Welcome

CONTENTS

GETTING STARTED

Power on your LattePanda

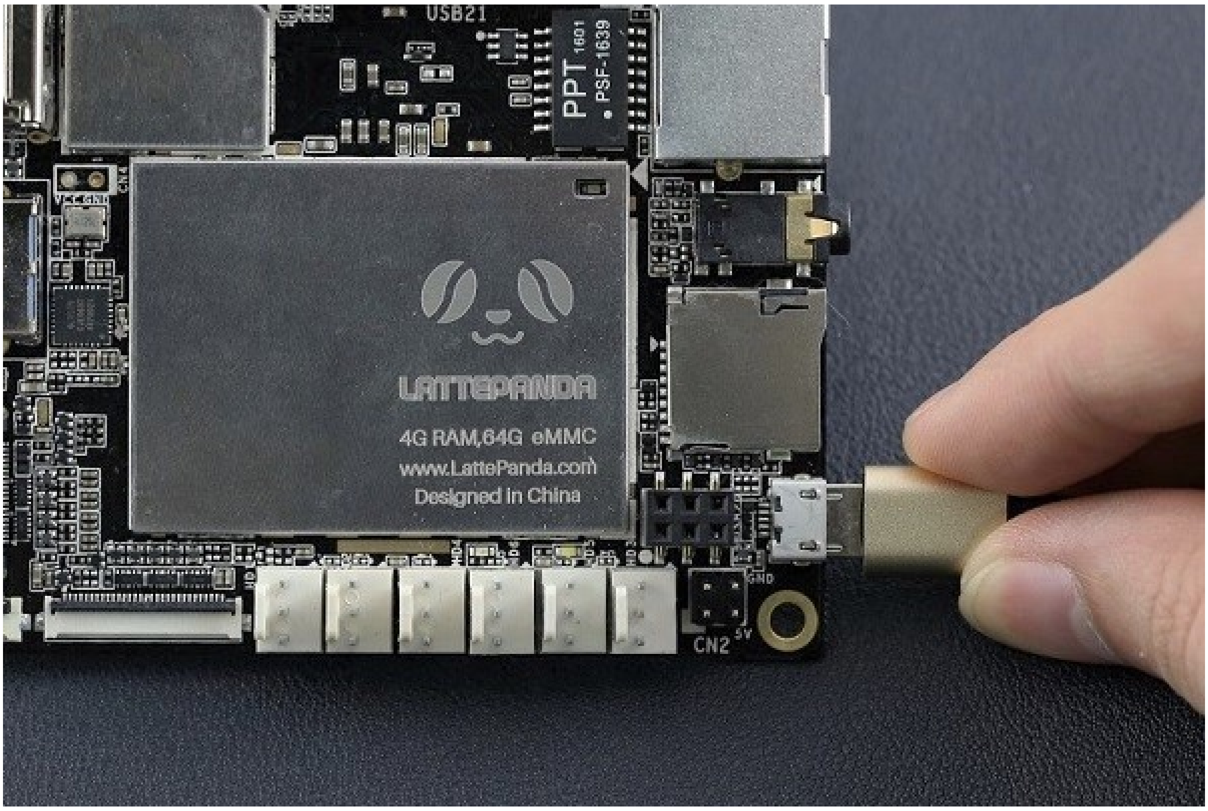
The LattePanda is powered through the micro USB port. Any standard USB adapter (such as a cell phone wall charger) with **at least 2A of current** can be used as a power supply for the LattePanda. A power adapter will not be bundled with the board, but you may get one from our online store or any electronics retailer.

Note: Check your power adapter and USB cable connection before use. Insufficient or unstable current may prevent your LattePanda from initializing.

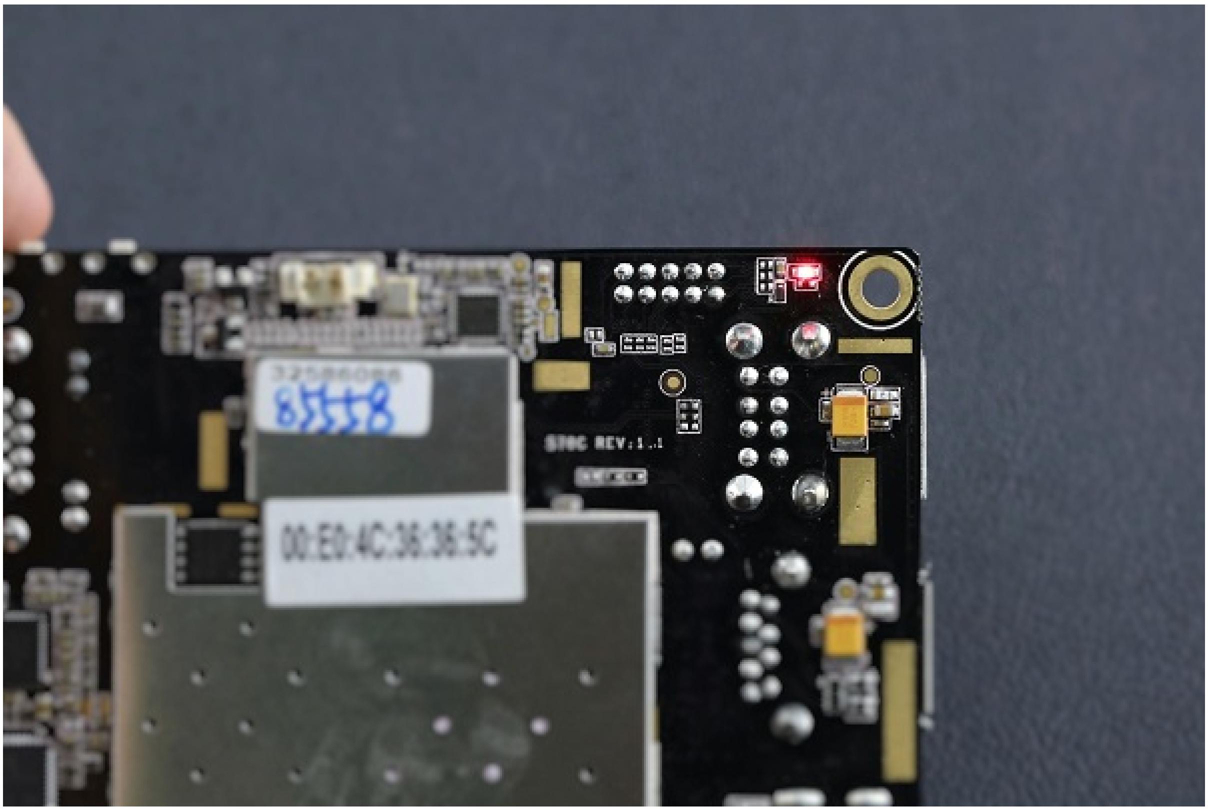
Here’s our list of recommended power adapters and cables

When you have an adapter and micro USB cable ready, follow the instructions below to get started with your LattePanda.

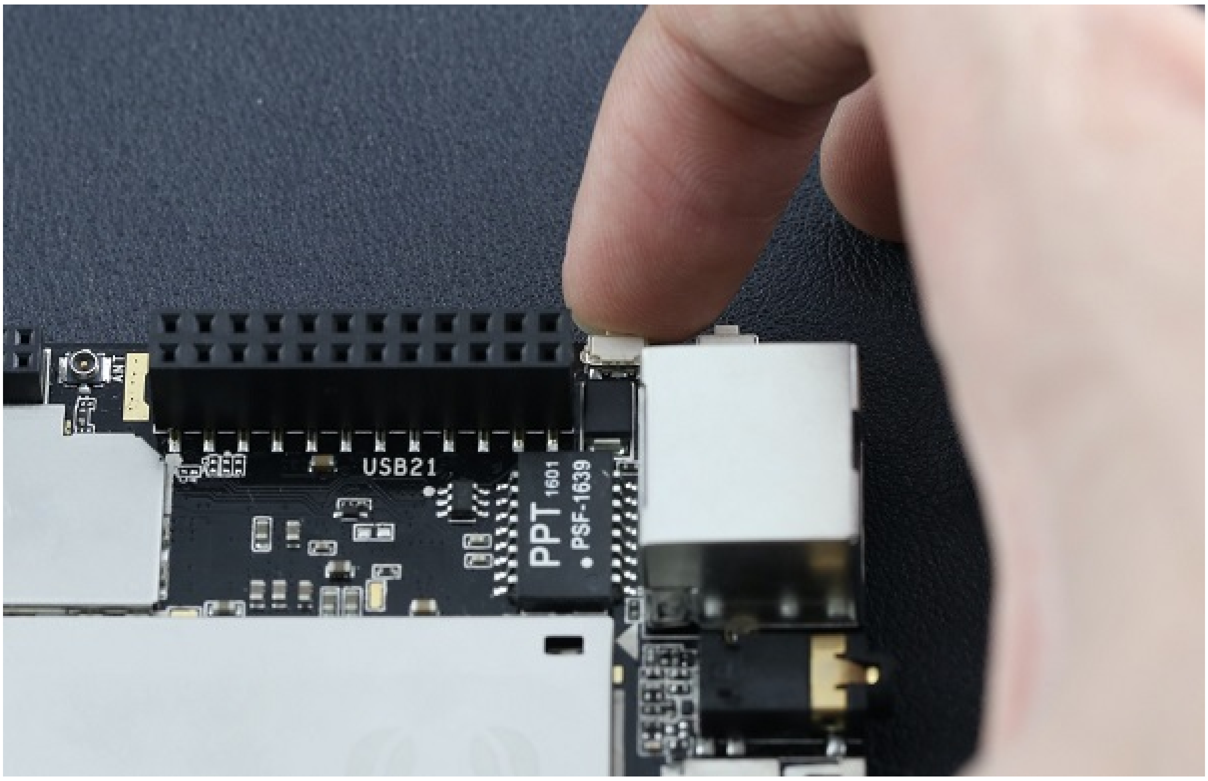
1.Plug the USB into the USB power adapter, and the microUSB into the micro USB port of the LattePanda (The micro USB port is located next to the SD card socket).
If you need IPS display and touch panel, please plug it first



2.When plugged in, you should see the red LED indicator lights up on the underside of the board. This means that the LattePanda is initializing. Wait patiently for a few seconds until the LED goes out.



3.When the LED turns off, press and hold the power button for one second to turn the LattePanda on. You should see the LED lights up again

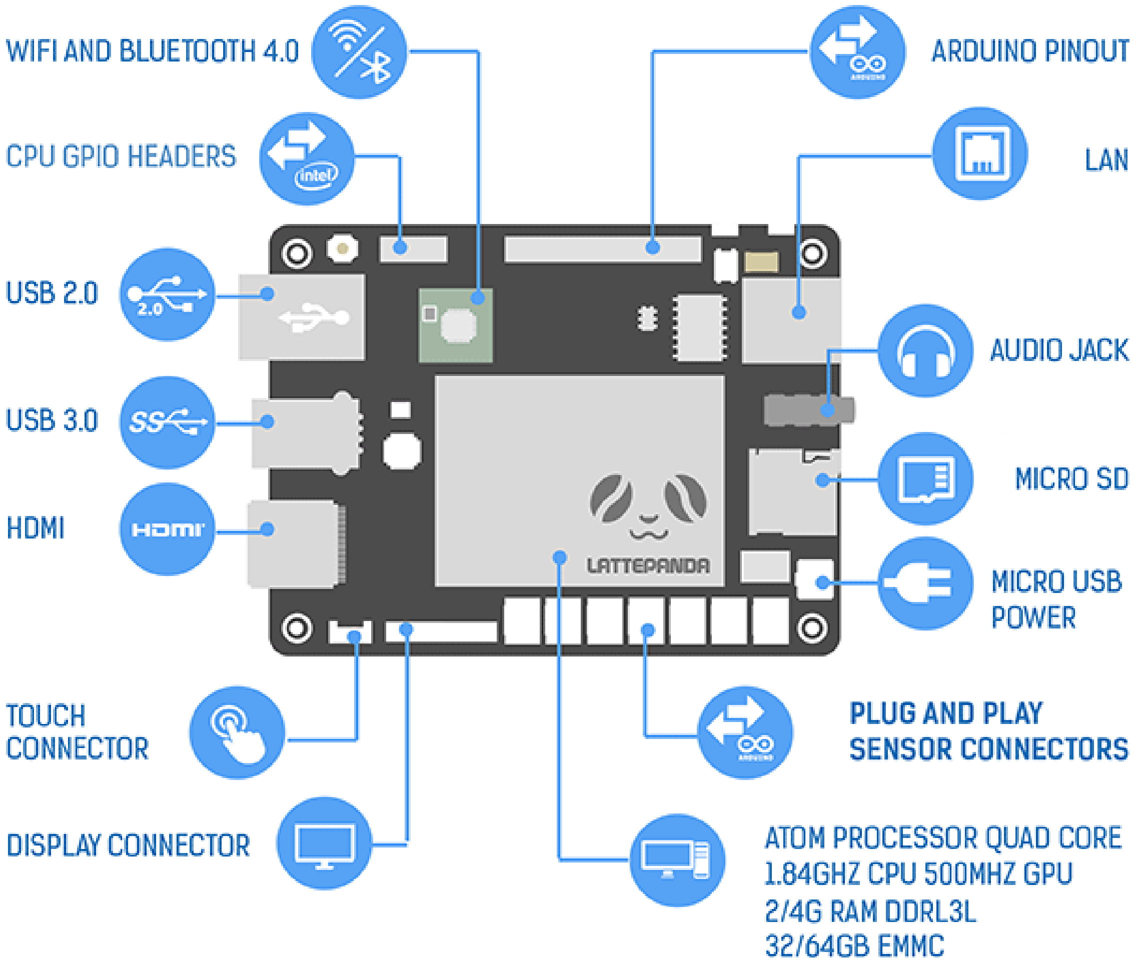


Connect with your peripheral device

LattePanda is compatible with a wide range of peripheral devices.

You may connect any USB supported devices, such flash drives, mouse and keyboard, or a webcam to USB 3.0 and 2.0 ports. The SD card socket supports extra storage from a mini SD card. You may also connect the LattePanda to an external speaker device through its 3.5mm audio jack.

The LattePanda’s Arduino compatible co-processor with plug and play headers and GPIO pins supports standard 5V sensors and actuators that enable it to interact with the physical world.



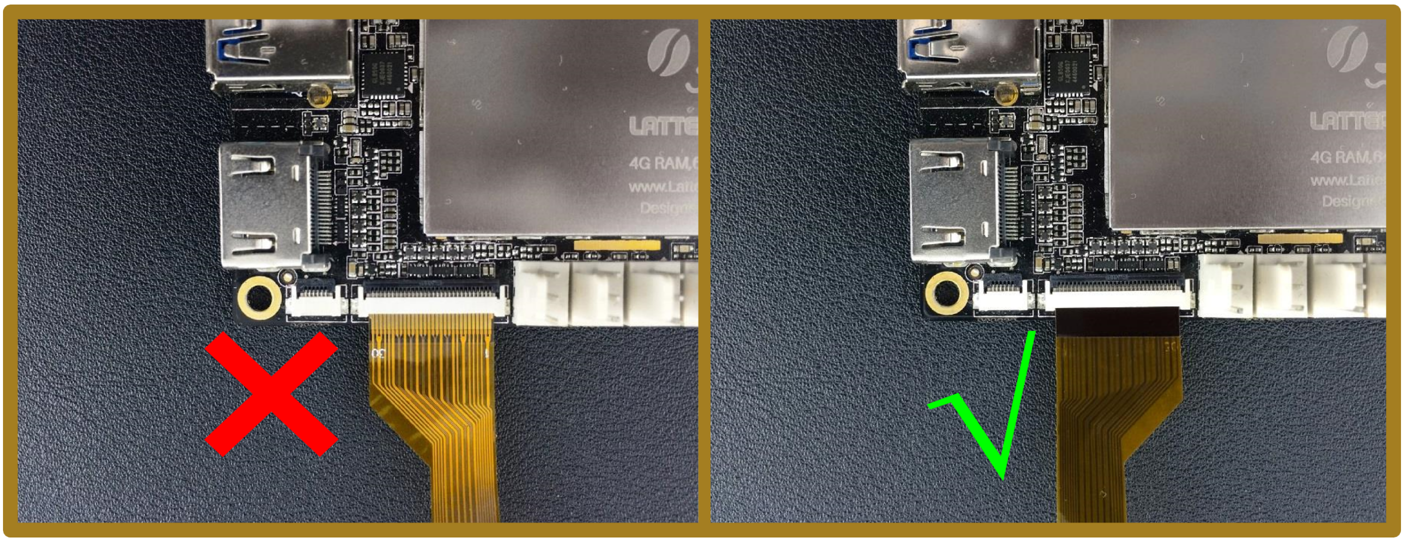
Connect to Wi-Fi

1.Install the Wi-Fi antenna by plugging the round shaped end into the socket labelled “ANT” located next to the GPIO pins on the board.

2.In Windows select a Wi-Fi connection by clicking the Wi-Fi icon in the system tray at the bottom right of the screen. Follow the wizard to setup a connection.

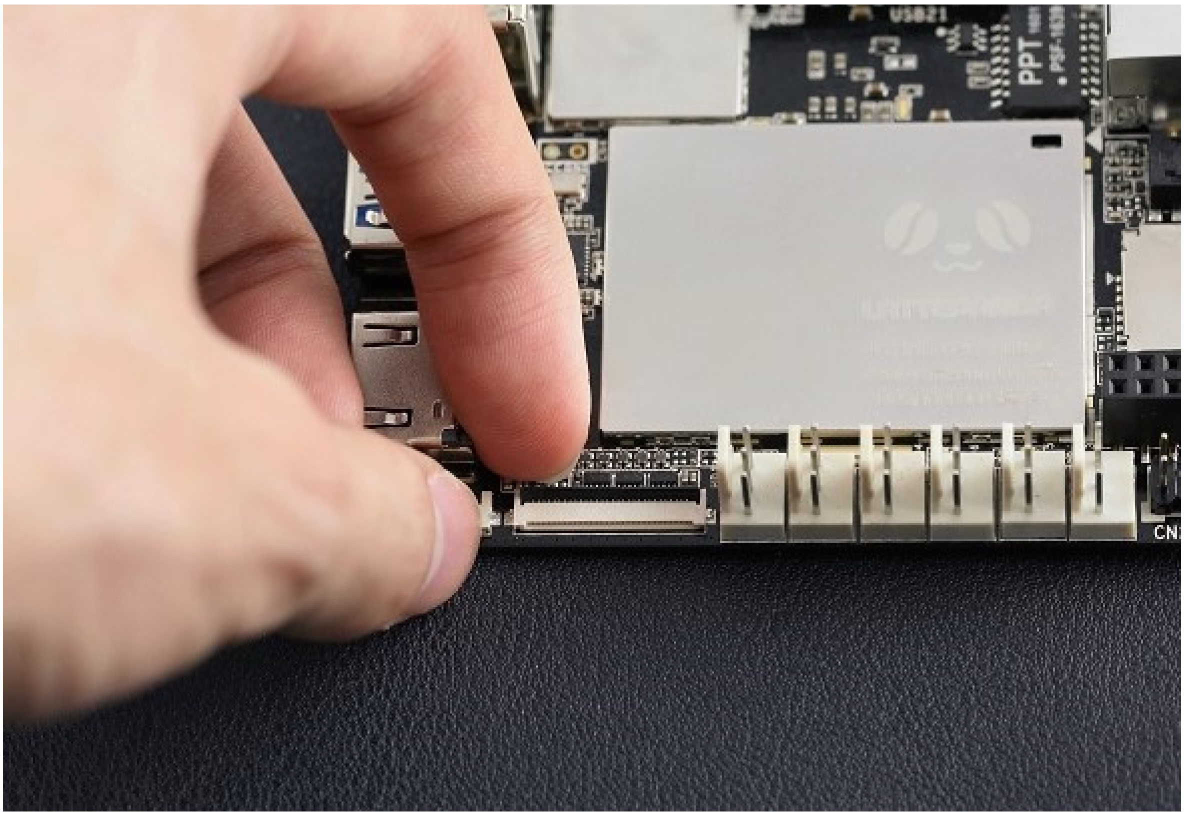
Note: A weak Wi-Fi signal may prevent a Wi-Fi connection being established. Check the antenna is connected properly to the board and make sure that a Wi-Fi network is available.

Connect with 7“ Display and Touch Panel Overlay



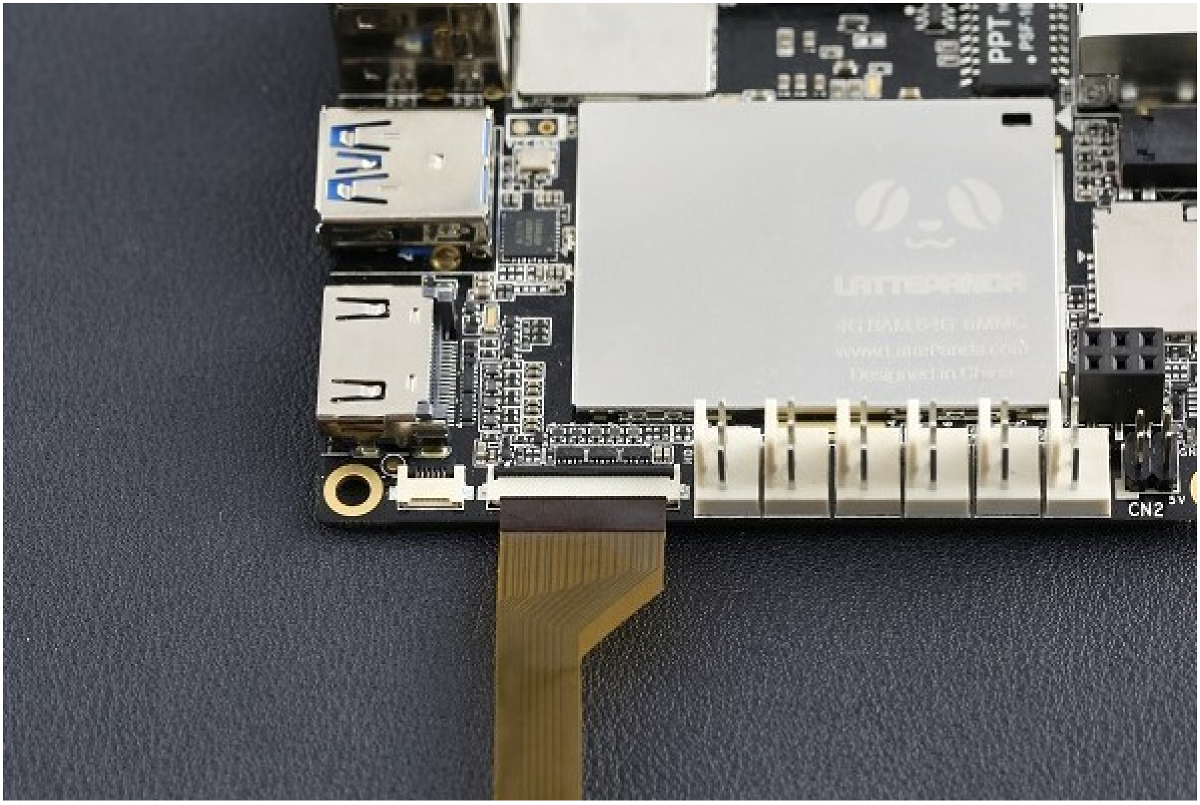
Note:Please connect it with LattePanda BEFORE power-on. And make sure the Golden Finger face the right side.
The contacts on FPC are so compact and in order. Please be careful that any dislocation connection may cause the LattePanda short circuit and the IPS display abnormal like ghosting or flicker.

