# การใช้งาน กล่องสมองกล







STEL CON

Dann Greet

# หลักการของระบบควบคุม





### ์ตัวอย่างระบบควบคุมอย่างง่าย ๆ



## เครื่องปรับอากาศ





# ชุดอุปกรณ์ในชุด IPST-SE มาตรฐาน 2









### โปรแกรมที่ใช้งาน



#### โปรแกรม wiring

sketch_jul01a	Arduino 0107 🛛 🗕 🗆 🗙
File Edit Sketch Tools Help	
	₽ <sup>.</sup>
sketch_jul01a	
#include <ipst.h></ipst.h>	^
<pre>void setup() {     OK();// Wait for OK button</pre>	
}	
void <b>loop</b> () {	
}	
<	×
1	IPST-SE, ATMega644P @ 16 MHz on COM1



โปรแกรม Arduino

# ติดตั้งซอฟต์แวร์





Arduino1.0.7\_Setup150326 ประกอบด้วย

- ซอฟต์แวร์ Arduino 1.0.7
- ไลบรารี่ IPST-SE
- ตัวอย่าง IPST-SE
- ไดรเวอร์ USB



# ติดตั้งซอฟต์แวร์

Setup - Arduino 1.0.7 Build 1070 (driver signed) -	Setup - Arduino 1.0.7 Build 1070 (driver signed) -	📚 Setup - Arduino 1.0.7 Build 1070 (driver signed) - 🗉 💶	
Welcome to the Arduino 1.0.7 Build 1070 (driver signed) Setup Wizard	Select Destination Location Where should Arduno 1.0.7 Build 1070 (driver signed) be installed?	Select Start Mene Folder Where should Setup place the program's shortcuts?	
This will install Anduno 1.0.7 Build 1070 (driver signed) on your computer. Recommended to REMOVE/UNINSTALL previous version of Advance, or CHAVEE POLICER NAME's for anoid.	Setup will install Arduno 1.0.7 Build 1070 (driver signed) into the following folder. To continue, dick Next. If you would like to select a different folder, click Browse.	Setup will create the program's shortcuts in the following Start Menu folder. To continue, didk Next. If you would like to select a different folder, didk Browne.	
Component/Moraries version conflict problem Click Heast to continue, or Cancel to exit Setue.	At least 288.9 MB of free dais space is required.	3	
Text > Cancel	< Back / Hent > Cancel	<back next=""> Cancel</back>	

0	Setup - Arduino 1.0.7 Build 1070 (driver signed) -
R	ady to Install Setup is now ready to begin installing Arduino 1.0.7 Build 1070 (driver signed) on your computer.
	Click Install to continue with the installation, or click Back if you want to review or change any settings.
	Destination location:
	Start Henu folder Arduins
	< Back Install Cancel







# ขั้นตอนติดตั้งไดรเวอร์

#### เพื่อให้คอมพิวเตอร์รู้จักกับบอร์ด IPST-SE





# เปิดโปรแกรมครั้งแรก



#### เลือกบอร์ด IPST-SE

	Set default sket	×	
	set IPST-SE as defau	lt	
C ARD			ن ن زر
AN OPEN BY MASS: GIANLUC/ BASED OF	PROJECT WRITTEN, DEBUGGED AND SUPP IMO BANZI, DAVID CUARTIELLES, TOM I A MARTINO AND DAVID MELLIS N PROCESSING BY CASEY REAS AND BEN	ORTED COE, FRY	

โปรแกรม Arduino ถูกเปิด



### ตรวจสอบว่าเลือกบอร์ด IPST-SE แล้ว

•	sketch_jul01a   Arduino 010	7	
File Edit Sketch	Tools Help		
sketch_jul01a	Auto Format Ctrl+T Archive Sketch Fix Encoding & Reload		
#include < <mark>ips</mark>	Serial Monitor Ctrl+Shift+N	1	Unicon board (Caterina)
void <b>setup</b> () {	ATX-2 Block		POP-XT (Caterina) POP-168 or RXB-168 (POP-BOT)
<mark>OK();</mark> // Wai	Board		
}	Serial Port	Þ	IPST-PlusPlus, ATMega16 @ 16 MHz
void <b>loop</b> () {	Programmer Burn Bootloader	Þ	ATX2, ATMega644P @ 20 MHz Arduino Uno



### เชื่อมต่อกับคอมพิวเตอร์





# เลือกพอร์ตอนุกรม



#### เลือก Serial Port ให้ตรง

Tools Window Help						
	Auto Format	Ctrl+T				
	Archive Sketch					
	Fix Encoding & Reload					
	Serial Monitor	Ctrl+Shift+M	file for			
	Board	•				
	Serial Port	l	✓ COM113			
	Burn Bootloader	•				
	Burn Bootloader	,				



Universal Senal Bus, controllers

### รูปแบบการทำงานโปรแกรม Arduino

void setup() สำหรับกำหนดค่า เกิดขึ้นครั้งเดียว void loop() โปรแกรมหลักทำงานต่อเนื่อง





### โฟลวชาร์ตแสดงการทำงานโปรแกรม Arduino



## โปรแกรม ที่ 1

```
#include <ipst.h>
void setup()
{
  glcd(0,0,"Hello World");
}
void loop()
{
}
```



#### ตรวจสอบไวยกรณ์และอัพโหลด



### ผลลัพธ์ที่จอภาพ IPST-SE



# คุณสมบัติของจอภาพ GLCD



### คำสั่ง GLCD

glcd แสดงข้อความที่จอ GLCD ได้ 21 ตัว 16 บรรทัด (size 1) รูปแบบ

```
glcd(x,y,*p,...)
พารามิเตอร์
x คือตำแหน่งบรรทัดมีค่าตั้งแต่ 0-15
y คือตำแหน่งตัวอักษรมีค่าตั้งแต่ 0-24
*p คือข้อความที่ต้องการนำมาแสดง
ค่าพิเศษ
```

%d แสดงตัวเลขจำนวนเต็มในช่วง -32,768 ถึง 32,767 %h แสดงตัวเลขฐานสิบหก %b แสดงตัวเลขฐานสอง %l แสดงตัวเลขจำนวนเต็มในช่วง -2,147,483,648 ถึง 2,147,483,647 %f แสดงผลตัวเลขจำนวนจริง (แสดงทศนิยม 3 หลัก)



