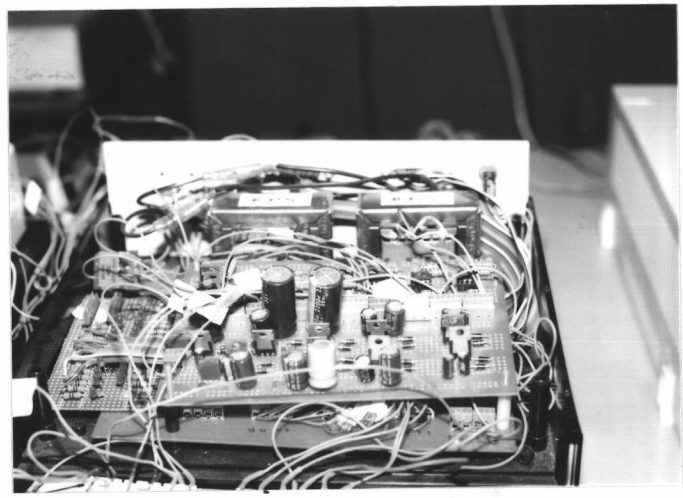
คาร์ดดิจิตอลอินพุทและเอาท์พุทใช้สำหรับการรับค่าสัญญาณจากดีโคดเดอร์ของสัญญาณเอ็นโคดเดอร์ ซึ่งคาร์ดที่ใช้ในที่นี้เป็นแบบ 48 บิท และมีไฟแสดงการทำงานทั้งหมด 16 ดวง ซึ่งใน 48 บิทนี้ ถูกแบ่งออกเป็น 3 พอร์ทด้วยกันคือพอร์ท A,B และ C พอร์ทละ 16 บิท โดยในที่นี้ใช้เพียง 2 พอร์ทคือ A และ B โดยคาร์ดดังกล่าวมีโครงสร้างตามรูปที่ ก.7



รูปที่ ก.7 แสดงดิจิตอลอินพุทและเอาท์พุทคาร์ด

วงจรที่ใช้ในการแบ่งสัญญาณจากคาร์ดให้เป็นบวกและลบ

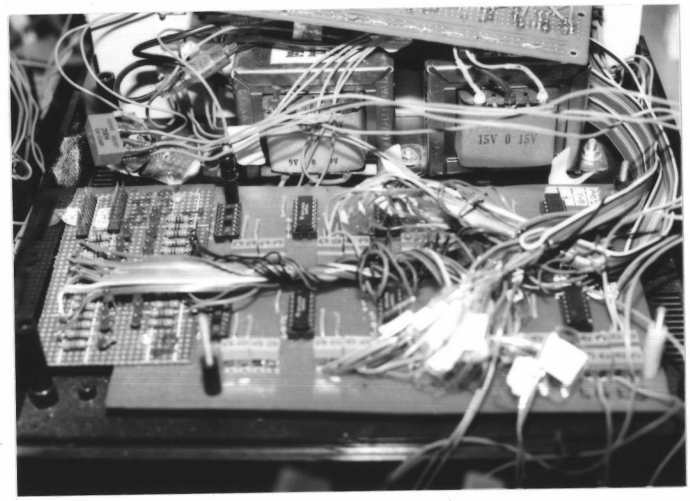
คาร์ดที่ใช้ในการควบคุมให้สัญญาณควบคุมเพียงด้านเดียว ขณะที่ระบบควบคุมมีสัญญาณที่เป็นทั้งทางด้านบวกและลบ ดังนั้นในการควบคุมต้องใช้วงจรแบ่งสัญญาณดังรูปที่ ก.8



รูปที่ ก.8 แสดงวงจรแบ่งสัญญาณคาร์ดจากระบบจริง

วงจรดีโคดเดอร์หรือวงจรถอดรหัส

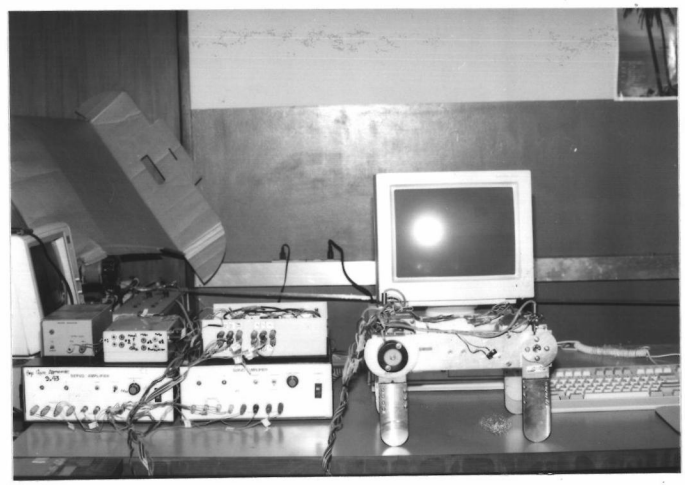
ใช้ถอดรหัสจากตัววัดตำแหน่งคือเอ็นโคดเดอร์ นำค่าของสัญญาณมาใช้ในการควบคุม โดยวงจรถอดรหัสเป็นตัวนับตำแหน่งเป็นดิจิตอลที่คอมพิวเตอร์สามารถนำค่าไปสร้างสัญญาณควบคุม ตามรูปที่ ก.9



รูปที่ ก.9 แสดงวงจรดีโคดเดอร์ของระบบจริง

การเชื่อมต่อของระบบควบคุมทั้งหมด

รูปที่ ก.10 แสดงการเชื่อมต่อของอุปกรณ์ในการควบคุมทั้งหมดของระบบภายใต้คอมพิวเตอร์ควบคุมซึ่งเป็นคอมพิวเตอร์รุ่น 486DX4-100



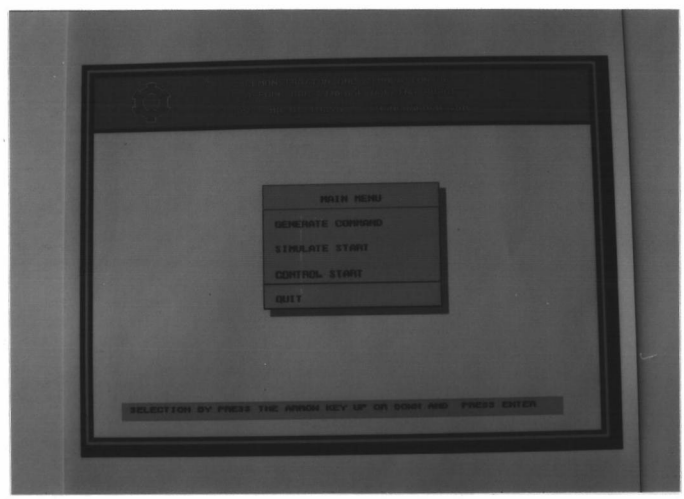
รูปที่ ก.10 แสดงระบบควบคุมของหุ่นยนต์เดินด้วยกลไกสี่ลิงค์โดยรวม

ภาคผนวก ข

แสดงหน้าจอและโปรแกรมควบคุมโดยรวม

ส่วนของโปรแกรมหลักในการควบคุม

ส่วนของเมนเมนูแสดงได้ดังรูปที่ ข.1 ซึ่งมีโครงสร้างและส่วนของโปรแกรมทั้งหมดดังนี้



รูปที่ ข.1 แสดงหน้าจอของเมนูหลักในการควบคุม

```
/* **************  Demonstration and Simulation of******************** */
/* ************** A FOUR BAR LINKAGE WALKING ROBOT************* */
#include <graphics.h>
#include <math.h>
#include <stdio.h>
#include <conio.h>
#include <time.h>
  /* 36000/2MHz */
#define count 36000
#define  T    0.018
```

```
#define pi    3.142857
#define base  0x220
#define CLI() (disable())
#define STI() (enable())
#define  PRESSED 1
/* *** Samplingtime(T)=count/2MHz *** */
float    T;
int      count;
float      S,s,C,D,E,A,data[7],w,a;
int      dat[5],ary1[5],ary2[5],er,fun,xx,yy,x1=150,y1=430,ST,SP;
int      choice,b,bb,c,cc,ee,e,i,ii,j,J,k,kk,m,p;
double   vf,vb;
char     y;
void   *saucer;
FILE   *infile,*outfile;
struct point {int x,y;};
extern   pcl812(int,int*,int*,int*);


main()
{
  int driver,mode;

    driver=DETECT;
    initgraph(&driver,&mode,"");
    write_pos(vf=0,vb=0);
    xx=getmaxx();  yy=getmaxy();
    p:
      Main_window();
      Gear();
```

```
p1:
  Main_menu();
  choice=bioskey(0);
  if(choice==0x5000){b++;bb+=30;goto p1;} /* i=case,ii=blink step */
  if(choice==0x4800){b--;bb-=30;goto p1;}
  if(b==4&&choice==0x1c0d){clearviewport();closegraph();exit(0);}
  if(choice==0x1c0d)
  {
      switch(b)
      {
          case 1:
                p22:
                  /* เป็นส่วนของการสร้างไฟล์ทางเดินแบบแบง-แบง*/
                   clearviewport();
                   comm_menu();
                   choice=bioskey(0);
                   if(choice==0x5000){ b++;bb+=30; goto p22; }
                   if(choice==0x4800){ b--;bb-=30; goto p22; }
                   if(b==2 && choice==0x1c0d){ clearviewport(); goto p;}
                   if(choice==0x1c0d)
                   {
                        system("gencom.exe");
                        goto p22;
                   }
                    else goto p22;
           case 2:
                 ps_1:
                      ground(0,0,xx,yy);
                      rectangle(10,10,xx-10,yy-10);
```

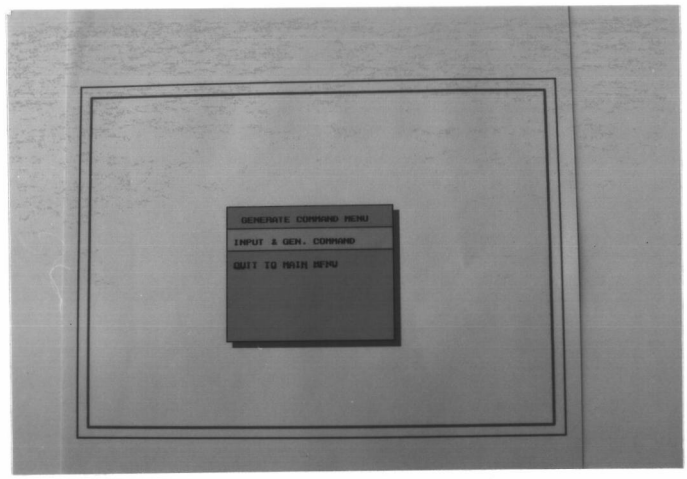
```
remark("ARROW UP OR DOWN AND PRESS ENTER FOR A
  SELECT",95,434,508,449);
shado(170,105,450,130);
outtextxy(190,110,"SELECT GEN. ACCORDING TO A
     MENU");

ps_2:
/* ส่วนของการแสดงผลลัพธ์จากไฟล์ทางเดินด้วยภาพบนหน้าจอ */
subsim_menu();
choice=bioskey(0);
if(choice==0x5000){b++;bb+=30;goto ps_2;}
if(choice==0x4800){b--;bb-=30;goto ps_2;}
if(b==3 && choice==0x1c0d)
{
    clearviewport();goto p;
}
if(choice==0x1c0d)
{
    switch(b)
    {
        case 1:
            ps_21:
                sub1_subsim_menu();
                choice=bioskey(0);
                if(choice==0x5000){c++;cc+=30;goto
                                    ps_21;}
                if(choice==0x4800){c--;cc-=30;goto
                                    ps_21;}
                if(c==2 && choice==0x1c0d)
```

```
                {
                    ground(220,150,600,350);
                    goto ps_2;
                }
                if(c==1 && choice==0x1c0d)
                {
                    sim_comm1();
                    if(ST>5) goto ps_1;
                    if(SP>5) goto ps_1;
                    if(ST>SP) goto ps_1;
                    outfile=fopen("d:\simstep.xls","w");
                    fprintf(outfile,"%2d\t%2d\n",ST,SP);
                    fclose(outfile);
                    setcolor(LIGHTCYAN);
                    system("simulate.exe");
                    goto ps_1;
                }
                 else goto ps_21;
            case 2:
                outfile=fopen("d:\simstep.xls","w");
                ST=1;SP=5;
                fprintf(outfile,"%2d\t%2d\n",ST,SP);
                fclose(outfile);
                system("simulate.exe");
                goto ps_1;
        }
    }
    else goto ps_2;
case 3:
```

```
                    /* ส่วนของการควบคุมระบบจริงๆ */
                    p1_3:
                        Ctrl_menu();
                        choice=bioskey(0);
                          if(choice==0x5000){e++;ee+=30;goto p1_3;}
                          if(choice==0x4800){e--;ee-=30;goto p1_3;}
                          if(e==2&&choice==0x1c0d)
                          {
                                clearviewport();
                                goto p;
                          }
                          if(e==1 && choice==0x1c0d)
                          {
                                Control_pos();
                                setfillstyle(SOLID_FILL,BLACK);
                                bar(0,0,xx,yy);
                                goto p;
                          }
                case 4:
                          /* ออกจากโปรแกรมควบคุม */
                           break;
                        }
            }
            else goto p1;
}
```