1.Lift up the actuator. Use thumb or index finger might be easier.



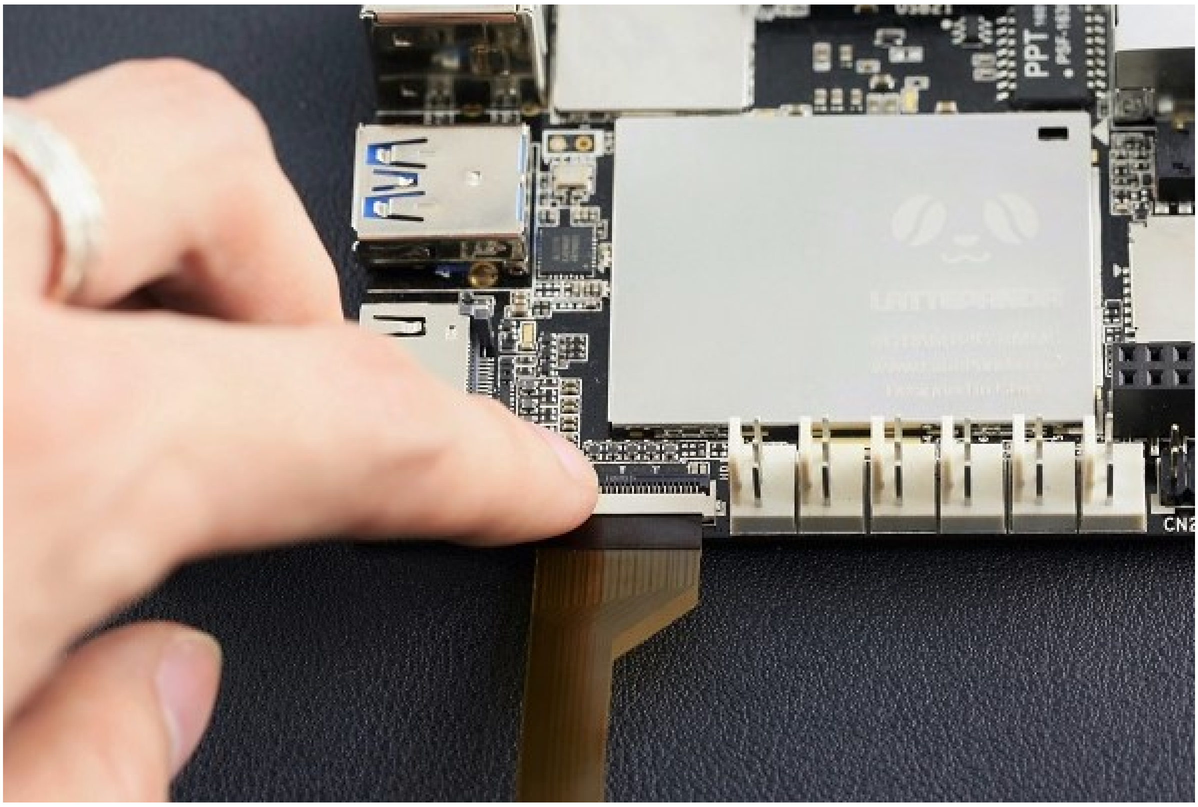
2.Insert display FPC in.

Place Golden Finger side down!



Note: The FPC must be fully inserted in the connector. If not fully inserted, the actuator will not close properly. Should this be the case, lift up the actuator and repeat the process (starting with Step 1 above)

3.Rotate down the actuator until firmly closed.



4.Insert the FPC of touch panel in
Same as the display. Place the Golden Finger side down too.

Software & Operating Systems

Setting up a VNC server on the LattePanda using TightVNC

INTRODUCTION

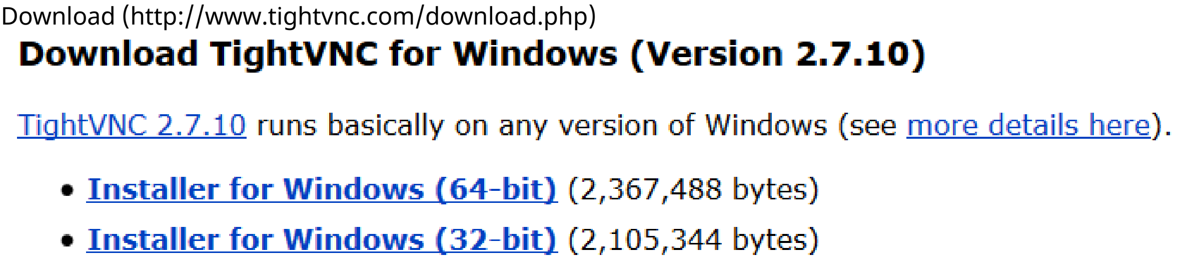
In this tutorial I will demonstrate how to install a VNC service on your LattePanda using Windows. This will enable you to access the GUI of Windows running on your LattePanda from a different PC on your local network. I will use TightVNC, a free and easy to set up service.
VNC stands for “Virtual Network Computing”. It is a way of transmitting the keyboard and mouse events of one computer to another - in other words using one computer to remote control another. This is useful because you might not have extra monitors, keyboards or mice lying around – using a VNC service enables you to access several computers on your local network using just one computer, monitor, keyboard and mouse. You might also have a headless server set up which doesn’t require constantly attached peripherals. Setting up a VNC server on your headless server is a handy way to interface with a GUI if and when you need it.

Let’s get started:

STEP 1 – INSTALLATION

1. Download and install TightVNC for Windows on your LattePanda. Choose 32-bit or 64-bit depending on your system architecture.
   (LattePanda Standard is 32-bit, LattePanda Enhanced is 64-bit)

Download (http://www.tightvnc.com/download.php)

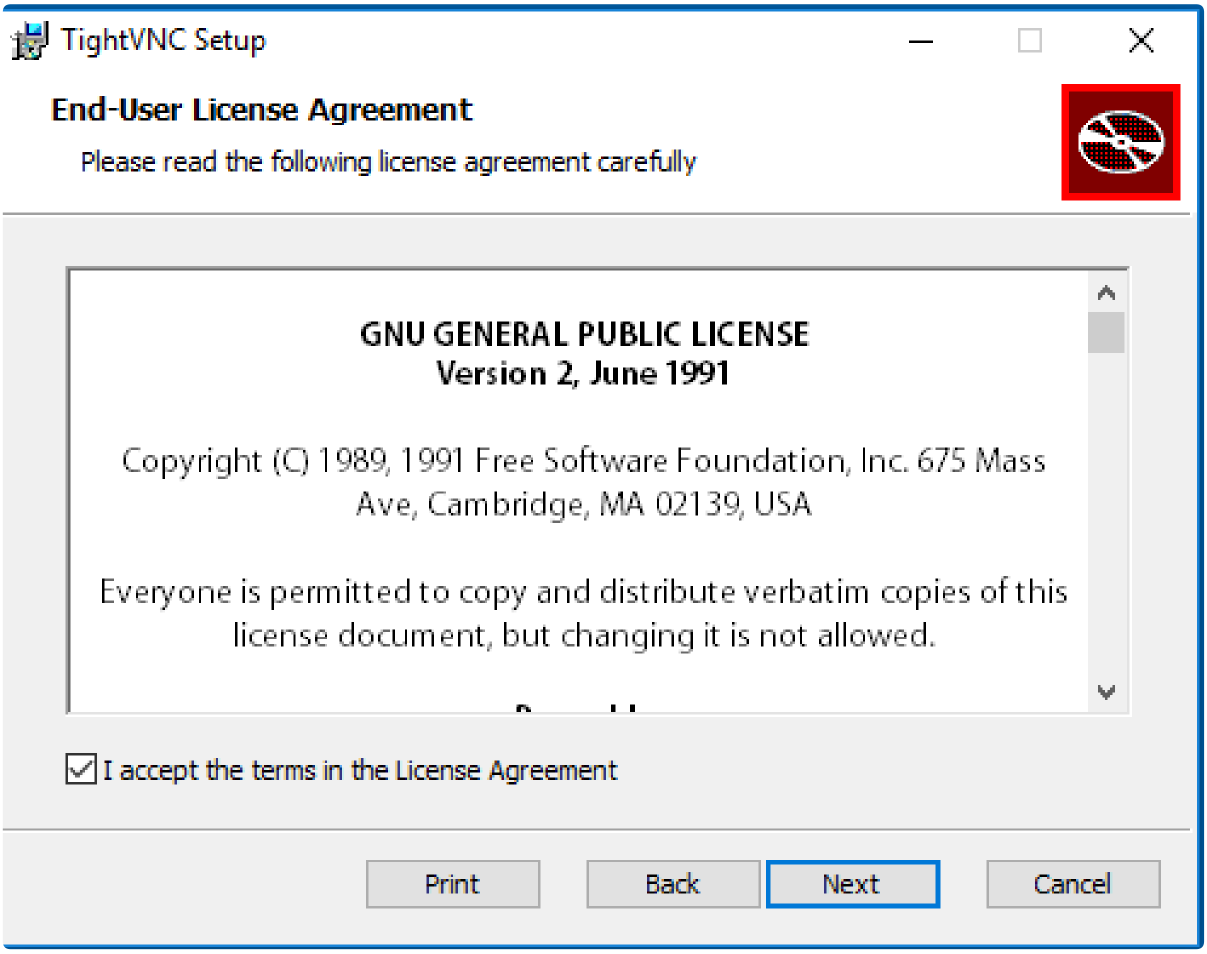


Download TightVNC for Windows (Version 2.7.10)

TightVNC 2.7.10 runs basically on any version of Windows (see more details here).

- **Installer for Windows (64-bit)** (2,367,488 bytes)
- **Installer for Windows (32-bit)** (2,105,344 bytes)

2. End-User Licence Agreement
   Accept the licence agreement and click next



End-User License Agreement

Please read the following license agreement carefully

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

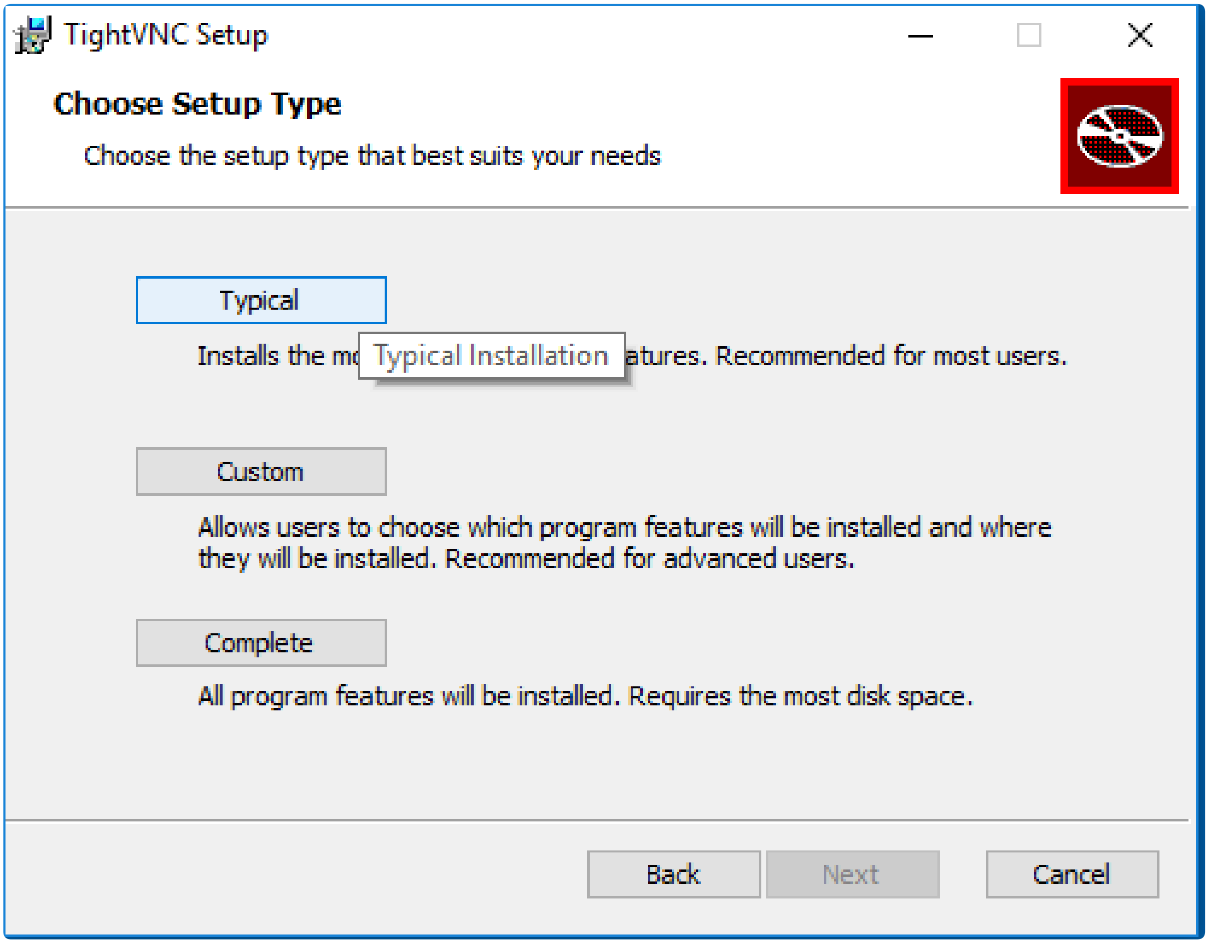
Copyright (C) 1989, 1991 Free Software Foundation, Inc. 675 Mass Ave, Cambridge, MA 02139, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

☑ I accept the terms in the License Agreement

3. Choose Setup Type
   Typical installation will install both TightVNC server and TightVNC viewer on your LattePanda
   Custom installation allows you to select which elements to install. Really all we need is the server, unless you would like

to be able to view other PCs on your network through the LattePanda, in which case you will need the viewer as well. For this tutorial we will just do the typical install.



TightVNC Setup

Choose Setup Type

Choose the setup type that best suits your needs

Typical

Installs the mo Typical Installation atures. Recommended for most users.

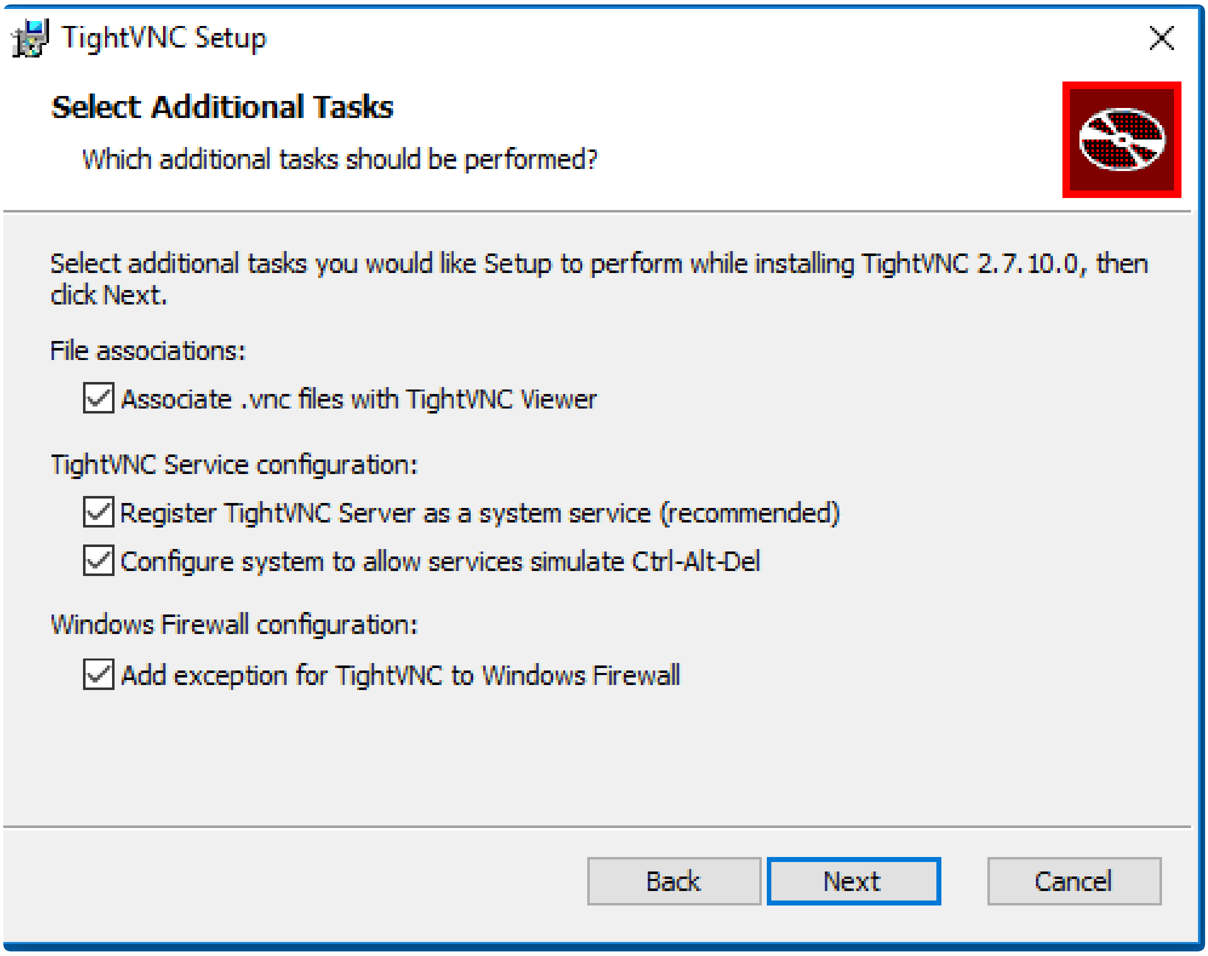
Custom

Allows users to choose which program features will be installed and where they will be installed. Recommended for advanced users.

Complete

All program features will be installed. Requires the most disk space.

4. Select Additional Tasks
   Check all the boxes



TightVNC Setup

Select Additional Tasks

Which additional tasks should be performed?

Select additional tasks you would like Setup to perform while installing TightVNC 2.7.10.0, then click Next.

File associations:

☑ Associate .vnc files with TightVNC Viewer

TightVNC Service configuration:

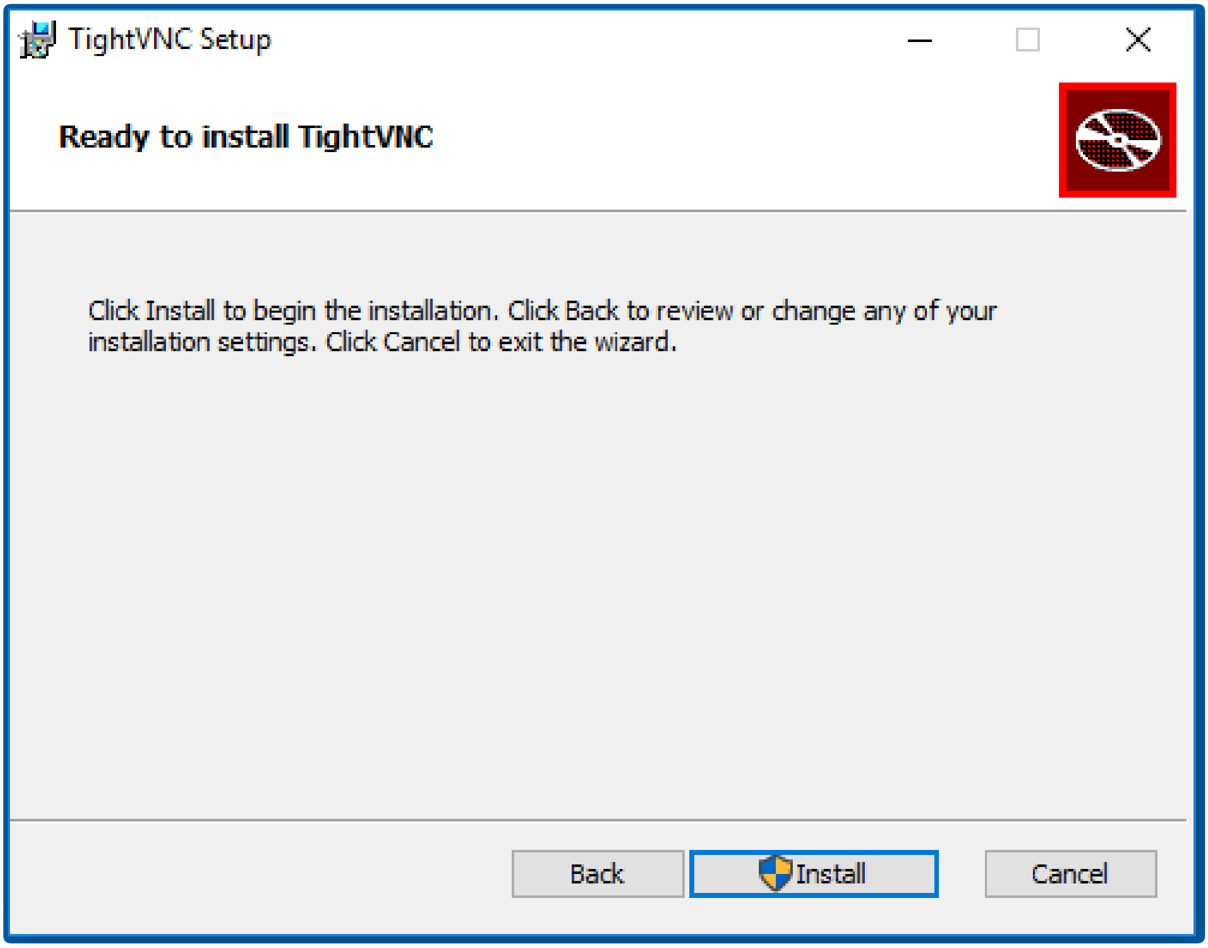
☑ Register TightVNC Server as a system service (recommended)

☑ Configure system to allow services simulate Ctrl-Alt-Del

Windows Firewall configuration:

☑ Add exception for TightVNC to Windows Firewall

5. Ready to Install TightVNC
   Click Install to begin!

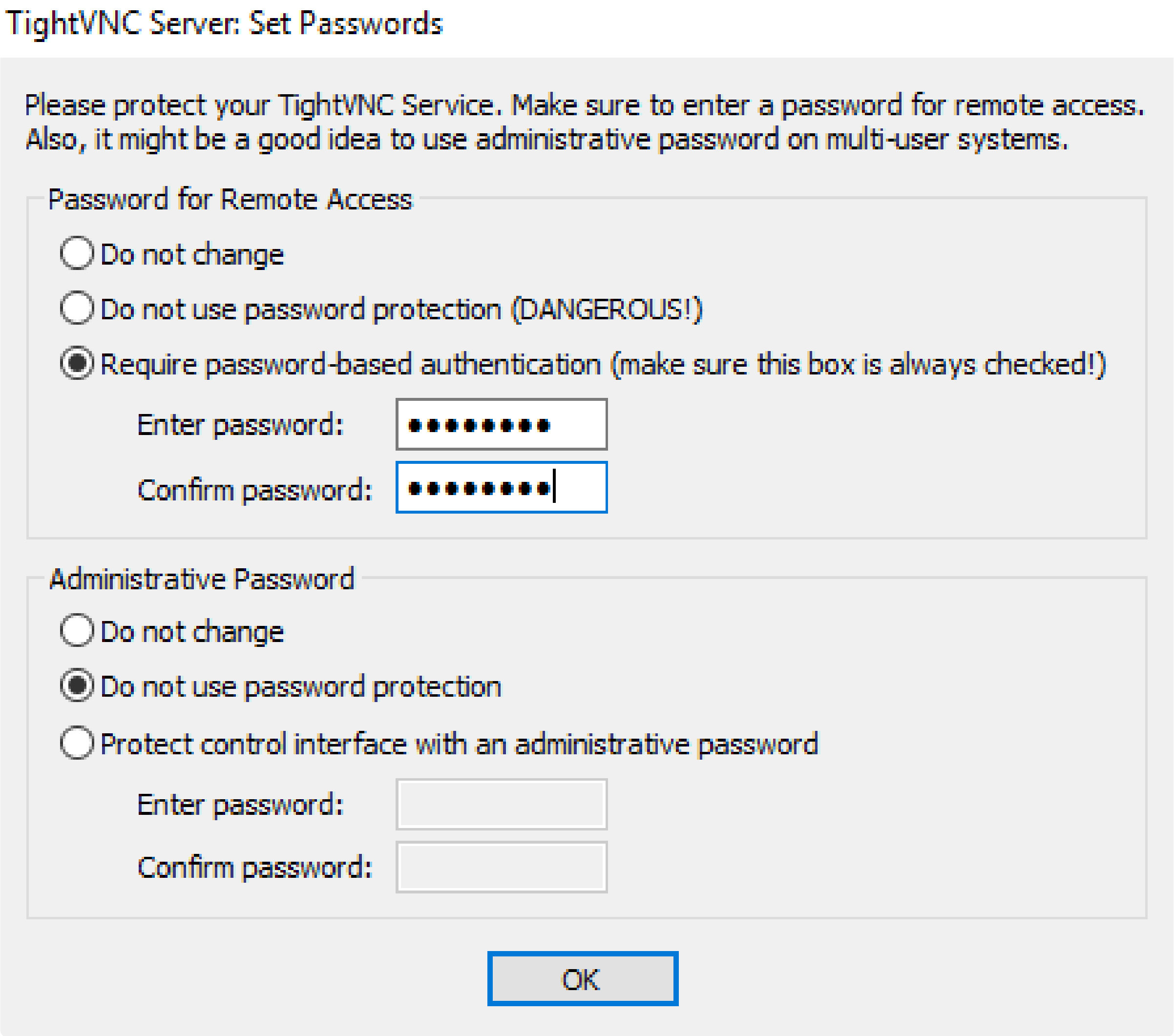


TightVNC Setup

Ready to install TightVNC

Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.