# คำสั่งอื่น ๆ ของ GLCD

#### glcd

setTextColor setTextBackgroundColor glcdClear glcdFillScreen glcdMode setTextSize

glcdPixel glcdRect glcdFillRect glcdLine glcdCircle glcdFillCircle glcdArc



### <mark>ค่าสีตัวอักษ</mark>ร

#### ตัวอย่าง

#include <ipst.h>
void setup(){
 setTextColor(GLCD\_WHITE);
 glcd(0,0,"Hello");
 setTextColor(GLCD\_GREEN);
 glcd(1,0,"World");

void loop(){}

#### $\texttt{setTextColor}\left(\texttt{COLOR}\right)$



}

GLCD RED, GLCD GREEN,

INEX

GLCD BLUE, GLCD YELLOW, GLCD BLACK, GLCD WHITE, GLCD CYAN, GLCD MAGENTA **GLCD ORANGE** GLCD LIME GLCD VIOLET GLCD PINK GLCD DOLLAR GLCD SKY GLCD BROWN GLCD DARKGREEN GLCD NAVY GLCD GRAY GLCD DARKGRAY

#### ปรับขนาดตัวอักษร





# คำสั่งของ glcdMode (หมุนหน้าจอ)



#### ปกติเป็น Mode 0

### ตัวอย่าง:glcdMode(1);



### ค่าสีพื้นหลังตัวอักษร

#### ตัวอย่าง

#include <ipst.h>
void setup(){
setTextBackgroundColor(GLCD\_RED);
setTextColor(GLCD\_YELLOW);
glcd(0,0,"Hello World");
}
void loop(){}



#### $\verb+setTextBackgroundColor(COLOR)$

World

#### glcdClear()

เคลียร์หน้าจอ

#### glcdFillScreen (COLOR)

เทสีลงบนหน้าจอทั้งหน้า

#### ตัวอย่าง

```
#include <ipst.h>
void setup(){}
void loop(){
glcdClear();
delay(500);
glcdFillScreen(color[0]);
delay(500);
 glcdFillScreen(color[1]);
delay(500);
 glcdFillScreen(color[2]);
 delay(500);
```



glcdRect(x,y,width,height,color)
glcdFillRect(x,y,width,height,color)
glcdCircle(x,y,radius,color)
glcdFillCircle(x,y,radius,color)

#### glcdLine(x1,y1,x2,y2,color)

x	ตำแหน่งแนวนอน	width	ความกว้าง
У	ตำแหน่งแนวตั้ง	height	ความสูง
		redius	รัศมี
		color	ଝ



### สร้างวงกลมสีแดงอยู่กึ่งกลางจอภาพ>รัศมีเต็มจอพอดี



#### แบบทดสอบ 2

# ลากเส้น 4 เส้นโดยมีจุดตัดอยู่กลางจอภาพพอดี





#### แบบทดสอบ 3

# สร้างสี่เหลี่ยมซ้อนกันดังรูป





#### การแสดงผลค่าตัวเลข

### glcd(0,0,"%d",100);





## ตัวแปร (ที่ใช้งานบ่อย ๆ)

- byte 0-255 (unsigned char)
- word 0-65535 (unsigned int)
- boolean 0-1 True False
- int -32768>ถึง 32767
- char -128 ถึง 127
- float -3.4 x 10<sup>38</sup> ถึง 3.4 x 10<sup>38</sup>

หาข้อมูลเพิ่มเติมจาก reference



### หลอด LED เอาต์พูตดิจิตอลอย่างง่าย



out (ch, state); ส่งค่าสถานะ(state) 0 หรือ 1 ออกไปยังตำแหน่งขา (ch)ที่ระบุ

เช่น out (17,1);



#### ตัวอย่าง : ไฟกะพริบ

#include <ipst.h>
void setup() { }
void loop() {
 out(17,1);
 delay(100);
 out(17,0);
 delay(100);




# กำหนดเสียง : ลำโพงเปียโซ

• ใช้ลำโพงเปียโซ มีอิมพีแดนซ์ 32w

• มีค่าความถี่ย่าน 300Hz ถึง 3000 Hz





**ZX-SPEAKER** 

## การต่อลำโพงเปียโซ



#### **ZX-SPEAKER**



<u>ต่อช่อง 16</u>



# คำสั่งสร้างเสียง

**beep** : กำเนิดเสียงความถี่ 500 Hz นาน 100 มิลลิวินาที beep (ch) ;

**sound** : กำเนิดเสียงตามความถี่และช่วงเวลาที่กำหนด sound (ch, freq, time) ; freq กำหนดค่าความถี่เสียง time กำหนดช่วงเวลากำเนิดเสียง มิลลิวินาที



### ตัวอย่าง : สร้างเสียง 1

```
#include <ipst.h>
void setup() {
}
void loop() {
    beep(19);
    delay(1000);
```

สร้างสัญญาณเสียงติ๊ดทุก ๆ 1 วินาที (ความถี่เสียง 500 Hz ดังนาน 0.1 วินาที)



### ตัวอย่าง : สร้างเสียง 2

```
#include <ipst.h>
void setup() {
}
void loop() {
   sound(19,1200,500);
   delay(1000);
```

### สร้างสัญญาณเสียงความถี่ 1200 Hz ดังนาน 0.5 วินาที เว้นทุก ๆ 1 วินาที



}

# สวิตช์ OK

# กดเป็น True ไม่กดเป็น False





# ดำสั่ง sw\_OK()

```
sw_OK() ตรวจสอบสวิตช์ OK บน IPST-SE
1 (True) เมื่อกดสวิตช์
0 (False) เมื่อไม่กดสวิตช์
หมายเหตุ การกดสวิตช์ทำให้ค่าที่อ่านได้จาก Knob มีค่าเป็น 0
ตัวอย่าง
```

```
if (sw_OK () )
{
beep (19) ;
}
ผลลัพธ์>เมื่อกดสวิตช์มีเสียงออกลำโพง
```



#### วนรอกดสวิตช์ OK เมื่อปล่อยสวิตช์ จะกระโดดไปทำคำสั่งบรรทัดถัดไป

#### ตัวอย่าง

#### sw\_OK\_press(); beep(19);

ผลลัพธ์ : รอกดสวิตช์>เมื่อกดส่งเสียงออกลำโพง



ฟังก์ชั้น <mark>OK()</mark>

แสดงข้อความที่หน้าจอ
 รอจนกระทั่งกดสวิตช์ OK
 ทำงานคำสั่งถัดไป

ตัวอย่าง
#include <ipst.h>
void setup() {
 OK();
}
void loop() {



Rummina

.



