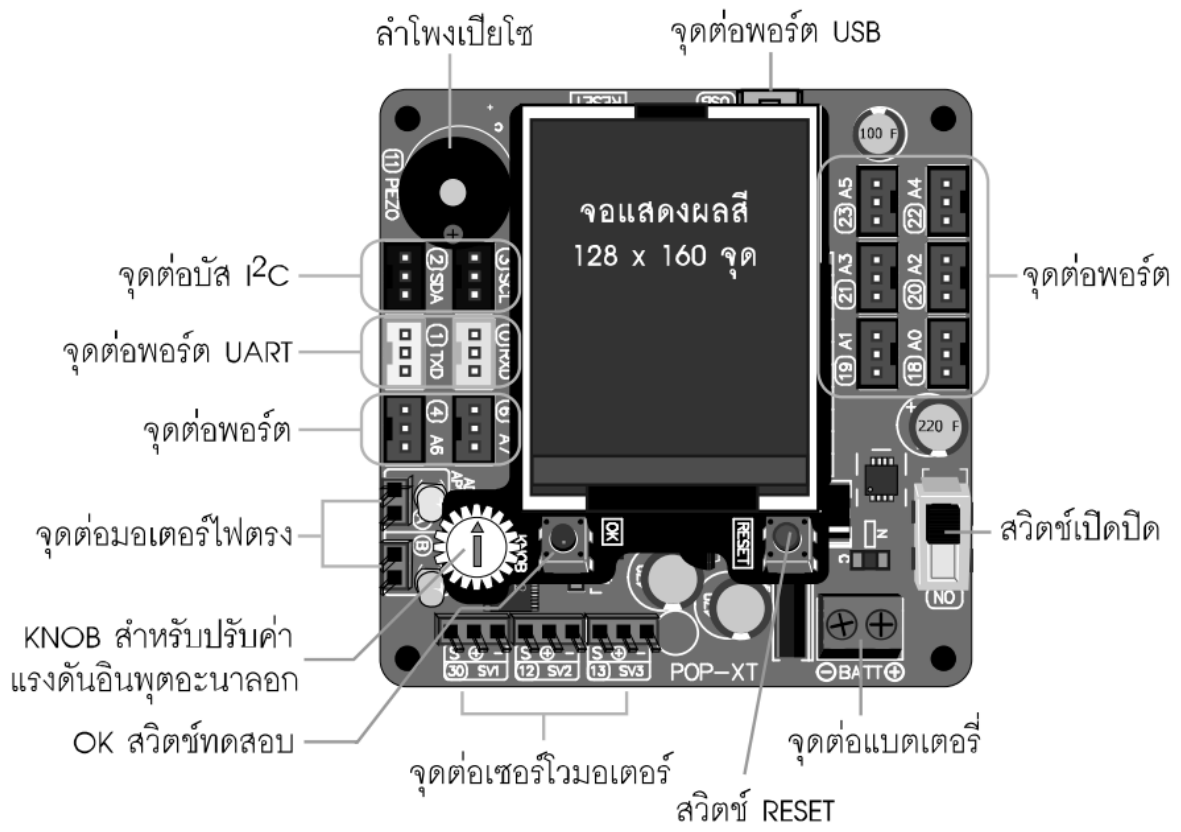
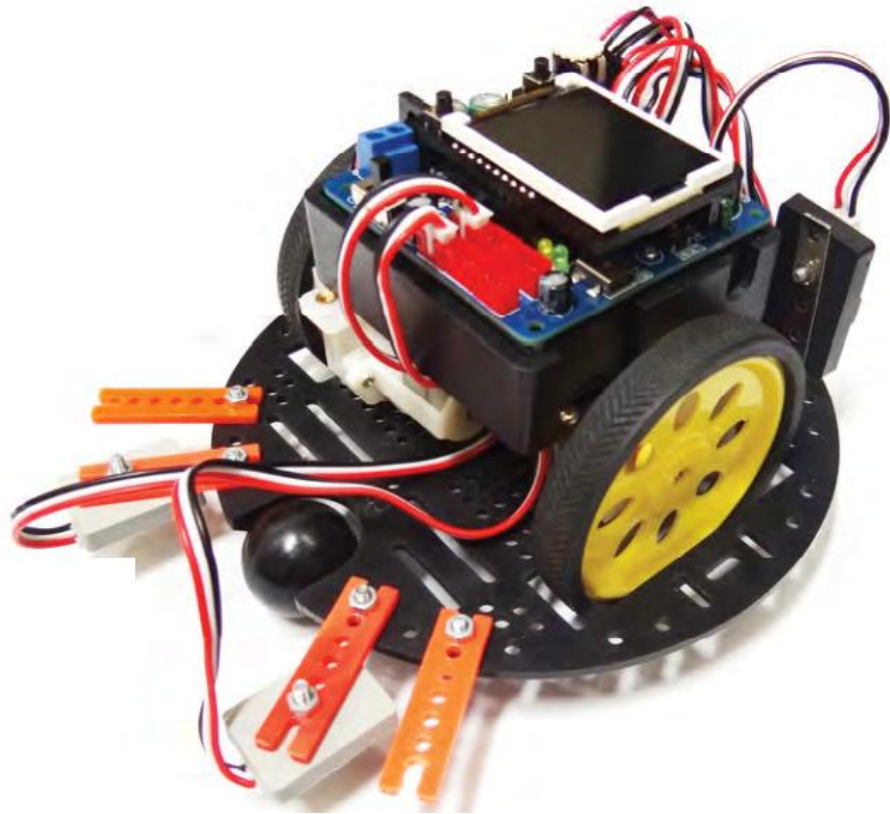
