App Inventor 的基本介绍和蓝芽链接功能的使用

一、专案建置与基本介绍

我们 MIT App Inventor 2,它是图形化接口的开发环境,只要拉对象和方 块图就能完成一个 APP,是不是很简单呢?

首先,进入网址 http://ai2.appinventor.mit.edu/ 并登入 google 账号后

点击 New Project 来新增专案,并打上专案名称点击 OK,如下图。

МІТ Ар	p Inventor 2 Beta	Project +	Connect -	Build +	Help +
New Project	Delete Project				
Projects					
Name		Date Cr	eated		
	Create r	new App Inve	ntor project		
	Proj	ject name: te	st		
	Ca	ancel	OK		

进入之后会看到开发接口,主要分为四项, Palette、Viewer、Components 和 Properties

Palette: 所有组件的放置区,需要那些功能就把它拖到 Viewer 区的屏幕 里,.net 玩家看了应该会觉得很亲切,很多功能都大同小异。

Viewer: 装置预览的画面, 把需要的组件拖曳进来并做排版, 但实际位置还 是有出入, 放到装置上看会比较准。

Components: 所有拖曳到预览画面的组件(包括隐藏的)都会出现在这里面,

86DUIND www.86duino.com

我们可以点击里面的组件到右边的 Properties 做修改。

Properties: 显示组件的属性,可以在预览画面中点击或 Components 里 点击即可编辑组件的属性。

Palette	Viewer		Components	Properties
User Interface		Display hidden components in Viewer	Screen1	Screen1
Button CheckBox DatePicker Image Label ListPicker ListView Notifie PasswordTextBox		♥.d ¥ 9:48 Screen1		AboutScreen AlignHorizontal Left AlignVertical Top BackgroundColor White BackgroundImage None
Sinder Spinner TextBox TimePicker WebViewer	0 0 0			CloseScreenAnimation Default Icon None OpenScreenAnimation
Layout			Rename Delete	Default

我们先来做一个按按钮会改变文字的程序,先把左侧的 button(按钮)点住并 拖曳到预览画面中,如下图,如果拖错组件可在 Components 框框内点击 Delete 出现对话框按 OK 即可删除,另外 Rename 可对组件做更名,但在写程序也就 是堆方块时也要找相同名字才能做调用的动作。

Palette	Viewer	Components
User Interface	Display hidden components in Viewer	😑 🛑 Screen1
Button 💿	الله 9:48	Button1
CheckBox 📀	Screen1	
DatePicker 📀	Text for Button1	
🎴 Image 📀		
A Label 🧿		
ListPicker 🕐		
ListView 📀		
🛕 Notifier 📀		
PasswordTextBox (?)		
🛄 Slider 💿		
📑 Spinner 🕐		
TextBox 🕐		
DimePicker 📀		
WebViewer 📀		
Layout		Rename Delete
Media		

再来我们到 Properties(属性) 框框中的 FontSize (字号)改为 20, 然后在

Viewer	Components	Properties
Display hidden components in Viewer	B Screen1	Button1
UDisplay hidden components in Viewer 零』 ● 9:48 Screent 請果皆我	Button1	Button1 BackgroundColor Default Enabled FontBold FontBold FontSize 20 FontTypeface default Image None Shape
		default ShowFeedback
	Rename Delete	
		Text
	Media	諸點我
	Upload File	TextAlignment

Text 选项中,改成想要的文字,例如:请点我,如下图。

拉霸往下拉或鼠标往下滚到下面可以看到 Width(宽) 与 Height(高) 的选项。

ledia	請點我	
Upload File	TextAlignment center 🔻	
	TextColor Default	
	Visible showing Visible	
	Width	
	Automatic	
	Height	
	Automatic	

点开可以看到三个控件: Automatic(自动)、Fill parent(充满)、pixels(自定 义大小),我们在 Width(宽)的属性改为充满,如下图。

Automa	tic nt
D	pixels
Cancel	OK

在 Height(高)的选项中选择自定义并输入 50, 单位为 pixels, 如下图。

Automa	tic
Fill pare	nt
50	pixels
Cancel	OK

完成图,记得属镜中的 Height(高)要比 FontSize(字号)大,不然会切到字喔。

		9:48 🖬 🖍 🦃	3
Screen1			
	請點我		

接着再把 Label 拖曳到预览画面中,如下图。

Pal	ette		Viewer
Us	er Interface		Display hidden components in Viewer
	Button	۲	جَيارا گا 9:48
1	CheckBox	۲	Screen1
1	DatePicker	۲	清累收
2	Image	۲	
A	Label	•	Text for Label1
	ListPicker	۲	
≡	ListView	۲	
Δ	Notifier	۲	
	PasswordTextBox	۲	
	Slider	۲	
8	Spinner	۲	
I	TextBox	۲	
0,0	TimePicker	۲	
	WebViewer	0	
La	yout		
М	edia		

在再来我们将 Properties 框框中的 FontSize 一样改为 20, 如下图, 接着

86DUIND www.86duino.com

看到右上角的 Designer 是设计程序 UI(接口)的地方也是目前所在的地方,而 Blocks 是堆积木区,也就是撰写程序的地方,我们程序接口也做的差不多了, 接着来点击 Blocks 到堆积木区来一探究竟吧!