## สวิตช์ SW1



สวิตซ์ SWI อ่านด่าด้วยฟังก์ชิ่น swl() แล: swl\_press()



# คำสั่ง <mark>sw1()</mark>

```
sw1() ตรวจสอบสวิตช์ SW1 บน IPST-SE
      1 (True) เมื่อกดสวิตช์
      0 (False) เมื่อไม่กดสวิตช์
ตัวอย่าง
      if(sw1())
             out(17,1);
```



#### วนรอกดสวิตช์ <mark>SW1</mark> เมื่อปล่อยสวิตช์ จะกระโดดไปทำคำสั่งบรรทัดถัดไป

#### ตัวอย่าง

### sw1\_press(); beep(19);

ผลลัพธ์ : รอกดสวิตช์ SW1 เมื่อกดส่งเสียงออกลำโพง



#### แบบทดสอบ 4

#### เขียนโปรแกรม ใช้สวิตช์ OK เปิด ใช้สวิตช์ SW1 ปิด



### แผงวงจรสวิตช์ : ZX-SWITCH01







อ่านค่าขาดิจิตอลจากพอร์ตใด ๆ ของบอร์ด IPST-SE

### in(ch);

<mark>ch</mark> คือขาพอร์ตที่ต้องการอ่านค่าอินพุต <mark>การคืนค่า</mark> คืนค่าสัญญาณดิจิตอลของตำแหน่งขาพอร์ตที่ อ่าน มีค่าเป็น 0 หรือ 1 เท่านั้น



### ทดสอบเขียนโปรแกรมกับ <mark>ZX-Switch01</mark>













# วงจรของโมดูล LED8



Inex

# การเชื่อมต่อโมดูล LED8 กับ IPST-SE





# ฟังก์ชั่น LED8()

#### ส่งข้อมูล 1 ไบต์ไปยังบอร์ด LED8

#### LED8 (pin,dat);

pin ขาพอร์ตที่ต้องการติดต่อ dat ข้อมูล 1 ไบต์ที่จะส่ง ค่า 0 LED ดับหมด ค่า 255 LED ติดทั้งหมด





# ตัวอย่าง ไฟวิ่งด้วยโมดูล LED8

```
#include <ipst.h>
byte x=1;
void setup() { }
void loop() {
  x=1;
  while (x<128) {
      LED8(20,x); x=x*2;
delay(200);
}
  while(x>1) {
    LED8 (20, x); x=x/2;
delay(200);
```





### หมุนเพื่อปรับค่า 80-1023 –



# ปุ่มปรับค่าอะนาลอก <mark>knob()</mark>

# คำสั่ง GLCD

glcd แสดงข้อความที่จอ GLCD ได้ 21 ตัว 16 บรรทัด (size 1) รูปแบบ

```
glcd(x,y,*p,...)
พารามิเตอร์
x คือตำแหน่งบรรทัดมีค่าตั้งแต่ 0-15
y คือตำแหน่งตัวอักษรมีค่าตั้งแต่ 0-24
*p คือข้อความที่ต้องการนำมาแสดง
ค่าพิเศษ
```

%d แสดงตัวเลขจำนวนเต็มในช่วง -32,768 ถึง 32,767 %h แสดงตัวเลขฐานสิบหก %b แสดงตัวเลขฐานสอง %l แสดงตัวเลขจำนวนเต็มในช่วง -2,147,483,648 ถึง 2,147,483,647 %f แสดงผลตัวเลขจำนวนจริง (แสดงทศนิยม 3 หลัก)



### การแสดงผลค่าตัวเลขด้วย glcd

# glcd(0,0,"%d",100);



#### ค่าพิเศษ

%d แสดงตัวเลขจำนวนเต็มในช่วง -32,768 ถึง 32,767 %h แสดงตัวเลขฐานสิบหก %b แสดงตัวเลขฐานสอง %l แสดงตัวเลขจำนวนเต็มในช่วง -2,147,483,648 ถึง 2,147,483,647 %f แสดงผลตัวเลขจำนวนจริง (แสดงทศนิยม 3 หลัก)



# คำสั่ง <mark>knob()</mark>

<mark>knob</mark> ใช้อ่านค่าตัวต้านทานปรับค่าได้บน IPST-SE ทำงานเหมือน คำสั่ง analog(7) ค่าอยู่ในช่วง 80-1023

#### ตัวอย่าง

int val=0; val=knob(); glcd("%d",val);



# คำสั่ง knob(x)

รูปแบบ

### knob(x); โดย x คือค่า Scale การคืนค่า ค่าที่อ่านได้จาก knob มีค่าระหว่าง 0 ไปจนถึงค่า x ตัวอย่าง

หน้าจอ glcd แสดงค่าตัวเลข 0-180



# คำสั่ง knob(x,y)

รูปแบบ
knob(x,y);
x คือค่า Scale ช่วงเริ่มต้น
y คือค่า Scale ช่วงท้าย
ตัวอย่าง
glcd(1,1,"%d ",knob(10,90));
หน้าจอจะแสดงค่า 10-90 ตามการหมุน knob()



# ตัวอย่าง bar graph เมื่อปรับ knob

```
#include <ipst.h> // include file for IPST-SE
int x;
void setup() {
  setTextSize(3);
}
void loop(){
   glcd(1,0,"%d ",knob(128));
   glcdFillRect(0,80,knob(128),10,GLCD RED);
   glcdFillRect(knob(128),80,128-knob(128),10,GLCD BLACK);
}
```



### แผงวงจร ตัวต้านทานปรับค่าได้



#### ตัวต้านทานปรับค่าได้แบบตัวนอน





ตัวด้านทานปรับค่าได้แบบตัวตั้ง



สัญลักษณ์

# แผงวงจร ตัวต้านทานปรับค่าได้



# ฟ้งก์ชั้น analog()

อ่านค่าอะนาลอกจากตำแหน่งพอร์ตที่ระบุ (A0-A6)

# analog(ch);

ch คือขาพอร์ตอะนาลอก (A0-A6) : ระบุเฉพาะตัวเลข

<mark>ผลลัพธ์์ :</mark> ค่า 0-1023 (10 บิต) จากตำแหน่งขาพอร์ตที่ต้องการ



### แผงวงจรตรวจจับแสง ZX-LDR







ใช้ตรวจจับแสงสว่าง เลือกเอาต์พุตได้ 2 แบบคือ แรงดันเอาต์พุตเพิ่ม เมื่อแสงตกกระทบมากขึ้น แรงดันเอาต์พุตลดลง เมื่อแสงตกกระทบมากขึ้น



## สวิตช์เปิดไฟกลางคืน



# <mark>ตัวอย่าง : ไฟฉายสั่งเปิด/ปิดไฟ</mark>

```
#include <ipst.h>
int x, y=0;
void setup(){}
void loop(){
 x=analog(6);
 glcd(0,0,"LDR=%d ",x);
 if(x<200){
  while(analog(6)<200);delay(300);</pre>
   if(y==0){
     out(16,1); y=1;
 }
   else{
     out(16,0);y=0;
 }}
```