6. TightVNC Server: Set Passwords
   Password-based A



TightVNC Server: Set Passwords

Please protect your TightVNC Service. Make sure to enter a password for remote access. Also, it might be a good idea to use administrative password on multi-user systems.

Password for Remote Access

○ Do not change

○ Do not use password protection (DANGEROUS!)

● Require password-based authentication (make sure this box is always checked!)

Enter password: ●●●●●●●●

Confirm password: ●●●●●●●●

Administrative Password

○ Do not change

● Do not use password protection

○ Protect control interface with an administrative password

Enter password:

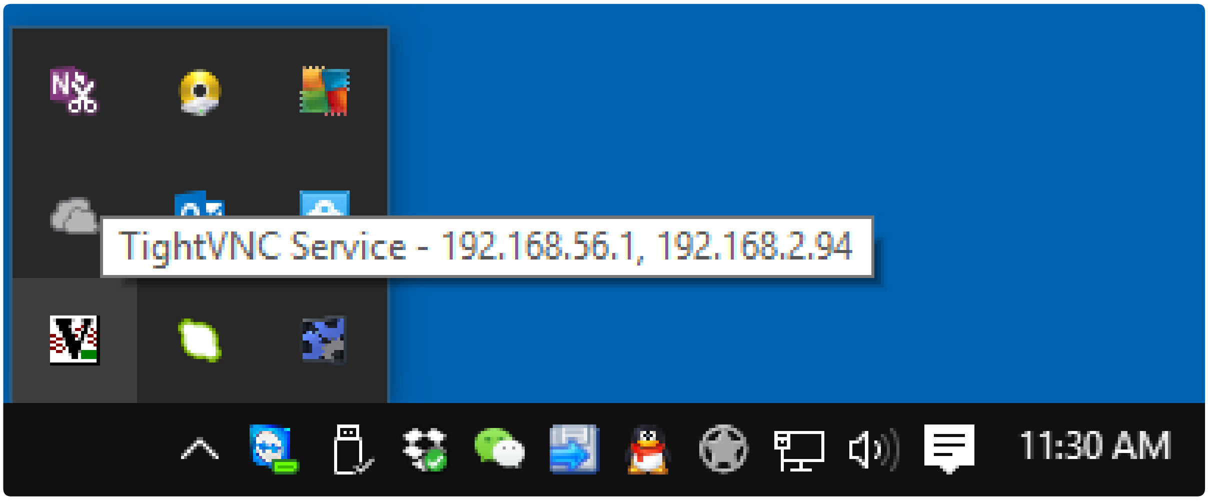
Confirm password:

   At this point it is wise to set a password for remote access. Point the radio button towards “Require password-based authentication” and choose a password. Retype it in the following box.
   I have set the password as “lattepan” (as the password can’t be longer than 8 characters)

7. Administrative Password
   This is not strictly necessary. In this tutorial I will not set an administrative password, but you may if you wish. If you set a password for this you will have to enter it before changing any configuration settings.
   When you are happy with your settings, click “OK”. Click “Finish” to exit the setup wizard.

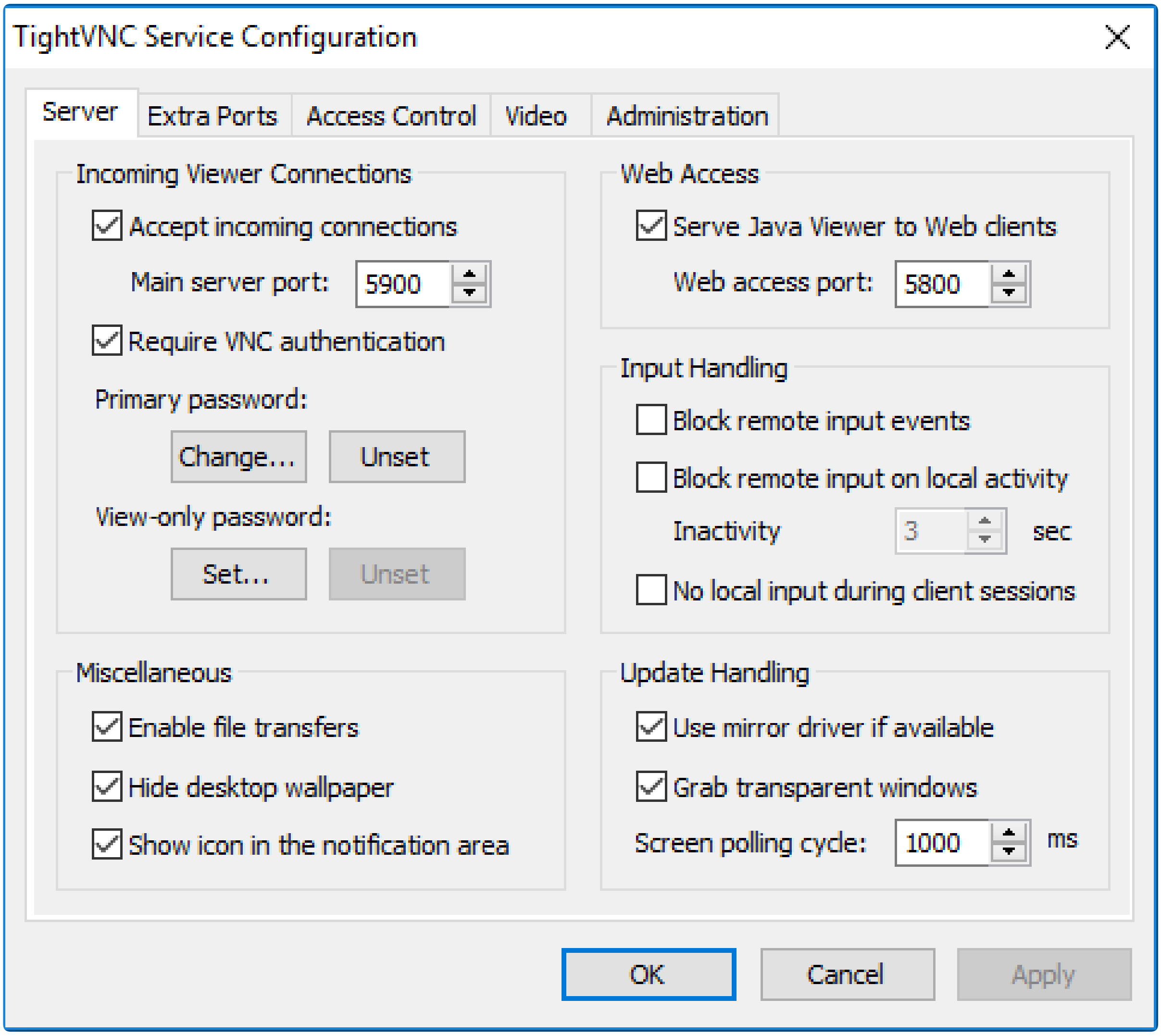
STEP 2 - CONFIGURATION

You should now see a new icon in your system tray. (If you don't, try logging out and logging back in to your PC).



TightVNC Service - 192.168.56.1, 192.168.2.94

Here you can see the IP address your PC is on.
Double click it to bring up the service configuration window. The default settings should be fine for our purposes.



TightVNC Service Configuration

Incoming Viewer Connections

☑ Accept incoming connections

Main server port: 5900

☑ Require VNC authentication

Primary password: Change... Unset

View-only password: Set... Unset

Web Access

☑ Serve Java Viewer to Web clients

Web access port: 5800

Input Handling

□ Block remote input events

□ Block remote input on local activity

Inactivity 3 sec

□ No local input during client sessions

Miscellaneous

☑ Enable file transfers

☑ Hide desktop wallpaper

☑ Show icon in the notification area

Update Handling

☑ Use mirror driver if available

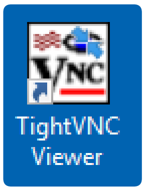
☑ Grab transparent windows

Screen polling cycle: 1000 ms

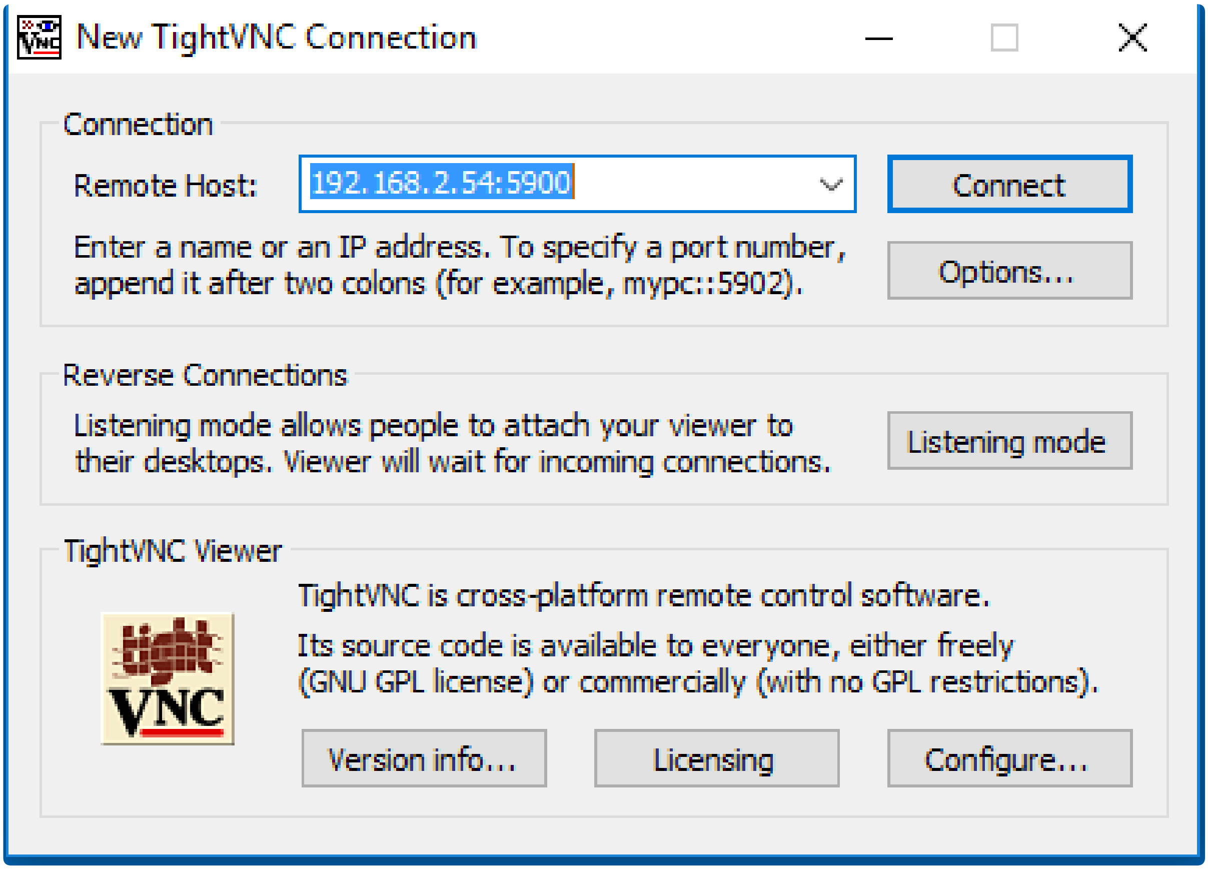
Next, you will need to go on to the computer you would like to control the LattePanda with and using the same installation package, install TightVNC viewer.

When you have successfully installed TightVNC viewer, check that you are on the same network as your LattePanda, otherwise it will not work!
When you are certain that your computer and the LattePanda are on the same local network, proceed to step 3

STEP 3 - TESTING



Open TightVNC Viewer. A window will appear for a new TightVNC Connection. At this point, you need to input the IP address of your LattePanda.


New TightVNC Connection

Connection

Remote Host: 192.168.2.54:5900

Enter a name or an IP address. To specify a port number, append it after two colons (for example, mypc::5902).

Reverse Connections

Listening mode allows people to attach your viewer to their desktops. Viewer will wait for incoming connections.

TightVNC Viewer

TightVNC is cross-platform remote control software.

Its source code is available to everyone, either freely (GNU GPL license) or commercially (with no GPL restrictions).

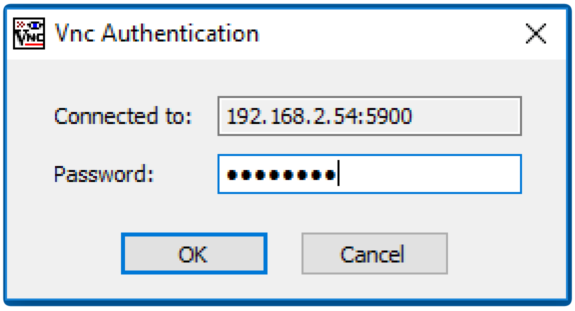
Tip: A quick way of finding this is if you hover over the system tray TightVNC icon on your LattePanda. A hint will pop up with “TightVNC Service - ” You could also go in to your router control interface and look for attached devices.

The next step is to input this IP address in the New TightVNC Connection Window, followed by the port number you set in the service settings. The default is port 5900.

```
<ip address of LattePanda>:<port number>
```

```
e.g. 192.168.2.60:5900
```

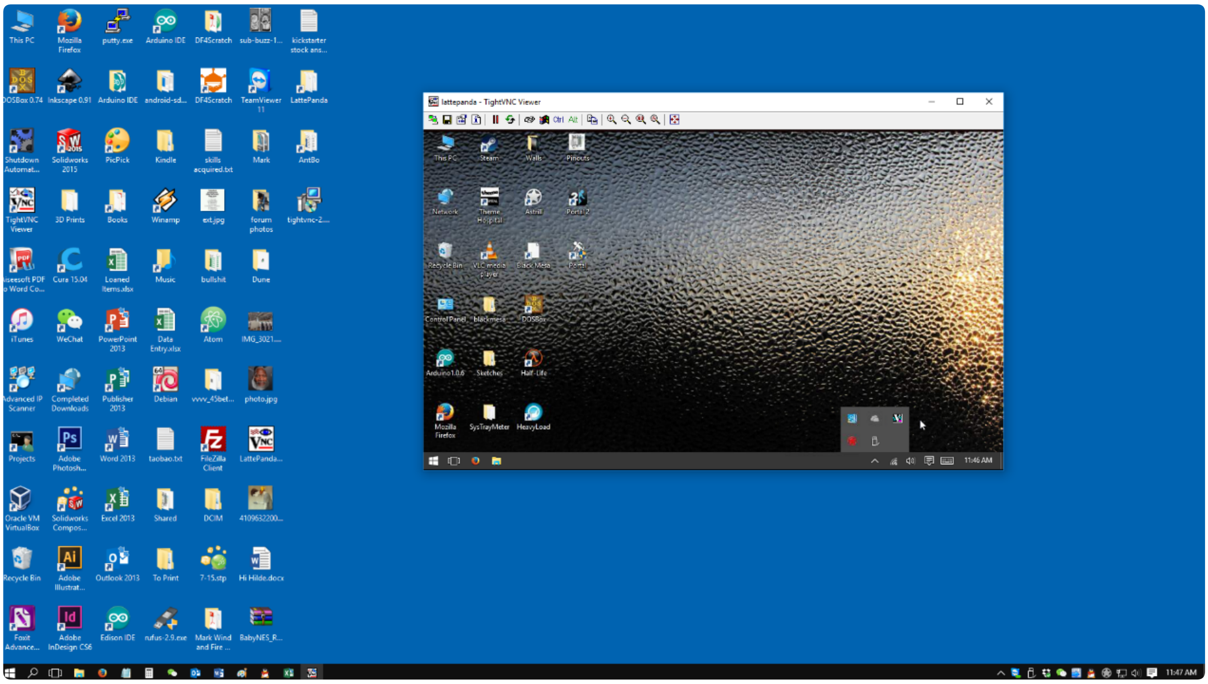
Click connect. If all goes well you will be prompted with a password input. Input the password you set earlier. My password was “lattepan”. Press enter.


Vnc Authentication

Connected to: 192.168.2.54:5900

Password: ••••••••

You will be greeted by a window containing your LattePanda’s GUI! You may now control it remotely!

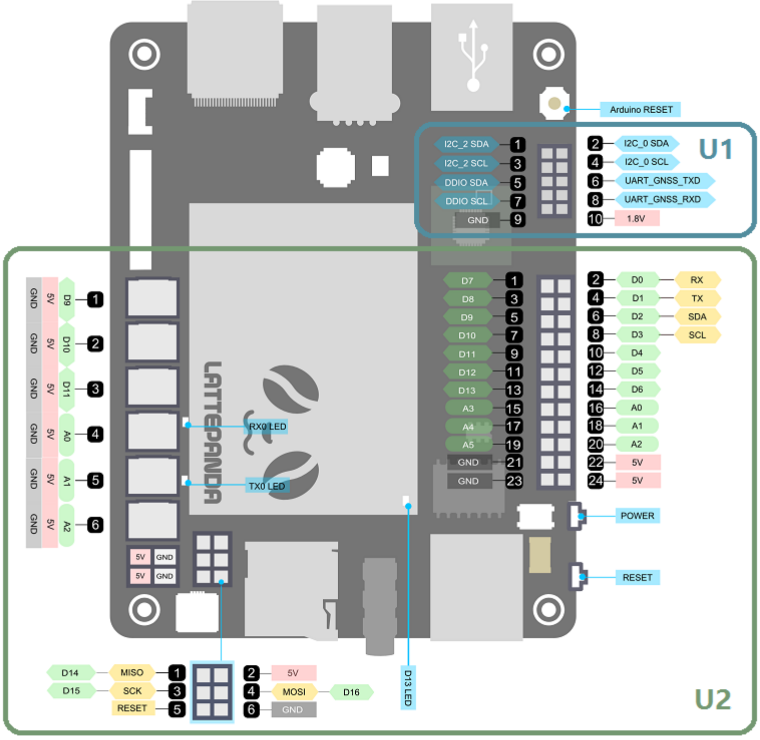


This concludes the LattePanda VNC tutorial. If you have any questions or comments please let us know in the forum. I hope this has been of help to you.

Hardware & Accessories

Inputs and Outputs

In this article we will discuss the images of the Lattepanda. Below is a basic diagram that displays all the pins:



Pinouts in area U1 are assigned to the X-Z8300 core.
Pinouts in area U2 are assigned to the ATmega32u4 core.

Each of the 20 digital pins (A0 - A5, D0 - D13) in area U2 can be used as an input or output, each operating at 5 volts. Each pin can output or receive 40 mA and each has an internal pull-up resistor (disconnected by default) of 20-50k ohm.
Caution: Exceeding 40mA on any I/O pin may cause permanent damage to the ATmega32u4.

Some pins have specialized functions:
Analog Inputs: A0 - A5, A6 - A11 (on D4, D6, D8, D9, D10, and D12). The LattePanda has 12 analog inputs, labeled A0 through A11, all of which can also be used as digital I/O. Each pin has a 10 bit resolution (i.e. 1024 different values). By default they measure from ground to 5 volts.
Serial: D0 (RX) and D1 (TX). Used to receive (RX) and transmit (TX) TTL serial data.
External Interrupts: D3 (interrupt 0), D2 (interrupt 1), D0 (interrupt 2), D1 (interrupt 3) and D7 (interrupt 4). These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
PWM: D3, D5, D6, D9, D10, and D13 provide 8-bit PWM output.
SPI: D16 (MOSI), D14 (MISO), D15 (SCK).
LED: D13 There is a built-in LED driven by digital pin 13. When the pin's val
TWI: D2(SDA), D3(SCL).

Other pins on the board:
Reset:
Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

How to access pinouts from Visual Studio

LATTEPANDA.FIRMATA

LattePanda.Firmata (https://github.com/LattePandaTeam/LattePanda-Development-Support/tree/master/LattePandaFirmata) is an open-source Firmata library provided by LattePanda, which is suitable for Windows apps developed in Visual Studio. This class allows you to control Arduino GPIO from Windows apps, with features including:
● Reading and writing to digital pins
● Reading analog inputs
● Controlling servo motors
● Sending data to devices and receiving data form devices through the I2C Bus

3 STEPS TO YOUR REMOTE ARDUINO PROJECT

1. Set up your PC
2. Set up the Arduino(It is pre-installed, unless you changed the Arduino program)
3. Create a project or use the sample project (http://www.lattepanda.com/wp-content/uploads/2016/02/blinkYourBoard.zip)

FUNCTIONALITY

CONSTRUCTOR

- `public Arduino();`
- `public Arduino(string serialPortName, Int32 baudRate, bool autoStart, int delay);`
- `public Arduino(string serialPortName);`
- `public Arduino(string serialPortName, Int32 baudRate);`
  There are four different constructors, you can choose the proper one depending on your needs
  Parameters
  `serialPortName:`Specify serial port name which you can find in the Device Manager
  `baudRate:` It specifies the serial port speed, the value must be the same as Arduino Firmata. The default value is 57600.