	Designer Blocks
Components	Properties
Screen1	Label1
Button1	BackgroundColor
	FontBold
	FontItalic
	FontSize
	20
	FontTypeface
	default 🔻

主要分为两大区域,分别是 Blocks 积木区和 Viewer 观看区也就是程序撰 写的区域,我们先来介绍 Blocks 的部分吧。

MIT App Inventor 2 Beta	Project Connect - Build - Help - My Projects Guide R *	eport an Issue
test	Screen1 - Add Screen Remove Screen	Designer Blocks
Blocks	Viewer	
 Built-in Control Logic Math Text Lists Colors Variables Procedures Screen1 Button1 Label1 Any component 	0 0 Show Warnings	

Blocks 积木区,主要会用到的部分是内建的 Built-in 及刚刚拉进预览窗口中

组件都会出现在 Screen1 中,我们现在点击 Built-in 中的 Control(控制)选项,如下图,有没有很熟悉呢?我们常用的判断式和循环都放在这里喔~

lu ol	
Blocks	Viewer
Built-in Control Cogic Cogic Math Text Lists Colors Variables	for each number from t 1 to t 5 by 1 do
Procedures Screen1 Button1 Label1 Any component	for each (item) in list (do while test () do if () then () Varnings else

接下去是Logic(逻辑),布尔运算大家应该都不陌生,这里就不多做介绍了。



Math(数学),里面包含了基本运算及随机变量,连三角函数都有喔。

www.86duino.com



Variables(变量), 宣告、改变、取得变量的指令都在里面。

.86duino.com	
Blocks	Viewer
⊖ Built-in	
Control	initialize global name to
Logic	aet 🗖
Math	
Text	set 🚺 to 🕨
Lists	
Colors	initialize local name to
Variables	in
Procedures	
Screen1	C 🔲 initialize local name to 🕻
Button1	in F
Label1	

86DUIND

接着要进入正题啦,当 Button 被点击之后会触发 Click 的指令,我们就可以针对这个控件做一些事情,点击 Button1 可以看到控件(积木)跳出来,我们拖曳.Click(点击)这个积木出来到右侧,如果拖错东西出来可以拖到右下角的垃圾桶把它删除喔,如下图。

Blocks	Viewer	
 Built-in Control Logic Math Text Lists Colors 	when Button1 Click do when Button1 GotFocus do	when Button1 Click
Variables Variables Procedures Screen1 Button1 A Label1	when Button1 .LongClick do when Button1 .LostFocus do	
Any component	when Button1 . TouchDown do A 0 when Button1 . TouchUp	

接着点击左侧的 Label1,把标有「set Label1.Text to」的积木拖曳到刚刚的积木里把它拼起来,如下图,它的功能为改变 Label1 显示的字。

www.86duino.com Viewer Blocks Built-in Label1 . BackgroundColor Control when Button1 .Click Logic set Label1 . Text To 🕻 set Label1 . BackgroundColor . to Math Text Label1 . FontSize Lists set Label1 . FontSize to Colors Variables Label1 · Height · Procedures 😑 📃 Screen1 set Label1 . Height . to Button1 A Label1 Label1 • . Text • Any component set Label1 . Text . to Label1 . TextColor set Label1 . TextColor to

86Duind

接着点击左侧的 Text,把第一个积木拖到右边拚进去。

Blocks	Viewer	
 Builtin Control Logic Math Text Lists Colors Variables Procedures Screen1 	<pre>when Button1 Cick do set Label1 · Text · to C * * * length is empty compare texts < < ></pre>	
Button1 Label1 Any component	trim upcase starts at text piece Show Warmings contains text	1

鼠标点击刚刚的积木中空白处把内容改为想要的字,例如:你好,这样 程序就大致完成了,是不是很简单呢?



如果刚刚有开喇叭,在积木接合的瞬间会有"喀"一声,另外当积木拼不进去 并弹开时,就代表你逻辑错误或拉错积木啦!在过程中观察一下会发现判断式和 一些控制指令都是^{see},而回传信息的积木都是¹,另外还要注意左下角有没 有错误呢,如下图,黄色惊叹号是警告,通常都是积木没拼上去,可以不理它, www.86duino.com

但是当有红色惊叹号出现,就表示程序严重错误,赶快找到错误并修正吧。



程序完成之后,目前有两种方案可以传到 Android 装置,方案一是藉由 WiFi 与计算机同一个局域网络可直接传档案,方案二是直接生成 apk 安装档,再丢 到 Android 装置中安装。

MIT App Inventor 2 Beta	Project - Col	nnect +	Build + Help +	
test	Screen1 - Add Screen		App (provide QR code for .apk)	
Blocks	Viewer	-	- the course aparto my compared	
Built-in Control Logic	when BL do set	utton1 •). Label1 •	Click . Text to ("你好"	
Math				

将 APP 传到装置之前, 先到 设定-> 安全性 把不明来源打勾, 如有出 现警告请按下确定。

<	🔯 安全性		
	裝置管理		
	裝 置管理員 查看或撤銷裝置管理員		
	不明的來源 ^{允許安裝來源不明的應用程式}	~	
	驗證應用程式 不允許安裝可能造成系統損壞的應 用程式,或在安裝前提出警告		
	憑證儲存空間		
	儲存空間類型 ^{僅限軟體}		
	信任的憑證		

如果选择方案一,必须先安装 MIT AI2 Companion 这个 APP,开启

PLAY 商店搜寻打 MIT 就有关键词可以选择了,如下图,安装之后就把它开 启吧。

〈 序 應用程式		Q	1	
× * *	VIT AI2 Cor AIT CENTER FOR	npan R мовіі 安裝	ion Le lea	
MIT App Inventor 2 Comparison MIT App Inventor 2 -Of- can the QR code SR Dyr Code Connect with code Connect with code Connect with code Code 20760720		122 Mit App I Mit App I Mi	werker 2 Companion IT App Inver type in the 6-digit or scan the QR co Sie Digit Code Innect with Scan QR cc or IP Address is: 1921 Version: 2.11 aZzz	
 ►	∽ ∩ ₹	」 2014年 2	5 C 7月5日 2.58MB	

先回到计算机中,点击 Build-> App(provide QR code for .apk),如下图。

MIT App Invent	or 2 Beta	Project *	Connect +	Build +	Help +
test		Screen1 +	Add Screen	App (pr	rovide QR code for .apk) ave .apk to my computer)
Palette		Viewer			
User Interface				Display	/ hidden components in Viewer
Button	0				
CheckBox	(?)			Screen	1
DatePicker	0				請點我
image	۲				
A Label	1			Text f	or Label1