## ์ตัวอย่าง : โปรแกรมนับคนเข้าห้องสมุด

```
#include <ipst.h>
int x, y=0;
void setup() { setTextSize(3) ; }
void loop() {
 x=analog(6);
 glcd(0,0,"LDR=%d ",x);
 if(x<200){
  while(analog(6)<200);delay(300);</pre>
  y++;
  qlcd(2,0,"%d ",y);
}}
```



# ไอซีวัดอุณหภูมิ MCP9701

แรงดันเอาต์พุตเปลี่ยนแปลง 19.5mV/องศา คำนวณจากค่าอะนาลอกที่อ่านได้ จากสูตร Temp = (val x 0.25) - 20.51 val ค่าอะนาลอกที่อ่านได้จาก IPST-SE




# คำสั่ง <mark>sw\_ok\_press()</mark>



# ไอซีวัดอุณหภูมิ MCP9701

```
#include <ipst.h>
int val, i;
void setup(){ glcdClear(); setTextSize(2); }
void loop()(
  glcd(1,2,"Digital");
  glcd(2,2,"THERMO");
  glcd(3,3,"METER");
  val=0;
  for (i=0;i<20;i++) {val = val+analog(3); }</pre>
  val = val/20;
  Temp = (float(val)*0.25) - 20.51 ;
  setTextSize(3);
  setTextColor (GLCD YELLOW);
  glcd(3,1,"%f",Temp);
  setTextColor (GLCD WHITE);
  setTextSize(2);
  glcd(6,2,"Celsius");
  delay(500);
```



74

Digital

THERMO

METER

Celsius

# การสื่อสารอนุกรม





UART1



75

# คำสั่งสำหรับสื่อสารอนุกรม

uart\_available()
uart\_getkey()
uart
uart
uart putc

ถ้ามีข้อมูลถูกป้อนเข้ามาเงื่อนไขเป็นจริง รับค่าข้อมูล 1 ไบต์ ส่งข้อมูลหลาย ๆ ไบต์ออกไป ส่งข้อมูลออกไปตัวเดียว

uart1\_available()
uart1\_getkey()
uart1
uart1\_putc

ถ้ามีข้อมูลถูกป้อนเข้ามาเงื่อนไขเป็นจริง รับค่าข้อมูล 1 ไบต์ ส่งข้อมูลหลาย ๆ ไบต์ออกไป ส่งข้อมูลออกไปตัวเดียว



### ตัวอย่าง : รับค่าจากคอมพิวเตอร์ 🔶 LED

```
#include <ipst.h>
byte x;
void setup() { setTextSize(4); }
void loop() {
  if (uart available()) {
    x=uart getkey();
    glcd(1,1,"%h ",x);
    if (x=='a') {out(17,1); }
    else if (x=='b') {out (17,0); }
```



### การเปิดหน้าต่าง Serial monitor





# สื่อสารระหว่างไมโครคอนโทรลเลอร์ 2 บอร์ด





### ตัวอย่างการรับ และ ส่งข้อมูล

### ภาครับ

```
#include <ipst.h>
int x;
void setup() {
```

```
}
void loop() {
    if (uart1_available()) {
        x=uart1_getkey();
        glcd(1,1,"%d ",x);
    }
}
```

### ภาคส่ง

}

#include <ipst.h>
int x=0;
void setup(){}
void loop(){
 uart1\_putc(x);
 x++;
 delay(300);



# 



motor (CH, POW)

CH 1-2>คือมอเตอร์ 1 หรือ 2
 ALL>คือทั้งมอเตอร์ 1>และ 2
 POW ความเร็ว -100>ถึง 100
 ค่าบวก>เดินหน้า
 ค่าลบ>ถอยหลัง

motor (1,80) motor (2,-50)



# คำสั่ง motor\_stop() : หยุดมอเตอร์

motor\_stop หยุดขับมอเตอร์ตามช่องที่กำหนด รูปแบบ

motor\_stop (ch)
 ch คือช่อง 1 หรือ 2 และ ALL กรณีหยุดพร้อมกัน
 motor\_stop (ALL) = ao ()
 กรณีหยุดมอเตอร์ 2 ตัวพร้อมกัน ใช้คำสั่ง ao() แทนได้

motor\_stop(1) ; มอเตอร์ 1 หยุด motor\_stop(2) ; มอเตอร์ 2 หยุด ao() ; มอเตอร์ 2 ตัวหยุดพร้อมกัน



# การขับเคลื่อนหุ่นยนต์เบื้องต้น



#### คำสั่งเดินหน้า

motor(1,Speed);
motor(2,Speed);

#### คำสั่งถอยหลัง motor (1 , -Speed) ;

motor(2,-Speed);



# คำสั่งขับเคลื่อนรูปแบบฟังก์ชั่น

fd(Speed); เดินหน้า bk (Speed); กอยหลัง sl (Speed); ເລີ້ຍวซ้าย sr (Speed); ເລີ້ຍวขวา tl (Speed); เลี้ยวซ้ายล้อเดียว tr (Speed); เลี้ยวขวาล้อเดียว ao() ; หยุด

Speed คือความเร็วหุ่นยนต์ 0-100



# ฟังก์ชั้นขับเคลื่อนหุ่นยนต์





# การปรับหุ่นยนต์ให้เคลื่อนที่ตรง

fd2 (Speed1, speed2) ; เดินหน้า bk2 (Speed1, speed2) ; ถอยหลัง Speed1 คือมอเตอร์ 1 และ 2 Speed2 คือมอเตอร์ 3 และ 4





# ใช้สวิตช์ควบคุมการเคลื่อนที่ iBOT



in(17) = ซ้าย in(16) = ขวา



# หุ่นยนต์ บังคับด้วยสวิตช์





```
#include <ipst.h>
void setup() {OK();}
void loop() {
    if(!in(16)&&!in(17)) {
        fd(40);
    }
}
```

}

}

}

else {

ao();

```
else if(!in(16)&& in(17)){
    sl(40);
```

```
else if(in(16)&&!in(17)){
    sr(40);
```



### แผงวงจรตรวจจับแสงสะท้อน ZX-03









# รูปแบบการติดตั้ง ZX-03 เข้าที่ด้านหน้าหุ่นยนต์











92



# การติดตั้ง ZX-03 ด้านใต้ของหุ่นยนต์



### การอ่านค่าสะท้อนแสงของพื้นผิว

```
#include <ipst.h>
void setup() {
    OK();
}
void loop() {
    glcd(1,1,"L=%d
    glcd(3,1,"R=%d
```



",analog(0)); ",analog(1));