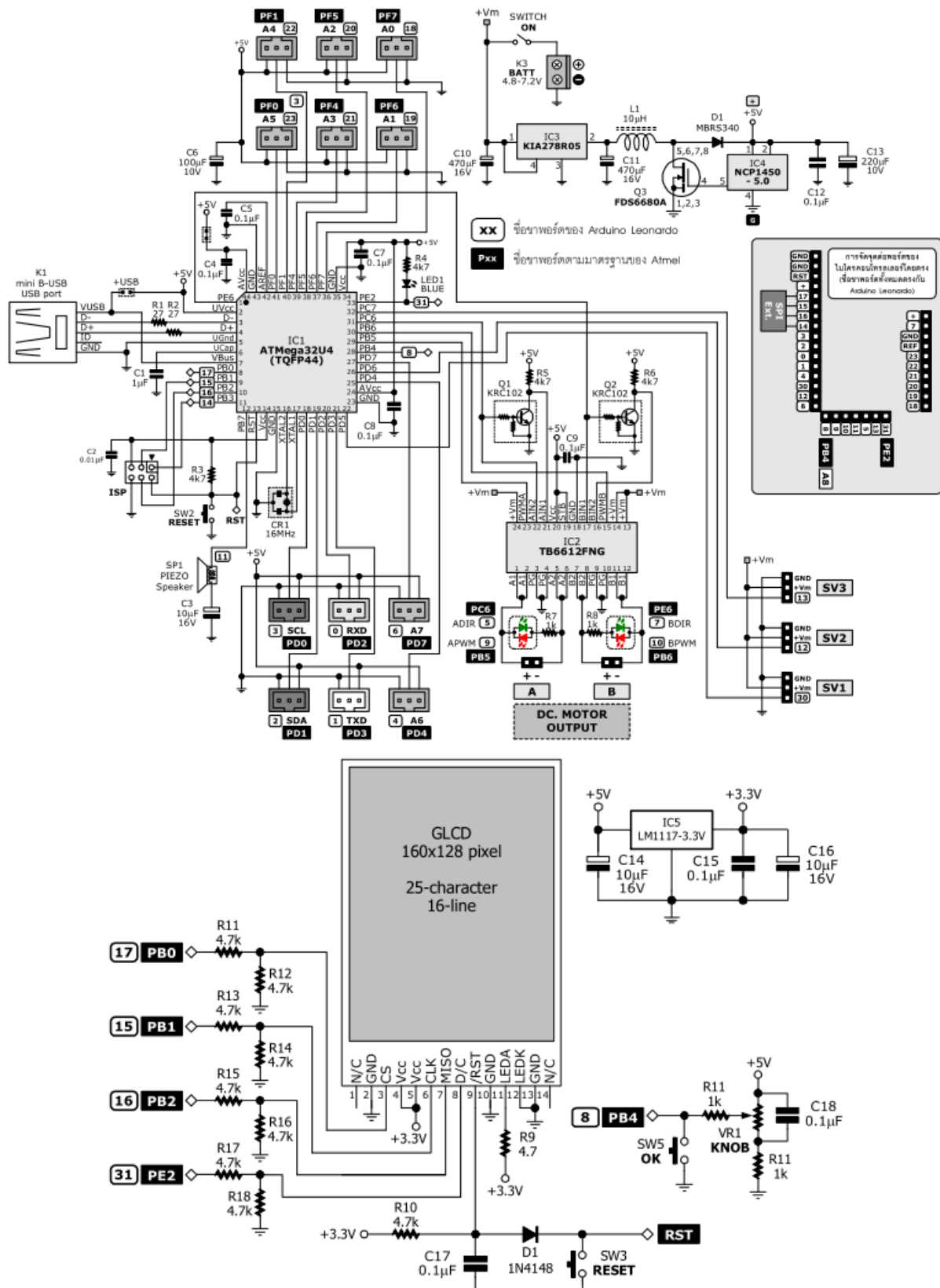