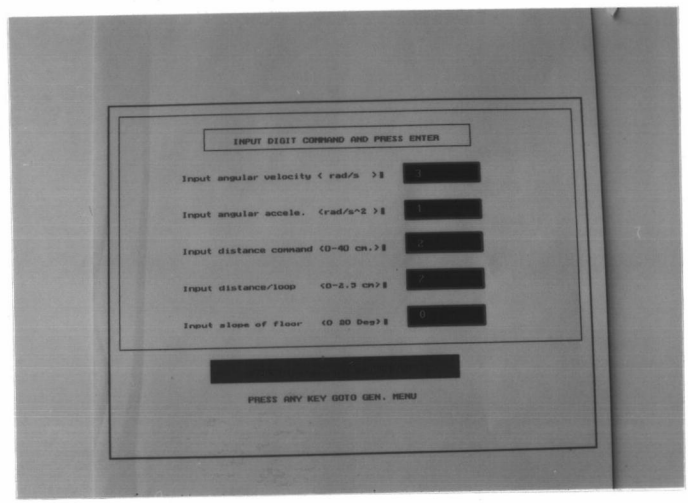
โปรแกรมที่ใช้ในการควบคุมนี้แบ่งออกเป็น 3 ส่วนหลักๆ คือ

1. ส่วนของการสร้างไฟล์ทางเดิน



รูปที่ ข.2 แสดงเมนูของการสร้างไฟล์คำสั่งทางเดินแบบแบง-แบง

จากเมนูหลักของการสร้างไฟล์ดังรูปที่ ข.2 ภายหลังจากที่เราทำการเลือกการสร้างไฟล์ทางเดิน(gencommand) ก็จะปรากฏหน้าจอดังรูปที่ ข.3 และถ้าเลือกออกจากเมนูของการสร้างไฟล์ก็จะกลับไปที่หน้าจอของเมนูหลัก ที่การเลือกสร้างไฟล์ทำการป้อนค่าคำสั่งคือความเร็วเชิงมุม,ความเร่งเชิงมุม,ระยะทางที่ก้าวเดินทั้งหมดและระยะทางที่ก้าวเดินต่อลูปของการควบคุมที่ความชันของพื้นเท่ากับเท่าไร ซึ่งเมื่อป้อนค่าสิ่งเหล่านี้เข้าไปจนครบแล้วกดปุ่มเอ็นเตอร์ระบบก็จะเริ่มทำการแสดงการสร้างไฟล์คำสั่งบนหน้าจอและแสดงผลการเสร็จสมบูรณ์ดังรูป