}

### ผลลัพธ์การอ่านค่าการสะท้อน



Inex

ด่าที่อ่านได้เปลี่ยนแปลงตามความสูงเซนเซอร์จากพื้น

# หุ่นยนต์เคลื่อนที่หยุดที่เส้นดำ

#### #include <ipst.h> void setup() { **OK()**; **fd(**100**);** while (analog (0)>250) ao(); **glcd(**1,1,"Stop... ");

#### void loop(){}

#### มาจากค่ากลาง ระหว่างขาวกับดำ

#### (450+50)/2





# if(L>250&&R>250){ fd(60);

# เมื่อซ้ายและขวาเจอขาว เดินดรงไป





}

if (L<250&&R>250) { **sl(**60**); delay(20);** 

# เมื่อซ้ายเจอดำและขวาเจอขาว เลื่าวข้าย



if(L>250&&R<250){ **sr(**60**); delay(20);** 

# เมื่อซ้ายเจอขาวและขวาเจอดำ



99

3

if (L<250&&R<250) { **fd(**60**)**; **delay(200);** 

### เมื่อซ้ายเจอดำและขวาเจอดำ

### เดินตรงข้ามทางแยก





### ทดสอบสนามเคลื่อนที่ตามเส้นอย่างง่าย



### ตัวอย่างการทำสนามด้วยเทปพันสายไฟ



### โปรแกรมทดสอบ

```
#include <ipst.h>
int L,R;
void setup() {OK();}
void loop() {
  L=analog(0);
  R=analog(1);
  if(L>250&&R>250) { fd(100); }
  else if (L<250&&R>250) {sl(100); delay(20); }
  else if (L>250&&R<250) {sr(100); delay(20); }
}
```



### เจอเส้นตัดส่งเสียงติดออกลำโพง



# โปรแกรมเจอเส้นตัดส่งเสียงติ๊ดออกลำโพง

```
#include <ipst.h>
int L,R;
void setup(){OK();}
void loop() {
  L=analog(0);
  R=analog(1);
  if(L>250&&R>250){ fd(100);}
  else if (L<250&&R<250) {
  fd(100); sound(16,500,200);
}
  else if (L<250&&R>250) {sl(100); delay(20); }
  else if (L>250&&R<250) {sr(100); delay(20); }
}
```



### เจอเส้นตัดแล้วเลี้ยวขวา



### โปรแกรมเจอเส้นตัดแล้วเลี้ยวขวา

```
#include <ipst.h>
int L,R;
void setup() {OK();}
void loop() {
  L=analog(0);
  R=analog(1);
  if(L>250&&R>250){ fd(100);
  else if (L<250&&R<250) {
    fd(100); sound(16,500,200);
    sr(100);delay(400);
```

else if(L<250&&R>250) {sl(100);delay(20); }
else if(L>250&&R<250) {sr(100);delay(20); }</pre>



}

### การเลี้ยว 90 องศาพอดีเมื่อเจอเส้นตัด


### รูปแบบการสร้างฟังก์ชั่น





#### ฟังก์ชั้นเจอเส้นตัดเลี้ยวขวา/ซ้าย

void R90() {
 fd(60);
 sound(16,1500,100);
 while(analog(1)>250){sr(60);}
 while(analog(1)<250){sr(60);}</pre>



#### ปรับแก้ เจอเส้นตัดแล้วเลี้ยวขวา

```
#include <ipst.h>
int L,R;
void setup(){OK();}
void loop() {
  L=analog(0); R=analog(1);
  if(L>250&&R>250) { fd(100); }
  else if (L<250&&R<250) {R90(); }
  else if (L<250&&R>250) {sl(100); delay(20); }
  else if (L>250&&R<250) {sr(100); delay(20); }
}
void R90(){
  fd(60);
  sound (16,1500,100);
  while (analog(1)>250) {sr(60);}
  while (analog(1)<250) {sr(60);}
```



}

# ລຸບໂກຣຄົມສົອນ











### แบตเตอรี่ลิเธียม โพลิเมอร์ : Li-Po



แดง บวก 

#### 2 ເชລ 7.4V

- กระแส 1100mA
- จ่ายกระแส 30 เท่า
- ชาร์จ 5 เท่า







### คุณสมบัติของแบตเตอรี่ Li-Po

2 เซล 7.4V กระแส 1100mAh จ่ายกระแส 30 เท่า ชาร์จ 5 เท่า



1 เซล 3.7V อนุกรมกัน 2 เซล = 7.4V จ่ายไฟ 1100 mA ต่อเนื่องได้ประมาณ 1 ชั่วโมง จ่ายกระแสชั่วขณะได้ 1100 x 30 = 33000mA O\_o! ชาร์จได้ 5 เท่า 1100x5 = 5500mA ใช้เวลาประมาณ 20 นาที



#### ข้อดีของแบตเตอรี่ Li-Po



#### ข้อดีของแบตเตอรี่แบบ Li-Po เมื่อน้ำมาใช้กับหุ่นยนต์

มีน้ำหนักเบาในเมื่อเทียบกับความจุ (mAh)
 จ่ายกระแสได้มากกว่าความจุ ทำให้หุ่นยนต์มีความเร็วเพิ่มขึ้นชัดเจน
 แรงดันคงที่ หุ่นยนต์ทำงานนิ่งตลอด จนหมดความจุ
 มีหลายรูปแบบขนาด ทำให้ยึดติดตั้งได้ง่าย
 คายประจุด้วยตัวเอง(Self Discharge) น้อย
 ชาร์จเต็มเร็วมาก



#### ข้อเสียของแบตเตอรี่ Li-Po



#### ข้อเสียของแบตเตอรี่แบบ Li-Po เมื่อน้ำมาใช้กับหุ่นยนต์

- 1. มีราคาแพงเมื่อเทียบกับแบตเตอรี่ชนิดอื่น ๆ
- 2. ต้องใช้เครื่องชาร์จที่มีความเฉพาะ ซึ่งบางแบบก็มีราคาแพง
- ต้องดูแลเป็นพิเศษ ถ้าเกิดการลัดวงจรจะเกิดความเสียหายใหญ่หลวง
   ต้องคอยดูแลเรื่องความจุ ถ้าใกล้หมด จะเกิดการสูญเสียแรงดันและแบตเตอรี่เกิด ความเสียหาย ต้องมีการตรวจวัดความจุของแบตเตอรี่อยู่เสมอเมื่อใช้งาน
   เมื่อไม่ใช้งานนาน ๆ ต้องไม่ให้แบต มีความจุดเต็มค้างไว้ ไม่งั้นแบตจะบวม



#### วัดแบต Li-Po และเตือนป้องกันแบตเสื่อม

แสดงไฟและเดือน



#### วัดโวลต์และเดือน



117

### ้เครื่องชาร์จแบตอเนกประสงค์กระแสสูง







#### การเชื่อมต่อกับบอร์ด IPST-SE



#### แขนจับ SM-GRIPPER

Senter 1

dini ile

113

#### **SERVO2: SV1**

B

#### SERVO1 : SV0



COTTO:

#### 1. ถอดโครงหุ่นยนต์ IPST-SE ออก





2. ติดตั้งเสารอง 32 มม.