หุ่นยนต์ POP-BOT (XT)



วงจรหุ่นยนต์ POP-BOT XT



โปรแกรมควบคุม

1. Hello world

```
#include <popxt.h>

void setup()
{
    glcd(1,0,"Hello World");
}

void loop()
{
}
```

2. แสดงข้อความหลายบรรทัด

```
#include <popxt.h>

int i,j;

void setup()
{
    glcdFillScreen(GLCD_WHITE);
    setTextColor(GLCD_BLACK);
    setTextBackgroundColor(GLCD_WHITE);
    for (i=0;i<16;i++)
    {
        glcd(i,0," Row %d ",i);
    }
}

void loop()
{}
```

3. กำหนดขนาดตัวอักษรและทิศทางการแสดงผล

```
#include <popxt.h>

int x,m;

void setup()
{
    setTextColor(GLCD_RED);
}
```

```

void loop()
{
  for (x=1;x<6;x++)
  {
    setTextSize(x);
    for(m=0;m<4;m++)
    {
      glcdClear();
      glcdMode(m);
      glcd(0,0,"dX",x);
      glcd(1,0,"M=%d",m);
      sleep(500);
    }
  }
}

```

4. แสดงกราฟฟีก

```

#include <popxt.h>
int i,j;
void setup()
{

}

void loop()
{
  glcdClear;
  sleep(300);
  for (i=0;i<160;i+=4)
  {
    glcdLine(0,0,128,i,GLCD_WHITE);
  }
  for (i=0;i<128;i+=4)
  {
    glcdLine(0,0,i,160,GLCD_RED);
  }
}

```

```

    sleep(2000);
    glcdRect(32,40,64,80,GLCD_BLUE);
    sleep(300);
    glcdFillCircle(32,40,31,GLCD_GREEN);
    glcdFillCircle(96,40,31,GLCD_YELLOW);
    glcdFillCircle(32,120,31,GLCD_MAGENTA);
    glcdFillCircle(96,120,31,GLCD_SKY);
    sleep(1000);
    glcdCircle(64,40,31,GLCD_GREEN);
    glcdCircle(32,80,31,GLCD_BLUE);
    glcdCircle(64,120,31,GLCD_YELLOW);
    glcdCircle(96,80,31,GLCD_SKY);
    sleep(1000);
    glcdFillRect(0,0,128,160,GLCD_YELLOW);
    sleep(1000);
}

```

5. เส้นโค้ง

```

#include <popxt.h>
int i;
void face()
{
    glcdFillCircle(64,70,50,GLCD_WHITE);
    glcdArc(48,60,16,30,150,GLCD_RED);
    glcdCircle(48,55,5,GLCD_BLUE);
    glcdCircle(80,55,5,GLCD_BLUE);
    glcdArc(80,60,16,30,150,GLCD_RED);
    glcdFillCircle(64,70,7,GLCD_YELLOW);
    glcdArc(64,80,30,220,320,GLCD_RED);
    glcdArc(64,80,29,220,320,GLCD_RED);
}
void setup()
{}
void loop()