รูปที่ ข.3 แสดงหน้าจอสมบูรณ์ของการสร้างไฟล์คำสั่งทางเดิน

จากรูปเป็นผลจากการคอมไพล์โปรแกรมดังนี้

```
#include "graphics.h"
#include "math.h"
#include "stdio.h"
#include "conio.h"
#define T 0.018
#define pi 3.142857
float  TT,w,a,x,S,tf1,tb1,s,D,E,A,
       Cf1f[7],Cb1f[7],Cf2[7],Cb2[7],RF[7],
       RB[7],Rf,Rb,data[7];
int    nf[7],nb[7],n1[7],nf1[7],nb1[7],P[7],xx,yy,m;
double Cf1a[7],Cb1a[7],Cf2a[7],Cb2a[7];
FILE   *infile,*outfile;
main()
{
  int driver,mode;

    driver=DETECT;
    initgraph(&driver,&mode,"");
    xx=getmaxx();
    yy=getmaxy();
    clearviewport();
    input_com();
    gen_comm();
}
/* ************************sub GENCOMMAND menu******************** */
comm_menu()
{ int b,bb;
      ground(0,0,xx,yy);
```

```
        setcolor(RED);
        setlinestyle(0,0xffff,3);
        rectangle(0,0,xx,yy);
        rectangle(15,15,xx-15,yy-15);
        setlinestyle(0,0xffff,1);
        shado(200,170,420,350);
        setfillstyle(1,WHITE);
        outtextxy(220,185,"GENERATE COMMAND MENU");
        rectangle(200,170,420,200);
        if(b>2){b=1;bb=0;}
        if(b<1){b=2;bb=30;}
        setfillstyle(1,YELLOW);
        bar(200,200+bb,420,230+bb);
        rectangle(200,200+bb,420,230+bb);
        outtextxy(210,215,"INPUT & GEN. COMMAND");
        outtextxy(210,245,"QUIT TO MAIN MENU");
}
/* ** use for input distance(s) centimetre ** */
Gen_menu()
{
 int ii;
        shado(230,260,450,320);
        setfillstyle(1,YELLOW);
        bar(230,260+ii,450,290+ii);
        rectangle(230,260+ii,450,290+ii);
        outtextxy(235,275,"Input command ");
        outtextxy(235,305,"Quit to command menu");
        shado(230,260,450,320);
        setcolor(BLUE);
```

```
        ground(220,250,460,360);
}
input_com()
{
    float C,Sl;

    ground(0,0,xx,yy);
    setcolor(GREEN);
    rectangle(130,35,490,65);
    setcolor(RED);
    setlinestyle(0,0xffff,3);
    rectangle(0,0,xx,yy);
    setlinestyle(0,0xffff,1);
    outtextxy(170,50,"INPUT DIGIT COMMAND AND PRESS ENTER");
    outtextxy(100,100 ,"Input angular velocity < rad/s  >□ ");
    scan_textbar(400,90,500,115);
    outtextxy(100,150,"Input angular accele.  <rad/s^2 >□ ");
    scan_textbar(400,140,500,165);
    outtextxy(100,200,"Input distance command <0-40 cm.>□ ");
    scan_textbar(400,185,500,210);
    outtextxy(100,250,"Input distance/loop    <0-2.5 cm>□ ");
    scan_textbar(400,235,500,260);
    outtextxy(100,300,"Input slope of floor   <0-20 Deg>□ ");
    scan_textbar(400,285,500,310);
    rectangle(15,15,xx-15,335);
    gotoxy(53,7);
    scanf("%f",&w);
    gotoxy(53,10);
    scanf("%f",&a);
```

```
    gotoxy(53,13);
    scanf("%f",&s);
    gotoxy(53,16);
    scanf("%f",&D);
    if(D>s) D=s;
    x=D;
    if(x>2.2) x=2.20;
    else      x=x;
    C=s/x;
    E=C-(int)C;
    if(E != 0 ) m=(int)C+1;
    else        m=C;
    gotoxy(53,19);
    scanf("%f",&Sl);
    S=Sl;
    if(S>20) S=20;
    else     S=S;
}

gen_comm()
{
  int     mm,nn,check,i,k,j;
  float   ml=0.7,mb=6.7,Ll=18.0,Lb=27.0,
            a1,b1,c1,O,mn,zf1,zb1,zb2,zf2,zb3,
            Ls,a2,b2,c2,d2,e2,z2,z;
  int     nf2[10],nb2[10],J,L;
  float   nf1_real,tf2,nf2_real,n_realf,ww,
            nb1_real,tb2,nb2_real,n_realb;
  double  Cf0,Cb0;
```

```
L=1;
if(E != 0) L=2;
 for(J=1;J<=L;J++)
 {
        /*step1*/
        S=S*pi/180;
        mm=0;
        do
        {
            if(J==2) x=s-(m-1)*D;
             O=(S+mm)*pi/180;
            mn=mm*pi/180;
            a1=ml*sin(O)*Ll/2+mb*sin(S)*Ll*cos(mn)+ml*sin(S)*Ll/2*cos(mn);
            b1=mb*cos(S)*(Lb/2-Ll*sin(mn))+ml*cos(S)*(Lb-Ll/2*sin(mn));
            c1=a1-b1;
            mm++;
        }while(fabs(c1)>=1.0);
        zf1=mm;
        zb1=zf1;
        /*step2*/
        Ls=x+Lb;
        a2=zb1*pi/180;
        nn=0;
        do{
            b2= nn*0.2*pi/180;
            c2=pow( (pow(Lb,2)-pow(Ll,2)*pow( (cos(a2)-cos(b2)),2 )),0.5);
            d2=Ll*sin(b2)-Ll*sin(a2);
            e2=Ls-(c2+d2);
```

```
    nn++;
}while(fabs(e2)>0.05);
z2=(nn-1)*0.2;
z=180/pi*asin( (( cos(zf1*pi/180)-cos(z2*pi/180) )*LI/26) );
zf2=z2+z;
zb3=zb1+z;
RF[0]= 0  ;RB[0]=0;
RF[1]=-zf1;RB[1]=-zb1;
RF[2]=-zf2;RB[2]=-zb3;
RF[3]= zb3;RB[3]= zf2;
RF[4]= zf1;RB[4]=zb1;
RF[5]= 0;  RB[5]=0;
for(i=1;i<=5;i++)
{
    tf1=w/a;  A=2*0.5*a*pow(tf1,2);
    tb1=tf1;
    if(fabs(RF[i]-RF[i-1])>A)
    {
          nf1_real=tf1/T;
          if(RF[i]<RF[i-1])
          {
               Cf1a[i]=RF[i-1]-0.5*a*pow(tf1,2);
               Cf2a[i]=RF[i]+pow(w,2)/2/a;
          }
          if(RF[i]>RF[i-1])
          {
               Cf1a[i]=RF[i-1]+0.5*a*pow(tf1,2);
               Cf2a[i]=RF[i]-pow(w,2)/2/a;
          }
```

```
}
if(fabs(RF[i]-RF[i-1])<=A)
{
        if(RF[i]<RF[i-1])
        {    Cf1a[i]= RF[i-1]-fabs(RF[i]-RF[i-1])*0.5;
             Cf2a[i]= Cf1a[i];
        }
        if(RF[i]>RF[i-1])
        {
             Cf1a[i]=RF[i-1]+ fabs(RF[i]-RF[i-1])*0.5;
             Cf2a[i]=Cf1a[i];
        }
        if(RF[i]==RF[i-1])
        {
             Cf1a[i]=RF[i-1];
             Cf2a[i]=Cf1a[i];
        }
        tf1=pow(2/a*fabs(Cf1a[i]-RF[i-1]),0.5);
        nf1_real=tf1/T;
   }
   if(fabs(RB[i]-RB[i-1])>A)
   {
          nb1_real=tb1/T;
          if(RB[i]<RB[i-1])
          {
               Cb1a[i]=RB[i-1]-0.5*a*pow(tf1,2);
               Cb2a[i]=RB[i]+pow(w,2)/2/a;
          }
          if(RB[i]>RB[i-1])
```

```
        {
            Cb1a[i]=RB[i-1]+0.5*a*pow(tf1,2);
            Cb2a[i]=RB[i]-pow(w,2)/2/a;
        }
}

if(fabs(RB[i]-RB[i-1])<=A)
{
        if(RB[i]<RB[i-1])
        {
            Cb1a[i]=RB[i-1]-fabs(RB[i-1]-RB[i])*0.5;
            Cb2a[i]=Cb1a[i];
        }
        if(RB[i]>RB[i-1])
        {
            Cb1a[i]=RB[i-1]+fabs(RB[i-1]-RB[i])*0.5;
            Cb2a[i]=Cb1a[i];
        }
        if(RB[i]==RB[i-1])
        {
            Cb1a[i]=RB[i-1];
            Cb2a[i]=Cb1a[i];
        }
            tb1=pow(2/a*fabs(Cb1a[i]-RB[i-1]),0.5);
            nb1_real=tb1/T;
   }

   nf1[i]=nf1_real;
   if(nf1_real-(int)nf1_real>0.5) nf1[i]=nf1_real+1;
```

```
nb1[i]=nb1_real;
if(nb1_real-(int)nb1_real>0.5) nb1[i]=nb1_real+1;
nb1[i]=nb1_real;
tf2=fabs((Cf2a[i]-Cf1a[i])/w);
tb2=fabs((Cb2a[i]-Cb1a[i])/w);
nf2_real=tf2/T;
nb2_real=tb2/T;
nf2[i]=nf2_real;
if(nf2_real-(int)nf2_real>0.5) nf2_real=nf2_real+1;
nb2[i]=nb2_real;
if(nb2_real-(int)nb2_real>0.5) nb2_real=nb2_real+1;
n_realf=2*nf1_real+nf2_real;
n_realb=2*nb1_real+nb2_real;
if(fabs(RF[i]-RF[i-1])>A) nf[i]=n_realf;
else   { nf[i]=2*nf1[i]; n_realf=nf[i]; nf2[i]=nf1[i];}
if(fabs(RB[i]-RB[i-1])>A) nb[i]=n_realb;
else   { nb[i]=2*nb1[i]; n_realb=nb[i]; nb2[i]=nb1[i];}
if (n_realf-nf[i] >= 0.5) nf[i] = nf[i]+1;
if (n_realb-nb[i] >= 0.5) nb[i] = nb[i]+1;
if (nf[i]==0) nf[i] = 1;
if (nb[i]==0) nb[i] = 1;
if (nf[i]<nb[i]) P[i]=nb[i];
else P[i]=nf[i];
if(J==1)
{
      if(i==1)
      {
          if( (outfile=fopen("d:\step11.xls","w"))==NULL){
              printf("Cannot open file\n");
```

```
                exit(0);
                }
        }
        if(i==2)
        {
            outfile=fopen("d:\step12.xls","w");
        }
        if(i==3)
        {
            outfile=fopen("d:\step13.xls","w");
        }
        if(i==4)
        {
            outfile=fopen("d:\step14.xls","w");
        }
        if(i==5)
        {
            outfile=fopen("d:\step15.xls","w");
        }
}
if(J==2)
{
        if(i==1)
        {
            outfile=fopen("d:\step21.xls","w");
        }
        if(i==2)
        {
            outfile=fopen("d:\step22.xls","w");
```

```
        }
        if(i==3)
        {
            outfile=fopen("d:\step23.xls","w");
        }
        if(i==4)
        {
            outfile=fopen("d:\step24.xls","w");
        }
        if(i==5)
        {
            outfile=fopen("d:\step25.xls","w");
        }
}
k=1;
while(k<=P[i])
{
        if(fabs(RF[i]-RF[i-1])>A)
        {
          if(k<=nf1[i])
          {
                if(RF[i]<RF[i-1]) Rf=RF[i-1]-0.5*a*pow(k*T,2);
                if(RF[i]>RF[i-1]) Rf=RF[i-1]+0.5*a*pow(k*T,2);
                if(k==nf1[i])     Cf1a[i]=Rf;
          }
          if(nf1[i]<k && k<=nf[i]-nf1[i])
          {
                if(RF[i]<RF[i-1]) Rf=Cf1a[i]-w*(k-nf1[i])*T;
                if(RF[i]>RF[i-1]) Rf=Cf1a[i]+w*(k-nf1[i])*T;
```

```
        if(k==nf[i]-nf1[i])Cf2a[i]=Rf;
    }
    if(nf[i]-nf1[i]<k && k<=nf[i])
    {
        if(RF[i]<RF[i-1])Rf=Cf2a[i]-w*(k-nf[i]+nf1[i])*T+0.5*a*pow((k-
                                nf[i]+nf1[i])*T,2);
        if(RF[i]>RF[i-1])Rf=Cf2a[i]+w*(k-nf[i]+nf1[i])*T-0.5*a*pow((k-
                                nf[i]+nf1[i])*T,2);
    }
    if(k>=nf[i]) Rf=RF[i];
}
if(fabs(RF[i]-RF[i-1])<=A)
{
  if(k<=nf1[i])
  {
        if(RF[i]<RF[i-1]) Rf=RF[i-1]-0.5*a*pow(k*T,2);
        if(RF[i]>RF[i-1]) Rf=RF[i-1]+0.5*a*pow(k*T,2);
        if(k==nf1[i])     Cf1a[i]=Rf;
  }
  if(nf1[i]<k && k<=nf[i])
  {
        ww=a*tf1;
        if(RF[i]<RF[i-1])Rf=Cf1a[i]-ww*(k-nf1[i])*T+0.5*a*pow((k-nf1[i])*T,2);
        if(RF[i]>RF[i-1])Rf=Cf1a[i]+ww*(k-nf1[i])*T-0.5*a*pow((k-nf1[i])*T,2);
  }
}
if(fabs(RB[i]-RB[i-1])>A)
{
  if(k<=nb1[i])
```

```
{
        if(RB[i]<RB[i-1]) Rb=RB[i-1]-0.5*a*pow(k*T,2);
        if(RB[i]>RB[i-1]) Rb=RB[i-1]+0.5*a*pow(k*T,2);
        if(k==nb1[i])     Cb1a[i]=Rb;
}
if(nb1[i]<k && k<=nb[i]-nb1[i] )
{
        if(RB[i]<RB[i-1]) Rb=Cb1a[i]-w*(k-nb1[i])*T;
        if(RB[i]>RB[i-1]) Rb=Cb1a[i]+w*(k-nb1[i])*T;
        if(k==nb[i]-nb1[i])Cb2a[i]=Rb;
}
if(nb[i]-nb1[i]<k && k<=nb[i])
{
        if(RB[i]<RB[i-1])Rb=Cb2a[i]-w*(k-nb[i]+nb1[i])*T+0.5*a*pow((k-
                                nb[i]+nb1[i])*T,2);
        if(RB[i]>RB[i-1])Rb=Cb2a[i]+w*(k-nb[i]+nb1[i])*T-0.5*a*pow((k-
                                nb[i]+nb1[i])*T,2);
 }
}
if(fabs(RB[i]-RB[i-1])<=A)
{
  if(k<=nb1[i])
  {
        if(RB[i]<RB[i-1]) Rb=RB[i-1]-0.5*a*pow(k*T,2);
        if(RB[i]>RB[i-1]) Rb=RB[i-1]+0.5*a*pow(k*T,2);
        if(k==nb1[i])     Cb1a[i]=Rb;
  }
  if(nb1[i]<k && k<=nb[i])
  {
```

```
                    ww=a*tb1;
                    if(RB[i]<RB[i-1])Rb=Cb1a[i]-ww*(k-nb1[i])*T+0.5*a*pow( (k-
                                        nb1[i])*T,2);
                    if(RB[i]>RB[i-1])Rb=Cb1a[i]+ww*(k-nb1[i])*T-0.5*a*pow( (k-
                                        nb1[i])*T,2);
                  }
                }

                fprintf(outfile,"%6.5f\t%6.5f\n",Rf,Rb);
                k++;
    }
    if((check=fclose(outfile))!=0)
    {
          printf("Error on output file close");
          exit(1);
    }
    setcolor(LIGHTRED);
    rectangle(135,350,465,380);
    setfillstyle(SOLID_FILL,BLUE);
    bar(135,350,165+60*i,380);
    if(i==5)
    {
          sound(300);
          outtextxy(186,365,"GENERATE FILE ARE COMPLETED");
          delay(800);
          nosound();
    }
}
TT=T;
```

```
outfile=fopen("d:\comm.xls","w");
for(j=1;j<=1;j++)
{
fprintf(outfile,"%5.3f\t%5d\t%5.3f\t%5.3f\t%5.3f\t%5.3f\t%5.3f\t%5.4f\n",D,m,E,s,S,w,a,
      TT);
}
if((check=fclose(outfile))!=0)
{
   printf("Error on output file close");
   exit(1);
}
if(J==1)
{
   outfile=fopen("d:\loop1.xls","w");
   for(j=1;j<=5;j++)
   {
        fprintf(outfile,"%5d\t%5.3f\t%5.3f\n",P[j],RF[j],RB[j]);
   }
   if((check=fclose(outfile))!=0)
   {
        printf("Error on output file close");
        exit(1);
   }

}
if(J==2)
{
   outfile=fopen("d:\loop2.xls","w");
   for(j=1;j<=5;j++)
```

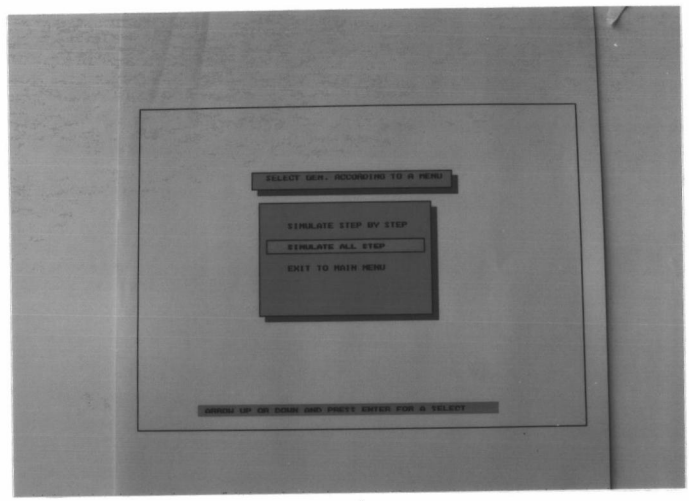
```
            {
                fprintf(outfile,"%5d\t%5.3f\t%5.3f\n",P[j],RF[j],RB[j]);
            }
            if((check=fclose(outfile))!=0)
            {
                printf("Error on output file close");
                exit(1);
            }
        }
    }
    outtextxy(185,400,"PRESS ANY KEY GOTO GEN. MENU");
    getch();

}

/* ***********General graphics func ************************/
scan_textbar(int x_lt1,int y_lt1,int x_lb2,int y_lb2)
{
    setcolor(LIGHTBLUE);
    rectangle(x_lt1,y_lt1,x_lb2,y_lb2);
    setfillstyle(1,BLACK);
    floodfill(x_lt1+1,y_lt1+1,LIGHTBLUE);
    setlinestyle(0,0xffff,3);
    rectangle(x_lt1,y_lt1,x_lb2,y_lb2);
    setlinestyle(0,0xffff,1);
    setcolor(RED);
}

shado(int a,int b,int c,int d)
```

```
{
    setfillstyle(1,LIGHTBLUE);
    bar(a+8,b+8,c+8,d+8);
    setfillstyle(1,LIGHTCYAN);
    bar(a,b,c,d);
    setcolor(BLACK);
    rectangle(a,b,c,d);
    setfillstyle(1,LIGHTGRAY);
     return;
}
ground(int x_t,int y_t,int x_l,int y_l)
{
    setfillstyle(1,WHITE);
    bar(x_t,y_t,x_l,y_l);
    setcolor(LIGHTCYAN);
    setfillstyle(1,LIGHTGRAY);
    setcolor(BLACK);
}
remark(char word[80],int x1,int y1,int x2,int y2)
{
        setfillstyle(1,LIGHTCYAN);
        bar(x1,y1,x2,y2);
        setcolor(BLACK);
        gprintf(x1+10,y1+5,"%.60s",word);
        setcolor(LIGHTBLUE);
}
gprintf( int xloc, int yloc, char *fmt, ... )
{
  va_list  argptr;
```

```
char str[150];
int cnt;
va_start( argptr, format );
cnt = vsprintf( str, fmt, argptr );
outtextxy( xloc,yloc, str );
yloc += textheight( "H" ) + 2;
va_end( argptr );
return( cnt );

}
```