接着会跑出进度条,正在编译 APP 中,如下图。



接着就是拿起装置开启刚刚安装的 MIT AI2 Companion 程序啦,如下图, 然后点击 scan QR code。



之后对着屏幕扫描,如下图。



扫完 QR code 后会跳回原来的画面,这时不要紧张也不要乱按,其实是 APP 正透过网络从计算机传到手机,需等待一段时间,成功了便会出现程序安装画面, 如下图,我们按点击安装。



如果选择方案二,点击 Build-> App(save.apk to my computer),



86DUIND www.86duino.com

> 进度条跑完之后就会跳出储存窗口,如下图,储存之后再把它传进装置 做安装,但装置必须安装档案管理 APP(有些装置会内建),浏览至档案目录 并点击安装,虽然方案二会比方案一还要费时,但多了安装档可以随时分享 给其他人使用,方案一适合程序除错时使用。

		另存新檔	1			×
€ 🦻 ד ↑ 🎚	▶ 本機 ▶ 文件 ▶ А	App Inventor 2	~ ¢	搜尋 App Invento	or 2 🍂	o
組合管理 ▼ 新:	増資料夾				HI • (0
☆ 我的最愛 ▶ 下載 ■ 桌面 ◎ 最近的位置	▲ 名稱	· 没	修改 有符合搜尋條件的 ¹	(日期 頁目	類型	
≪ 家用群組 √ 家用群組 √ 本機 ○ 下載	ł					
 □ 文件 □ 資件 □ 音樂 □ 具面 	~ <					>
檔案名稱(N): 存檔類型(T):	<mark>test.apk</mark> APK 檔案 (.apk)					* *
▲ 陽藏資料夾				存楣(S)	取消]

安装完成后,我们就来验收结果啦,如下图,这是点击 Button 前的样子。

Screen1

請點我

Text for Label1

点击 Button 之后, Label 就变成 "你好"了,如下图。

Screen1

請點我

你好

是不是觉得程序没有很完整呢?例如可以在 Screen1 属性中的 Title 项, 把 "Screen1" 改成 "我的第一个程序",还有在 Label1 属性中的 Text 项, 把里面的文字去掉,这样只有在点击 button 之后才会出现文字,如下图, 屏幕有没有干净许多呢?



三、第二个程序

我们来做结合 EduCake 的程序吧,最简单方便的沟通方式就非蓝芽莫 属啦~首先开新项目点击 Project-> Start new project

est		My Projects
		Start new project
Palette		Import project (.aia) from my computer
User Interface		Import project (.aia) from a repository
Button	0	Delete project
CheckBox	0	Save project
DatePicker	0	Save project as
		<u> </u>
专案名称	就取 "Bluetoo ⁻ Create new App	th_Connect"吧。 Inventor project
	Project name:	Bluetooth_Connect

接着从左侧拖曳一个 ListPicker 组件到预览画面中。

Palette		Viewer
User Interface		Display hidden components in Viewer
Button	0	?⊋.ıl № 9:48
CheckBox	0	Screen1
DatePicker	0	Text for ListPicker1
🔛 Image	0	
\Lambda Label	0	
ListPicker	0	
ListView	•	
🔼 Notifier	0	
PasswordTextBox	0	
Slider Slider	0	
📑 Spinner	0	
TextBox	0	

在 Properties(属性)字段中的 FontSize 改为 20, Text 改为"选择蓝芽

装置",Width(宽)改为 Fill parent(充满),Height(高)改为 50 pixels, 改完之后会跟下图一样。

		(h)	9:48	
	Screen1			
	選擇藍	芽裝置		
1				
L				

这里先接绍一下 ListPicker 的功能,它功能有点像 Button,不过它是 具有选单的 Button,点击它就会跳出我们指定的选项,如下图,然后我们 可以针对点击其中的选项再做后续的动作。

Bluetooth_Connect	
選項一	
選項二	
選項三	
選項四	1

选单设定的方法有两种,一种是在 Properties(属性)字段中的

ElementsFromString 里来新增选项,如下图,选项跟选项之间用半角逗号 隔开即可,使用的前提是有默认的选项才能预先新增,这里的步骤可以不用 做,要测试也是可以的。

	Properties	
	ListPicker1	
	BackgroundColor Default	
	ElementsFromString 選項一,選項二,選項三,選	
方法二,如下图	日,可用在程序中指定选项,弹	性较高。
set ListPicker1 . Elen	nentsFromString 🔹 to 📴 選項-	,選項二,選項三,選項四
方法二的另一种	中表示法,一切以写程序方便就	好。
set ListPicker1 •	. Elements • to 📔 🔲 make a	list <u>"選項一"</u> "選項二" "選項三"

接下来进入正题,到 Palette(组件区),点击 Connectivity 会跳出组件, 接着把 BluetoothClient 拖曳到预览屏幕内,如下图。

×-3	MIT App Invento	or 2 Beta	Project •	Connect •	• Build • Help •	
Blue	tooth_Conne	ct	Screen1 •	Add Screen	een Remove Screen	
Palet	tte		Viewer			
Use	r Interface				Display hidden components in Viewer	
Lay	out				🦻 🛍 🦉 9:48	
Med	lia		Screen1			
Drawing and Animation					選擇藍芽裝置	
Sensors						
Soci	ial					
Stor	age					
Con	nectivity				8	
Ş 1	ActivityStarter	۲			BluetoothClient1	
10 E	BluetoothClient	۲				
10 E	BluetoothServer	۲				
• 1	Web	۲				
LEG	08 MINDSTORM	50				

拉进去之后,因为没有实体,所以只会显示在预览屏幕的下面,也不需 要更改属性。

www.86duino.com

Screen1 - Add Screen Remove Screen		Designer Blo
Viewer	Components	Properties
Display hidden components in Viewer Skreen1 選擇藍芽裝置	9.43	BluetoothClient1 CharacterEncoding UTF-8 DelimiterByte 0 HighByteFirst Secure
Non-visible components	Rename Delete Media	