```

```
{
  for(i=0;i<4;i++)
  {
    glcdClear();
    glcdMode(i);
    face();
    sleep(1000);
  }
}
```

6. กำเนิดเสียง 1

```
#include <popxt.h>
void setup()
{
}
void loop()
{
  beep();
  sleep(1000);
}
```

7. กำเนิดเสียง 2

```
#include <popxt.h>
void setup()
{
}
void loop()
{
  sound(500,500);
  sound(2500,500);
}
```

8. อ่านค่าจากปุ่ม KNOB และสวิตช์ OK

```
#include <popxt.h>
void setup()
{
  glcdClear;
  setTextSize(2);
}
```

```

    glcd(1,1,"Press OK");
    sw_ok_press();
    beep();
    glcdClear;
}
void loop()
{
    if (sw_ok())
    {
        sound(500,500);
    }
    glcd(1,0,"Knob value");
    setTextSize(3);
    glcd(2,2,"%d ",knob());
    setTextSize(2);
}

```

9. LED ติดดับ

```

#include <popxt.h>
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    sw_ok_press();
}
void loop()
{
    out(4,1);
    out(6,0);
    sleep(400);
    out(4,0);
    out(6,1);
    sleep(400);
}

```

10. พื้น GLCD เป็นสีแดง

```
#include <popxt.h>

int colors;

void setup()
{
    int colors;
    colors=colorRGB(31,0,0);
    glcdFillScreen(colors);
}

void loop()
{
}
```

11. แสดงตัวหนังสือแนวตั้งแนวนอน

```
#include <popxt.h>

void setup()
{
    setTextSize(2);
}

void loop()
{
    glcdClear();
    glcdMode(0);
    glcd(0,0,"POP-BOTXT");
    sw_ok_press();
    glcdClear();
    glcdMode(1);
    glcd(0,0,"POP-BOTXT");
    sw_ok_press();
    glcdClear();
    glcdMode(2);
    glcd(0,0,"POP-BOTXT");
    sw_ok_press();
    glcdClear();
    glcdMode(3);
}
```

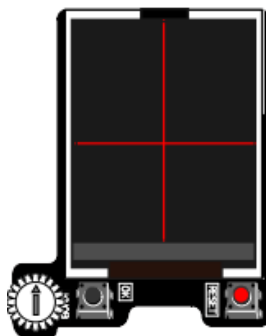


```
    glcd(0,0,"POP-BOTXT");
    sw_ok_press();
}
```

12. กำหนดขนาดตัวอักษร

```
#include <popxt.h>
void setup()
{
    setTextSize(1);
    setTextColor(GLCD_GREEN);
    glcd(0,0,"Size1");
    setTextSize(2);
    glcd(1,0,"Size2");
    setTextSize(3);
    glcd(2,0,"Size3");
    setTextSize(4);
    glcd(3,0,"Size4");
}
void loop()
{}
```

13. สร้างเส้น



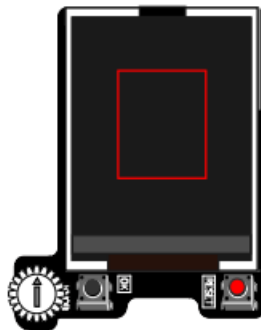
```
#include <popxt.h>
int i;
void setup()
{
    for (i=0;i<128;i+=4)
    {
```

```

        lcdPixel(i,80,GLCD_RED);
    }
    for (i=0;i<160;i+=4)
    {
        lcdPixel(64,i,GLCD_RED);
    }
}
void loop()
{}

```

14. สร้างสี่เหลี่ยม

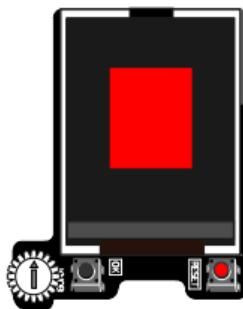


```

#include <popxt.h>
void setup()
{
    lcdRect(32,40,64,80,GLCD_RED);
}
void loop()
{}

```

15. สี่เหลี่ยมสีพื้นแดง



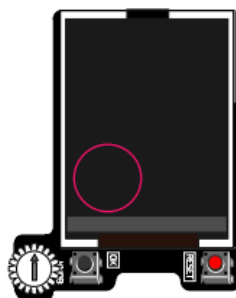
```
#include <popxt.h>
void setup()
{
    glcdFillRect(32,40,64,80,GLCD_RED);
}
void loop()
{
}
```