`autoStart:` Connect the Arduino automatically when it is true. Default value: True
`delay:` Set the maximum interval in seconds of serial connection timeout. Default time: 8 Seconds

CONFIGURATION

- public void pinMode (int pin, byte mode);
  Sets the mode of the specified pin as you wish
  Parameters
  `pin:` the number of the pin whose mode you wish to set
  `mode:` `Arduino.OUTPUT`, `Arduino.INPUT`, `Arduino.PWM`, `Arduino.SERVO`
  Returns
  None

DIGITAL

- public void digitalWrite (int pin, byte value);
  Write to a digital pin that has been toggled to output mode with pinMode() method
  Parameters
  `pin:` The digital pin to write to
  `value:` `Arduino.HIGH`, `Arduino.LOW`
  Returns
  None
- public int digitalRead(int pin);
  Returns the last known state of the digital pin
  Parameters
  `pin:` The Arduino digital input pin
  Returns
  `Arduino.HIGH` or `Arduino.LOW`
- public event DigitalPinUpdated digitalPinUpdated;
  Call this Callback function when the digital value of the pin whose mode has been set as `Arduino.INPUT` update
  Delegate
  `public delegate void DigitalPinUpdated(byte pin, byte state);`
  Parameters
  `pin:` the pin whose value will update
  `state:` `Arduino.HIGH`, `Arduino.LOW`
  Returns
  None

ANALOG

- `public void analogWrite(int pin, int value);`
  Write to an analog pin using Pulse-width modulation (PWM)
  Parameters
  `pin:` Analog output pin
  `value:` PWM frequency from 0 (always off) to 255 (always on)
  Returns
  None
- public int analogRead(int pin);
  Returns the last known value of the analog pin
  Parameters
  `pin:` The Arduino analog input pin(0~5)
  Returns
  `int:`A value representing the analog value between 0 (0V) and 1023 (5V)
- `public event AnalogPinUpdated analogPinUpdated;`
  Call this function when the analog value of pin whose mode has been set as `Arduino.INPUT` update
  Delegate
- `public delegate void AnalogPinUpdated(int pin, int value);`
  Parameters
  `pin:` the pin whose value update
  `value:` A value representing the analog value between 0 (0V) and 1023 (5V)

Returns
None

SERVO

- `public void servoWrite(int pin, int angle);` Write the angle to specified pin which has been set the mode as Servo
  Parameters
  `pin:` Servo output pin.
  `value:` Angle from 0 to 180.
  Returns
  None

I2C/TWOWIRE

- `public void wireBegin(Int16 delay);`
  Initiate the Wire library and join the I2C bus as a master. This should normally be called only once
  Parameters
  `delay:` the number of milliseconds to pause (ms).
  Returns
  None
- `public void wireRequest(byte slaveAddress,Int16 slaveRegister, Int16[] data,byte mode);`
  Request I2C devices to send or receive data, then call `didI2CDataReveive` event to handle data when receive data.
  Parameters

  `slaveAddress:` the 7-bit address of the device to request bytes from
  `slaveRegister:` Specify the register that you want to store data or read data, choose `Arduino.NONE` if there is empty.
  `data`:

  1. when set the Parameter `mode` as `Arduino.I2C_MODE_WRITE` ,the data such as `{0x00,0x01}` will be sending to devices through I2C BUS.
  2. when set the Parameter `mode` as`Arduino.I2C_MODE_READ_ONCE` or `Arduino.I2C_MODE_READ_CONTINUOUSLY` , data specify the length of received data.

  `mode`:

  1. `Arduino.I2C_MODE_WRITE` :Send data
  2. `Arduino.I2C_MODE_READ_ONCE` : Receive data once
  3. `Arduino.I2C_MODE_READ_CONTINUOUSLY` :Receive data continuous
  4. `Arduino.I2C_MODE_STOP_READING` :Stop receiving data

Returns
None

- `public event DidI2CDataReveive didI2CDataReveive;`
  Call this event when receive I2C data.
  Delegate
  `public delegate void DidI2CDataReveive(byte address ,byte register, byte[] data);`
  Parameters
  `address:` : Specify slave computer address
  `register:`: Specify register address
  `data:` : Data returned
  Returns
  None

EXAMPLES

DIGITALWRITE

In this example, we will blink the LED which is connected with digital pin (D0 - D13)

API Required :

1. public Arduino();
2. public void pinMode(int pin, byte mode);

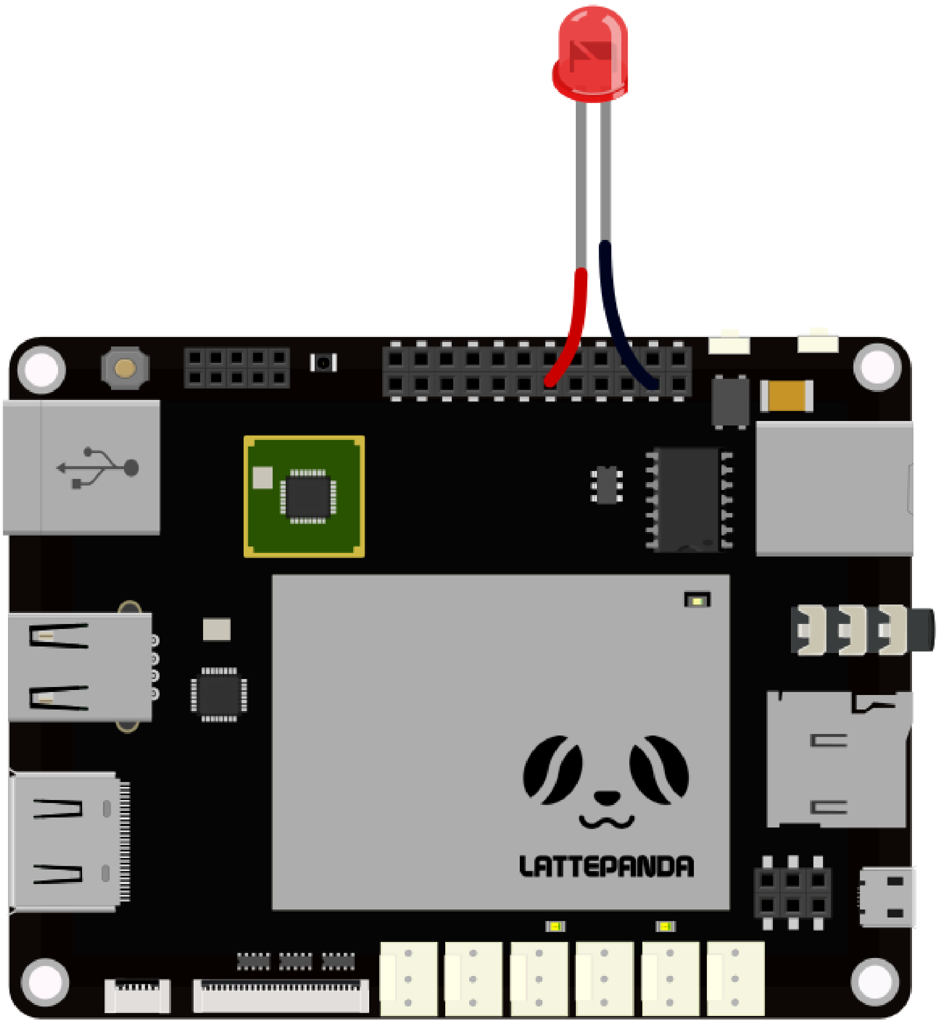
3. public void digitalWrite(int pin, byte value);

Hardware Required:

1. LattePanda x 1
2. led x 1 (or you can use the LED attached to pin 13 on the Arduino board itself)

Circuit:

1. LED inserted directly into pin 13



Code:

1. Create a new project in Visual Studio, Refer to Create a project
2. Main function code :

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Threading;
using LattePanda.Firmata;
namespace blinkYourBoard//project name
{
class Program
{
static Arduino arduino = new Arduino();//create an instance and initialize with the default parameters
static void Main(string[] args)
{
arduino.pinMode(13, Arduino.OUTPUT);//Set the digital pin 13 as output
while (true)
{
// ==== set the led on or off
arduino.digitalWrite(13, Arduino.HIGH);//set the LED  on
Thread.Sleep(1000);//delay a seconds
arduino.digitalWrite(13, Arduino.LOW);//set the LED  off
Thread.Sleep(1000);//delay a seconds
}
}
}
}
```

Test :

1. Click Debug button to execute, the LED will start blinking.

DIGITALREAD

This example detects the Button state through digital pin (D0-D13).

API required:

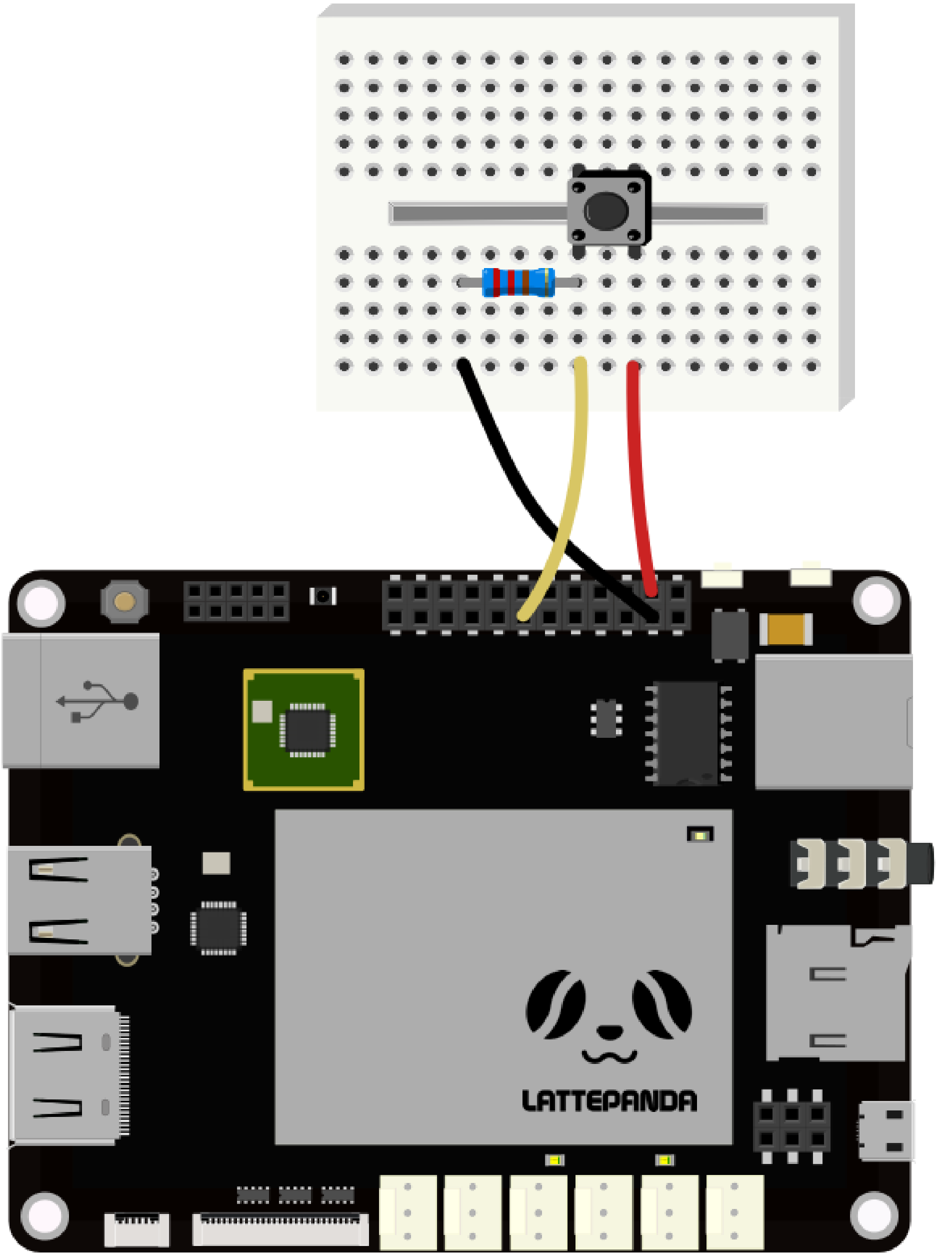
1. public Arduino();
2. public void pinMode(int pin, byte mode);
3. public int digitalRead(int pin);

Hardware Required:

1. LattePanda x 1
2. Button x 1
3. Resistor (Resistance value greater than 1KΩ) x 1

Circuit:

1. Connect button to pin 12 as following figure shows



Code:

1. Create a new project in Visual Studio, refer to Create a project
2. Main function code

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Threading;
using LattePanda.Firmata;
namespace buttonDemo//your project name
{
class Program
{
```

```
static Arduino arduino = new Arduino();//create an instance and initialize with the default parameters
static void Main(string[] args)
{
arduino.pinMode(12, Arduino.INPUT);// Set the digital pin 12 as input
int Value = arduino.digitalRead(12);// Read the state of pin 12 once.
Console.WriteLine(Value);
arduino.digitalPinUpdated += Arduino_digitalPinUpdated;//
Add Event Listeners and call it when the digital input update.
}
private static void Arduino_digitalPinUpdated(byte pin, byte state)
{
Console.WriteLine(pin);
Console.WriteLine(state);
}
}
}
```

Test:

1. Click Debug to execute, then the screen will print the value when you push down or release the button

PWM

This example assigns a pulse width modulation (PWM) value to an output pin (D3, D5, D6, D9, D10, D11) to dim or brighten an LED

API Required:

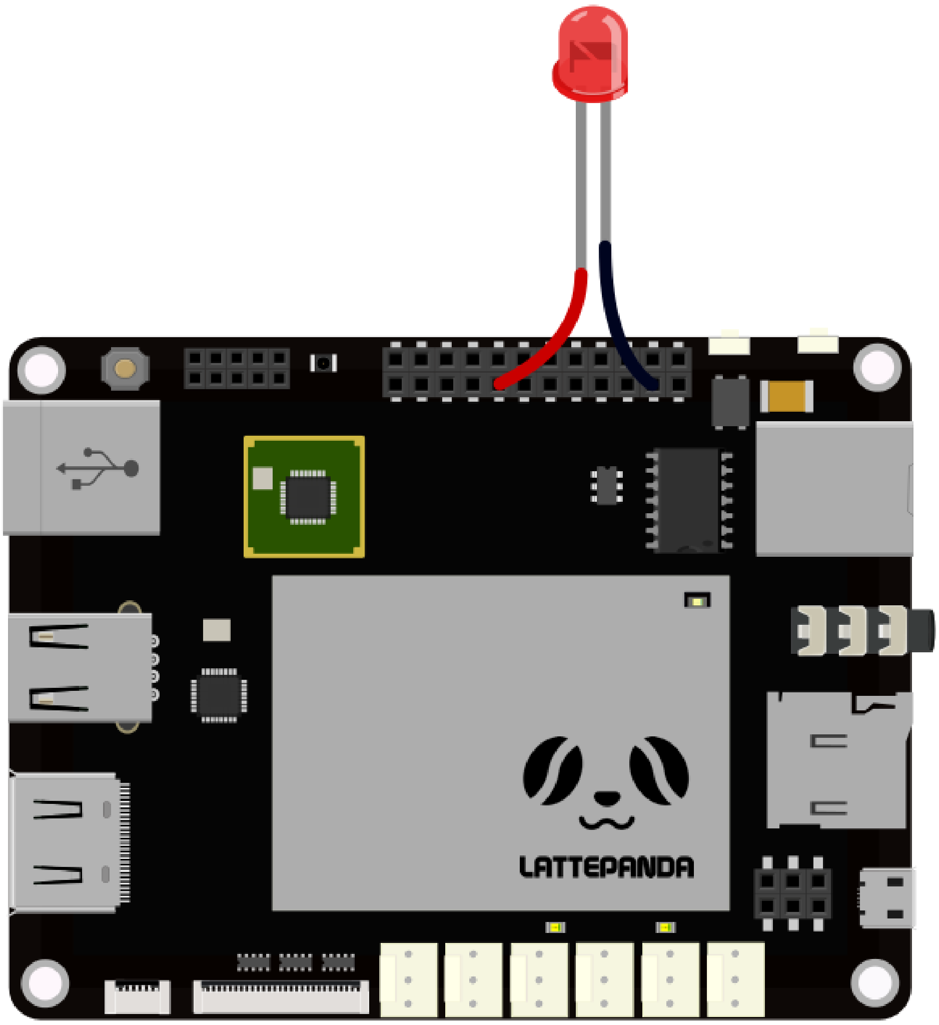
1. public Arduino();
2. public void pinMode(int pin, byte mode);
3. public void analogWrite(int pin, int state);

Hardware Required:

1. LattePanda x 1
2. LED x 1

Circuit:

1. LED connected directly into pin 11 as following figure shows



Code:

1. Create a new project in Visual Studio, refer to Create a project
2. Main function code

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Threading;
using LattePanda.Firmata;
namespace analogWriteExample
{
class Program
{
static Arduino arduino = new Arduino();//create an instance and initialize with the default parameters
static void Main(string[] args)
{
arduino.pinMode(11, Arduino.PWM);
while (true)
{
```

```
for (int i = 0; i <= 255; i++)
{
arduino.analogWrite(11, i);
Thread.Sleep(4);//delay 4ms
}
for (int i = 255; i >= 0; i--)
{
arduino.analogWrite(11, i);
Thread.Sleep(4);//delay 4ms
}
}
}
}
}
```

Test:

1. Click Debug to execute, you will find the LED brightness vary form dim to bright and then back again.

ANALOGREAD

This example detect the value of analog pin (A0-A5) where a potentiometer is connected, and then print the value

API Required:

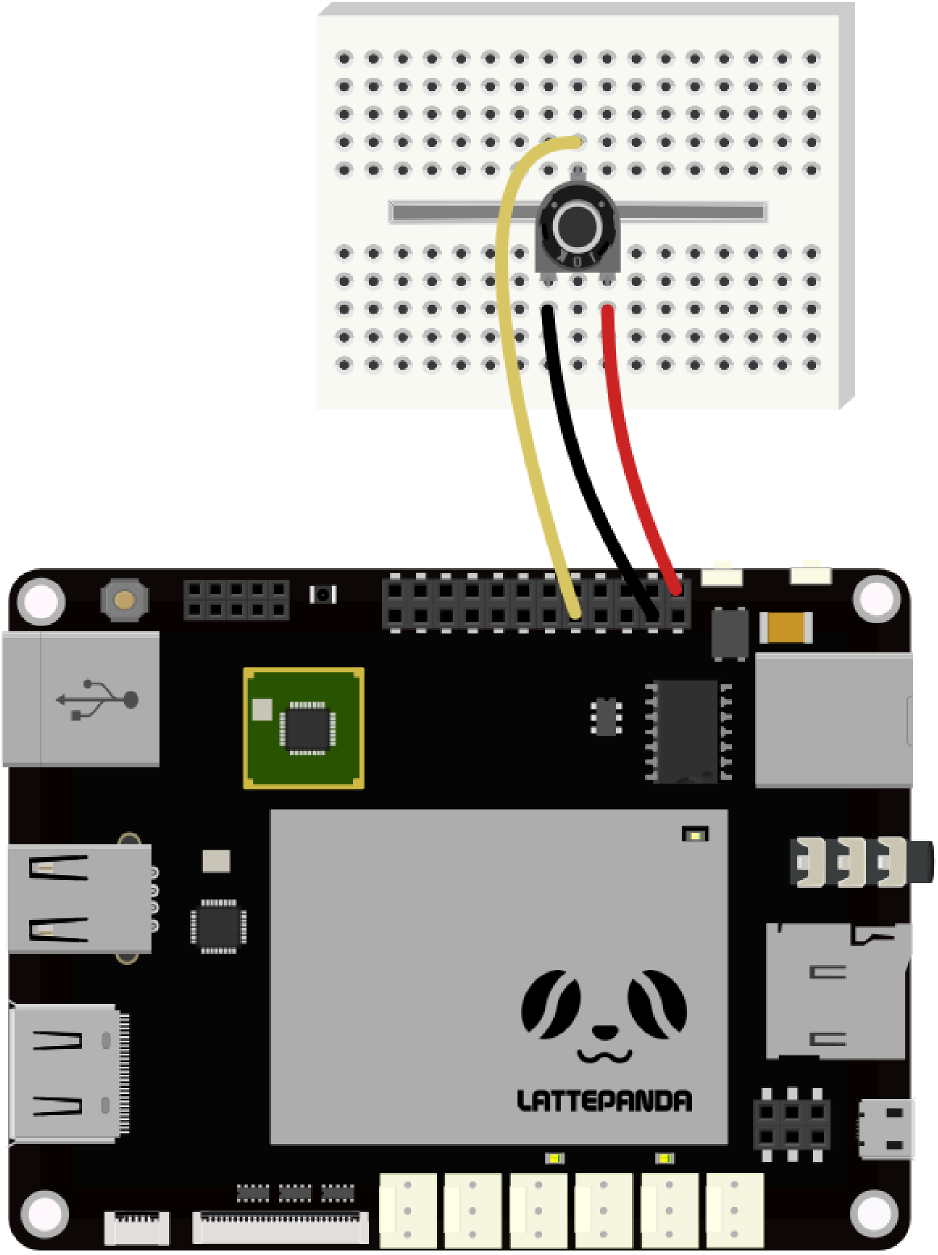
1. public Arduino();
2. public int analogRead(int pin);
3. public event AnalogPinUpdated analogPinUpdated;

Hardware Required:

1. LattePanda x 1
2. Potentiometer x 1

Circuit:

1. Connect the potentiometer to pin 3 as following figure shows：



Code :

1. Create a new project in Visual Studio, refer to Create a project
2. Main function Code：

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Threading;
using LattePanda.Firmata;
namespace analogWriteExample
{
class Program
{
static Arduino arduino = new Arduino();//create an instance and initialize with the default parameters
static void Main(string[] args)
{
int Value = arduino.analogRead(3);//Read the state of pin 3
Console.WriteLine(Value);
arduino.analogPinUpdated += Arduino_analogPinUpdated; ;//Add Event Listeners and call it when the analog input update.
}
private static void Arduino_analogPinUpdated(int pin, int value)
{
if(pin==3)
{
Console.WriteLine(pin);
Console.WriteLine(value);
}
}
}
}
```

Test :

1. Click Debug to execute, the state of potentiometer will print when you rotate it.

SERVO

In this example, we will sweep the servo motor back and forth across 180 degrees.