#### 3. ยึดบอร์ด IPST-SE เข้ากับ SM-GRIPPER





4. วาง SM-GRIPPER เข้ากับฐานหุ่นยนต์ ใช้สกรูขันยึด





5. น้ำแบตเตอรี่ Li-Po ติดตั้งเข้ากับ SM-GRIPPER





125

6. เสียบสาย SERVO เข้าที่ช่อง SV0 และ SV1



#### การใช้งานเซอร์โวมอเตอร์



```
servo(CH, POS);
CH ช่องที่ใช้ขับ = 1 ถึง 4
POS ตำแหน่งองศาเซอร์โว = 0-180 ,-1
     ด่า -1 หมายถึงหยุดจ่ายสัญญาณให้เซอร์โว
     เซอร์โวจะไม่ล็อกแกน
```



#### โปรแกรมทดสอบเซอร์โว 1

```
#include <ipst.h>
int x;
void setup() {
  OK();glcdClear();
}
void loop() {
  x=knob(180);
  servo(1, x);
  glcd(1,1,"%d ",x);
```

หมุน knob ทดสอบ







### โปรแกรมทดสอบเซอร์โว 2

```
#include <ipst.h>
int x;
void setup() {
  OK();glcdClear();
void loop() {
  x=knob(180);
  servo(2, x);
  glcd(1,1,"%d ",x);
```

#### หมุน knob ทดสอบ





## ฟังก์ชั้นให้เซอร์โวอยู่ในตำแหน่ง home

```
#include <ipst.h>
int x;
void setup() {
  OK();
  sHome();
}
void loop(){}
void sHome(){
  servo(0,43);servo(1,51);
  delay(1000);
  servo(0,-1);servo(1,-1);
```



#### ชุด Wireless Joy Stick Kit





## ติดตั้ง Joy PSX เข้ากับหุ่นยนต์ IPST-SE



DAT ต่อกับขา 30 CMD ต่อกับขา 29 CLK ต่อกับขา 27

SEL ต่อกับขา 28



133

#### เสียบตัวรับเข้ากับ ZX-PSX2





DAT ต่อกับขา 30 CMD ต่อกับขา 29 SEL ต่อกับขา 28 CLK ต่อกับขา 27

#### รูปแบบการกำหนดส่วนหัวโปรแกรม

#include <ipst.h>
#include <PS2X\_lib.h>
#define PS2\_DAT 30
#define PS2\_CMD 29
#define PS2\_SEL 28
#define PS2\_CLK 27
PS2X ps2x ;

พยายามเชื่อมต่อ เก็บค่าผลลัพธ์ ที่ตัวแปร error

int error = ps2x.config\_gamepad(PS2\_CLK, PS2\_CMD, PS2\_SEL, PS2\_DAT,0,0);

#### ถ้า error = 0 แสดงว่าเชื่อมต่อได้



## คำสั่งเรียกชื่อปุ่มต่าง ๆ

PSB SELECT PSB L3 PSB R3 PSB START PSB PAD UP PSB PAD RIGHT PSB PAD DOWN PSB PAD LEFT PSB L2 PSB R2 PSB L1 PSB R1 PSB TRIANGLE PSB CIRCLE PSB CROSS PSB SQUARE

0x0001 0x0002 0x0004 0x008 0x0010 0x0020 0x0040 0x0080 0x0100 0x0200 0x0400 0x0800 0x1000 0x2000 0x4000 0x8000

#### อ่านค่าการกด PSX แสดงผลออก GLCD

```
#include <ipst.h>
#include <PS2X lib.h>
#define PS2 DAT 30
#define PS2 CMD 29
#define PS2 SEL 28
#define PS2 CLK 27
PS2X ps2x ;
void setup() {
  delay(1000);
  setTextSize(2); glcd(0,0,"Connecting");
  while(true){
    int error = ps2x.config gamepad(PS2 CLK, PS2 CMD, PS2 SEL, PS2 DAT, 0, 0);
    if(error==0){
      glcd(0,0, "OK ");
      delay(1000); glcdClear(); break;
    } delay(500);
 }
}
void loop(){
  ps2x.read gamepad(0,0);
  if (ps2x.Button (PSB CIRCLE)) {glcd(1, 1, "Circle "); }
  else if(ps2x.Button(PSB CROSS)) {glcd(1, 1, "Cross ");}
  else if (ps2x.Button (PSB SQUARE)) {glcd(1, 1, "Square ");}
  else if (ps2x.Button (PSB TRIANGLE)) {glcd(1, 1, "Triangle");}
  else {glcd(1, 1, " "); }
  delay(50);
```



}

#### อ่านค่าจอยอะนาลอก แสดงผลออก GLCD

```
#include <ipst.h>
int LX, LY, RX, RY;
#include <PS2X lib.h>
#define PS2 DAT 30
#define PS2 CMD 29
#define PS2 SEL 28
#define PS2 CLK 27
PS2X ps2x ;
void setup() {
  delay(1000);
  setTextSize(2); glcd(0,0,"Connecting");
  while(true){
    int error = ps2x.config gamepad(PS2 CLK, PS2 CMD, PS2 SEL, PS2 DAT, 0, 0);
    if(error==0){
      glcd(0, 0, "OK ");
      delay(1000); glcdClear(); break;
 }
    delay(500);
 }
}
void loop(){
  ps2x.read gamepad(0,0);
  LX=ps2x.Analog(PSS LX); glcd(0,0,"LX=%d ",LX);
  RX=ps2x.Analog(PSS RX); glcd(1,0,"RX=%d ",RX);
  LY=ps2x.Analog(PSS LY); glcd(2,0,"LY=%d ",LY);
  RY=ps2x.Analog(PSS RY); glcd(3,0,"RY=%d ",RY);
}
```