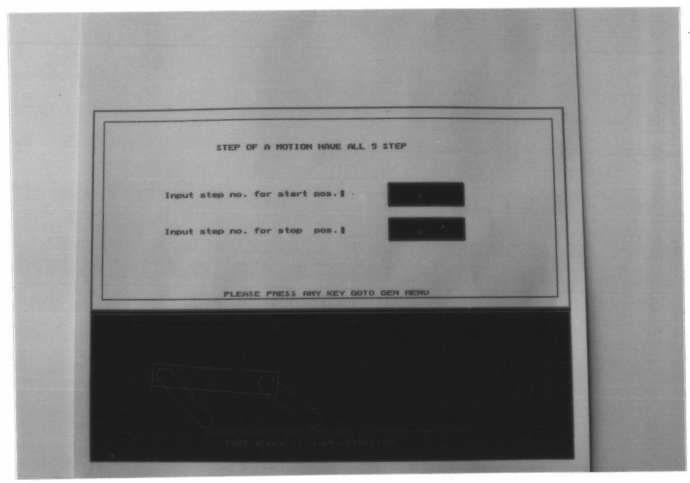
2. ส่วนของการแสดงผลไฟล์ทางเดินด้วยการแสดงบนหน้าจอ

ขั้นตอนนี้เป็นการตรวจสอบผลลัพธ์ของไฟล์ทางเดินว่ามีสิ่งผิดปกติเกิดขึ้นหรือไม่ โดยการตรวจสอบการเดินนี้สามารถพิจารณาได้เป็น 2 โหมด คือโหมดแรกเป็นการพิจารณาทีละลำดับของการเดิน ส่วนโหมดที่ 2 เป็นการพิจารณาทีเดียวทั้งกระบวนการตั้งแต่เริ่มต้นจนจบซึ่งภายใต้เมนูหลักของระบบเมื่อเราเลือกการแสดงผลทางหน้าจอ(Simulation) ก็จะเข้าสู่หน้าจอของเมนูหลักของการแสดงผลทางหน้าจอซึ่งมีทางเลือกอยู่ทั้งหมด 3 ทางด้วยกัน คือทางแรกเป็นการเลือกแสดงผลทีละลำดับและทางเลือกที่สองคือแสดงทีเดียวทั้งกระบวนการและทางเลือกสุดท้ายคือออกจากเมนูหลักของการแสดงผลทางหน้าจอมาสู่เมนูหลักของระบบ ซึ่งหน้าจอของเมนูการแสดงผลทดลองทางหน้าจอเป็นดังรูปที่ ข.4



รูปที่ ข.4 แสดงหน้าจอของเมนูซิมมูเลชัน(Simulationmenu)

จากรูปที่ ข.4 หลังจากที่เราทำการเลือกโหมดของการแสดงผลว่าจะแสดงผลทีละลำดับตามลำดับที่ต้องการพิจารณาหรือเลือกพิจารณาการแสดงจบกระบวนการก็จะปรากฏหน้าจอดังรูปที่ ข.5 เหมือนๆ กัน ต่างกันตรงที่ท่าทางของการเดินของหุ่นยนต์ที่ค้างไว้คนละลำดับการเดิน