API Required:

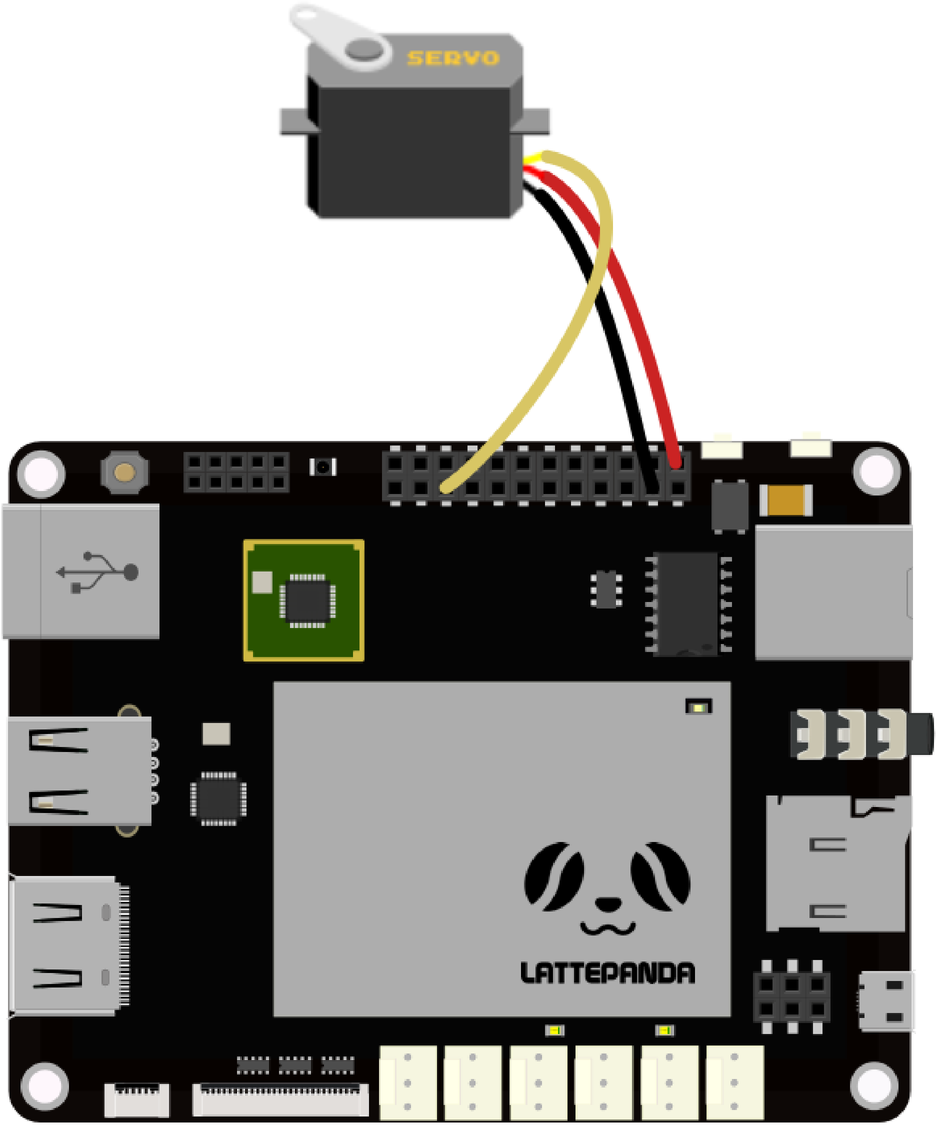
1. public Arduino();
2. public void pinMode(int pin, byte mode);
3. public void servoWrite(int pin, int angle);

Hardware Required:

1. LattePanda x 1
2. Servo Motor x 1

Circuit:

1. Servo inserted directly into pin D9 :



Code :

1. Create a new project in Visual Studio, Refer to Create a project
2. Main function code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Threading;
using LattePanda.Firmata;
namespace analogWriteExample
{
class Program
{
static Arduino arduino = new Arduino();//create an instance and initialize with the default parameters
static void Main(string[] args)
{
arduino.pinMode(9, Arduino.SERVO);
while (true)
{
arduino.servoWrite(9, 180);//tell the servo motor go to the position in 180 degrees
Thread.Sleep(1000);//delay a seconds
arduino.servoWrite(9, 0);//tell the servo motor go to the position in 0 degrees
Thread.Sleep(1000);//delay a seconds
}
}
}
}
```

Test:

1. Click debug to execute, you will find the motor sweeping forth and back continuously.

`I2C`

This example will show you how to use I2C to get the data form 3-axis accelerometer ADXL345

API Required:

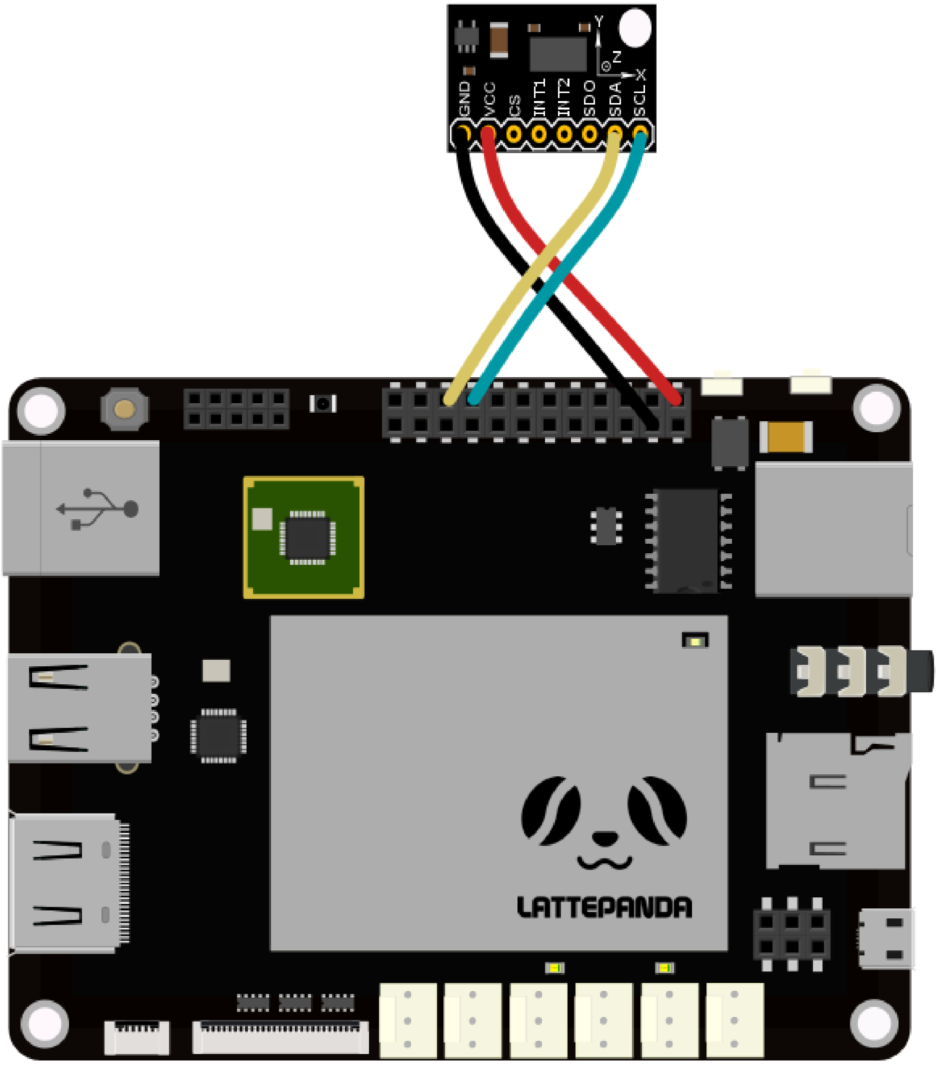
1. public Arduino();
2. public void wireBegin(Int16 delay);
   3.public void wireRequest(byte slaveAddress,Int16 slaveRegister, Int16[] data,byte mode);
3. public event DidI2CDataReveive didI2CDataReveive;

Hardware Required:

1. LattePanda x 1
2. ADXL345 x 1

Circuit:

1. The following is a figure describing which pins on the LattePanda should be connected to the pins on the accelerometer.



Code :

1. Create a new project in Visual Studio, Refer to Create a project
2. Main function code:

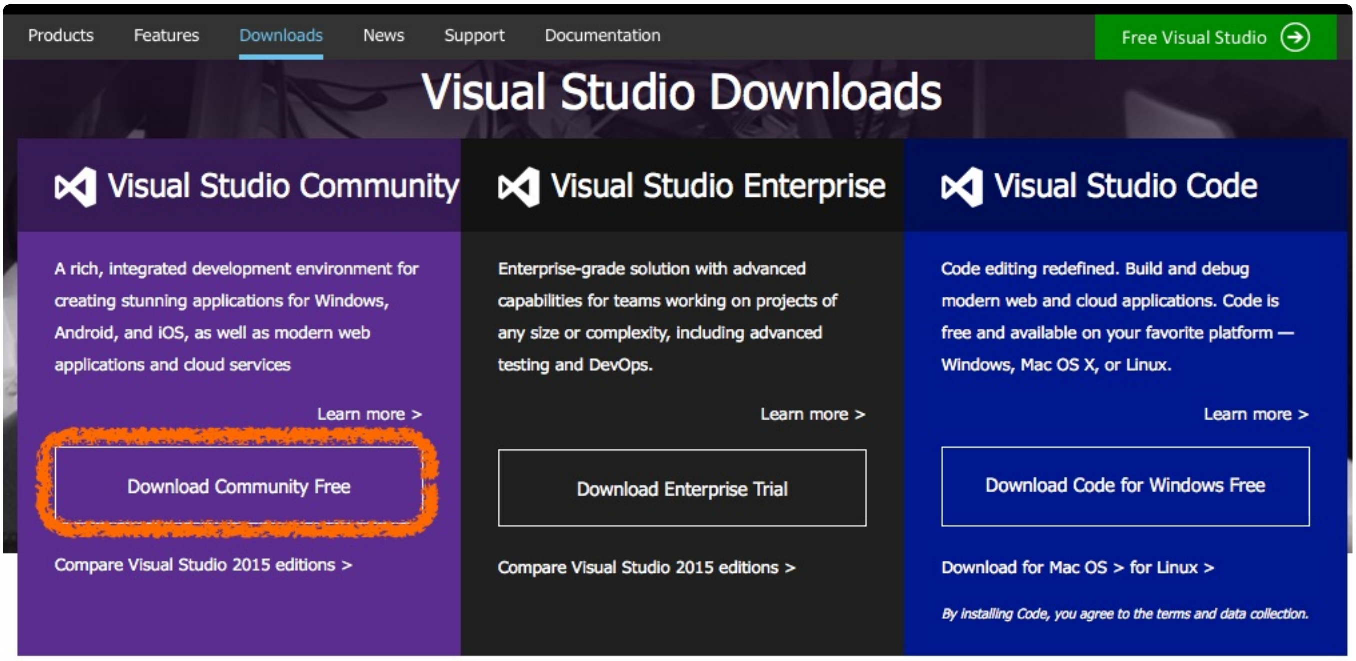
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Threading;
using LattePanda.Firmata;
namespace analogWriteExample
{
class Program
{
static Arduino arduino = new Arduino();//create an instance and initialize with the default parameters
static void Main(string[] args)
{
arduino.wireBegin(200);
arduino.wireRequest(0x53, 0x2D, new Int16[] { 8 }, Arduino.I2C_MODE_WRITE);//Write data{8} to I2C
arduino.didI2CDataReveive += Arduino_didI2CDataReveive;//did I2C Data Reveive
arduino.wireRequest(0x53, 0x32, new Int16[] { 6 }, Arduino.I2C_MODE_READ_CONTINUOUSLY);//Read data form I2C data
}
private static void Arduino_didI2CDataReveive(byte address, byte register, byte[] data)
{
Console.WriteLine(BitConverter.ToInt16(data, 0));
Console.WriteLine(BitConverter.ToInt16(data, 2));
Console.WriteLine(BitConverter.ToInt16(data, 4));
}
}
}
```

Test:

1. Click Debug to execute, the 3-axis acceleration data will be printing continuous.

SET UP YOUR PC

1. Download Visual Studio 2015 Here (https://www.visualstudio.com/downloads/download-visual-studio-vs)



Visual Studio Downloads

Visual Studio Community

A rich, integrated development environment for creating stunning applications for Windows, Android, and iOS, as well as modern web applications and cloud services

Visual Studio Enterprise

Enterprise-grade solution with advanced capabilities for teams working on projects of any size or complexity, including advanced testing and DevOps.

Learn more >
Download Enterprise Trial
Compare Visual Studio 2015 editions >

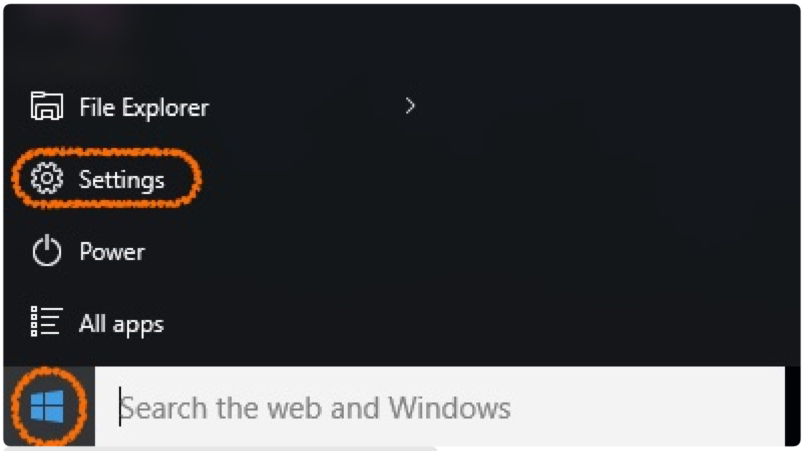
Visual Studio Code

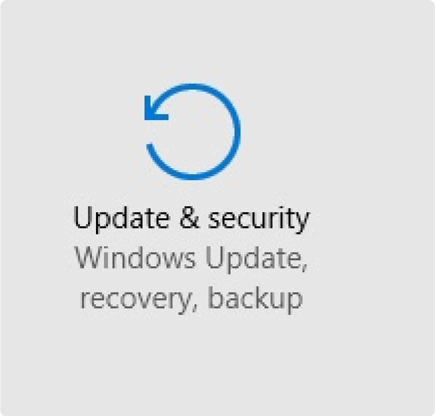
Code editing redefined. Build and debug modern web and cloud applications. Code is free and available on your favorite platform — Windows, Mac OS X, or Linux.

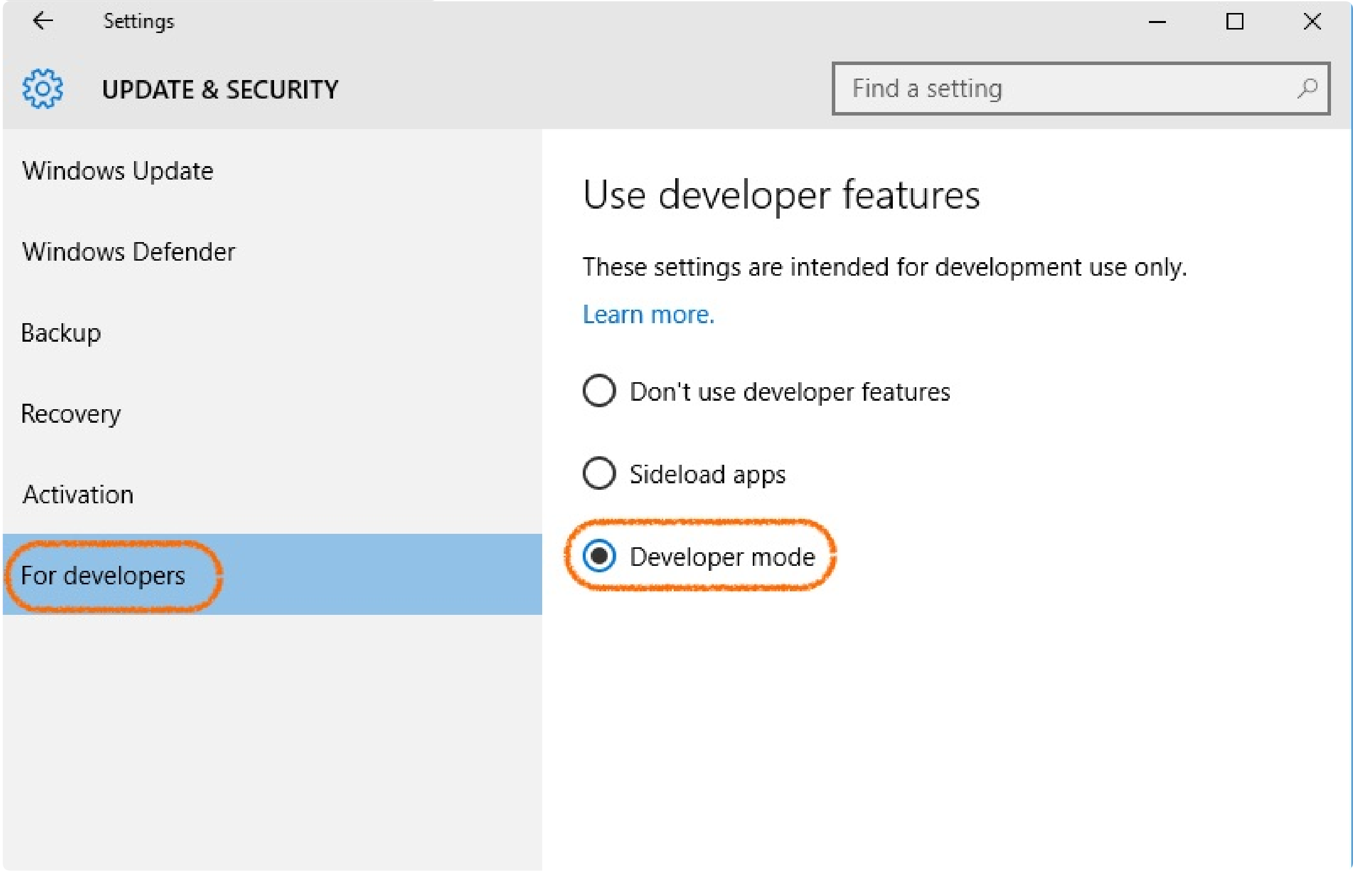
Learn more >
Download Code for Windows Free
Download for Mac OS > for Linux >

By installing Code, you agree to the terms and data collection.

2. Enable developer mode on your operating system







Use developer features

These settings are intended for development use only.

Learn more.

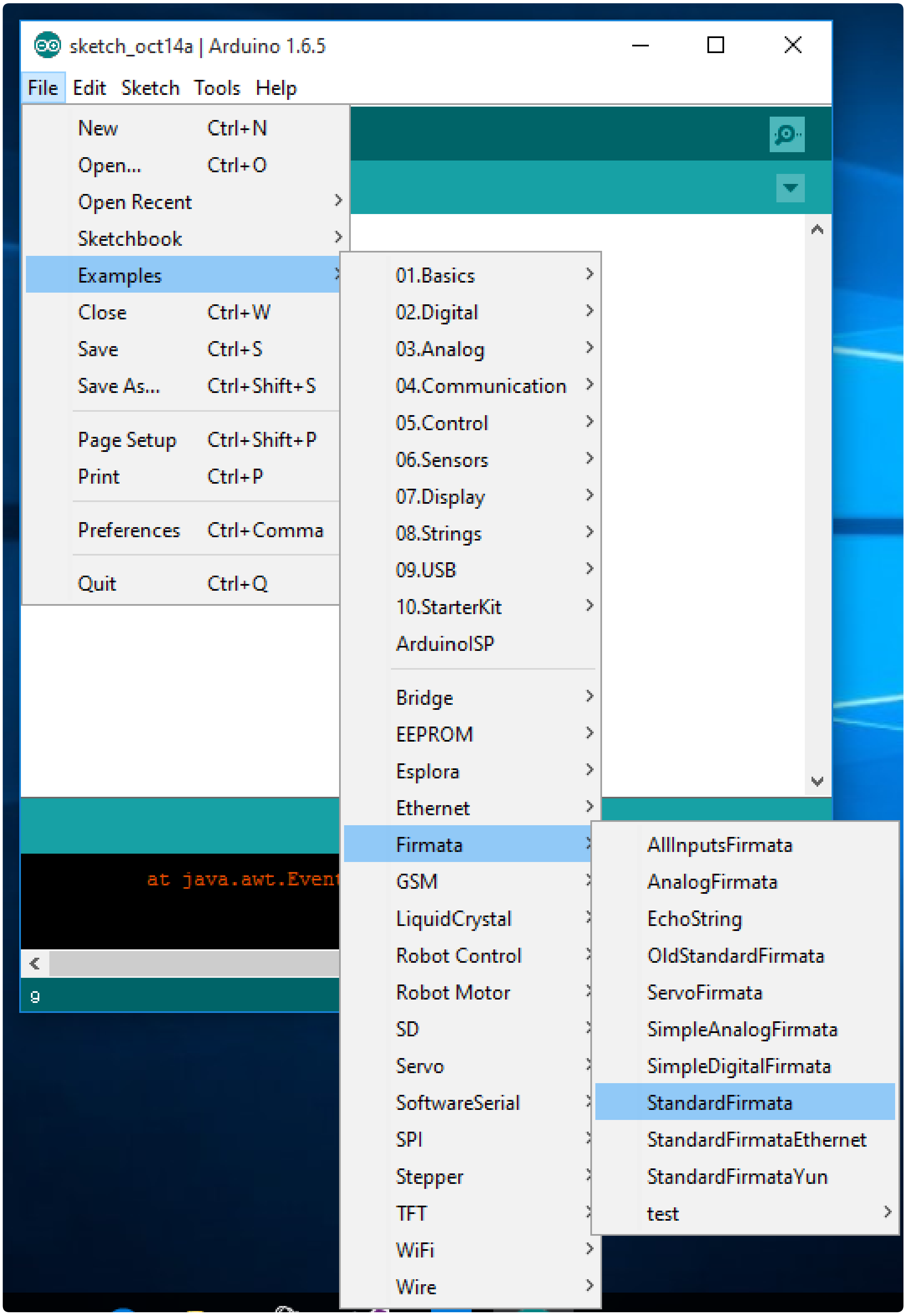
○ Don't use developer features

○ Sideload apps

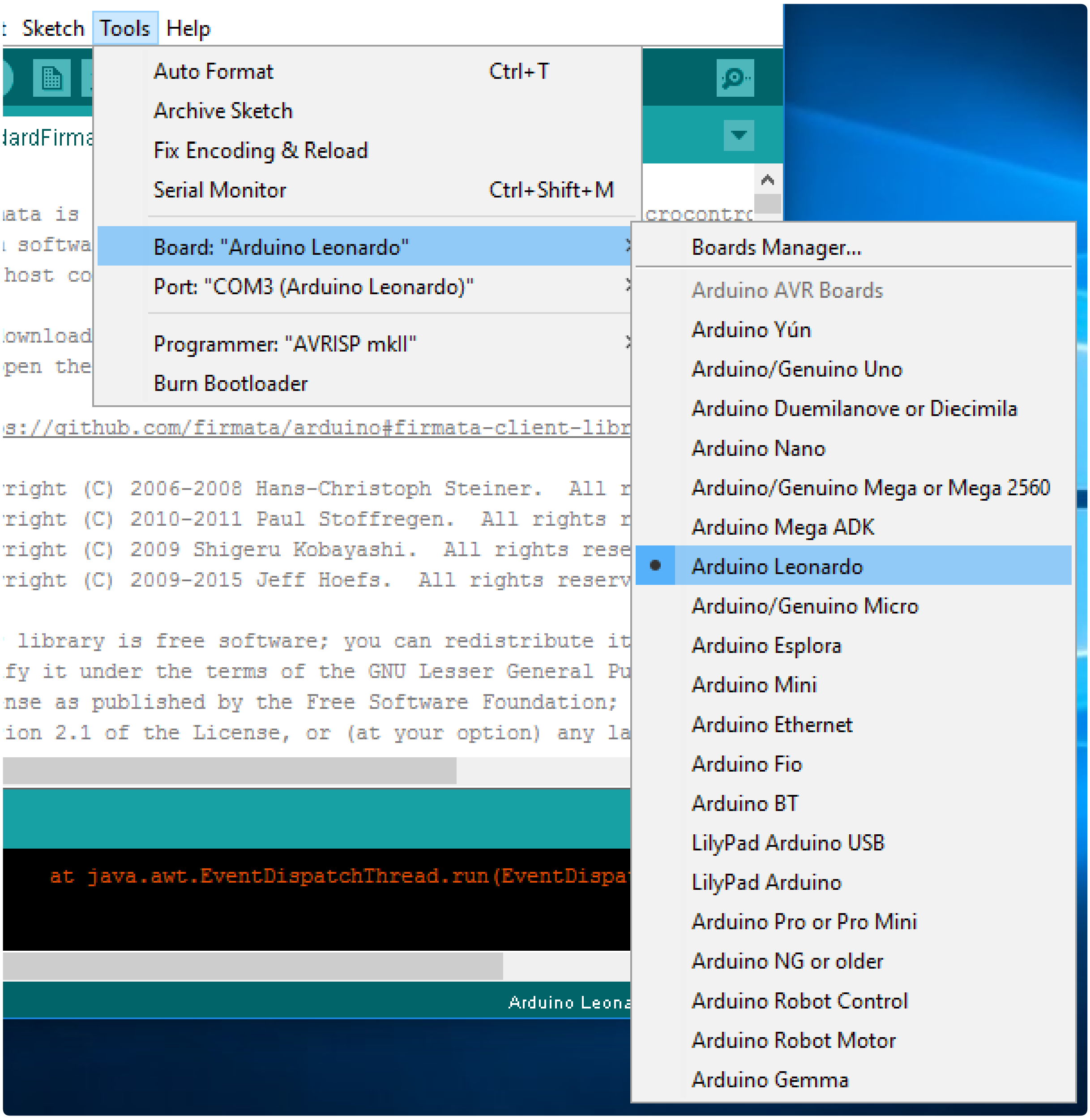
● Developer mode

SET UP THE ARDUINO

1. Open Arduino. And select the “StandardFirmata”



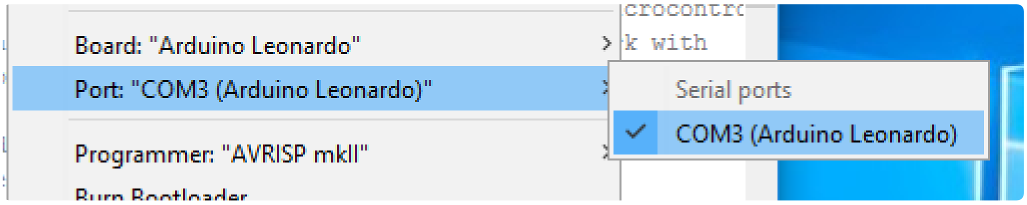
2. Select “Arduino Leonardo”