接下来从左侧拖曳一个 Button 组件到预览画面中, Properties(属性) 字段中的 FontSize 改为 20, Text 改为"请点我", Width(宽) 改为 Fill parent(充满), Height(高) 改为 50 pixels, 改完之后会跟下图一样。

Palette	View	er
User Interface		Display hidden components in Viewer
Button	0	≈₁1 🖻 9:48
CheckBox	0	Screen1
DatePicker	0	選擇酸芽裝署
🌌 Image	0	
A Label	0	諸點我
ListPicker	0	and a summary of a
ListView	0	
🛆 Notifier	0	
PasswordTextBox	0	
Slider	0	
🗑 Spinner	0	
I TextBox	0	
TimePicker		
WebViewer	0	
Layout		
Media		Non-visible components
Drawing and Animatio	n	8
Concern		BluetoothClient1

在 Button1 的 Properties(属性)字段中 Visible 改为 hidden(隐藏),用 意是执行 APP 的时候先隐藏,等到蓝芽联机之后才做显现,接下来点击

86DUIND www.86duino.com

Blocks 到积木区吧。

1	Text		
1	請點我		
г	FextAlignment		
(center 🔻		
1	TextColor		
1	Default		
1	/isible		
	hidden 🔻		
N	Vidth		
	Fill parent		
ł	leight		
1	50 pixels		

左侧点击 ListPicker1 出现控件,拖曳 BeforePicking 控件出来,如下 图,它的触发条件是点击 ListPicker"前"会触发,我们可以派它做一些事情,例如:更改或新增 ListPicker 的选单。

Blocks	Viewer
 Built-in Control Logic Math Text Lists Colors Variables 	when ListPicker1 AfterPicking do when ListPicker1 BeforePicking do when ListPicker1 GotFocus
Procedures Screen1 ListPicker1 Button1 BluetoothClient1	do when ListPicker1 .LostFocus do
Any component	when ListPicker1 TouchDown

再次点击 ListPicker1 出现控件,滚动条往下拉才看的到 Elements 控件,把它拖曳出来组合进去,如下图。

www.86duino.com



左侧点击 BluetoothClient1 出现控件,一样将滚动条往下拉,看到 AddressesAndNames,把它拖曳到右侧接上,如下图,功能是将 Android 所配对过的蓝芽设备用 list 的方式列出来,我们这样做可以在程序刚开始执 行的时候触发点击 ListPicker1 前的控件,然后做新增蓝芽列表的选项。



左侧点击 ListPicker1 出现控件,拖曳 AfterPicking 控件出来,如下图, 它的触发条件是点击 ListPicker"后"就会触发 AfterPicking 控件。

Blocks	Viewer	
Built-in Control	when ListPicker1 AfterPicking do	when ListPicker1 .BeforePicking do set ListPicker1 .Elements to Bluetooth
Math Text Lists	when ListPicker1 .BeforePicking do	when ListPicker1 AfterPicking
Variables Procedures Screen1	when ListPicker1 .GotFocus do	
ListPicker1 BluetoothClient1 Any component	when ListPicker1 LostFocus do	
	when ListPicker1 TouchDown do	

86DUIND www.86duino.com

Blocks	Viewer
Built-in Control Logic Math Text Lists Colors Variables Procedures Screen1 ListPicker1 Button1	when ListPicker BeforePicking do cet ListPicker Cite for each number from i 1 to i 6 by 1 do for each item in list i do
BluetoothClient1	while test # do

左侧点击 Control(控制),把 if 判断式拖曳至右侧接合,如下图。

左侧点击 BluetoothClient1,将 Connect address 控件拖曳到右侧跟 判断式接合,如下图,功能是给它蓝芽的地址就可以联机,成功回传 true, 失败回传 false。

Blocks	Viewer
 Built-in Control Logic Math Text Lists Colors Variables Procedures Screen1 ListPicker1 Buttoon1 BluetoothClient1 	call BluetoothClient1 .BytesAvailableToReceive call BluetoothClient1 .Connect address
Any component	call BluetoothClient1 • .ReceiveSigned1ByteNumber call BluetoothClient1 • .ReceiveSigned2ByteNumber

左侧点击 ListPicker1,将 Selection 拖曳至右侧结合,如下图,它的功能是回传 ListPicker1 选单中我们选择的文字。



左侧点击 Button1,将.Visible 控件拖曳到右侧接合,如下图,因为我 们刚开始是把 Button1 做隐藏,当蓝芽联机成功会回传 true 去处发 if 的条 件,我们就可以把按纽显现出来了。

Blocks	Viewer
Built-in Control Logic Math Logic Math Lists Colors Variables Procedures Colors Scren1 ListPicker1 ListPicker1 Button1 Blutton1 Ary component Ary component	set Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Ext • to •] Button1 • [Visible •] Set Set Button1 • [Visible •] Set Button1 • [Visible •] Button1 • [Visible •] [Ext •] Button1 • [Ext •] [Ext •] Button1 • [Ext •] [Ext •] Button1 • [Ext •] [Ext •]
接着 Blocks	点击 Logic,拖曳 true 到右侧接合,如下图。
Built-in Control Control Cogic Math Text Lists Colors Variables Procedures Screen1 ListPicker1 Button1 Button1 Buttont	<pre>when ListPicker1 BeforePicking do set ListPicker1 Clent1 AddressesAndNames when ListPicker1 Clent1 Connect when ListPicker1 AfterPicking do if call BluetoothClient1 Connect address [ListPicker1 Selection then set Buitton1 Clent1 Connect then set Buitton1 Clent1 Connect </pre>
BluetoothClient1 Any component	A 0 A 0 Show Warnings