16. เส้นทแยง



```
#include <popxt.h>
void setup()
{
    glcdLine(0,0,127,159,GLCD_RED);
}
void loop()
{
}
```

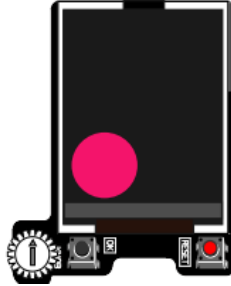
17. สร้างวงกลม



```
#include <popxt.h>
void setup()
{
    glcdCircle(32,120,31,GLCD_MAGENTA);
}
}
```

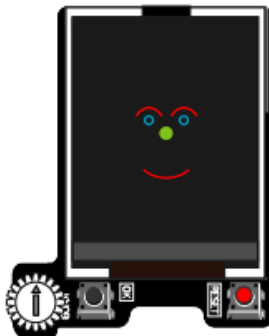
```
void loop()
{
```

18. สร้างวงกลมพื้นชมพู



```
#include <popxt.h>
void setup()
{
    glcdFillCircle(32,120,31,GLCD_MAGENTA);
}
void loop()
{
```

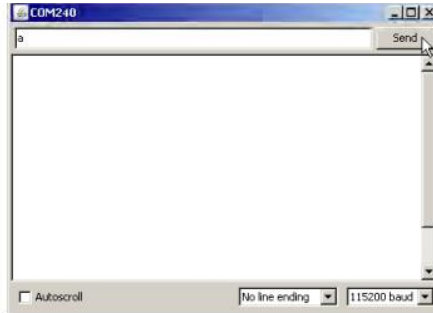
19. สร้างเส้นต่าง ๆ



```
#include <popxt.h>
void setup()
{
    glcdArc(48,80,16,30,150,GLCD_RED);
    glcdCircle(48,75,5,GLCD_BLUE);
    glcdCircle(80,75,5,GLCD_BLUE);
    glcdArc(80,80,16,30,150,GLCD_RED);
    glcdFillCircle(64,90,7,GLCD_GREEN);
    glcdArc(64,100,30,220,320,GLCD_RED);
}
```

```
void loop()
{
```

20. การสื่อสารผ่าน uart



```
#include <popxt.h>
void setup()
{
    glcdClear();
    setTextSize(2);
    glcdMode(1);
}
void loop()
{
    if(uart_available())
    {
        if(uart_getkey()=='a')
        {
            glcd(1,0,"Key a Active!");
            sleep(1000);
        }
        else
        {
            glcdClear;
        }
    }
}
```

21. หุ่นยนต์เดินหน้า ถอยหลัง

```
#include <popxt.h>
void setup()
{
}
void loop()
{
    fd(80);
    sleep(1000);
    bk(80);
    sleep(1000);
}
```

22. หุ่นยนต์เคลื่อนที่เป็นวงกลม

```
#include <popxt.h>
void setup()
{
    fd2(30,90);
    sw_ok_press();
    ao();
}
void loop()
{
}
```

23. หุ่นยนต์เคลื่อนที่เป็นสี่เหลี่ยม

```
#include <popxt.h>
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    glcd(2,1,"to Start");
    sw_ok_press();
    glcdClear();
    glcd(1,1,"Moving...");
}
```

```

void loop()
{
    fd(80);
    sleep(900);
    tr(80);
    sleep(400);
}

```

24. หุ่นยนต์เคลื่อนที่หลายทิศทาง

```

#include <popxt.h>
void setup()
{
    glcdClear();
    setTextSize(2);
    glcd(1,1,"Press OK");
    glcd(2,1,"to Start");
    sw_ok_press();
    beep();
    glcdClear();
}
void loop()
{
    glcdFillScreen(GLCD_RED);
    fd(80);
    sleep(600);
    glcdFillScreen(GLCD_GREEN);
    tl(60);
    sleep(400);
    glcdFillScreen(GLCD_BLUE);
    bk(80);
    sleep(600);
    glcdFillScreen(GLCD_YELLOW);
    sr(60);
    sleep(300);
}