## ควบคุมหุ่นยนต์ด้วยปุ่ม Analog แบบง่าย ๆ

```
#include <ipst.h>
int LX, LY, RX, RY;
#include <PS2X lib.h>
#define PS2 DAT 30
#define PS2 CMD 29
#define PS2 SEL 28
#define PS2 CLK 27
PS2X ps2x ;
void setup() {
  delay(1000);
  setTextSize(2); glcd(0,0,"Connecting");
  while(true){
    int error = ps2x.config gamepad(PS2 CLK, PS2 CMD, PS2 SEL, PS2 DAT,0,0);
    if(error==0){
      glcd(0, 0, "OK ");
      delay(1000); glcdClear(); break;
  }
    delay(500);
 }
}
void loop(){
  ps2x.read gamepad(0,0);
  motor(1,map(ps2x.Analog(PSS LY),0,255,100,-100));
  motor(2,map(ps2x.Analog(PSS RY),0,255,100,-100));
  delay(100);
}
```

139

## <u>ติดตั้งบอ</u>ร์ด IPST-SE บนโครง SUMO



## ติดตั้ง ZX-03 ตรวจจับการออกนอกสนามซูโม



141







## การพวงสายมอเตอร์ ด้วยขั้วต่อ ID<u>C 2 ขา</u>



#### การพวงมอเตอร์ ซ้ายและขวา

#### $A+B \longrightarrow motor1$ $C+D \longrightarrow motor2$





ซูโม่อย่างง่าย

#include <ipst.h> int CL=500, CR=500; void setup() {OK(); } void loop() { fd(60); if(analog(0)>CL) { bk(60);delay(300); sr(60);delay(800); if(analog(1)>CR) { bk(60);delay(500); sl(60);delay(900);


# ควบคุมด้วยจอย PSX





### โปรแกรม APP INVENTOR

### สร้างโปรแกรมควบคุม IPST-SE แบบไร้สายผ่าน Android ด้วยโมดูล BlueStick และซอฟต์แวร์ App Inventor 2





### เปิดใช้งาน App Inventor 2 (Online)



เข้าไปที่ http://appinventor.mit.edu/ จากนั้นกด Create

## ลงชื่อเข้าใช้งานด้วยบัญชี Google



### ลงชื่อเข้าใช้

#### Welcome to App Inventor!



Got an Android phone or tablet? Find out how to Set up and connect an Android device.

Don't have an Android device? Find out how to Set up and run the Android emulator.

(Emulator and USB connections are currently for Mac and Windows only. Support for Linux is *coming soon!*) (Support for Internet Explorer is *coming soon!*)



### แนะนำการตั้งค่า Android

กด Continue ไปต่อ



# สร้าง Project ใหม่

	Beta	Project *	Connect *	Build *	Help *	My Projects	Gui
Vew Project Dele	te Proiect						0111110
		teristi (fati sun teristi sun su					
Name	C	ate Created				Date Modified▼	

create new App	inventor project
Project name:	BlueStick
Cancel	OK



Privacy Policy and Terms of Use



## หน้าต่าง Designer

#### Designer เป็นที่สำหรับวางปุ่มหรือข้อความสำหรับติดต่อผู้ใช้



Privacy Policy and Terms of Use

# หน้าจอสำหรับการเขียนโค้ดที่เรียกว่า Blocks

151

MIT App Inventor 2 Beta	Project *	Connect *	Build *	Help *		My Projects	Guide	Report an Issue	inex.robot@gmai	.com *
BlueStick	Screen1 •	Add Screen	Remove Scre	en					Designer	Blocks
Blocks	Viewer	antening vin endt	Pok British Harden Stat							1
<ul> <li>Built-in</li> <li>Control</li> <li>Logic</li> <li>Math</li> <li>Text</li> <li>Lists</li> <li>Colors</li> <li>Variables</li> <li>Procedures</li> <li>Screen1</li> <li>Any component</li> </ul>						สำห	ารับก	ารเขียนโค้	ດແບບບລິຍ	'n
	▲ 0 Show T	▲ 0 Warnings				_			ĺ	
Kename Delete										
Media										
Upload File										
			Priva	cy Policy and	d Terms of Use	1				



# เชื่อมต่อ App Inventor กับแอนดรอยด์ 3 แบบ

แบบที่ 1 เชื่อมต่อด้วย WIFI (เราจะเลือกใช้วิธีนี้) แบบที่ 2 เชื่อมต่อผ่านซอฟต์แวร์อีมูเลเตอร์ แบบที่ 3 เชื่อมต่อตรงผ่านสาย USB





ดูผลการเปลี่ยนแปลงได้ทันที

ขียนโปรแกรมบนคอมพิวเตย



# 1.ขั้นตอนการเชื่อมต่อผ่าน WIFI

- 1. ดาวน์โหลดและติดตั้งโปรแกรม MIT Al2 Companion จาก Play Store บนแอนดรอยด์
- 2. ให้คอมพิวเตอร์และแอนดรอยด์ใช้ WIFI ชุดเดียวกัน



### App Inventor 2 ให้เลือกเชื่อมต่อแบบ AI Companion

2	MIT App Inventor 2 Beta	Project *	Connect * Bui	ld *	Help *		
Blu	JeStick	Screen1 •	Al Companion	s Scree	en		
Pa	lette	Viewer	USB				
U	ser Interface	Display I	Reset Connection	ewer			
	Button		Hard Reset		\$.d	9:48	
	CheckBox @	Screen1					
201	DatePicker						
	Image (?						
A	Label						
E	ListPicker (?						
H	ListPicker						
	Label (3						

Z

### จะมีหน้าต่างแสดง QRCODE และเลขรหัสดังรูป

ที่แอนดรอยด์ให้เปิดแอพ MIT Al2 Companion เลือกแสกน QRCODE หรือป้อนรหัสที่ปรากฏก็ได้

#### **Connect to Companion**

Launch the MIT Al2 Companion on your device and then scan the barcode or type in the code to connect for live testing of your app. Need help finding the Companion App?



Your code is:

sjgmlh

2

Cancel

**MIT App Inventor 2 Companion** 

#### MIT App Inventor 2

type in the 6-digit code -orscan the QR code

#### sjgmlh

#### connect with code





### การเปลี่ยนแปลงหน้าจอสัมพันธ์กับโปรแกรม

เมื่อกด Connect หน้าจอ แอนดรอยด์จะเหมือนกับหน้าจอ ออกแบบของ App Inventor Screen1

52

156





# การเขียนโปรแกรมเพื่อติดต่อกับโมดูลบลูทูธ

# ขั้นตอนออกแบบ

BlueStick		Screen1 •	Add Screen	Remove Screen
Palette		Viewer		
User Interface			[	Display hidden components in Viewer
Button	0			9:48 📓 🗐
CheckBox	0			Screen1
DatePicker	1		_	Text for ListPicker1
📔 Image	1			
A Label	0	/		ann ListPicker มาวาม
ListPicker	0	-		
ListView	٢			
A Notifier	1			
PasswordTextBox	0			
Slider	0			
10 Stider	۲	-		
PasswordTextBox				