侧点击 Button1,将 .Click 拖曳到右侧,如下图。

www.86duino.com

Blocks	Viewer
Built-in Control Logic Math	when (Button1 · Click do
Text Lists Colors	when Buttont .GotFocus do if do if call BluetoothClient1 .Connect address .call BluetoothClient1 .congClick then
Procedures Creen1 ListPicker1 Button1	do set Label1 . Text • to ListPicker1 . Selection • when Button1 LostFocus do do do
BluetoothClient1	when Button1 · TouchDown do a 0

接下来点击 BluetoothClient1,将 .SendText text 拖曳至右侧接合,如下图,它的功能是可以传字符串给联机中的蓝芽装置。

Blocks	Viewer	
Built a Control Contr	call BluetoothClient1: Send2ByteNumber number of set EstPicker1: call BluetoothClient1: Send2ByteNumber number number call BluetoothClient1: Send2ByteS ist set EstimClient1: venter set EstimClient1: Send1: Send1: Send1:	nes •

接下来点击 Text,我们拖曳一个空白字符串到右侧,并在里面输入"a",如下图,这样每按一次 Button 就会传"a"的字符给联机中的蓝芽装置。



接下来介绍 EduCake 蓝芽模块的部分,这里使用的蓝芽模块是较为大 众化的: HC-06。

HC-06 正面

www.86duino.com



HC-06 反面



接好电路以后,我们先来设定 HC-06 的一些参数



首先开启 IDE 接口,输入以下程序:程序上传好了以后,就可以利用 IDE 接口里面的 COM 监视器(Serial Monitor)来和蓝芽通讯,画面如下



到底要输入什么哩?其实 HC-06 在通电未联机情况下,灯光会闪烁并处于 AT command mode,当在这个状态的时候,我们就可以任意更改名称、密码,甚至连通讯速度(baud rate)都可以改,以下介绍几个常用的 AT 指令:

- ▶ AT:测试是否正常工作,响应「OK」
- ▶ AT+NAMEaaa:将装置名称改为「aaa」
- ▶ AT+PIN1234 : 将联机密码换为「1234」
- ▶ AT+VERSION : 响应韧体的版本
- ▶ AT+BAUD1:将 baud rate 换为 1200
- ▶ AT+BAUD2 : 将 baud rate 换为 2400
- ▶ AT+BAUD3 : 将 baud rate 换为 4800
- ▶ AT+BAUD4 : 将 baud rate 换为 9600
- ▶ AT+BAUD5 : 将 baud rate 换为 19200
- > AT+BAUD6:将 baud rate 换为 38400
- ▶ AT+BAUD7:将 baud rate 换为 57600

上面前三个 AT 指令比较常用到,如果要改名称为 abc,就输入 AT+NAMEabc,成功会显示 OKsetname,设定密码为 1234,就输入 AT+PIN1234,成功会显示 OKsetPIN,这里要注意的是更改名称之后要断 电重开才会生效而且不能使用中文(会乱码),而改完密码不需重开即生效。

86DUIND www.86duino.com

3	COM5	- 🗆 🗙
AT		Send
Enter AT commands: OK		
🖌 Autoscroll	No line end	ling 🗸 9600 baud 🗸

刚刚做出蓝芽联机的 APP, 再加上设定完 HC-06, 终于要进入正题啦, 我们要做出用手机控制一个 LED 灯, 程序代码如下:

86DUIND www.86duino.com



程序上传之后,我们转移阵地到 Android 装置中吧,首先确认装置已 经跟 HC-06 配对完成,如下图,本书以 HC-06 作为蓝芽名称方便辨识。

	く藍牙	開啟	:
	Android 装置 只向配對的裝置顯示		
	配對裝置		
	HC-06		₽
	搜尋裝置	P	
接	等着开启程序,如下图,点击"选择	释蓝芽装置"。	
Scre	en1	_	_
	選擇藍芽	裝置	
	跳出蓝芽选单,如下图,请 Bluetooth_Connect	选择 HC-06 蓝芽	模块联机。
	00:15:FF:F4:0B:7	2 JY-MCI	J
	98:D3:31:40:03:2	F HC-06	

联机成功后会出现"请点我"的 Button,如下图,可以观察到 HC-06 从 原本的闪烁变化为恒亮,接下来点击"请点我",EduCake 上的 LED 灯是否 亮起来了呢?再点一下就会熄灭了。



四、第三个程序

我们再度回到 App Inventor 中,延续刚刚的程序继续修改,如果想保 留刚刚的作品,可以利用 Project-> Save project as ... 来另存一个新项目 做编辑,接着删除原本的 Button1,如下图。

iewer	Deleting this component will delete all blo with it in the Blocks Editor. Are you sure yo delete?	ocks associated u want to	ponents Screen1 El ListPicker1
Screen1	確定	取滴	BluetoothClient1
			Rename Delete

在刚刚的程序中只有控制一颗 LED,这次我们要一次控制五颗,并在 EduCake 上加入 3 颗按纽,当想要把五颗按键摆成横的却没办法,这时候 就需要用到 HorizontalArrangement,在 Layout 中,将它拖曳至预览屏幕 中,如下图,这个组件的概念有点像图层管理,在框框里面的原件就只能在 里面活动,并随着 HorizontalArrangement 隐藏而隐藏,是个方便管理接 口的好工具。

Jser Interface	Display hidden components in Viewer 🦻 📶 🖻 9:4
ayout	╤₁1 2 9:4
HorizontalArrangement (2)	Screen1
TableArrangement ③	選擇藍芽裝管
VerticalArrangement ⑦	
Media	
Drawing and Animation	
Sensors	
Social	
Storage	
Connectivity	
ECOR MINDSTORMSR	

在 HorizontalArrangement 的属性中,将 Width(宽)改为 Fill parent, 如下图。

Properties			
Properties HorizontalArrangement1 AlignHorizontal Left AlignVertical Top Visible showing Width Fill parent			
Height Automatic			