```

```

    glcdFillScreen(GLCD_MAGENTA);
    bk(80);
    sleep(600);
    glcdFillScreen(GLCD_WHITE);
    sl(60);
    sleep(300);
}

```

25. การปรับความเร็วของมอเตอร์ด้านซ้าย-ขวา

```

#include <popxt.h>
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    glcd(2,1,"to Start");
    sw_ok_press();
    glcdClear();
    glcd(1,1,"Moving...");
    fd2(70,70);
}
void loop()
{}

```

26. สวิตช์ตรวจสอบการชน

```

#include <popxt.h>
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    glcd(2,1,"to Start");
    sw_ok_press();
    glcdClear();
    glcd(1,1,"Let's go!");
}

```



```

void loop()
{
  if(in(22)==0)
  {
    sound(1000,300);
    glcdFillScreen(GLCD_RED);
  }
  if(in(23)==0)
  {
    sound(2000,300);
    glcdFillScrrren(GLCD_YELLOW);
  }
}

```

27. สวิตช์เป็นรีโมทควบคุมการเคลื่อนที่ของหุ่นยนต์

```

#include <popxt.h>
void setup()
{
  setTextSize(2);
  glcd(1,1,"Press OK");
  glcd(2,1,"to Start");
  sw_ok_press();
  glcdClear();
  glcd(1,1,"Let's go!");
}
void loop()
{
  if(in(22)==0&&in(23)==0)
  {
    fd(60);
  }
  else if(in(22)==0)
  {
    tl(60);
  }
}

```

```

    }
else if(in(23)==0)
    {
        tr(60);
    }
else
    {
        ao();
    }
}

```

28. หลบหลีกวัตถุเมื่อถูกชน

```

#include <popxt.h>
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    glcd(2,1,"to Start");
    sw_ok_press();
    glcdClear();
    glcd(1,1,"Let's go!");
}
void loop()
{
    fd(70);
    if(in(22)==0)
    {
        bk(80);
        sleep(300);
        sr(80);
        sleep(200);
    }
    if(in(23)==0)
    {

```

```

        bk(80);
        sleep(400);
        sl(80);
        sleep(400);
    }
}

```

29. แสดงค่าตัวเลขจากเส้นขาวดำ

```

#include <popxt.h>
void setup()
{
    setTextSize(3);
    glcdMode(3);
    glcd(1,1,"Press OK");
    sw_ok_press();
}
void loop()
{
    glcd(1,0,"L=%d ",analog(0));
    glcd(3,0,"R=%d ",analog(1));
    sleep(100);
}

```

30. หุ่นยนต์ร้องและแสดงที่จอเป็นสีแดง เมื่อถูกยกจากพื้น

```

#include <popxt.h>
void setup()
{}
void loop()
{
    if(analog(0)<20)
    {
        sound(2500,500);
        glcdFillScreen(GLCD_RED);
    }
}

```

```

else
{
    glcdClear();
}
}

```

31. หุ่นยนต์เคลื่อนที่ไม่ตกโต๊ะ

```

#include <popxt.h>
int L,R;
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    glcd(2,1,"to Start");
    sw_ok_press();
    glcdClear();
    glcd(1,1,"Moving...");
}
void loop()
{
    fd(60);
    L=analog(0);
    R=analog(1);
    if (L<500&&R<500)
    {
        bk(80);
        sound(1000,100);
        sleep(500);
        sr(60);
        sleep(300);
    }
    else if(L<500)
    {
        bk(40);
        sound(1500,100);
        sleep(400);
    }
}

```

```

        sr(60);
        sleep(300);
    }
    else if(R<500)
    {
        bk(80);
        sound(1500,100);
        sleep(300);
        sl(60);
        sleep(500);
    }
}

```

32. หุ่นยนต์หยุดเมื่อเจอเส้นสีดำ

```

#include <popxt.h>
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    sw_ok_press();
    glcd(1,1,"Moving...");
    fd(60);
    while(analog(0)>500);
    ao();
}
void loop()
{}

```

33. หุ่นยนต์หยุด(ถอยหลังเล็กน้อย)เมื่อเจอเส้นสีดำ

```

#include <popxt.h>
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    sw_ok_press();
    glcd(1,1,"Moving...");
}