รูปที่ ข.5 แสดงหน้าจอผลลัพธ์ของการทดลองเดินบนหน้าจอ

ภายใต้ผลลัพธ์ของการแสดงผลดังกล่าว ภายใต้โปรแกรมดังนี้

```
#include <graphics.h>
#include <math.h>
```

```
#include <stdio.h>
#include <dos.h>
#define pi 3.142857
FILE *infile,*outfile;
int xx,yy,ST,SP;
float w,a,T;

main()
{
  int driver,mode;
   driver=DETECT;
   initgraph(&driver,&mode,"");
   xx=getmaxx();
   yy=getmaxy();
   setcolor(LIGHTCYAN);
   sim_robo();
}
/* **********************sub SIMULATE menu ********************** */
sim_comm()
{
      ground(0,0,xx,275);
      setcolor(RED);
      rectangle(15,15,xx-15,260);
      outtextxy(170,50,"STEP OF A MOTION HAVE ALL 5 STEP");
      outtextxy(100,115,"Input step no. for start pos.□ ");
      scan_textbar(400,105,500,135);
      outtextxy(100,165,"Input step no. for stop  pos.□ ");
      scan_textbar(400,153,500,182);
      setcolor(LIGHTRED);
```

```
    gprintf(430,120,"%2d",ST);
    gprintf(430,168,"%2d",SP);
    setlinestyle(0,0xffff,3);
    rectangle(0,0,xx,275);
    rectangle(0,280,xx,yy);
    setlinestyle(0,0xffff,1);
}
sim_robo()
{
 int    M,N,K,loop,r=10,m,x1=150,y1=430,P[7];
 float  Rf1,Rb1,Rff,Rbb,RF0[7],RB0[7],x11,y11,Ll=90,Lb=135,zeta,
        X[20],Y[20],LX[10],LY[10],l,Xg,zf,Zf,zb,Zb,Rfc,Rfs,Rbc,Rbs,
        x,D,E,s,S,RF[7],RB[7],Rf,Rb;
    infile=fopen("d:\simstep.xls","r");
    fseek(infile,0,0);
    while(!feof(infile))
    {
        if(ferror(infile))
        {
            printf(10,10,"File error \n");
            exit(1);
        }
        fscanf(infile,"%d\t%d\n",&ST,&SP);
    }
    fclose(infile);
    sim_comm();
    infile=fopen("d:\comm.xls","r");
    fseek(infile,0,0);
    while(!feof(infile))
```

```
{
    if(ferror(infile))
    {
        printf(10,10,"File error \n");
        exit(1);
    }
    fscanf(infile,"%f\t%d\t%f\t%f\t%f\t%f\t%f\t%f\n",&D,&m,&E,&s,&S,&w,&a,&T);
}
fclose(infile);
setcolor(LIGHTBLUE);
outtextxy(180,450,"THIS SCALE IS 1 CM./DIVISION");
line(x1,y1+7,x1+350,y1+7);
line(x1,y1+7,x1+350,y1+7-350*tan(S));
K=1;
while(K<=70)
{
    line(x1+K*5*cos(S),y1+7-K*5*sin(S),\
        x1+K*5*cos(S)+4*sin(S),y1+7-K*5*sin(S)+4*cos(S));
    K++;
}
setlinestyle(4,0xffff,1);
x=D;
Xg=x*5;
M=m;
if(M>=10) M=10;
loop=1;
    while(loop<=M)
    {
        if(E==0)
```

```
{
    infile=fopen("d:\loop1.xls","r");
    fseek(infile,0,0);
    N=1;
    while(!feof(infile))
    {
        if(ferror(infile))
        {
            gprintf(10,100,"File error \n");
            exit(1);
        }
        fscanf(infile,"%d\t%f\t%f\n",&P[N],&RF[N],&RB[N]);
        N++;
    }
    fclose(infile);
}
if(loop==M && E!=0)
{
    infile=fopen("d:\loop2.xls","r");
    fseek(infile,0,0);
    N=1;
    while(!feof(infile))
    {
      if(ferror(infile))
      {
          gprintf(10,100,"File error \n");
          exit(1);
      }
      fscanf(infile,"%d\t%f\t%f\n",&P[N],&RF[N],&RB[N]);
```

```
        N++;
    }
    fclose(infile);
}
x11=x1+Xg*cos(S)*(loop-1);
y11=y1-Xg*sin(S)*(loop-1);
for(I=ST;I<=SP;I++)
{
    if(I==1)
    {
        infile=fopen("d:\step11.xls","r");
        if(E!=0 && loop==M) {infile=fopen("d:\step21.xls","r");}
        fseek(infile,0,0);
        N=1;
        while(!feof(infile))
        {
            if(ferror(infile))
            {
                gprintf(10,100,"File error \n");
                exit(1);
            }
            fscanf(infile,"%f\t%f\n",&Rf,&Rb);
            Rf=Rf*pi/180;  Rb=Rb*pi/180;
            Rf1=fabs(Rf);  Rb1=fabs(Rb);
            X[1]=x11; Y[1]=y11;
            X[2]=X[1]-LI*sin(Rf1); Y[2]=Y[1]-LI*cos(Rf1);
            zf=atan((X[1]-X[2])/(Y[1]-Y[2])); Zf=zf*180/pi;
            zb=zf; Zb=Zf;
            Rfs=r*sin(zf); Rfc=r*cos(zf);
```

```
Rbs=r*sin(zb); Rbc=r*cos(zb);
X[3]=X[1]+Lb-Ll*sin(Rb1);Y[3]=Y[1]-Ll*cos(Rb1);
X[4]=X[1]+Lb;Y[4]=Y[1];
LX[1]=X[2]-Rfc;LY[1]=Y[2]+Rfs;
LX[2]=X[1]-Rfs-Rfc;
LY[2]=Y[1]-Rfc+Rfs;
LX[3]=X[2]+Rfc;LY[3]=Y[2]-Rfs;
LX[4]=X[1]-Rfs+Rfc;
LY[4]=Y[1]-Rfc-Rfs;
LX[5]=X[3]-Rfc;LY[5]=Y[3]+Rfs;
LX[6]=X[4]-Rfs-Rfc;
LY[6]=Y[4]-Rfc+Rfs;
LX[7]=X[3]+Rfc;
LY[7]=Y[3]-Rfs;
LX[8]=X[4]-Rfs+Rfc;
LY[8]=Y[4]-Rfc-Rfs;
setfillstyle(SOLID_FILL,BLACK);
bar(x11-100,y11-120,x11+250,y11+5);
setcolor(LIGHTCYAN);
arc(X[2],Y[2],Zf,180+Zf,r);
arc(X[1]-Rfs,Y[1]-Rfc,180+Zf,360+Zf,r);
line(LX[1],LY[1],LX[2],LY[2]);
line(LX[3],LY[3],LX[4],LY[4]);
arc(X[3],Y[3],Zb,180+Zb,r);
arc(X[4]-Rfs,Y[4]-Rfc,180+Zf,360+Zf,r);
line(LX[5],LY[5],LX[6],LY[6]);
line(LX[7],LY[7],LX[8],LY[8]);
rectangle(X[2]-15,Y[2]-15,X[3]+15,Y[3]+15);
N++;
```

```
        }
        fclose(infile);
    }
    if(I==2)
    {
        infile=fopen("d:\step12.xls","r");
        if(E!=0 && loop==M)
        {
            infile=fopen("d:\step22.xls","r");
        }
        fseek(infile,0,0);
        N=1;
        while(!feof(infile))
        {
           if(ferror(infile))
           {
                gprintf(10,100,"File error \n");
                exit(1);
           }
           fscanf(infile,"%f\t%f\n",&Rf,&Rb);
           Rf=Rf*pi/180;  Rb=Rb*pi/180;
           Rf1=fabs(Rf); Rb1=fabs(Rb);
           zeta=asin( LI/Lb*(cos(Rb1)-cos(Rf1)) );
           X[1]=x11-LI*sin(Rb1);Y[1]=y11-LI*cos(Rb1);
           X[2]=X[1]+Lb*cos(zeta); Y[2]=Y[1]+Lb*sin(zeta);
           X[3]=X[2]+LI*sin(Rf1); Y[3]=y11;
           zf=atan((x11-X[1])/(y11-Y[1]));Zf=zf*180/pi;
           zb=atan((X[3]-X[2])/(Y[3]-Y[2]));Zb=zb*180/pi;
           Rfs=r*sin(zf); Rfc=r*cos(zf);
```

```
Rbs=r*sin(zb); Rbc=r*cos(zb);
X[4]=x11-Rfs;Y[4]=y11-Rfc;
X[5]=X[1]-Rfc;Y[5]=Y[1]+Rfs;
X[6]=x11-Rfs-Rfc;
Y[6]=y11-Rfc+Rfs;
X[7]=X[1]+Rfc;Y[7]=Y[1]-Rfs;
X[8]=x11-Rfs+Rfc;
Y[8]=y11-Rfc-Rfs;
X[9]=X[1]+Lb*cos(zeta);Y[9]=Y[1]+Lb*sin(zeta);
X[10]=X[3]-Rbs; Y[10]=Y[3]-Rbc;
X[11]=X[2]-Rbc;Y[11]=Y[2]+Rbs;
X[12]=X[3]-Rbs-Rbc;
Y[12]=Y[3]-Rbc+Rbs;
X[13]=X[2]+Rbc;Y[13]=Y[2]-Rbs;
X[14]=X[3]-Rbs+Rbc;
Y[14]=Y[3]-Rbc-Rbs;
setfillstyle(SOLID_FILL,BLACK);
bar(x11-100,y11-120,x11+250,y11+5);
setcolor(LIGHTCYAN);
arc(X[1],Y[1],Zf,180+Zf,r);
arc(X[4],Y[4],180+Zf,360+Zf,r);
line(X[5],Y[5],X[6],Y[6]);
line(X[7],Y[7],X[8],Y[8]);
arc(X[9],Y[9],Zb,180+Zb,r);
arc(X[10],Y[10],180+Zb,360+Zb,r);
line(X[11],Y[11],X[12],Y[12]);
line(X[13],Y[13],X[14],Y[14]);
X[15]=X[1]-15*cos(zeta)+15*sin(zeta);
Y[15]=Y[1]-15*(sin(zeta)+cos(zeta));
```

```
            X[16]=X[15]-30*sin(zeta); Y[16]=Y[15]+30*cos(zeta);
            X[17]=X[15]+(Lb+30)*cos(zeta);Y[17]=Y[15]+(Lb+30)*sin(zeta);
            X[18]=X[16]+(Lb+30)*cos(zeta);Y[18]=Y[16]+(Lb+30)*sin(zeta);
            line(X[15],Y[15],X[16],Y[16]);
            line(X[16],Y[16],X[18],Y[18]);
            line(X[18],Y[18],X[17],Y[17]);
            line(X[17],Y[17],X[15],Y[15]);
            N++;
        }
        fclose(infile);
}
if(I==3)
{
        infile=fopen("d:\step13.xls","r");
        if(E!=0 && loop==M)
        {
            infile=fopen("d:\step23.xls","r");
        }
        fseek(infile,0,0);
        N=1;
        while(!feof(infile))
        {
            if(ferror(infile))
            {
                gprintf(10,100,"File error \n");
                exit(1);
            }
            fscanf(infile,"%f\t%f\n",&Rf,&Rb);
            Rf=Rf*pi/180;  Rb=Rb*pi/180;
```

```
Rf1=fabs(Rf);  Rb1=fabs(Rb);
zeta=asin( Ll/Lb*(cos(Rb)-cos(Rf)));
setcolor(LIGHTCYAN);
X[1]=x11+Ll*sin(Rb);Y[1]=y11-Ll*cos(Rb);
X[2]=X[1]+Lb*cos(zeta); Y[2]=Y[1]+Lb*sin(zeta);
X[3]=X[2]-Ll*sin(Rf);
zb=atan((x11-X[1])/(y11-Y[1])); Zb=zb*180/pi;
zf=atan((X[3]-X[2])/(y11-Y[2])); Zf=zf*180/pi;
Rfs=r*sin(zf);Rfc=r*cos(zf);
Rbs=r*sin(zb);Rbc=r*cos(zb);
X[4]=x11-Rbs; Y[4]=y11-Rbc;
setfillstyle(SOLID_FILL,BLACK);
bar(x11-100,y11-120,x11+250,y11+5);
arc(X[1],Y[1],Zb,180+Zb,r);
arc(X[4],Y[4],180+Zb,360+Zb,r);
line(X[1]-Rbc,Y[1]+Rbs,x11-Rbs-Rbc,y11-Rbc+Rbs);
line(X[1]+Rbc,Y[1]-Rbs,x11-Rbs+Rbc,y11-Rbc-Rbs);
arc(X[2],Y[2],Zf,180+Zf,r);
arc(X[3]-Rfs,y11-Rfc,180+Zf,360+Zf,r);
line(X[2]-Rfc,Y[2]+Rfs, X[3]-Rfs-Rfc,y11-Rfc+Rfs);
line(X[2]+Rfc,Y[2]-Rfs,X[3]-Rfs+Rfc,y11-Rfc-Rfs);
X[5]=X[1]-15*cos(zeta)+15*sin(zeta);
Y[5]=Y[1]-15*(sin(zeta)+cos(zeta));
X[6]=X[5]-30*sin(zeta); Y[6]=Y[5]+30*cos(zeta);
X[7]=X[5]+(Lb+30)*cos(zeta);
Y[7]=Y[5]+(Lb+30)*sin(zeta);
X[8]=X[6]+(Lb+30)*cos(zeta);
Y[8]=Y[6]+(Lb+30)*sin(zeta);
line(X[5],Y[5],X[6],Y[6]);
```

```
            line(X[6],Y[6],X[8],Y[8]);
            line(X[8],Y[8],X[7],Y[7]);
            line(X[7],Y[7],X[5],Y[5]);
            N++;
        }
        fclose(infile);
}
if(l==4)
{
        infile=fopen("d:\step14.xls","r");
        if(E!=0 && loop==M)
        {
            infile=fopen("d:\step24.xls","r");
        }
        fseek(infile,0,0);
        N=1;
        while(!feof(infile))
        {
            if(ferror(infile))
            {
                gprintf(10,100,"File error \n");
                exit(1);
            }
            fscanf(infile,"%f\t%f\n",&Rf,&Rb);
            Rf=Rf*pi/180;  Rb=Rb*pi/180;
            zeta=asin(Ll/Lb*(cos(Rf)-cos(Rb)));
            setcolor(LIGHTCYAN);
            X[1]=x11+Lb+Xg+Ll*sin(Rf)-Lb*cos(zeta)-Ll*sin(Rb);
            Y[1]=y11-r;
```

```
X[2]=X[1]+Ll*sin(Rb); Y[2]=y11-Ll*cos(Rb);
X[3]=X[2]+Lb*cos(zeta);
Y[3]=Y[2]-Lb*sin(zeta);
X[4]=x11+Lb+Xg;
zf=atan((X[3]-X[4])/(y11-Y[3]));Zf=zf*180/pi;
zb=atan((X[2]-X[1])/(y11-Y[2]));Zb=zb*180/pi;
Rfs=r*sin(zf);Rfc=r*cos(zf);
Rbs=r*sin(zb);Rbc=r*cos(zb);
X[5]=X[1]+Rbs;Y[5]=y11-Rbc;
X[6]=X[4]+Rfs;Y[6]=y11-Rfc;
setfillstyle(SOLID_FILL,BLACK);
bar(x11-100,y11-120,x11+250,y11+5);
arc(X[2],Y[2],360-Zb,180-Zb,r);
arc(X[5],Y[5],180-Zb,360-Zb,r);
line(X[2]-Rbc,Y[2]-Rbs,X[1]+Rbs-Rbc,y11-Rbc-Rbs);
line(X[2]+Rbc,Y[2]+Rbs,X[1]+Rbs+Rbc,y11-Rbc+Rbs);
arc(X[3],Y[3],360-Zf,180-Zf,r);
arc(X[6],Y[6],180-Zf,360-Zf,r);
line(X[3]-Rfc,Y[3]-Rfs,X[4]+Rfs-Rfc,y11-Rfc-Rfs);
line(X[3]+Rfc,Y[3]+Rfs,X[4]+Rfs+Rfc,y11-Rfc+Rfs);
X[7]=X[2]-15*cos(zeta)+15*sin(zeta);
Y[7]=Y[2]+15*(sin(zeta)-cos(zeta));
X[8]=X[7]+30*sin(zeta); Y[8]=Y[7]+30*cos(zeta);
X[9]=X[7]+(Lb+30)*cos(zeta);
Y[9]=Y[7]-(Lb+30)*sin(zeta);
X[10]=X[8]+(Lb+30)*cos(zeta);
Y[10]=Y[8]-(Lb+30)*sin(zeta);
line(X[7],Y[7],X[8],Y[8]);
line(X[8],Y[8],X[10],Y[10]);
```

```
            line(X[10],Y[10],X[9],Y[9]);
            line(X[9],Y[9],X[7],Y[7]);
            N++;
        }
        fclose(infile);
    }
    if(I==5)
    {
        infile=fopen("d:\step15.xls","r");
        if(E!=0) infile=fopen("d:\step25.xls","r");
        fseek(infile,0,0);
        N=1;
        while(!feof(infile))
        {
            if(ferror(infile))
            {
                gprintf(10,100,"File error \n");
                exit(1);
            }
            fscanf(infile,"%f\t%f\n",&Rf,&Rb);
            Rf=Rf*pi/180;  Rb=Rb*pi/180;
            setcolor(LIGHTCYAN);
            setfillstyle(SOLID_FILL,BLACK);
            bar(x11-100,y11-120,x11+250,y11+5);
            X[1]=x11+Xg;
            X[2]=X[1]+LI*sin(Rb); Y[2]=y11-LI*cos(Rb);
            X[3]=x11+Lb+Xg+LI*sin(Rf);Y[3]=y11-LI*cos(Rf);
            X[4]=X[3]-LI*sin(Rf);
            zf=atan((X[3]-X[4])/(y11-Y[3]));Zf=zf*180/pi;
```

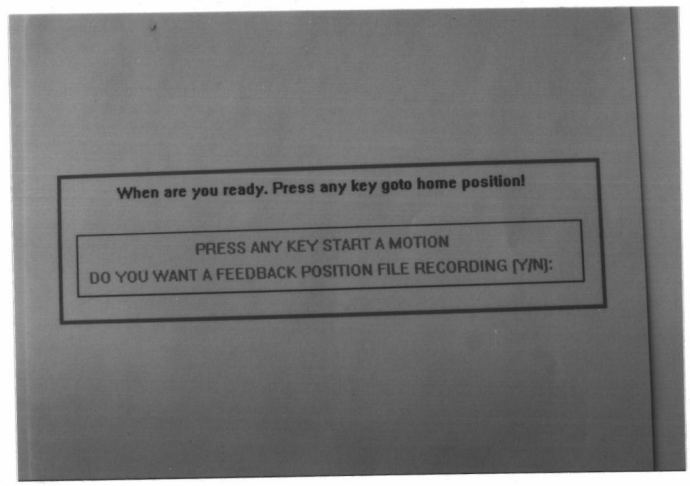
```
                    zb=atan((X[2]-X[1])/(y11-Y[2]));Zb=zb*180/pi;
                    Rfs=r*sin(zf);Rfc=r*cos(zf);
                    Rbs=r*sin(zb);Rbc=r*cos(zb);
                    X[5]=X[1]+Rbs;Y[5]=y11-Rbc;
                    X[6]=X[4]+Rfs;Y[6]=y11-Rfc;
                    arc(X[2],Y[2],360-Zb,180-Zb,r);
                    arc(X[5],Y[5],180-Zb,360-Zb,r);
                    line(X[2]-Rbc,Y[2]-Rbs,X[1]+Rbs-Rbc,y11-Rbc-Rbs);
                    line(X[2]+Rbc,Y[2]+Rbs,X[1]+Rbs+Rbc,y11-Rbc+Rbs);
                    arc(X[3],Y[3],360-Zf,180-Zf,r);
                    arc(X[6],Y[6],180-Zf,360-Zf,r);
                    line(X[3]-Rfc,Y[3]-Rfs,X[4]+Rfs-Rfc,y11-Rfc-Rfs);
                    line(X[3]+Rfc,Y[3]+Rfs,X[4]+Rfs+Rfc,y11-Rfc+Rfs);
                    rectangle(X[2]-15,Y[2]-15,X[3]+15,Y[3]+15);
                    N++;
                }
                fclose(infile);
            }
        }
        loop++;
    }
    sound(200);
    delay(200);
    nosound();
    setcolor(LIGHTRED);
    outtextxy(180,250,"PLEASE PRESS ANY KEY GOTO GEN MENU");
    getch();
    setlinestyle(0,0xffff,1);
}
```

```
gprintf( int xloc, int yloc, char *fmt, ... )
{
  va_list  argptr;
  char str[150];
  int cnt;
  va_start( argptr, format );
  cnt = vsprintf( str, fmt, argptr );
  outtextxy( xloc,yloc, str );
  yloc += textheight( "H" ) + 2;
  va_end( argptr );
  return( cnt );
}
scan_textbar(int x_lt1,int y_lt1,int x_lb2,int y_lb2)
{
      setcolor(LIGHTBLUE);
      rectangle(x_lt1,y_lt1,x_lb2,y_lb2);
      setfillstyle(1,BLACK);
      floodfill(x_lt1+1,y_lt1+1,LIGHTBLUE);
      setlinestyle(0,0xffff,3);
      rectangle(x_lt1,y_lt1,x_lb2,y_lb2);
      setlinestyle(0,0xffff,1);
      setcolor(RED);
 }
 ground(int x_t,int y_t,int x_l,int y_l)
 {
      setfillstyle(1,WHITE);
      bar(x_t,y_t,x_l,y_l);
      setcolor(LIGHTCYAN);
      setfillstyle(1,LIGHTGRAY);
```

```
    setcolor(BLACK);
}
```