把 Button 拖曳到刚刚的框框内,在属性中,将 Text 改为 LED1, Height(高)改成 50 pixels,如下图。

≈₁ i i 9:4	8	Screen1 ListPicker1 HorizontalArrangement Button1
╤₄∥ 🖻 9:4	8	ListPicker1 HorizontalArrangement Button1
		HorizontalArrangement
		Button1
		Division all Official
		BluetoothClient1
	4	

跟刚刚一样的动作,再拉四颗按钮出来,属性的部分除了名字之外其他

www.86duino.com

都相同,如下图。

選擇藍芽	耧置		
			_
D2 LED3	LED4	LED5	
	選擇藍す D2 LED3	選擇藍芽裝置 D2 LED3 LED4	選擇藍芽裝置 D2 LED3 LED4 LED5

感觉按钮是不是没有排正呢?再点击每个 button,在属性中的 Width(宽)改为 Fill parent,改完之后就会如下图一样。

Screen1			li.	Ral 🛿 9:4	18
	選	擇藍芽裝	置		
LED1	LED2	LED3	LED4	LED5	1/17

接下来再拖曳一个 HorizontalArrangement 到预览屏幕中,并在属性中的 Width(宽)改为 Fill parent,如下图。

Palette		Viewer					Components		Properties
User Interface Layout		Display h	idden comp	onents in Vi	ewer	🔊 📶 📓 9:48	© Screen1	1	HorizontalArrangement2 AlignHorizontal
 HorizontalArrangement TableArrangement VerticalArrangement 	© ©	Screen1	選	擇藍芽裝	b置		Button	IlArrangement1 1 2 3	AlignVertical Top
Media Drawing and Animation Sensors Social Storage Connectivity		LED1	LED2	LED3	LED4	LED5	Button Button Horizonta Bluetooth	4 5 Il <mark>Arrangement2</mark> Client1	Visible showing Width Fill parent Height Automatic
LEGO® MINDSTORMS®							∢ Rename	Delete	

在左侧点击 User Interface 接着拖曳一个 CheckBox 到刚刚的框框中, 在属性中将 Text 改为按钮 1,如下图,它的功能是左侧的框框可以打勾也



可以取消勾选,我们这里是用来要观看 EduCake 的按钮是否有被按下。

Palette	Viewer	Components	Properties
User Interface	Display hidden components in Viewer	🛛 🗍 Screen 1	CheckBox1
Button CheckBox DatePicker Image Label ListPicker ListView Notifier PasswordTextBox Slider Spinner TextBox TimePicker	⑦ Screen1 ⑦ 送揮藍芽裝置 ⑦ 上ED1 LED1 LED2 LED1 LED3 1 1	E ListPicker1 HorizontalArrangement1 Button1 Button2 Button3 Button3 Button5 HorizontalArrangement2 CheckBox1 BluetoothClient1	BackgroundColor None Checked Enabled PontBold FontBold Fonttalic I4.0 FontTypeface default Text
WebViewer	•	Rename Delete	按鎖1 TextColor Black

继续刚才的动作再拖曳两个 CheckBox, 属性中的 Text 分别改为按纽 2

及按纽3,最后再将三个 CheckBox 属性中的 Width(宽)都改为 Fill parent, 如下图,此动作是为了要让组件与画面对齐。

2011	DatePicker	U		選	擇藍芽裝	置		Butto	on2	Checked
-	Image	0						Butto	on3	Enchled
A	Label	3	LED1	LED2	LED3	LED4	LED5	Butto	on4	
	ListPicker	0						Butto	on5	FontBold
≡	ListView	0	□按鈕1	E I	安鈕2	回按鈕	3	😑 🖾 Horizon	talArrangement2	
	Notifier	0						Chec	:kBox1	FontItalic
-	PasswordTextBox	(7)						Chec	kBox2	
-								Chec	kBox3	FontSize
11	Slider	0						🚺 Bluetoo	thClient1	14.0
8	Spinner	0								FontTypeface
1	TextBox	0								default
8:10	TimePicker	0								Text
	WebViewer	0								按鈕3
									E F	TextColor
La	yout							Rename	Delete	Black
M	edia			Non-v	isible con	nponents				Visible
Dr	awing and Animation				8			Media		showing 🔻
Se	nsors			E	BluetoothCli	ent1		Upload	File	Width
So	ocial									Automatic Fill parent
St	orage									o pixels
Co	onnectivity									Cancel OK

之后再拉一个 Button 到预览屏幕中, Text 改为中断联机, Width(宽) 改为 Fill paren, 如下图。

			II.	P.al 📓 9:48			
Screen1							
	選	擇藍芽裝	置				
LED1	LED2	LED3	LED4	LED5			
□按鈕1	□按鈕2 □按鈕3						
		中斷連線	Į				

因为蓝芽在联机之前除了联机的 button 显示之外,其他的组件都要隐藏起来,所以我们将 HorizontalArrangement1、

HorizontalArrangement2及Button6都隐藏起来,将属性中的Visible改为hidden,如下图。

Components	Properties
😑 📃 Screen1	HorizontalArrangement1
EistPicker1	AlianHorizontal
HorizontalArrangement1	Left
Button1	A March Frankland
Button2	Alignvertical
Button3	ioh .
Button4	Visible
Button5	showing T
😑 🔤 HorizontalArrangement2	showing
CheckBox1	Fill parent
CheckBox2	Height
CheckBox3	Automatic
Button6	
🔢 BluetoothClient1	

接着在左侧 Sensors 底下有 Clock 组件,将他拖曳至预览屏幕内,因为它也是没有实体所以只能显示在下面。

www.86duino.com

Palette		Viewer
User Interface		Display hidden components in Viewer
Layout		🦻 🦛 9:48
Media		Screen1
Drawing and Animation		選擇藍芽裝置
Sensors		
AccelerometerSensor	0	
BarcodeScanner	(?)	
💆 Clock	0	
LocationSensor	0	
🕨 NearField	0	
OrientationSensor	(?)	
Social		
Storage		
Connectivity		
LEGO® MINDSTORMS®	9	
		Non-visible components
		BluetoothClient1 Clock1