```

```

        fd(60);
        while(analog(0)>500);
        bk(100);
        sleep(100);
        ao();
    }
void loop()
    {}

```

34. หุ่นยนต์เดินตามเส้น โดยใช้ตัวเซนเซอร์ 2 ตัว

```

#include <popxt.h>
int L,R;
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    sw_ok_press();
    glcd(1,1,"Moving...");
}
void loop()
{
    L=analog(0);
    R=analog(1);
    if (L<500&&R<500)
    {
        fd(60);
        sound(2000,200);
    }
    else if(L>500&&R>500)
    {
        fd(60);
    }
    else if(L<500)
    {
        sl(60); sleep(20);
    }
}

```

```

    }

    else if(R<500)
    {
        sr(60);
        sleep(20);
    }
}

```

35. หุ่นยนต์เดินตามเส้น เมื่อเจอสีแยกแล้วเลี้ยวขวา โดยใช้ตัวเซนเซอร์ 2 ตัว

```

#include <popxt.h>
int L,R;
void R90()
{
    fd(60);
    sound(1500,100);
    while(analog(1)>500)
    {
        sr(60);
    }
    while(analog(1)<500)
    {
        sr(60);
    }
}
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    sw_ok_press();
    glcd(1,1,"Moving...");
}
void loop()
{
    L=analog(0);
    R=analog(1);
    if (L<500&&R<500)

```

```

    {
        R90();
    }
else if(L>500&&R>500)
    {
        fd(60);
    }
else if(L<500)
    {
        sl(60);
        sleep(20);
    }
else if(R<500)
    {
        sr(60); sleep(20);
    }
}

```

36. หุ่นยนต์เดินตามเส้น เมื่อเจอแยกให้เลี้ยวตามที่กำหนด โดยใช้ตัวเซนเซอร์ 2 ตัว

```

#include <popxt.h>
int L,R,x=0;
void R90()
{
    fd(60);
    sound(1500,100);
    while(analog(1)>500)
    {
        sr(60);
    }
    while(analog(1)<500)
    {
        sr(60);
    }
}
void L90()

```



```

{
  fd(60);
  sound(1500,100);
  while(analog(0)>500)
  {
    sl(60);
  }
  while(analog(0)<500)
  {
    sl(60);
  }
}

void setup()
{
  setTextSize(2);
  glcd(1,1,"Press OK");
  sw_ok_press();
  glcd(1,1,"Moving...");
}

void loop()
{
  L=analog(0);
  R=analog(1);
  if (L<500&&R<500)
  {
    if(x==6)
    {
      x=0;
    }
    x++;
    if (x==3)
    {
      R90();
    }
  }
  else

```

```

        {
            L90();
        }
    }

    else if(L>500&&R>500)
    {
        fd(60);
    }
    else if(L<500)
    {
        sl(60);
        sleep(20);
    }
    else if(R<500)
    {
        sr(60);
        sleep(20);
    }
}

```

37. การอ่านค่าจากโมดูลวัดระยะทาง GP2D120

```

#include <popxt.h>
#include <gp2d120_lib.h>
int raw,dist;
void setup()
{
    setTextSize(2);
    glcdMode(1);
    glcd(1,1,"Press OK");
    sw_ok_press();
    glcdClear();
}
void loop()

```

```

{
  raw = analog(20);
  dist = getdist(20);
  glcd(2,1,"RAW Data = %d ",raw);
  glcd(3,1,"Distance = %d cm",dist);
  sleep(100);
}

```

38. หุ่นยนต์หลบหลีกวัตถุแบบไม่สัมผัส

```

#include <popxt.h>
void setup()
{
  setTextSize(2);
  glcd(1,1,"Press OK");
  glcd(2,1,"to Start");
  sw_ok_press();
  glcdClear();
  glcd(1,1,"Let's go!");
}
void loop()
{
  fd(70);
  if(analog(20)>250)
  {
    bk(80);
    glcdFillScreen(color[random(8)]);
    sleep(300);
    sl(70);
    sleep(600);
  }
}

```

39. การใช้งานไมโครคอนโทรลเลอร์ ATmega32U4 กับเซอร์โวมอเตอร์

```
#include <popxt.h>
int x;
void setup()
{
  glcdMode(1);
  setTextSize(2);
  glcd(1,1,"Press OK");
  sw_ok_press();
  glcdClear();
}
void loop()
{
  x=map(knob(),80,1023,0,180);
  glcd(2,1,"Servo = %d ",x);
  servo(1,x);
}
```