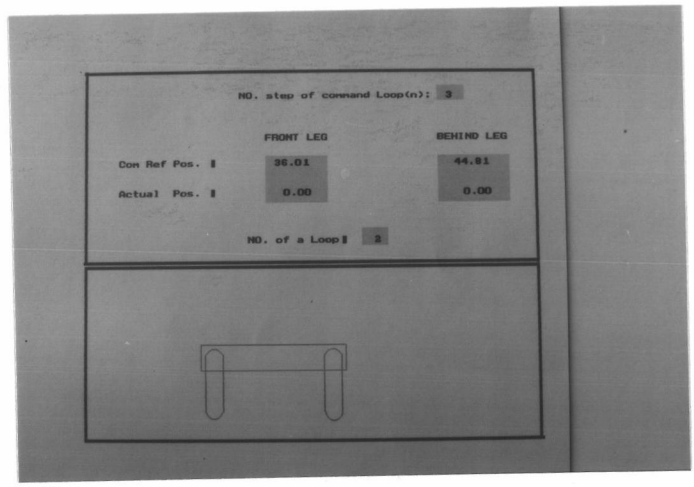
3. ส่วนของการควบคุมระบบ

เป็นการนำค่าคำสั่งในการควบคุมโดยการอ่านค่าคำสั่งจากไฟล์ที่สร้างไว้ภายใต้การควบคุมแบบแบง-แบง ที่ พีควบคุมซึ่งจากเมนูหลักเมื่อเลือกควบคุมระบบและให้กดคีย์อีกครั้ง ระบบก็จะเคลื่อนตัวเข้าสู่ตำแหน่งเริ่มต้นของระบบ หรือ ยางเลือออกจากโปรแกรมระบบก็ ออกจากโปรแกรมการควบคุมมาสู่เมนูหลักของการควบคุม ซึ่งหน้าจอการควบคุมถูกแสดงผลดังรูปที่ ข.6



รูปที่ ข.6 แสดงหน้าจอการควบคุมที่ระบบวิ่งเข้าสู่ตำแหน่งเริ่มต้น

และจากหน้าจอดังกล่าวเมื่อต้องการควบคุมระบบให้เลือกประเภทของการควบคุมว่าต้องการการควบคุม โดยมีการบันทึกผลการตอบสนองหรือไม่ถ้าต้องการก็กดคีย์ Y และถ้าไม่ต้องการก็กดคีย์ N ก็จะปรากฏหน้าจอของการควบคุมระบบจริงๆ ดังรูปที่ ข.7