将属性中的 TimerEnabled 取消勾选,我们不让他在开始就自动执行, TimerInterval 是跑完一次之后的时间间隔,我们将它改为 10,单位为毫秒,如 下图,接下来点击 Blocks 到积木区吧。

Properties	
Clock1	
TimerAlwaysFires	
TimerEnabled	
10	
	Properties Clock1 TimerAlwaysFires TimerEnabled TimerInterval 10

86DUIND www.86duino.com

跟上一个程序一样,我们联机之后要显现刚刚隐藏的组件,所以我们把 HorizontalArrangement1、HorizontalArrangement2及Button6的.Visible 控件拉出来,而Clock1是拉.TimerEnabled出来,如下图。



从左侧 Logic 中拖曳三个 true 过去接合,如下图。

Blocks	Viewer
Built-in Control Control Cogic Math Text Lists Colors Variables Procedures Screen1 ListPicker1 ListPicker1 Communications Button1	true when ListPicker1 BeforePicking do set ListPicker1 Cents to (BluetoothClient1 AddressesAndNames) not mot not if call BluetoothClient1 Connect address (ListPicker1 Selection) do if call BluetoothClient1 Connect set HorizontalArrangement1 Visible to (true) set HorizontalArrangement2 Visible to (true) set Button6 Visible to (true) set Clock1 TimerEnabled to (true) set Clock1 TimerEnabled to (true)
Button2 Button3 Button4 Button5	▲ 0 ▲ 0 Show Warnings

接下来要制作用五个按钮去控制 EduCake 上的 5 颗 LED,我们先拖曳 Button1.Click 出来再拖曳蓝芽的.SendText 出来接合,如下图。



86DUIND www.86duino.com



接下来就如法炮制出另外四个,并依顺序将 Text 控件改为 2、3、4、5,如下图,这样的用意是让等下撰写 EduCake 程序时能方便判断 Android 装置到底按了哪个按钮。

when Button1 . Click	
do call [BluetoothClient1] .SendText	
text (1 1	when Button4 .Click
	do call BluetoothClient1 .SendText
when Button2 . Click	text () * (4) *
do call BluetoothClient1 .SendText	
text (2 "	when Button5 . Click
	do call BluetoothClient1 .SendText
when Button3 . Click	text 15
do call BluetoothClient1 .SendText	
text (* 3 *	

接下来要制作 Android 接收 EduCake 传回来按钮的讯号, 左侧点击 Clock1 拉出.Timer 出来, 如下图。



接下来我们可以点击上面的.SendText做Ctrl + C 复制,Ctrl + V贴上, 然后再拖曳进去Timer 做接合,再将Text 的部分改为a,如下图。

uden Rutter 20 Chair	10AL -
do call BluetoothClient1 · .SendText text (* 2)*	when Button5 . Click do call BluetoothClient1 . SendText
when Button3 .Click do call BluetoothClient1 .SendText text (* 3 *	
when Clock1 . Timer do call BluetoothClient1 . SendText text (" a "	
0 0 0 Show Warnings	

然后从左侧的 Control 中拉出一个 if 判断式接在下面,如下图。



www.86duino.com

从左侧	的 Math 中拉出一个比较	式接在 if 的判断中,并点击中	间改成
">",如下图	د. ٥		
Blocks	Viewer		
Built-in Control Logic Math Text Lists Colors Variables Procedures Screen1 ListPicker1 ListPicker1 HorizontalArrangemer Button1 Button2		when Button2 • Click do call BluetoothClient1 • SendText text (* 2 * when Button3 • Click do call BluetoothClient1 • SendText text (* 3 * • • • • • • • • • • • • • • • • • • •	when But do call (
Button3 Button4 Button5	random integer from 1 to 100	then	

从左侧 Math 拉出数字项,拖曳至比较式内右边空格中接合,如下图。



从左侧 BluetoothClient1 拉出.BytesAvaliableToReceive,拖曳至比较 式内左边空格中接合,如下图,这串比较式可以判断是否有信息回传回来。



当有数据回传时,我们就要定义一个空间去储存它,从左侧 Variables 中拖曳 initialize global name to 出来到右侧放哪里都可以,并把 name 改成 text,方

www.86duino.com

便程序阅读,	如下图。
Blocks	Viewer
 Built-in Control Logic Math Text Lists Colors Variables Procedures Screen1 ListPricker1 SutPricker1 Button1 Button2 	<pre>initialize global name to f get initialize global (ext to f get initialize global (ext to f get initialize local name to f in in initialize local name to f in in in in in</pre>
Button3 Button4 Button5	Image: A transmission of the second secon
C C Line and a line an	

跟任写何程序一样,我们一定要赋予变量型别,我们就从左侧的 Text 拖曳一个空白的 Text 出来接在后面,如下图。

Blocks	Viewer
Built-in Control Cogic Math Text Logic Math Text Lists Colors Variables Procedures Screen1 Screen1 ListPicker1 Screent MicrontalArrangemer	<pre>initialize global Text to C * * * * initialize global Text to C * * * * initialize global Text to C * * * * initialize global Text to C * * * * initialize global Text to C * * * * * initialize global Text to C * * * * * initialize global Text to C * * * * * * * * * * * * * * * * * *</pre>
Button1 Button2 Button3 Button4 Button5 MorizontalArrangemer	Upcase

接着从左侧 Variables 中拖曳 set to 控件出来接在 if 判断式内,如下 图,并点击中间空白的部分选择刚刚的字符串变量,这样红色惊叹号就会消 失了。

Blocks	Viewer
U Variables U Variables Procedures Screen1 StistPicker1 Button1 Button2 Button3 Button5 Uton5 CheckBox1	initialize global (name to) get : set to ; initialize global (ext to) * * * when Clock1 .Timer do call BluetoothClient1 .Send Text text / * * initialize local name to ; in fill call BluetoothClient1 .BytesAvailableToReceive > (0) then A set to ; global text
CheckBox2 CheckBox3 Button6 BluetoothClient1 Clock1	A 0 A 1 Show Warnings