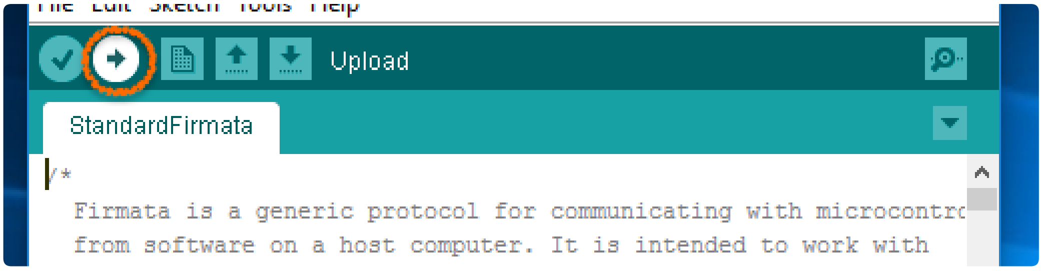
```
ata is
 softwa
host co

ownload
pen the
s://github.com/firmata/arduino#firmata-client-libr

right (C) 2006-2008 Hans-Christoph Steiner.  All r
right (C) 2010-2011 Paul Stoffregen.  All rights r
right (C) 2009 Shigeru Kobayashi.  All rights rese
right (C) 2009-2015 Jeff Hoefs.  All rights reserv

 library is free software; you can redistribute it
fy it under the terms of the GNU Lesser General Pu
nse as published by the Free Software Foundation;
ion 2.1 of the License, or (at your option) any la
```

```
at java.awt.EventDispatchThread.run(EventDispa
```

3. Select your COM port

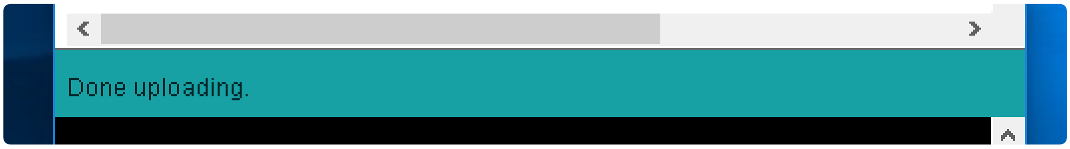

Board: "Arduino Leonardo"
Port: "COM3 (Arduino Leonardo)"
Programmer: "AVRISP mkII"
Burn Bootloader
Serial ports
✓ COM3 (Arduino Leonardo)
crocontrc
k with

4. Upload the sketch



```
/*
  Firmata is a generic protocol for communicating with microcontro
  from software on a host computer. It is intended to work with
```

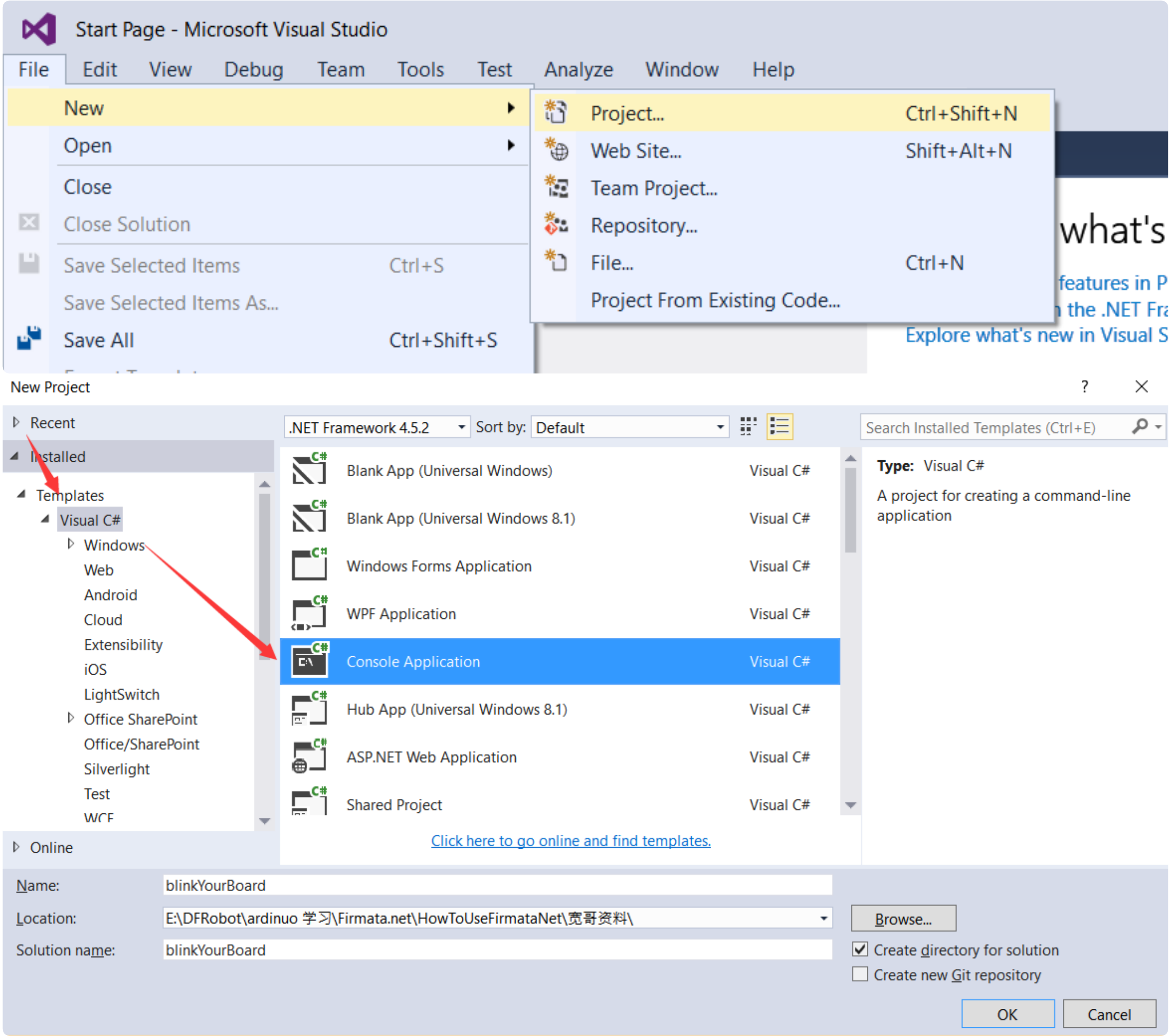
5. Upload Done!



Done uploading.

BLINK YOUR BOARD

1. Open Visual Studio 2015 and create a new app:



Start Page - Microsoft Visual Studio
File Edit View Debug Team Tools Test Analyze Window Help
New
Open
Close
Close Solution
Save Selected Items Ctrl+S
Save Selected Items As...
Save All Ctrl+Shift+S
Project... Ctrl+Shift+N
Web Site... Shift+Alt+N
Team Project...
Repository...
File... Ctrl+N
Project From Existing Code...
what's
features in P
the .NET Fra
Explore what's new in Visual S
New Project
Recent
Installed
Templates
Visual C#
Windows
Web
Android
Cloud
Extensibility
iOS
LightSwitch
Office SharePoint
Office/SharePoint
Silverlight
Test
WCF
Online
.NET Framework 4.5.2
Sort by: Default
Search Installed Templates (Ctrl+E)
Blank App (Universal Windows) Visual C#
Blank App (Universal Windows 8.1) Visual C#
Windows Forms Application Visual C#
WPF Application Visual C#
Console Application Visual C#
Hub App (Universal Windows 8.1) Visual C#
ASP.NET Web Application Visual C#
Shared Project Visual C#
Click here to go online and find templates.
Browse...
OK
Cancel

Type: Visual C#

A project for creating a command-line application

Name: blinkYourBoard

Location: E:\DFRobot\ardinuo 学习\Firmata.net\HowToUseFirmataNet\宽哥资料\

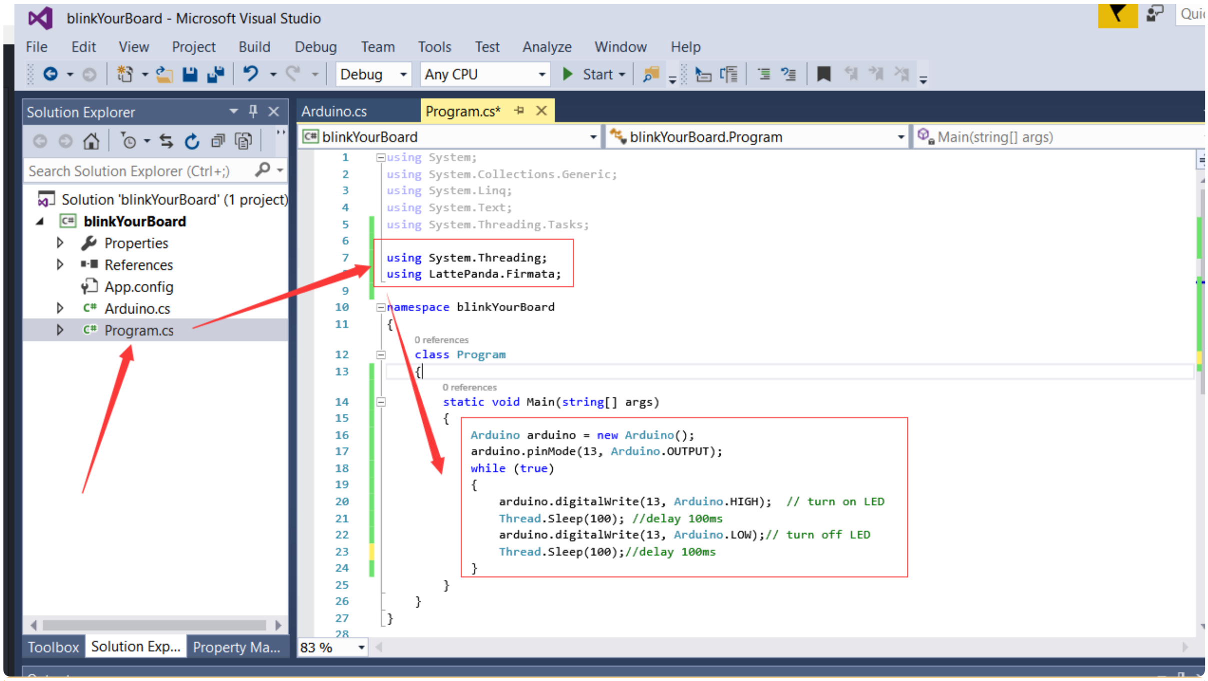
Solution name: blinkYourBoard

☑ Create directory for solution

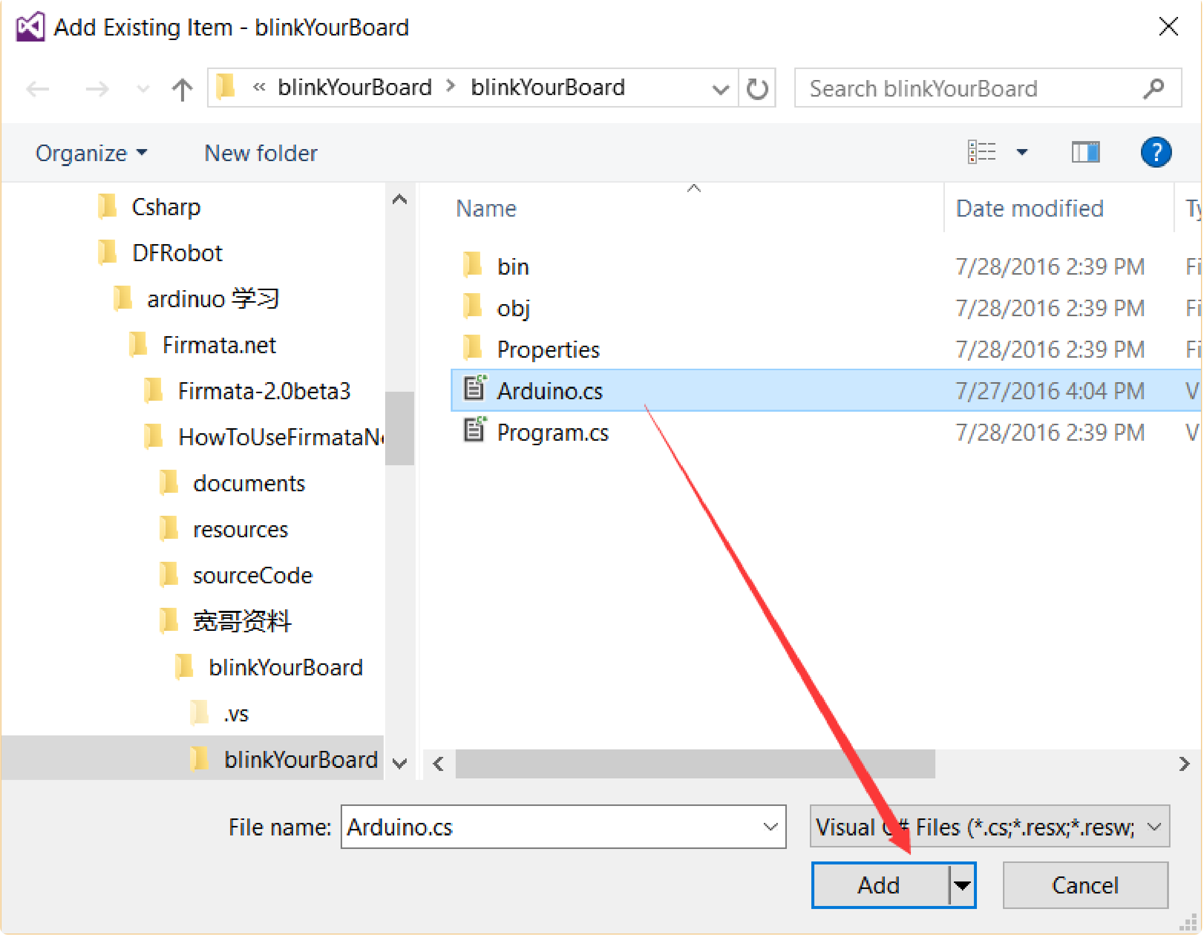
□ Create new Git repository

2. Download the LattePanda.Firmata class library (https://github.com/LattePandaTeam/LattePanda-Development-Support/tree/master/LattePandaFirmata)

3. Add the downloaded class library to your project Open your Solution Explorer and right-click in the blank area, then add existing item.



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using System.Threading;
using LattePanda.Firmata;

namespace blinkYourBoard
{
    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
            Arduino arduino = new Arduino();
            arduino.pinMode(13, Arduino.OUTPUT);
            while (true)
            {
                arduino.digitalWrite(13, Arduino.HIGH);  // turn on LED
                Thread.Sleep(100); //delay 100ms
                arduino.digitalWrite(13, Arduino.LOW);// turn off LED
                Thread.Sleep(100);//delay 100ms
            }
        }
    }
}
```



Name	Date modified	Ty
bin	7/28/2016 2:39 PM	Fi
obj	7/28/2016 2:39 PM	Fi
Properties	7/28/2016 2:39 PM	Fi
Arduino.cs	7/27/2016 4:04 PM	V
Program.cs	7/28/2016 2:39 PM	V

File name: Arduino.cs

Visual C# Files (*.cs;*.resx;*.resw;

Add | Cancel

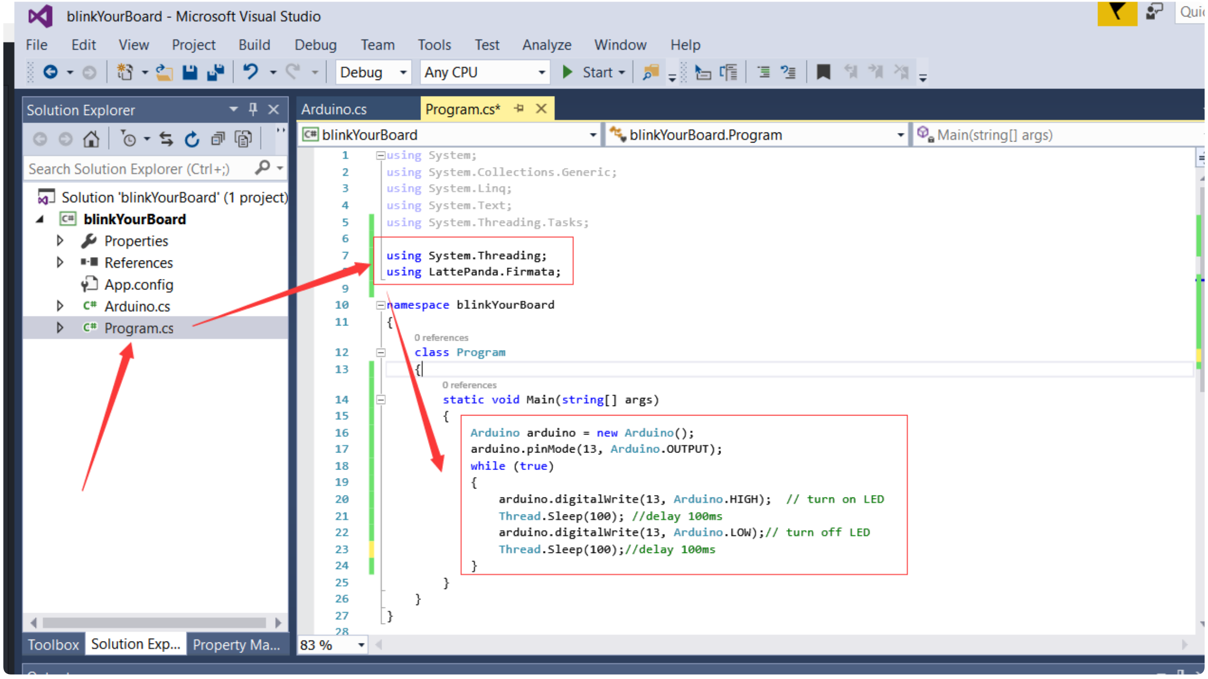
4. Add the following code to Program.cs.

- Add the following two lines code before the namespace blinkYourBoard.
  `using System.Threading;`
  `using LattePanda.Firmata;`
  The first namespace contributes to delay and second is LattePanda.Firmata class library namespace

- Add the following code in the Main Function

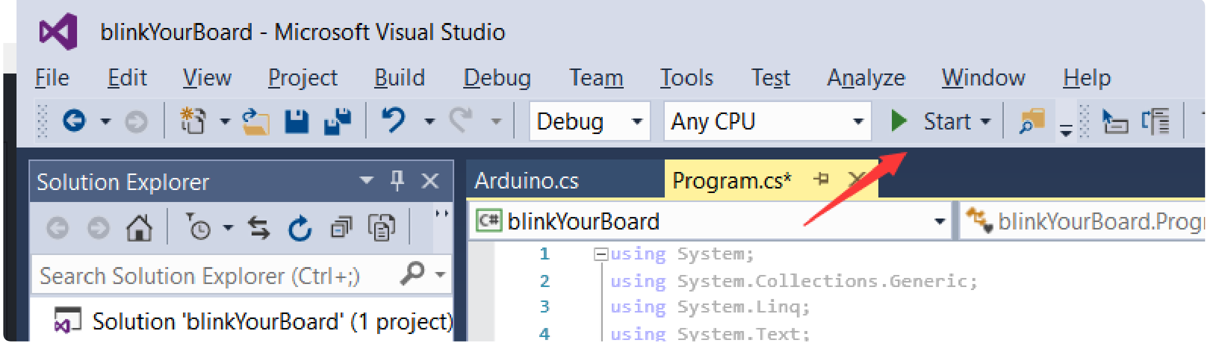
```
Arduino arduino = new Arduino();
arduino.pinMode(13, Arduino.OUTPUT);//Set the digital pin 13 as output
while (true)
{
// ==== set the led on or off
arduino.digitalWrite(13, Arduino.HIGH);//set the LED  on
Thread.Sleep(1000);//delay a seconds
arduino.digitalWrite(13, Arduino.LOW);//set the LED  off
Thread.Sleep(1000);//delay a seconds
}
```

The complete code is as follows:



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using System.Threading;
using LattePanda.Firmata;

namespace blinkYourBoard
{
    class Program
    {
        static void Main(string[] args)
        {
            Arduino arduino = new Arduino();
            arduino.pinMode(13, Arduino.OUTPUT);
            while (true)
            {
                arduino.digitalWrite(13, Arduino.HIGH);  // turn on LED
                Thread.Sleep(100); //delay 100ms
                arduino.digitalWrite(13, Arduino.LOW);// turn off LED
                Thread.Sleep(100);//delay 100ms
            }
        }
    }
}
```