40. หุ่นยนต์เมื่อพบวัตถุ ให้ตีวัตถุ

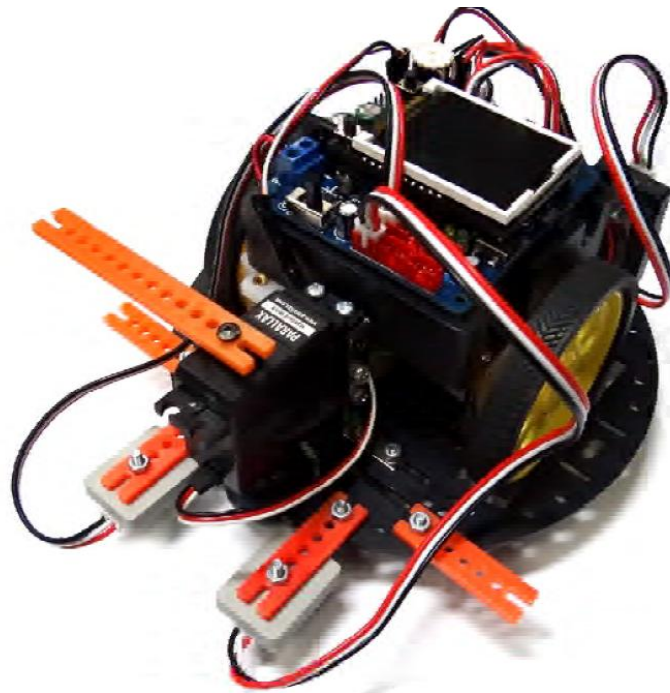
```
#include <popxt.h>
int L,R,x=0;
void setup()
{
  setTextSize(2);
  glcd(1,1,"Press OK");
  glcd(2,1,"to Start");
  sw_ok_press();
  glcdClear();
  glcd(1,1,"Let's go!");
  servo(1,0);
  sleep(1000);
  fd(60);
}
void loop()
```

```

{
  if(analog(0)<500)
  {
    ao();
    servo(1,180);
    sleep(1000);
    servo(1,0);
    while(1);
  }
}

```

41. หุ่นยนต์เดินตามเส้นที่มีทางแยก เคลื่อนที่ไปตามกำหนด เมื่อพบวัตถุ ให้ตัววัตถุแล้วเคลื่อนที่ไปตามกำหนด



```

#include <popxt.h>
int L,R,x=0,y=0;
void FF()
{
  fd(60);
  sleep(300);
}
void R90()
{

```

```

    fd(60);
    sound(1500,100);
    while(analog(1)>500)
    {
        sr(60);
    }
    while(analog(1)<500)
    {
        sr(60);
    }
}
void L90()
{
    fd(60);
    sound(1500,100);
    while(analog(0)>500)
    {
        sl(60);
    }
    while(analog(0)<500)
    {
        sl(60);
    }
}
void hit()
{
    bk(100);
    sleep(50);
    ao();
    servo(1,180);
    sleep(1000);
    servo(1,0);
    bk(60);
    sleep(200);
}

```

```

        R90();
    }
void setup()
{
    setTextSize(2);
    glcd(1,1,"Press OK");
    glcd(2,1,"to Start");
    sw_ok_press();
    glcdClear();
    glcd(1,1,"Let's go!");
}
void loop()
{
    L=analog(0);
    R=analog(1);
    if (L<500&&R<500)
    {
        if(x==5)
        {
            x=0;
            y++;
            if (y==4)
            {
                ao();
                servo (1,-1);
                while(1);
            }
        }
        x++;
        if (x==1)
        {
            L90();
        }
        else if(x==2||x==4)

```

```
    {
        hit();
    }

else if (x==3)
    {
        FF();
    }
else if (x==5)
    {
        R90();
    }
}
else if(L>500&&R>500)
    {
        fd(60);
    }
else if(L<500)
    {
        sl(60);
        sleep(20);
    }
    else if(R<500)
    {
        sr(60);
        sleep(20);
    }
}
```