从左侧 BluetoothClient1 拉.ReceiveText numberOfBytes 出来,接 合在 set global text to 后面,如下图,它的功能是接收并回传所指定的字 符串长度。



从左侧 Math 拉出数字接在.ReceiveText numberOfBytes 后面,如下图, 并改为1,我们只要接收一个字符即可



接下来从 Control 里拖出一个 if 判断式接在 set global text to 下面, 如下图,这次的判断式比较特别我们要再加入例外的动作,点击判断式左上角蓝色四方形,把左侧的 else 拉进去到 if 里面,下面就会多出一个 else 了。

Builtin Control Cogic Math Colors Variables Soreen1 UistPicker1 UistPicker	for each number from 1 1 to 1 6 by 1 do	when Clock1 Timer do call EllectoninClicn1 SendTed else f f clse EluctoothClicn1 Receive Set f f numberOfBytes f f then else
Button2 Button3 Button4 Button5	while test if do	Û

www.86duino.com

从 Logi	c 拖曳一个比较式出来,接在 if 的判断中,如下图。
Blocks	Viewer
 Builtin Control Logic Math Text Lists Colors Variables Procedures Screen1 ListPicker1 MehrizontalArrangemer Button1 	<pre>when Clock1 Timer do call BluetoothClient1 SendText text a call BluetoothClient1 BytesAvailableToReceive >> (0) then set global text to call BluetoothClient1 ReceiveText numberOfBytes (1) if call bluetoothClient1 numberOfBytes (1)</pre>
Button2 Button3 Button4 Button5	A 1 A 0 Show Warnings

接着从左侧 Variables 中拉出 get 出来接在比较式左边空格中,如下图, 并点击 get 空白处,选择刚刚定义的 global text 字符串变量。



我们从左侧 Text 中拉出一个空白 text 出来接在比较式右边空白处,如下图,并在空白中填入1。

BIOCKS	Viewer
Built-in Control Logic Math Text Lists Colors Variables Procedures Screen1 ListPicker1 Screen1 HorizontalArrangemer Button1 Dutter2	<pre>when Clock1 Timer do call BluetoothClient1 SendText text * a *</pre>
Button2 Button3 Button4 Button5	starts at text piece show Warnings contains text
HorizontalArrangemer	

www.86duino.com

从左侧 CheckBox1 中拉出.Checked 出来接到 if 判断式内,如下图。		
Viewer		
when CheckBox1 LostFocus do CheckBox1 LostFocus when Clock1 Timer do cal BluetoothClient1 SendText text (*2) f cal BluetoothClient1 BluetoothClient1 Receive Country then set global text to cal BluetoothClient1 Receive Country then set checkBox1 Checked to cal BluetoothClient1 Checked to cal B		
CheckBox1 . FontSize set CheckBox1 . FontSize to		

从 Logic 拖曳 true 到右边接合,如下图。

Blocks	Viewer
 Builtin Control Logic Math Text Lists Colors Variables Procedures Screen1 Screen1 Screen1 Button1 Putico 	<pre>when Clock1 • Timer do call BluetoothClient1 • SendText text (* 2 *</pre>
Button3 Button4 Button5 HorizontalArrangemer	Image: Constraint of the second se

撰写程序就是要善用复制(Ctrl + C)与贴上(Ctrl + V),我们点击 CheckBox1.Checked 之后按 Ctrl + C 复制,然后再按 Ctrl + V 贴上,就会 再多出一组一样的组合,如下图,我们把它拖曳到 else 字段中吧。



在 else 字段中,把后面的 True 改为 false,如下图,在 App Inventor 中,只要控件中有箭头都可以随时修改喔。



接下来就点击刚刚做的 if 判断式,复制(Ctrl + C)并贴上(Ctrl + V),然 后再移动到原本的 if 判断式下接合,之后再重复做一次,所以总共会有三个 if 判断式,如下图。



我们先来修改比较式中右边的数字,第一个不用改,第二和第三改为2 和3,如下图。



接着将第二和第三判断式内的 CheckBox1, 改为 CheckBox2 及 CheckBox3, 如下图。



接下来最后中断联机的按钮,也就是 Button6,拖曳.Click 事件出来,然后再从左侧 BluetoothClient1 中拖曳.Dissconnect 出来接合,如下图。



除了中断联机之外还要隐藏按钮和停止 Clock,就直接复制刚开始做的 控件显示过去,再全部改成 false 就好了,如下图,做到这边就算完成了, 接下来可以将程序传到 Android 装置中。



会不会觉得很奇怪,为什么要用 Clock 去接收处理 EduCake 所发送的 讯息呢?因为我们需要实时的处理这些送过来的讯息,而必须隔一小段时间 扫描一次看看有无讯息,而为什么要先送出"a"再做接收呢,因为 EduCake 的速度实在是太快了,一直传讯息会造成蓝芽反应不及,这时候我们就用 Arduino 牵制它,当接收到"a"的时候再送信息过去就好了,这样可以确保 资料有确实的传递。

接下来我们将 EduCake 的 LED 从一个接到五个,再增加三个按钮,接脚图如下。



接着开启 IDE 接口,输入以下程序



86DUIND www.86duino.com



因为我们传输的是字符,我们要把它变成一半数字最简单的方法就是直接减去0的字符就好,至于减去1是因为数组刚刚定义5个实际是从0~4, 所以要减去1,举个例子来说,当我们在 Android 装置中按下 LED1,它就 会传送一个"1"的字符回来,我们用 if 判断数组中的第0个位置自否为低电 位,然后再做相应的动作。 接下来就是测试时间啦,联机之后,点击 LED1~5 会有相应的 LED 会 亮起,而按下 EduCake 上的按钮会反应在按钮 1~3 上。