รูปที่ ข.7 แสดงหน้าจอของการควบคุมระบบจริง

ซึ่งจากรูปผลแสดงทางหน้าจอดังกล่าวเป็นไปตามโปรแกรมดังต่อไปนี้ ที่ภายใต้การควบคุมแบบพี. ควบคุม

```
/* ***************** 3 spline ctrl ********************** */
Control_pos()
{
   int        ii,K;
   unsigned bit;
infile=fopen("d:\comm.xls","r");
fseek(infile,0,0);
while(!feof(infile))
{
    if(ferror(infile))
    {
          printf(10,10,"File error \n");
          exit(1);
```

```
    }
    fscanf(infile,"%f\t%d\t%f\t%f\t%f\t%f\t%f\t%f\n",&D,&m,&E,&s,&S,&w,&a,&T);
}
  fclose(infile);
  count=T*2*pow(10,6);
  clearviewport();
  ground(15,130,xx-15,260);
  setcolor(RED);
  setlinestyle(0,0xffff,3);
  rectangle(15,130,xx-15,260);
  setlinestyle(0,0xffff,1);
  setcolor(BLUE);
  write_pos(vf=0,vb=0);
  outtextxy(100,150,"When Are you ready? PRESS ANY KEY GOTO HOME POSITION!");
  getch();
  dat[0]=base;
  dat[1]=2;
  dat[2]=1;
  fun=0;
  er=pcl812(fun,&dat[0],&ary1[0],&ary2[0]);
  goto_home();
  decode_reset(bit=1);
  delay(20);
  decode_reset(bit=0);
  sound(300);
  rectangle(90,190,540,245);
  setcolor(MAGENTA);
  outtextxy(186,200,"PRESS ANY KEY START A MOTION");
  delay(800);
```

```
nosound();
outtextxy(100,230,"DO YOU WANT A FEEDBACK POSITION FILE RECORDING (Y/N):");
y=getch();
ctrl_text();
setcolor(LIGHTBLUE);
outtextxy(180,450,"THIS SCALE IS 1 CM./DIVISION");
line(x1,y1+7,x1+350,y1+7);
line(x1,y1+7,x1+350,y1+7-350*tan(S));
K=1;
while(K<=70)
{
    line(x1+K*5*cos(S),y1+7-K*5*sin(S),\
    x1+K*5*cos(S)+4*sin(S),y1+7-K*5*sin(S)+4*cos(S));
    K++;
}
for(J=1;J<=m;J++)
{
    i=1;
    while(i<=5)
    {
        ctrl_pos();
        i++;
      }
      goto_home();
      decode_reset(bit=1);
      delay(20);
      decode_reset(bit=0);
  }
  write_pos(vf=0,vb=0);
```

```
    setcolor(LIGHTRED);
    outtextxy(150,250,"END OF CONTROL PRESS ANY KEY GOTO MAIN MENU");
    sound(300);
    delay(800);
    nosound();
    getch();
}
ctrl_text()
{
      ground(0,0,xx,275);
      setcolor(RED);
      rectangle(15,15,xx-15,260);
      outtextxy(190, 30,"NO. step of command Loop(n):");
      outtextxy(220,80,"FRONT LEG");
      outtextxy(420,80,"BEHIND LEG");
      outtextxy( 50,110,"Com Ref Pos. □ ");
      outtextxy( 50,145,"Actual  Pos. □ ");
      setlinestyle(0,0xffff,3);
      rectangle(0,0,xx,275);
      rectangle(0,280,xx,yy);
      setlinestyle(0,0xffff,1);
  }
  state_home() /* *check cw. or ccw.* */
  {
   int   HI,LO,n,kk;
   int   ch,start,stp;
         outportb(base+11,0);
         start=0;
         stp=3;
```

```
        inportb(base+4);
        n=3;
        ch=start;
        for (kk=0;kk<=n;kk++)
        {
                outportb(base+10,ch);
                outportb(base+11,6);
                do
                {
                        HI=inportb(base+5);
                }while (HI > 15 );
                LO=inportb(base+4);
                data[kk]=(HI*256+LO-2048)*5.0/2048;
                if (stp > ch)
                ch=ch+1;
                else
                ch=start;
           }
           outportb(base+11,0);
}

read_pos(double *cf,double *cb) /* Encode. 800 Pulse/rev */
{
  int LO1,HI1,LO2,HI2;

     outportb(0x1b3, 0x9b);  /* initial 8255 #1 mode15 */
     LO1=0xff&inportb(0x1b0);
     LO2=0xff&inportb(0x1b1);
     outportb(0x1b7,0x9b);   /* initial 8255 #2 mode15 */
```

```
    HI1=0xf&inportb(0x1b4);
    HI2=0xf&inportb(0x1b5);
    *cf=(LO1+HI1*256)*.45;
    if(*cf>1350)  {*cf=*cf-1842.75;}
    *cb=(LO2+HI2*256)*.45;
    if(*cb>1350)  {*cb=*cb-1842.75;}
}
write_pos(double VF,double VB)
{
    int    LO1,HI1,LO2,HI2;
    double X1,X2;

          X1=VF*2048/2.5+2048;
          HI1=X1/256;
          LO1=X1-(HI1*256);
          X2=VB*2048/2.5+2048;
          HI2=X2/256;
          LO2=X2-(HI2*256);
          outportb(base+4,LO1);
          outportb(base+5,HI1);
          outportb(base+6,LO2);
          outportb(base+7,HI2);
 }
 decode_reset(unsigned bit)
 {
    int dat[5],ary1[5],ary2[5],y;
    int er,fun;
        fun=21;
        dat[1]=bit/256;
```

```
        dat[0]=bit-dat[1]*256;
        er=pcl812 (fun,&dat[0],&ary1[0],&ary2[0]);
}
/* ************************sub menu CONTROL ************************ */
Ctrl_menu()
{
      shado(380,275,580,335);
        if(e>2){e=1,ee=0;}
        if(e<1){e=2,ee=30;}
      bar(380,275+ee,580,305+ee);
      setfillstyle(1,YELLOW);
      bar(380,275+ee,580,305+ee);
      setcolor(LIGHTMAGENTA);
      rectangle(380,275+ee,580,305+ee);
      setcolor(BLACK);
      setfillstyle(1,WHITE);
      outtextxy(415,290," CONTROL POSITION");
      outtextxy(415,320," QUIT TO MAIN MENU");
 }
 /* ***********General graphics func ***********************/
 scan_textbar(int x_lt1,int y_lt1,int x_lb2,int y_lb2)
 {
       setcolor(LIGHTBLUE);
       rectangle(x_lt1,y_lt1,x_lb2,y_lb2);
       setfillstyle(1,BLACK);
       floodfill(x_lt1+1,y_lt1+1,LIGHTBLUE);
       setlinestyle(0,0xffff,3);
       rectangle(x_lt1,y_lt1,x_lb2,y_lb2);
       setlinestyle(0,0xffff,1);
```

```
    setcolor(RED);
}
shado(int a,int b,int c,int d)
{
    setfillstyle(1,LIGHTBLUE);
    bar(a+8,b+8,c+8,d+8);
    setfillstyle(1,LIGHTCYAN);
    bar(a,b,c,d);
    setcolor(BLACK);
    rectangle(a,b,c,d);
    setfillstyle(1,LIGHTGRAY);
     return;
}
ground(int x_t,int y_t,int x_l,int y_l)
{
    setfillstyle(1,WHITE);
    bar(x_t,y_t,x_l,y_l);
    setcolor(LIGHTCYAN);
    setfillstyle(1,LIGHTGRAY);
    setcolor(BLACK);
}
remark(char word[80],int x1,int y1,int x2,int y2)
{
        setfillstyle(1,LIGHTCYAN);
        bar(x1,y1,x2,y2);
        setcolor(BLACK);
        gprintf(x1+10,y1+5,"%.60s",word);
        setcolor(LIGHTBLUE);
}
```

```
gprintf( int xloc, int yloc, char *fmt, ... )
{
  va_list  argptr;
  char str[150];
  int cnt;
  va_start( argptr, format );
  cnt = vsprintf( str, fmt, argptr );
  outtextxy( xloc,yloc, str );
  yloc += textheight( "H" ) + 2;
  va_end( argptr );
  return( cnt );
}

 goto_home()   /* *Single spline (a=const)* */
 {
  int    c;
  float    wf=0.4,KPf=0.24,KIf=0,KDf=0,K0F,K1F,K2F,RF,
           wb=0.4,KPb=.24,KIb=0,KDb=0,K0B,K1B,K2B,RB;
  double Cf,vF,EF,EF2,PF,EF1,MF,MF1,CF0,Cfi,Cbi,
           Cb,vB,EB,EB2,PB,EB1,MB,MB1,CB0;
           EF=PF=EF1= 0;
           K0F=KPf+(KIf*T)+(KDf/T);
           K1F=-KPf-(2*KDf/T);
           K2F=KDf/T;
           EB=PB=EB1=0.0;
           K0B=KPb+(KIb*T)+(KDb/T);
           K1B=-KPb-(2*KDb/T);
           K2B=KDb/T;
           dat[0]=0;
```

```
fun=10;
er=pcl812 (fun,&dat[0],&ary1[0],&ary2[0]);
dat[0]=count;
fun=11;
er=pcl812 (fun,&dat[0],&ary1[0],&ary2[0]);
read_pos(&CF0,&CB0);
Cfi=CF0;
Cbi=CB0;
c=1;
while(c<1200)
{
    read_pos(&CF0,&CB0);
    Cf=CF0;
    Cb=CB0;
    RF = 0.5*wf*pow(T*c,2);
    RB = 0.5*wb*pow(T*c,2);
    state_home();
    if (data[2]<2.0 ) EF = Cfi-RF-Cf;
    if (data[2]>=2.0) EF = Cfi+RF-Cf;
    if (data[3]<2.0 ) EB = Cbi-RB-Cb;
    if (data[3]>=2.0) EB = Cbi+RB-Cb;
    MF = PF+(K0F*EF);
    MB = PB+(K0B*EB);
    vF = MF;
    vB = MB;
    if(fabs(vF)>2.5) vF=vF/fabs(vF)*2.5;
    if(fabs(vB)>2.5) vB=vB/fabs(vB)*2.5;
    if(data[0]>1.0) vF=0;
    if(data[1]>1.0) vB=0;
```

```
        do
        {
            dat[0]=1;
            fun=12;
        }while(dat[1]<0);
        write_pos(vF,vB);
        dat[0]=count;
        fun=11;
        er=pcl812 (fun,&dat[0],&ary1[0],&ary2[0]);
        state_home();
        if( data[0]>1.0 && data[1]>1.0 ) break;
        if(data[0]>1.0) write_pos(vF=0,vB);
        if(data[1]>1.0) write_pos(vF,vB=0);
        EF2 = EF1; EF1= EF ;
        EB2 = EB1; EB1= EB ;
        MF1 = MF;
        MB1 = MB;
        PF  = MF1+(K1F*EF1)+(K2F*EF2);
        PB  = MB1+(K1B*EB1)+(K2B*EB2);
        c++;
    }
    write_pos(vF=0,vB=0);
}
ctrl_pos()
{
  float   Kpf=0.24,Kif=0,Kdf=0,
          Kpb=0.24,Kib=0,Kdb=0;
  float   Rf,RFF[4000],Cf,CF,CF1,CFs,CFc,Rf1,RF[7],Zf,zf,K0f,K1f,K2f,K1,
          Rb,RBB[4000],Cb,CB,CB1,CBs,CBc,Rb1,RB[7],Zb,zb,K0b,K1b,K2b,K;
```

```
float    Xg,x11,y11,Ll=90,Lb=135,zeta,l,X[20],Y[20],LX[10],LY[10],tim,t;
double   Ef,Eb,E1f,E1b,E2f,E2b,Mf,Mb,M1f,M1b,Pf,Pb,Cf0,Cb0;
int      r=10,P[7];
         setlinestyle(0,0xffff,3);
         Xg=D*5;
         setfillstyle(SOLID_FILL,LIGHTGRAY);
         bar(330,190,360,210);
         setcolor(RED);
         outtextxy(200,200,"NO. of a Loop□ ");
         setcolor(BLUE);
         gprintf(345,200,"%d",J);
         C=s/D;
         infile=fopen("d:\loop1.xls","r");
         fseek(infile,0,0);
         j=1;
         while(!feof(infile))
         {
            if(ferror(infile))
            {
                  printf(10,10,"File error \n");
                  exit(1);
            }
            fscanf(infile,"%d\t%f\t%f\n",&P[j],&RF[j],&RB[j]);
            j++;
         }
         fclose(infile);
         if(E!=0 && J==m )
         {
            Xg=s-D*(m-1)*5;
```

```
  infile=fopen("d:\loop2.xls","r");
  fseek(infile,0,0);
  j=1;
  while(!feof(infile))
  {
        if(ferror(infile))
        {
            printf(10,10,"File error \n");
            exit(1);
        }
        fscanf(infile,"%d\t%f\t%f\n",&P[j],&RF[j],&RB[j]);
        j++;
   }
   fclose(infile);
}
x11=x1+Xg*cos(S)*(J-1);
y11=y1-Xg*sin(S)*(J-1);
bar(220,105,290,160);
bar(420,105,500,160);
setcolor(BLUE);
Cf=0; Cb=0; Rf=0; Rb=0;
gprintf(230,110,"%5.2f                    %5.2f",Rf,Rb);
gprintf(233,145,"%5.2f                    %5.2f",Cf,Cb);
for(i=1;i<=5;i++)
{
   setfillstyle(SOLID_FILL,LIGHTGRAY);
   bar(420,23,450,40);
   setcolor(BLUE);
   gprintf(430,30,"%d",i);
```

```
Ef=E1f=E2f=0;
Eb=E1b=E2b=0;
Mf=M1f=Pf=0;
Mb=M1b=Pb=0;
K0f=Kpf+(Kif*T)+(Kdf/T);
K1f=-Kpf-(2*Kdf/T);
K2f=Kdf/T;

K0b=Kpb+(Kib*T)+(Kdb/T);
K1b=-Kpb-(2*Kdb/T);
K2b=Kdb/T;
if(i==1)
{
    infile=fopen("d:\step11.xls","r");
    if(E != 0 && J==m)
    {
        infile=fopen("d:\step21.xls","r");
    }
}
if(i==2)
{
    infile=fopen("d:\step12.xls","r");
    if(E !=0 && J==m)
    {
        infile=fopen("d:\step22.xls","r");
    }
}
if(i==3)
{
```

```
    infile=fopen("d:\step13.xls","r");
    if(E !=0 && J==m)
    {
          infile=fopen("d:\step23.xls","r");
    }
}
if(i==4)
{
     infile=fopen("d:\step14.xls","r");
     if(E !=0 && J==m)
     {
           infile=fopen("d:\step24.xls","r");
     }
 }
 if(i==5)
 {
      infile=fopen("d:\step15.xls","r");
      if(E !=0 && J==m)
      {
            infile=fopen("d:\step25.xls","r");
      }
  }
  dat[0]=0;
  fun=10;
  er=pcl812 (fun,&dat[0],&ary1[0],&ary2[0]);
  dat[0]=count;
  fun=11;
  er=pcl812 (fun,&dat[0],&ary1[0],&ary2[0]);
  if( (y=='y') )
```

```
{
    if(J==1)
    {
        if(i==1)
        {
            outfile=fopen("d:\pos11.xls","w");
        }
        if(i==2)
        {
            outfile=fopen("d:\pos12.xls","w");
        }
        if(i==3)
        {
            outfile=fopen("d:\pos13.xls","w");
        }
        if(i==4)
        {
            outfile=fopen("d:\pos14.xls","w");
        }
        if(i==5)
        {
            outfile=fopen("d:\pos15.xls","w");
        }
    }
    if(E!=0 && J==m )
    {
        if(i==1)
        {
            outfile=fopen("d:\pos21.xls","w");
```

```
        }
        if(i==2)
        {
            outfile=fopen("d:\pos22.xls","w");
        }
        if(i==3)
        {
            outfile=fopen("d:\pos23.xls","w");
        }
        if(i==4)
        {
            outfile=fopen("d:\pos24.xls","w");
        }
        if(i==5)
        {
            outfile=fopen("d:\pos25.xls","w");
        }
    }
}
setfillstyle(SOLID_FILL,BLACK);
bar(170,290,500,310);
setfillstyle(SOLID_FILL,LIGHTGRAY);
fseek(infile,0,0);
j=1;
while(!feof(infile))
{
    fscanf(infile,"%f\t%f\n",&Rf,&Rb);
    RFF[j]=Rf;    RBB[j]=Rb;
    j++;
```

```
}
fclose(infile);
k=1;
while(k<=P[i] && !kbhit() )
{
    Rf=RFF[k];   Rb=RBB[k];
    setfillstyle(SOLID_FILL,LIGHTGRAY);
    read_pos(&Cf0,&Cb0);
    Cf=Cf0; Cb=Cb0;
    if( (y=='y') )
    {
        fprintf(outfile,"%5.4f\t%5.4f\n",Cf,Cb);
    }
    CF=Cf*pi/180; CB=Cb*pi/180;
    K1=k; K1=K1/30-k/30;
    if( K1==0 )
    {
        bar(220,105,290,160);
        bar(420,105,500,160);
        setcolor(BLUE);
        gprintf(230,110,"%5.2f                    %5.2f",Rf,Rb);
        gprintf(233,145,"%5.2f                    %5.2f",Cf,Cb);
    }
    K=k; K=K/25-k/25;
    if( (i==1 || i==2) && K==0 )
    {
        CF1=fabs(CF); CB1=fabs(CB);
        zeta=asin( Ll/Lb*(cos(CB1)-cos(CF1)) );
        X[1]=x11-Ll*sin(CB1);Y[1]=y11-Ll*cos(CB1);
```

```
X[2]=X[1]+Lb*cos(zeta); Y[2]=Y[1]+Lb*sin(zeta);
X[3]=X[2]+Ll*sin(CF1); Y[3]=y11;
zf=atan((x11-X[1])/(y11-Y[1]));Zf=zf*180/pi;
zb=atan((X[3]-X[2])/(Y[3]-Y[2]));Zb=zb*180/pi;
CFs=r*sin(zf); CFc=r*cos(zf);
CBs=r*sin(zb); CBc=r*cos(zb);
X[4]=x11-CFs;Y[4]=y11-CFc;
X[5]=X[1]-CFc;Y[5]=Y[1]+CFs;
X[6]=x11-CFs-CFc;
Y[6]=y11-CFc+CFs;
X[7]=X[1]+CFc;Y[7]=Y[1]-CFs;
X[8]=x11-CFs+CFc;
Y[8]=y11-CFc-CFs;
X[9]=X[1]+Lb*cos(zeta);Y[9]=Y[1]+Lb*sin(zeta);
X[10]=X[3]-CBs; Y[10]=Y[3]-CBc;
X[11]=X[2]-CBc;Y[11]=Y[2]+CBs;
X[12]=X[3]-CBs-CBc;
Y[12]=Y[3]-CBc+CBs;
X[13]=X[2]+CBc;Y[13]=Y[2]-CBs;
X[14]=X[3]-CBs+CBc;
Y[14]=Y[3]-CBc-CBs;
setfillstyle(SOLID_FILL,BLACK);
bar(x11-100,y11-120,x11+250,y11+5);
setcolor(LIGHTCYAN);
arc(X[1],Y[1],Zf,180+Zf,r);
arc(X[4],Y[4],180+Zf,360+Zf,r);
line(X[5],Y[5],X[6],Y[6]);
line(X[7],Y[7],X[8],Y[8]);
arc(X[9],Y[9],Zb,180+Zb,r);
```

```
        arc(X[10],Y[10],180+Zb,360+Zb,r);
        line(X[11],Y[11],X[12],Y[12]);
        line(X[13],Y[13],X[14],Y[14]);
        X[15]=X[1]-15*cos(zeta)+15*sin(zeta);
        Y[15]=Y[1]-15*(sin(zeta)+cos(zeta));
        X[16]=X[15]-30*sin(zeta); Y[16]=Y[15]+30*cos(zeta);
        X[17]=X[15]+(Lb+30)*cos(zeta);Y[17]=Y[15]+(Lb+30)*sin(zeta);
        X[18]=X[16]+(Lb+30)*cos(zeta);Y[18]=Y[16]+(Lb+30)*sin(zeta);
        line(X[15],Y[15],X[16],Y[16]);
        line(X[16],Y[16],X[18],Y[18]);
        line(X[18],Y[18],X[17],Y[17]);
        line(X[17],Y[17],X[15],Y[15]);
}
if(i==3 && K==0)
{
        CF1=fabs(CF);  CB1=fabs(CB);
        zeta=asin( Ll/Lb*(cos(CB)-cos(CF)));
        setcolor(LIGHTCYAN);
        X[1]=x11+Ll*sin(CB);Y[1]=y11-Ll*cos(CB);
        X[2]=X[1]+Lb*cos(zeta); Y[2]=Y[1]+Lb*sin(zeta);
        X[3]=X[2]-Ll*sin(CF);
        zb=atan((x11-X[1])/(y11-Y[1])); Zb=zb*180/pi;
        zf=atan((X[3]-X[2])/(y11-Y[2])); Zf=zf*180/pi;
        CFs=r*sin(zf);CFc=r*cos(zf);
        CBs=r*sin(zb);CBc=r*cos(zb);
        X[4]=x11-CBs; Y[4]=y11-CBc;
        setfillstyle(SOLID_FILL,BLACK);
        bar(x11-100,y11-120,x11+250,y11+5);
        arc(X[1],Y[1],Zb,180+Zb,r);
```

```
arc(X[4],Y[4],180+Zb,360+Zb,r);
line(X[1]-CBc,Y[1]+CBs,x11-CBs-CBc,y11-CBc+CBs);
line(X[1]+CBc,Y[1]-CBs,x11-CBs+CBc,y11-CBc-CBs);
arc(X[2],Y[2],Zf,180+Zf,r);
arc(X[3]-CFs,y11-CFc,180+Zf,360+Zf,r);
line(X[2]-CFc,Y[2]+CFs, X[3]-CFs-CFc,y11-CFc+CFs);
line(X[2]+CFc,Y[2]-CFs,X[3]-CFs+CFc,y11-CFc-CFs);
X[5]=X[1]-15*cos(zeta)+15*sin(zeta);
Y[5]=Y[1]-15*(sin(zeta)+cos(zeta));
X[6]=X[5]-30*sin(zeta); Y[6]=Y[5]+30*cos(zeta);
X[7]=X[5]+(Lb+30)*cos(zeta);
Y[7]=Y[5]+(Lb+30)*sin(zeta);
X[8]=X[6]+(Lb+30)*cos(zeta);
Y[8]=Y[6]+(Lb+30)*sin(zeta);
line(X[5],Y[5],X[6],Y[6]);
line(X[6],Y[6],X[8],Y[8]);
line(X[8],Y[8],X[7],Y[7]);
line(X[7],Y[7],X[5],Y[5]);
}
if( (i==4 || i==5) && K==0 )
{
    zeta=asin(Ll/Lb*(cos(CF)-cos(CB)));
    setcolor(LIGHTCYAN);
    X[1]=x11+Lb+Xg+Ll*sin(CF)-Lb*cos(zeta)-Ll*sin(CB);
    Y[1]=y11-r;
    X[2]=X[1]+Ll*sin(CB); Y[2]=y11-Ll*cos(CB);
    X[3]=X[2]+Lb*cos(zeta);
    Y[3]=Y[2]-Lb*sin(zeta);
    X[4]=x11+Lb+Xg;
```