#### ในส่วน Properties หัวข้อ Text เปลี่ยนชื่อเป็น Connect

Properties		Connect	
ListPicker1			
BackgroundColor	1		
Default	12	3	
ElementsFromString	11 mar		
Enabled			
	1		
Text			
Connect	1.5		
TextAlignment	1		
center 👻	(E =	3	
TextColor	14 C		
Default			
Varias Dranortica			
uidiij Properiles			



### ลากปุ่ม Button มาวางอีกสองปุ่ม เปลี่ยนชื่อเป็น ON และ OFF





### ไปที่หัวข้อ Connectivity เลือก BluetoothClient1 มาวาง



#### เลือกเปลี่ยนหน้าต่างไปที่หน้า Blocks



#### ลากบล็อกชุดแรกสำหรับเรียกรายการของอุปกรณ์บลูทูธมาแสดง



### ลากบล็อกชุดที่ 2 หลังจากเลือกรายการจาก List แล้ว ให้ Connect บลูทูธจากแอดเดรสที่เลือก

Procedures		
😑 📋 Screen1	when ListPicker1 AfterPicking	
E ListPicker1	do set ListPicker1 . Selection to t call BluetoothClient1 . Connect	
Button1	address (ListPicker1 ). Selection	1
Button2		
BluetoothClient1		
Any component		

### ชุดบล็อกชุดที่ 3 เมื่อกดปุ่ม ON ให้ส่งค่าเลข 1 ออกไปยังบลูทูธ





ชุดบล็อกชุดที่ 4 เมื่อกดปุ่ม OFF ให้ส่งค่าเลข 0 ออกไปยังบลูทูธ





### รวมโค้ดทั้งหมดก็จะเป็นดังนี้





# การต่อโมดูล BlueStick เข้ากับบอร์ด IPST-SE



### การเขียนโปรแกรมรับค่าจาก BlueStick

บอร์ด มีฟังก์ชั่นสื่อสารอนุกรม ใช้ชื่อว่า UART

**uart1 ()** สำหรับส่งข้อความออกไป

# uart1\_available()

สำหรับดูผลว่ามีข้อมูลถูกส่งเข้ามาหรือไม่

uart1\_getkey() สำหรับอ่านค่าข้อมูลที่ถูกส่งมา



### ตัวอย่างโปรแกรม รับค่าและแสดงผลที่ GLCD

```
#include <ipst.h>
byte x;
void setup() {
  setTextSize(2);
  glcd(0,0,"You Press:");
  setTextSize(3);
}
void loop() {
  if(uart1 available()) {
    x=uart1 getkey();
    if(x==1) {qlcd(1,0,"ON "); }
    else if (x==0) { glcd (1, 0, "OFF"); }
```



### ทดสอบผลการทำงานของ App ที่เขียนขึ้น



# ຈັບคู่บลูทูธ



ให้คลิกเลือก ใส่ PIN ซึ่งในที่นี้เลือกใช้ 1234 เป็นค่ามาตรฐาน



### หลังจากจับคู่แล้ว



# กด Connect และเลือกชื่อที่สร้าง

* ② ● ● ● * ?	* ② ● ●
Connect	B4:52:7D:F2:97:B3 SmartWatch 2
OFF	00:12:06:31:81:13 BlueRover0002
	<i>โ</i> เลือกตัวนี้

### หลัง Connect ไฟที่ BlueStick จะติดค้าง



เมื่อกดสวิตช์ ON ที่หน้าจอ ข้อความที่ GLCD จะแสดงข้อความ "ON" ส่วนเมื่อกดสวิตช์ OFF ที่หน้าจอ ข้อความที่ GLCD จะแสดงข้อความ "OFF"

# การติดตั้ง App ไปที่แอนดรอยด์

		App (provide QR code for .apk)	Relational in the state of the second s
Screen 1 • Add Screen		App ( save .apk to my computer )	
Viewer			
	[	Display hidden components in Viewer	
	[	Display hidden components in Viewer	🔊 📶 📓 9:48
		Display hidden components in Viewer Screen1	<b>≋.₁  🖁</b> 9:48

### ี้เมื่อทดสอบโค้ดเป็นที่พอใจแล้ว ต้องการติดตั้ง App ไปยังแอนดรอยด์ ให้ไปที่เมนู Build เลือกทำได้ 2 ทาง



# การติดตั้ง App ไปที่แอนดรอยด์ (วิธีแรก)

#### **Barcode link for BlueStick**



#### OK

Note: this barcode is only valid for 2 hours. See the FAQ for info on how to share your app with others.

วิธีที่ 1 App (provide QR code for .apk) จะมีหน้าต่างแสดง QR code ลิงก์ดาวน์โหลดไฟล์ .apk สำหรับติดตั้ง ใช้ได้ 2 ชั่วโมง

🖸 🖪 📀 🔁 0 × N ? 12 V A5 MIT App Inventor 2 Companion MIT App Inventor 2 type in the 6-digit code -orscan the QR code Six Digit Code connect with code scan QR code Your IP Address is: 192.168.1.5 Version: 2.20ai2zx1

Θ

## การติดตั้ง App ไปที่แอนดรอยด์ (วิธีที่ 2)

Opening BlueStick.apk	
You have chosen to open:	Pauge BlueStick คุณต้องการดิดตั้งแอปพลิเคชันนี้หรือไม่ แอปพลิเคชันจะเข้าถึง:
which is: apk File (1.3 MB) from: http://ai2.appinventor.mit.edu	ข้อมูลส่วนบุคคล อ่านสถานะและ ID โทรศัพท์
What should Firefox do with this file?	ปู แก้ไข/ลบเนื้อหาในความจำในเครื่อง
O Open with Browse	การเข้าถึงอุปกรณ์
Save File     Do this <u>a</u> utomatically for files like this from now on.	<ul> <li>การเขาใช้อินเตอร์เน็ดเดิมรูปแบบ</li> <li>ดูสถานะ Wi-Fi</li> <li>ดูสถานะเครือข่าย</li> </ul>
OK Cancel	๙ การตั้งค่าบลูทูธ สร้างการเชื่อมต่อบลูทูธ
OK Carcer	<u>- เ</u> ทดสอบการเข <sup>้</sup> าถึงพื้นที่จัดเก็บที่มีการป้องกัน
วิธีที่ 2 บันทึกไฟล์ลงคอมพิวเตอร์ แล้ว	
ค่อยคัดลอกใส่เครื่องไปติดตั้ง ซึ่งอาจจะ	
ไปติดตั้งเครื่องอื่น ๆก็ได้	
	ຍກເລົກ ดิดตั้ง

175