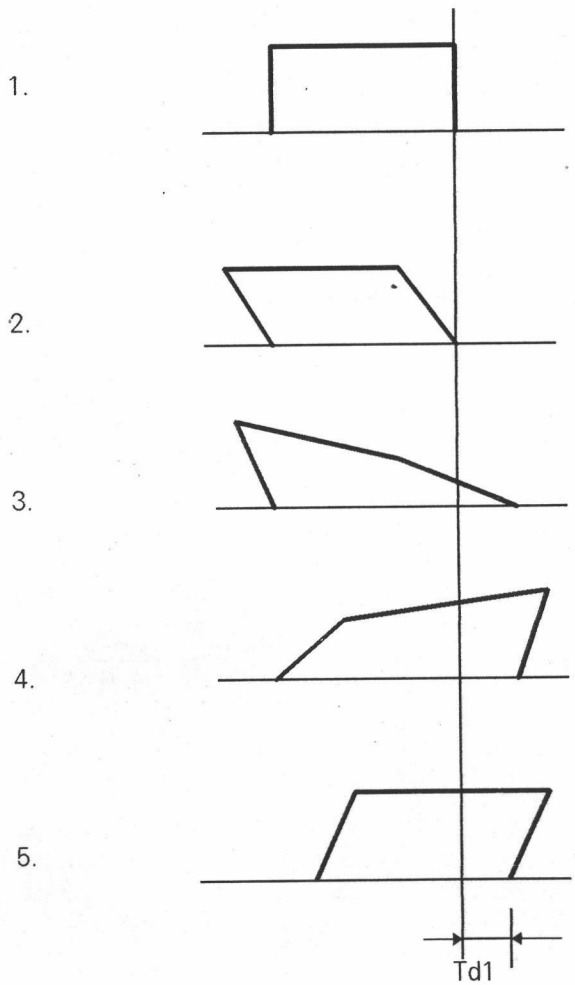
```
zf=atan((X[3]-X[4])/(y11-Y[3]));Zf=zf*180/pi;
zb=atan((X[2]-X[1])/(y11-Y[2]));Zb=zb*180/pi;
CFs=r*sin(zf);CFc=r*cos(zf);
CBs=r*sin(zb);CBc=r*cos(zb);
X[5]=X[1]+CBs;Y[5]=y11-CBc;
X[6]=X[4]+CFs;Y[6]=y11-CFc;
setfillstyle(SOLID_FILL,BLACK);
bar(x11-100,y11-120,x11+250,y11+5);
arc(X[2],Y[2],360-Zb,180-Zb,r);
arc(X[5],Y[5],180-Zb,360-Zb,r);
line(X[2]-CBc,Y[2]-CBs,X[1]+CBs-CBc,y11-CBc-CBs);
line(X[2]+CBc,Y[2]+CBs,X[1]+CBs+CBc,y11-CBc+CBs);
arc(X[3],Y[3],360-Zf,180-Zf,r);
arc(X[6],Y[6],180-Zf,360-Zf,r);
line(X[3]-CFc,Y[3]-CFs,X[4]+CFs-CFc,y11-CFc-CFs);
line(X[3]+CFc,Y[3]+CFs,X[4]+CFs+CFc,y11-CFc+CFs);
X[7]=X[2]-15*cos(zeta)+15*sin(zeta);
Y[7]=Y[2]+15*(sin(zeta)-cos(zeta));
X[8]=X[7]+30*sin(zeta); Y[8]=Y[7]+30*cos(zeta);
X[9]=X[7]+(Lb+30)*cos(zeta);
Y[9]=Y[7]-(Lb+30)*sin(zeta);
X[10]=X[8]+(Lb+30)*cos(zeta);
Y[10]=Y[8]-(Lb+30)*sin(zeta);
line(X[7],Y[7],X[8],Y[8]);
line(X[8],Y[8],X[10],Y[10]);
line(X[10],Y[10],X[9],Y[9]);
line(X[9],Y[9],X[7],Y[7]);
}
Ef=Rf-Cf;
```

```
                Eb=Rb-Cb;
                Mf=Pf+ K0f*Ef;
                Mb=Pb+ K0b*Eb;
                vf=Mf;
                if(fabs(vf)>2.5) vf=(2.5*vf)/fabs(vf);
                vb=Mb;
                if(fabs(vb)>2.5) vb=(2.5*vb)/fabs(vb);
                do
                {
                    dat[0]=1;
                    fun=12;
                    er=pcl812 (fun,&dat[0],&ary1[0],&ary2[0]);
                }while(dat[1]<0);
                write_pos(vf,vb);
                dat[0]=count;
                fun=11;
                er=pcl812 (fun,&dat[0],&ary1[0],&ary2[0]);

                E2f=E1f;E1f=Ef;
                E2b=E1b;E1b=Eb;
                M1f=Mf;
                M1b=Mb;
                Pf=M1f+ K1f*E1f+ K2f*E2f;
                Pb=M1b+ K1b*E1b+ K2b*E2b;
                k++;
            }
            fclose(outfile);
        }setlinestyle(0,0xffff,1);
}       /*  * สิ้นสุดโปรแกรมควบคุม * */
```

วิธีการเรียกใช้โปรแกรมในการควบคุม

โปรแกรมที่ใช้ในการควบคุมเขียนขึ้นจากภาษาซีซึ่งองค์ประกอบของโปรแกรมประกอบไปด้วยส่วนหลัก ๆ 3 ส่วนด้วยกัน คือ

1. การสร้างไฟล์ทางเดินหรือตำแหน่งต่างๆ ของการก้าวเดิน การสร้างไฟล์ของทางเดินนี้เป็นการสร้างคำสั่งในการเดินจากการคำนวณหาจากการป้อนระยะทางและความลาดชันของพื้นในการก้าวเดินโดยแบ่งลำดับการเดินออกเป็นทั้งหมด 5 ลำดับด้วยกันดังรูปที่ 4.9



รูปที่ ข.8 แสดงลำดับการเดินของหุ่นยนต์เดินด้วยกลไกสี่ลิงค์

จากรูปที่ 4.10 คำสั่งในการก้าวเดินในแต่ละสเต็ปเป็นไปตามการคำนวณหาจุดศูนย์ถ่วงโดยการประมาณทางสถิตย์ศาสตร์ซึ่งเป็นไปตามสมการที่ (2.44) ถึง (2.47)

หลังจากที่ได้มุมหรือตำแหน่งในการก้างเดินแต่ละลำดับเราก็นำคำสั่งดังกล่าวมาแบ่งตามการสุ่มหรือการแซมปลิง โดยการแบ่งเป็นไปตามสมการๆ เคลื่อนที่ซึ่งประกอบไปด้วยส่วนของความเร่งคงที่,ความเร็วคงที่ และความหน่วงคงที่ตามสมการที่ (2.24) ถึง(2.31) ซึ่งโปรไฟล์ของคำสั่งควบคุมมีลักษณะดังรูปที่ 2.19 ตรงขั้นนี้ไฟล์ข้อมูลที่ได้ประกอบไปด้วยไฟล์ชื่อstep11,step12, step13,step14 และstep15.xls ซึ่งอยู่บนแรมดิสค์ไดรฟ์

2. การตรวจสอบไฟล์ทางเดินด้วยการแสดงผลการก้าวเดินบนหน้าจอ ด้วยภาพ หลังจากที่เราทำการสร้างไฟล์คำสั่งดังกล่าวเสร็จแล้วก่อนที่จะทำการควบคุมจริงๆ เราสามารถตรวจสอบไฟล์ดังกล่าวด้วยการทดลองแสดงผลการควบคุมของไฟล์คำสั่งเป็นรูปจำลองลักษณะการเคลื่อนที่ของหุ่นยนต์บนหน้าจอคอมพิวเตอร์ ซึ่งในส่วนนี้มีให้เลือก 2โหมดด้วยกันคือเลือกดูแต่ละลำดับหรือเลือกทั้งกระบวนการ

3. เริ่มทำการควบคุมการก้าวเดินตามไฟล์ทางเดินที่สร้างขึ้น ภายหลังจากการตรวจสอบไฟล์คำสั่งจนแน่ใจแล้ว ก็เริ่มทำการควบคุมระบบจริงๆ ซึ่งในส่วนนี้ก็จะทำการอ่านไฟล์คำสั่งไปทำการควบคุมตามลำดับของการเดินพร้อมทั้งบันทึกผลลัพธ์การควบคุมเพื่อนำมาวิเคราะห์ซึ่งประกอบด้วยไฟล์ชื่อ pos11,pos12,pos13,pos14,และ pos15.xls ซึ่งเรียงตามลำดับ

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

ประวัตผู้เขียน

นายวิทยา พันธุ์เจริญศิลป์ เกิดเมื่อวันที่ 27 พฤษภาคม พ.ศ. 2513 ที่กิ่งอำเภอนาบอน จังหวัด นครศรีธรรมราช สำเร็จการศึกษาระดับปริญญาตรี วิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมเครื่องกล จากมหาวิทยาลัยสยาม เมื่อปีการศึกษา 2535 และได้ เข้าศึกษาในระดับปริญญาโท สาขาวิศวกรรมมเครื่องกล คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อปีการศึกษา 2535 และได้ทำงานที่

-มหาวิทยาลัยศรีปทุม

ตำแหน่งอาจารย์ประจำภาควิชาวิศวกรรมเครื่ องกล คณะวิศวกรรมศาสตร์ตั้งแต่ปี พ.ศ. 2536