5. Finally debug your project
   Connect your Arduino and click Start



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
```

Soon you will find the LED of your Arduino blinking

Project Examples

Turn a LattePanda in to a 3D Print Server Using Octoprint

OctoPrint is an open source software which is designed to manage 3D printers. It enables you to control and monitor a 3D printer remotely from any web browser just as if you were sitting in front of it, even if it’s on the other side of your house, in your garage, or on the other side of the world. In addition to this it can slice models by dragging and dropping .stl or .obj

files on to the web UI and you can view print jobs in progress and create time lapses when you connect an ordinary USB webcam!
In this tutorial we will discuss how to control and monitor a 3D printer remotely using OctoPrint and LattePanda.

What We Need

Before we begin you will need to prepare the following equipment:

1. LattePanda
2. A 3D printer (We have opted for the Overlord 3D Printer from DREAMMAKER)

What is the DREAMMAKER Overlord?

The DREAMMAKER Overlord is a 3D printer that is affordable, smart and stylish. It has many advantages such as:

1. A fusion of oriental classic and minimalism design
2. Up to 16 Pantone standard colors for your unique taste
3. Optimized operation on Windows/Mac OS/Linux
4. Auto calibration
5. Industrial Delta armature that enables a high level of repeated accuracy

The following section will walk through how to install OctoPrint and connect your 3D printer

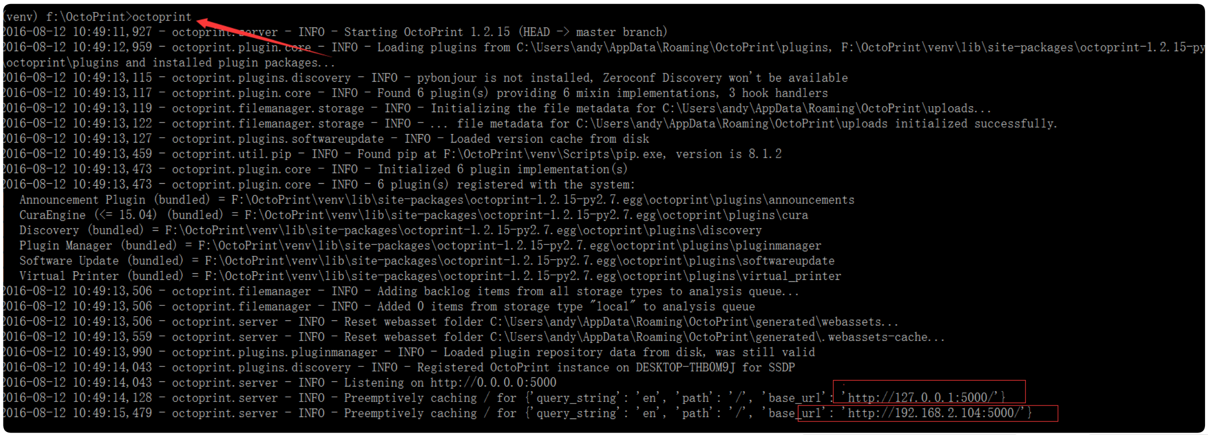
Let's get started:

STEP 1: INSTALL OCTOPRINT

To install OctoPrint follow the official tutorial:
Windows https://github.com/foosel/OctoPrint/wiki/Setup-on-Windows
In the tutorial you need to install Python 2.7 and Microsoft Visual C++ Compiler for Python 2.7 and then execute some command statements

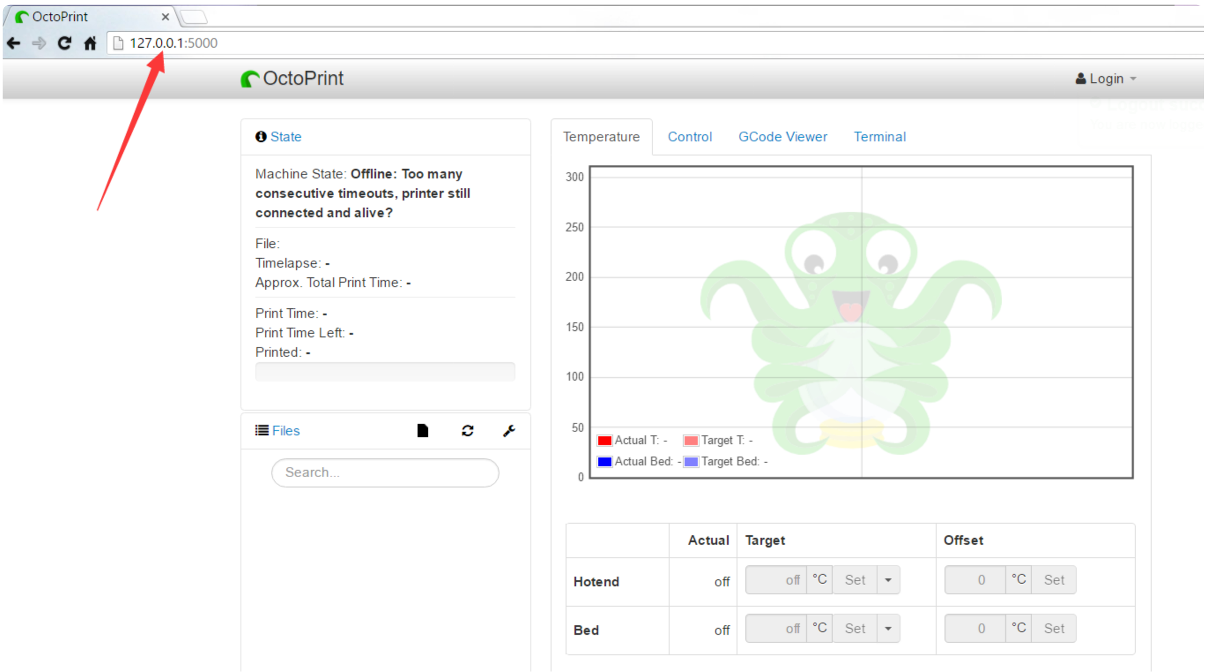
STEP 2: START OCTOPRINT IN A WEB BROWSER

When you complete last step in the section 1, the command window will show some information as following figures show:



```
(venv) f:\OctoPrint>octoprint
2016-08-12 10:49:11,927 - octoprint.server - INFO - Starting OctoPrint 1.2.15 (HEAD -> master branch)
2016-08-12 10:49:12,959 - octoprint.plugin.core - INFO - Loading plugins from C:\Users\andy\AppData\Roaming\OctoPrint\plugins, F:\OctoPrint\venv\lib\site-packages\octoprint-1.2.15-py
\octoprint\plugins and installed plugin packages...
2016-08-12 10:49:13,115 - octoprint.plugins.discovery - INFO - pybonjour is not installed, Zeroconf Discovery won't be available
2016-08-12 10:49:13,117 - octoprint.plugin.core - INFO - Found 6 plugin(s) providing 6 mixin implementations, 3 hook handlers
2016-08-12 10:49:13,119 - octoprint.filemanager.storage - INFO - Initializing the file metadata for C:\Users\andy\AppData\Roaming\OctoPrint\uploads...
2016-08-12 10:49:13,122 - octoprint.filemanager.storage - INFO - ... file metadata for C:\Users\andy\AppData\Roaming\OctoPrint\uploads initialized successfully.
2016-08-12 10:49:13,127 - octoprint.plugins.softwareupdate - INFO - Loaded version cache from disk
2016-08-12 10:49:13,459 - octoprint.util.pip - INFO - Found pip at F:\OctoPrint\venv\Scripts\pip.exe, version is 8.1.2
2016-08-12 10:49:13,473 - octoprint.plugin.core - INFO - Initialized 6 plugin implementation(s)
2016-08-12 10:49:13,473 - octoprint.plugin.core - INFO - 6 plugin(s) registered with the system:
  Announcement Plugin (bundled) = F:\OctoPrint\venv\lib\site-packages\octoprint-1.2.15-py2.7.egg\octoprint\plugins\announcements
  CuraEngine (<= 15.04) (bundled) = F:\OctoPrint\venv\lib\site-packages\octoprint-1.2.15-py2.7.egg\octoprint\plugins\cura
  Discovery (bundled) = F:\OctoPrint\venv\lib\site-packages\octoprint-1.2.15-py2.7.egg\octoprint\plugins\discovery
  Plugin Manager (bundled) = F:\OctoPrint\venv\lib\site-packages\octoprint-1.2.15-py2.7.egg\octoprint\plugins\pluginmanager
  Software Update (bundled) = F:\OctoPrint\venv\lib\site-packages\octoprint-1.2.15-py2.7.egg\octoprint\plugins\softwareupdate
  Virtual Printer (bundled) = F:\OctoPrint\venv\lib\site-packages\octoprint-1.2.15-py2.7.egg\octoprint\plugins\virtual_printer
2016-08-12 10:49:13,506 - octoprint.filemanager - INFO - Adding backlog items from all storage types to analysis queue...
2016-08-12 10:49:13,506 - octoprint.filemanager - INFO - Added 0 items from storage type "local" to analysis queue
2016-08-12 10:49:13,506 - octoprint.server - INFO - Reset webasset folder C:\Users\andy\AppData\Roaming\OctoPrint\generated\webassets...
2016-08-12 10:49:13,559 - octoprint.server - INFO - Reset webasset folder C:\Users\andy\AppData\Roaming\OctoPrint\generated\.webassets-cache...
2016-08-12 10:49:13,990 - octoprint.plugins.pluginmanager - INFO - Loaded plugin repository data from disk, was still valid
2016-08-12 10:49:14,043 - octoprint.plugins.discovery - INFO - Registered OctoPrint instance on DESKTOP-THB0M9J for SSDP
2016-08-12 10:49:14,043 - octoprint.server - INFO - Listening on http://0.0.0.0:5000
2016-08-12 10:49:14,128 - octoprint.server - INFO - Preemptively caching / for {'query_string': 'en', 'path': '/', 'base_url': 'http://127.0.0.1:5000/'}
2016-08-12 10:49:15,479 - octoprint.server - INFO - Preemptively caching / for {'query_string': 'en', 'path': '/', 'base_url': 'http://192.168.2.104:5000/'}
```

In the lower right corner of the command window there are two URLs `http://127.0.0.1:5000` and `http://192.168.2.104:5000`
They are highlighted with red rectangles in the image.
The first URL is your localhost URL. You may access the OctoPrint web UI on the PC connected to the Printer using this URL.



Machine State: **Offline: Too many consecutive timeouts, printer still connected and alive?**

File:
Timelapse: -
Approx. Total Print Time: -

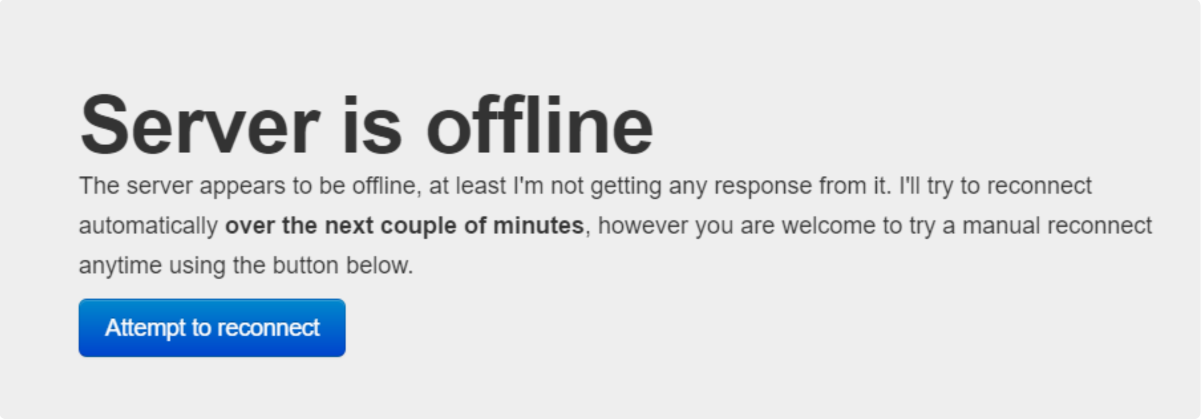
Print Time: -
Print Time Left: -
Printed: -

	Actual	Target	Offset
Hotend	off	off °C Set	0 °C Set
Bed	off	off °C Set	0 °C Set

The second URL is used to Connect to your 3D printer remotely from another PC on the same local Network as your LattePanda. The address 192.168.2.104 is actually your LattePanda's IP address. If the CMD window doesn't emerge the second URL, you can look for the IP address in Network and Sharing Center and attach the port number `:5000` at the end of IP address.

PLEASE NOTE:

1. We recommend using Google Chrome as a browser. This has been tested in Microsoft Edge, and wasn't able to work without some further calibration.
2. Do not close the command window when you attempt to open OctoPrint in your web browser, or you will see the following figure.



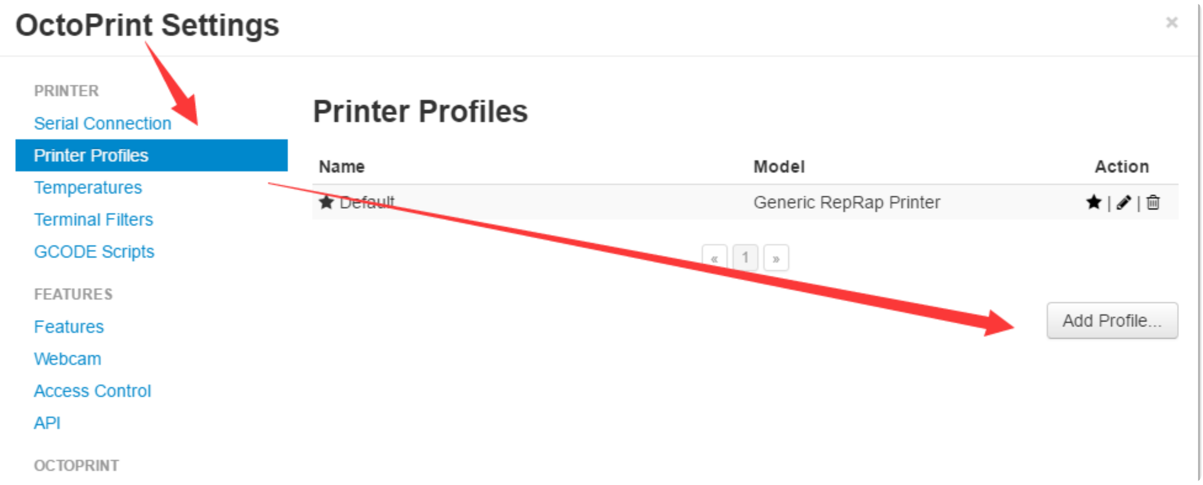
Server is offline

The server appears to be offline, at least I'm not getting any response from it. I'll try to reconnect automatically **over the next couple of minutes**, however you are welcome to try a manual reconnect anytime using the button below.

3. How to Start OctoPrint
   When we set up OctoPrint, there is no desktop icon or launcher. How can we open it when we need to use it? Just open the directory `F:\OctoPrint\venv\Scripts` and find the file `octoprint.exe`, double click it.
   Then the familiar command window will emerge where you can find the two URLs in the lower right corner. Input the URLs in to your web browser as before.

STEP 3: CONNECT TO YOUR 3D PRINTER

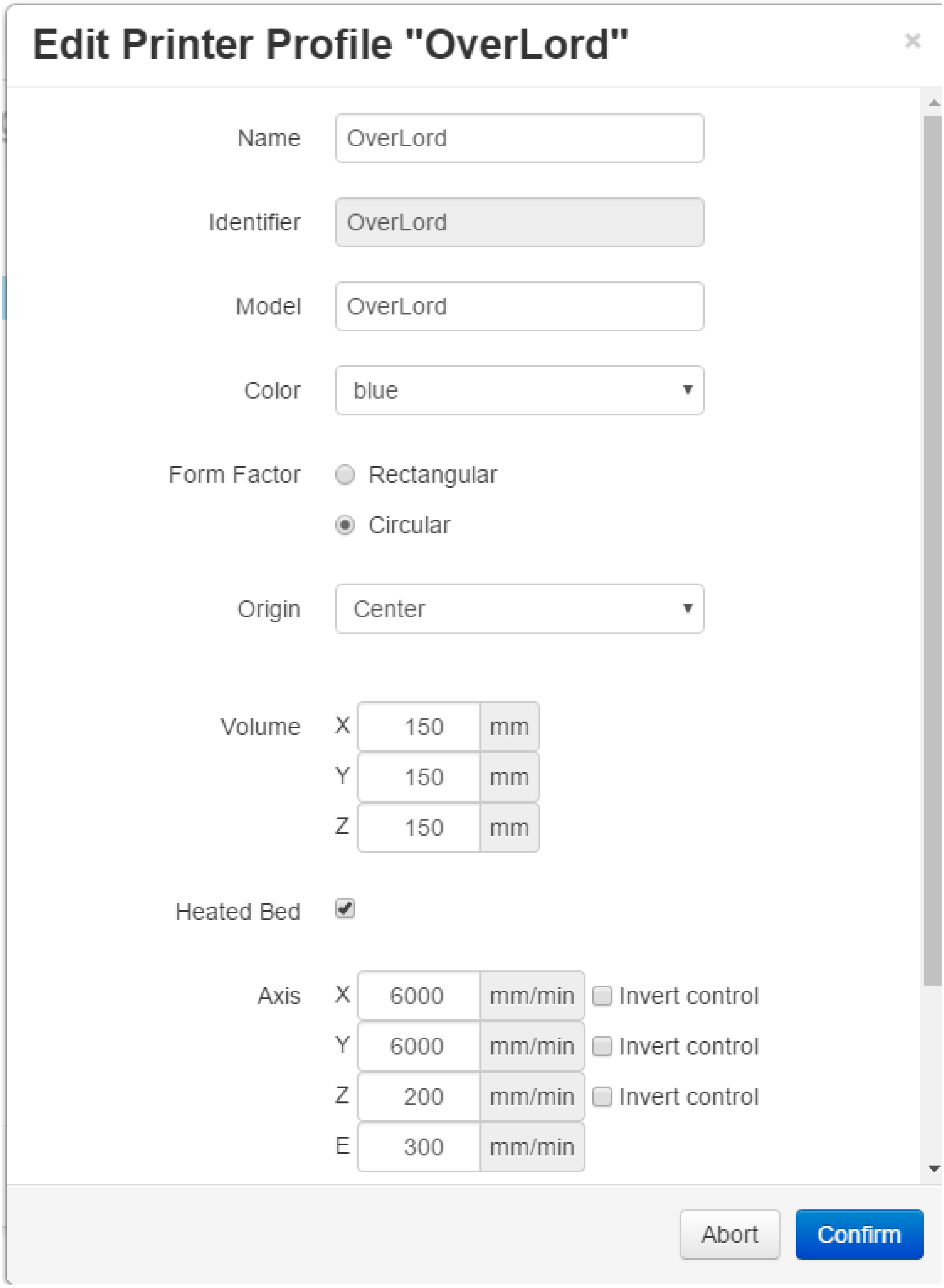
Log in your account (create one if you don't have one) and click Settings. Select "Printer Profiles" to add your printer. Choose your printer name and set some parameters similar to the following:



OctoPrint Settings

Printer Profiles

Name	Model	Action
★ Default	Generic RepRap Printer	★ | ✎ | 🗑



Edit Printer Profile "OverLord"

Name: OverLord

Identifier: OverLord

Model: OverLord

Color: blue

Form Factor: ○ Rectangular ● Circular

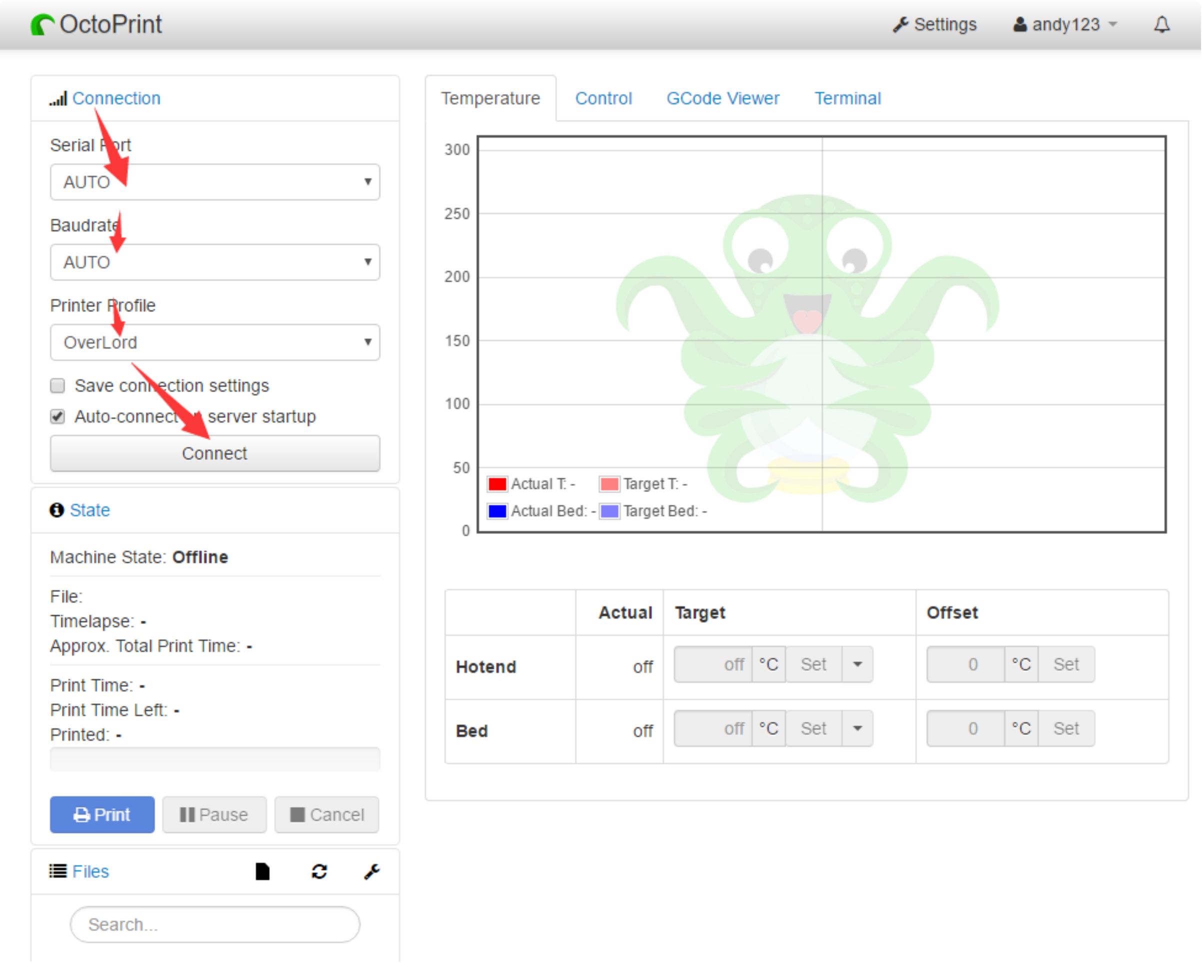
Origin: Center

Volume:
X 150 mm
Y 150 mm
Z 150 mm

Heated Bed: ☑

Axis:
X 6000 mm/min □ Invert control
Y 6000 mm/min □ Invert control
Z 200 mm/min □ Invert control
E 300 mm/min

Set the Serial Port and BaudRate as AUTO and select the Printer Profile "Overlord" that you created on the previous step



Serial Port
AUTO

Baudrate
AUTO

Printer Profile
OverLord

□ Save connection settings
☑ Auto-connect on server startup

Connect

State

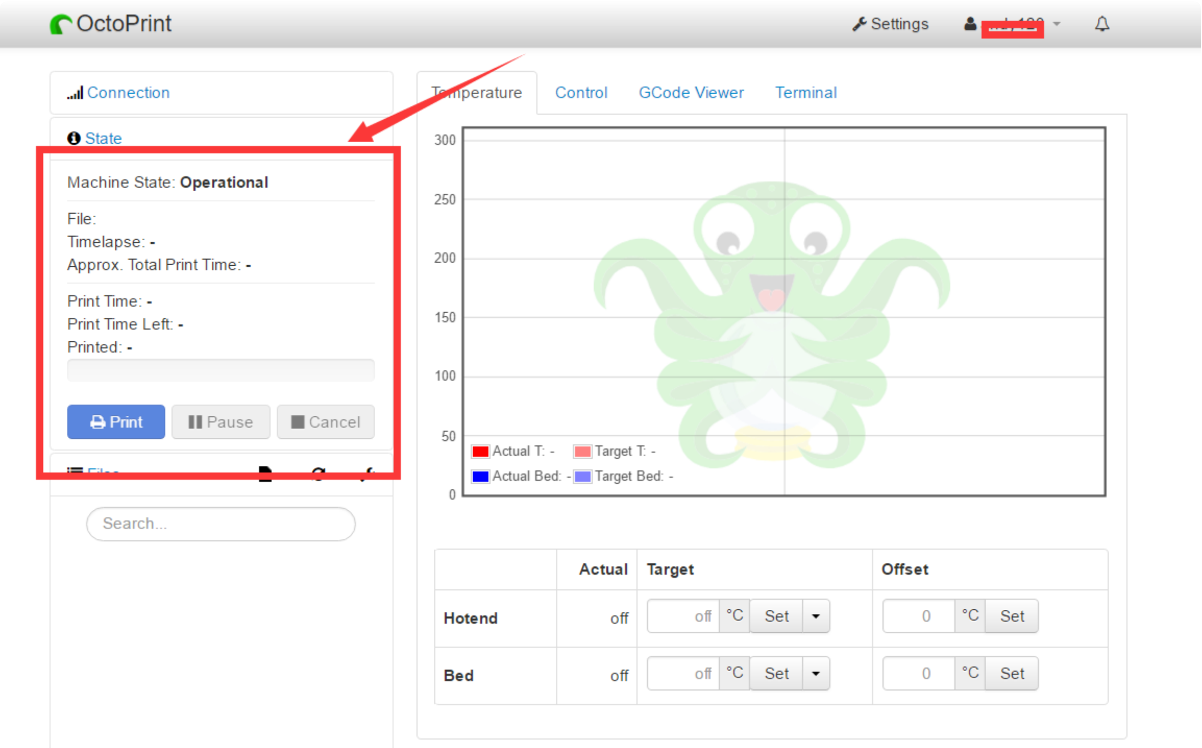
Machine State: **Offline**

File:
Timelapse: -
Approx. Total Print Time: -

Print Time: -
Print Time Left: -
Printed: -

	Actual	Target	Offset
Hotend	off	off °C Set	0 °C Set
Bed	off	off °C Set	0 °C Set

When you connect to the Overlord successfully, you should see the following information.



Machine State: **Operational**

File:
Timelapse: -
Approx. Total Print Time: -

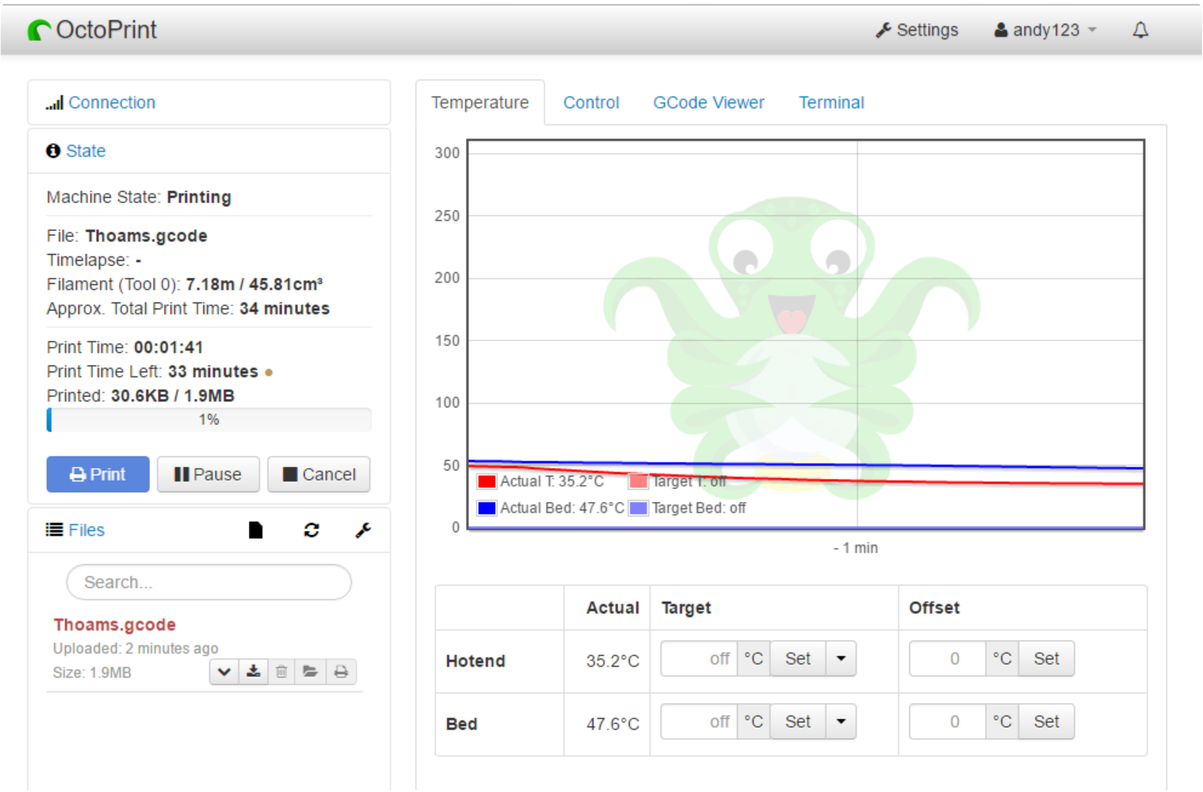
Print Time: -
Print Time Left: -
Printed: -

	Actual	Target	Offset
Hotend	off	off °C Set	0 °C Set
Bed	off	off °C Set	0 °C Set

If Octoprint fails to connect try restarting your 3D printer, refresh the OctoPrint web interface or disconnect and reconnect the USB cable.

STEP 4 : HOW TO UPLOAD FILES AND PRINT

Open the OctoPrint Web interface and press the upload button in the lower left corner to upload a .gcode file. When the upload is complete, you can click the print button, then the printer will begin to print the file. You can monitor the print time elapsed, remaining print time, nozzle temperature and also pause or cancel a running job as the following figure shows:



Machine State: **Printing**

File: **Thoams.gcode**
Timelapse: **-**
Filament (Tool 0): **7.18m / 45.81cm³**
Approx. Total Print Time: **34 minutes**

Print Time: **00:01:41**
Print Time Left: **33 minutes**
Printed: **30.6KB / 1.9MB**
1%

Thoams.gcode
Uploaded: 2 minutes ago
Size: 1.9MB

	Actual	Target	Offset
Hotend	35.2°C	off °C Set	0 °C Set
Bed	47.6°C	off °C Set	0 °C Set

You can now open OctoPrint on other computer or phone (connected to the same local Network as your LattePanda) and log in to monitor and control your printer remotely!

Feel free to contact us if you have any queries. We will give advice as soon as we can.

REFERENCES

1. http://www.dfrobot.com.cn/community/thread-13090-1-1.html
2. https://github.com/foosel/OctoPrint/wiki/Setup-on-Windows
3. http://www.lattepanda.com/product-detail/?pid=2
4. http://www.dreammaker.cc/

Face Detection using OpenCV

What is OpenCV

OpenCV (Open Source Computer Vision) is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing.
Adopted all around the world, OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding 9 million. Usage ranges from interactive art, to mines detection, online maps and advanced robotics.

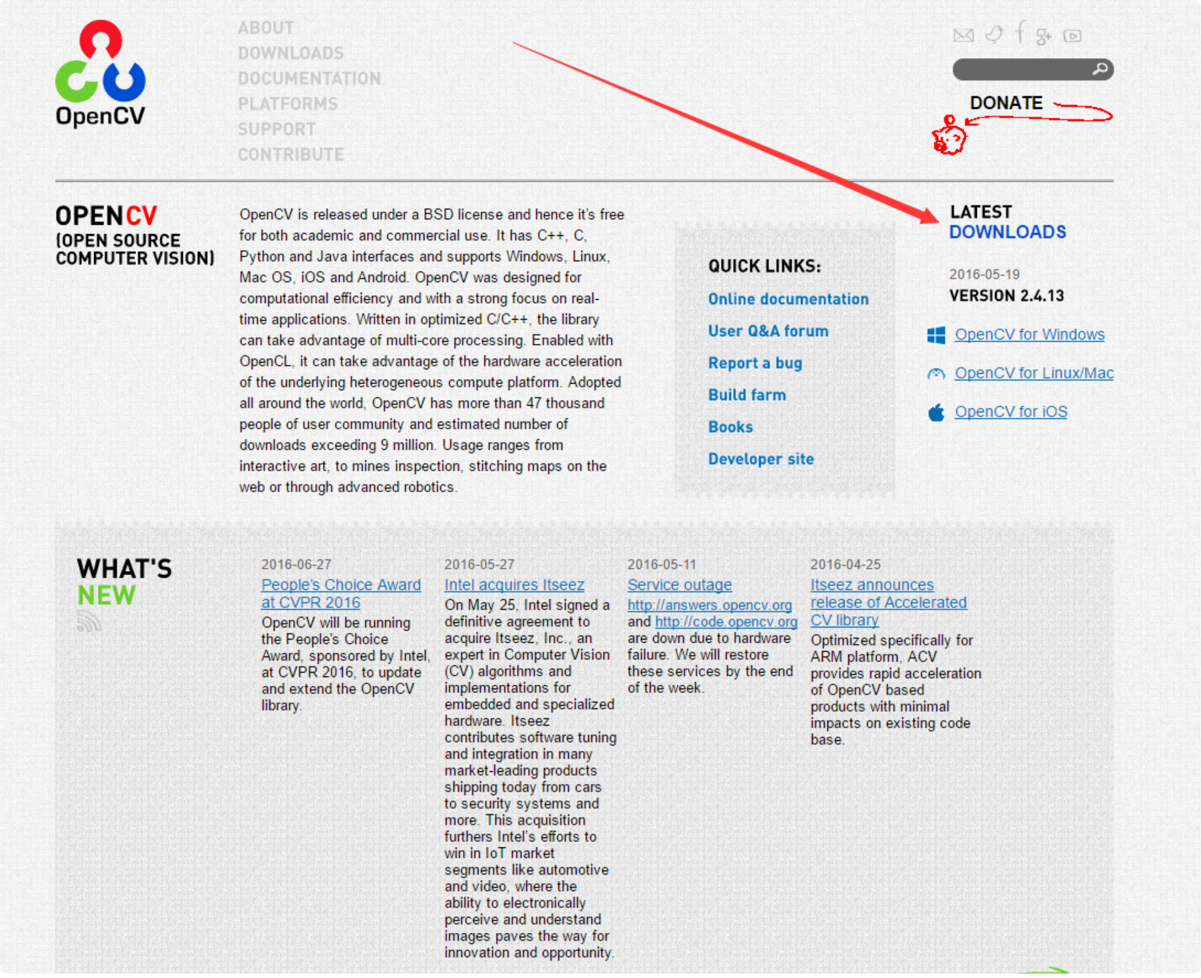
Let's get started:

STEP 1 : INSTALL VISUAL STUDIO 2015 AND OPENCV

1. Install Visual Studio 2015 on your computer
   Head over to https://www.visualstudio.com/products/visual-studio-professional-with-msdn-vs and download Visual

Studio Professional 2015. Unzip the downloaded file and double-click the 'vs_professional.exe', then the installation process will begin.

2. Install OpenCV Head over to the site: http://www.opencv.org and download the latest version of OpenCV (shown in the following figure). Choose the version according to your operating system.
   In this tutorial we are going to install OpenCV 3.1 using Visual Studio 2015 professional on a 64-bit system running Windows 10.



OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can take advantage of the hardware acceleration of the underlying heterogeneous compute platform. Adopted all around the world, OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding 9 million. Usage ranges from interactive art, to mines inspection, stitching maps on the web or through advanced robotics.

2016-05-19
VERSION 2.4.13

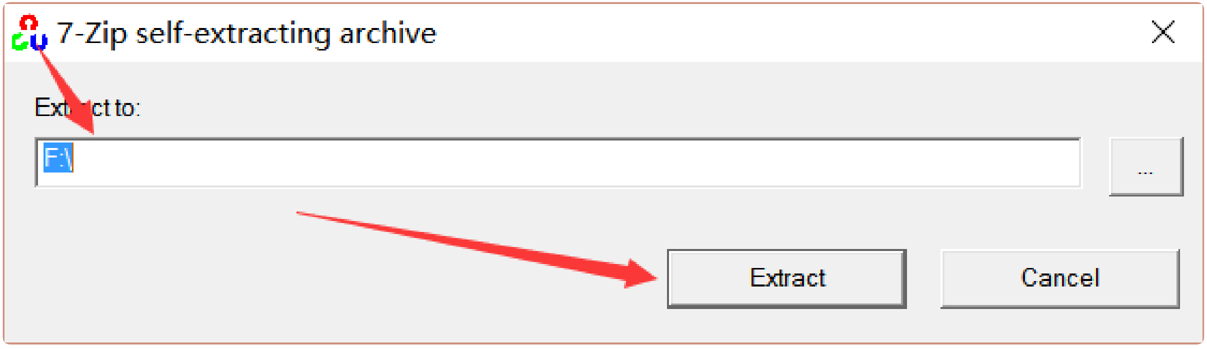
2016-06-27
People's Choice Award at CVPR 2016
OpenCV will be running the People's Choice Award, sponsored by Intel, at CVPR 2016, to update and extend the OpenCV library.

2016-05-27
Intel acquires Itseez
On May 25, Intel signed a definitive agreement to acquire Itseez, Inc., an expert in Computer Vision (CV) algorithms and implementations for embedded and specialized hardware. Itseez contributes software tuning and integration in many market-leading products shipping today from cars to security systems and more. This acquisition furthers Intel's efforts to win in IoT market segments like automotive and video, where the ability to electronically perceive and understand images paves the way for innovation and opportunity.

2016-05-11
Service outage
http://answers.opencv.org and http://code.opencv.org are down due to hardware failure. We will restore these services by the end of the week.

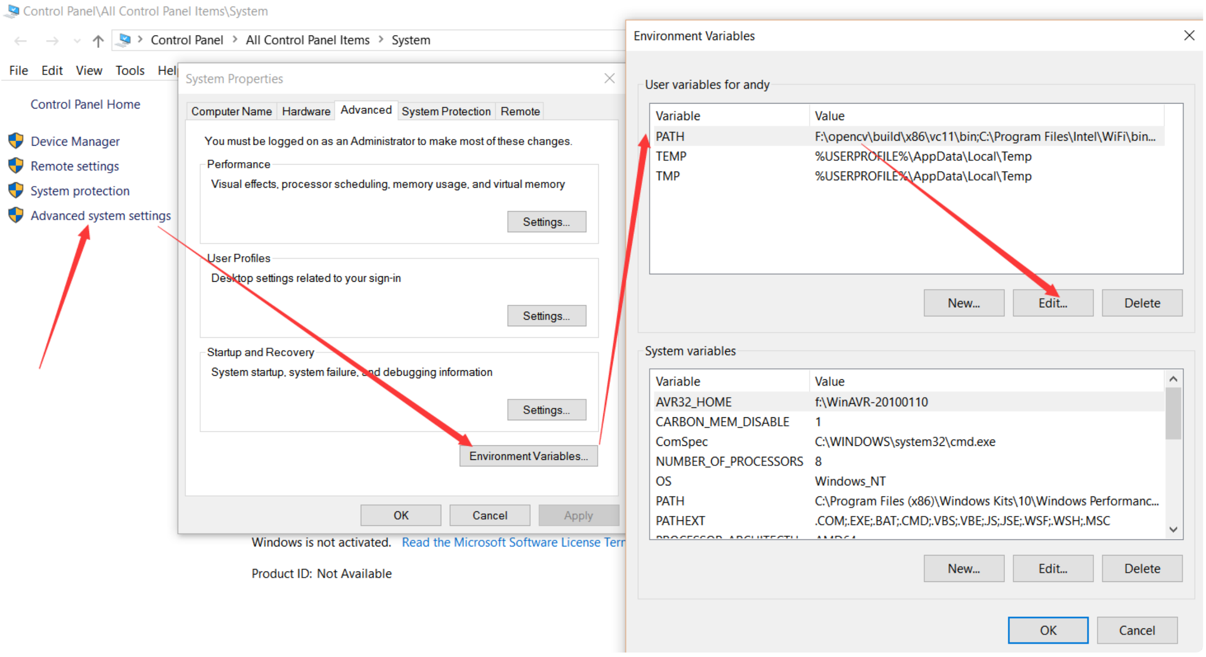
2016-04-25
Itseez announces release of Accelerated CV library
Optimized specifically for ARM platform, ACV provides rapid acceleration of OpenCV based products with minimal impacts on existing code base.

Extract the downloaded OpenCV file
Double click the downloaded OpenCV file, and then extract it



STEP 2: SET THE ENVIRONMENT VARIABLES

1. To do this step, open the Control Panel and then System. Click the Advanced System Settings, last Environment Variables in turns as show in the following figure.



You must be logged on as an Administrator to make most of these changes.

Visual effects, processor scheduling, memory usage, and virtual memory

Desktop settings related to your sign-in

System startup, system failure, and debugging information

Windows is not activated.

Product ID: Not Available

User variables for andy

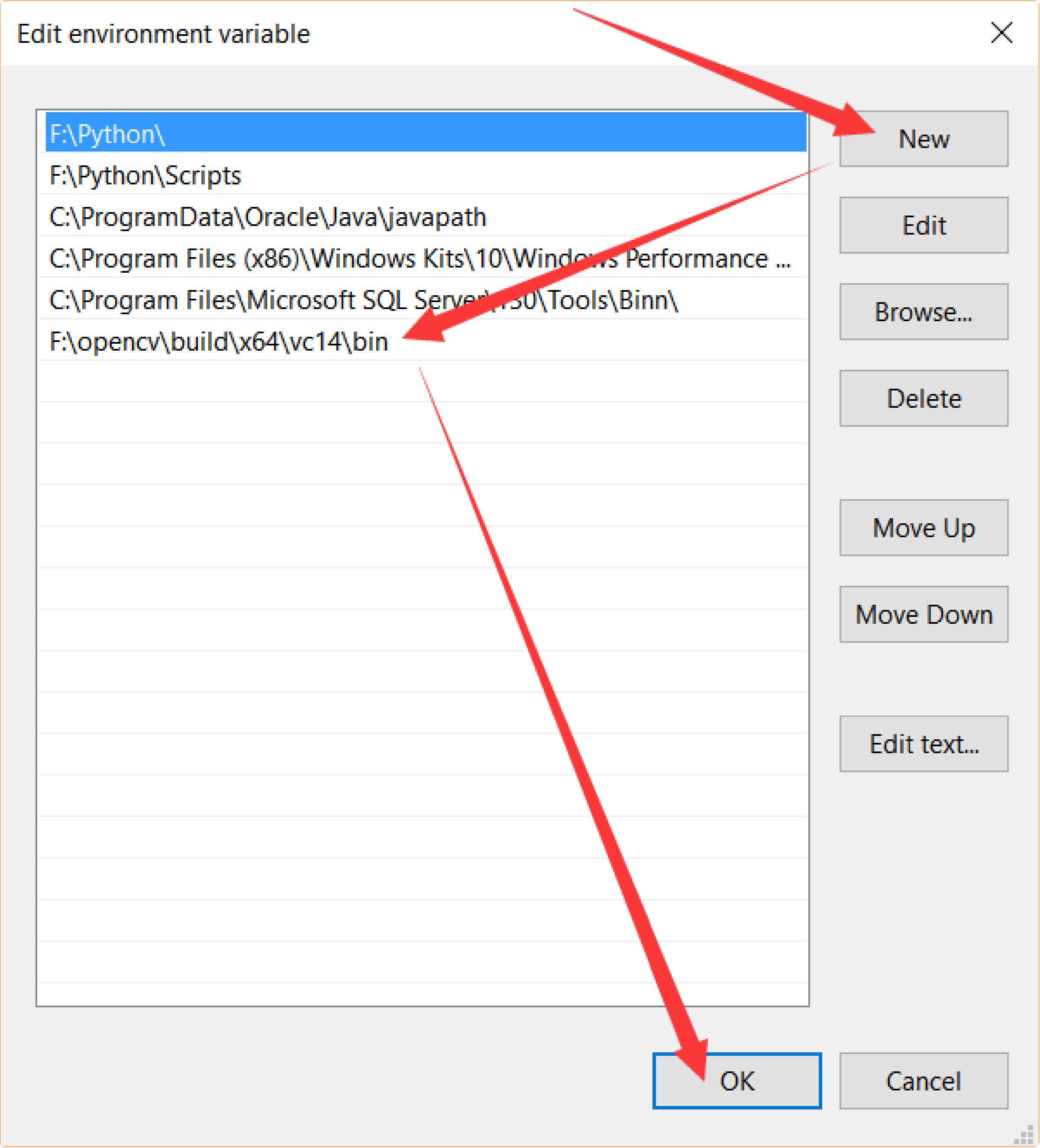
Variable	Value
PATH	F:\opencv\build\x86\vc11\bin;C:\Program Files\Intel\WiFi\bin...
TEMP	%USERPROFILE%\AppData\Local\Temp
TMP	%USERPROFILE%\AppData\Local\Temp

System variables

Variable	Value
AVR32_HOME	f:\WinAVR-20100110
CARBON_MEM_DISABLE	1
ComSpec	C:\WINDOWS\system32\cmd.exe
NUMBER_OF_PROCESSORS	8
OS	Windows_NT
PATH	C:\Program Files (x86)\Windows Kits\10\Windows Performanc...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTU...	AMD64

2. Edit the PATH environment variables and Add a new environment variable, then give it the value of `F:\opencv\build\x64\vc14\bin`. Note that change the value depends on the path where you have extracted your OpenCV in

step 2.

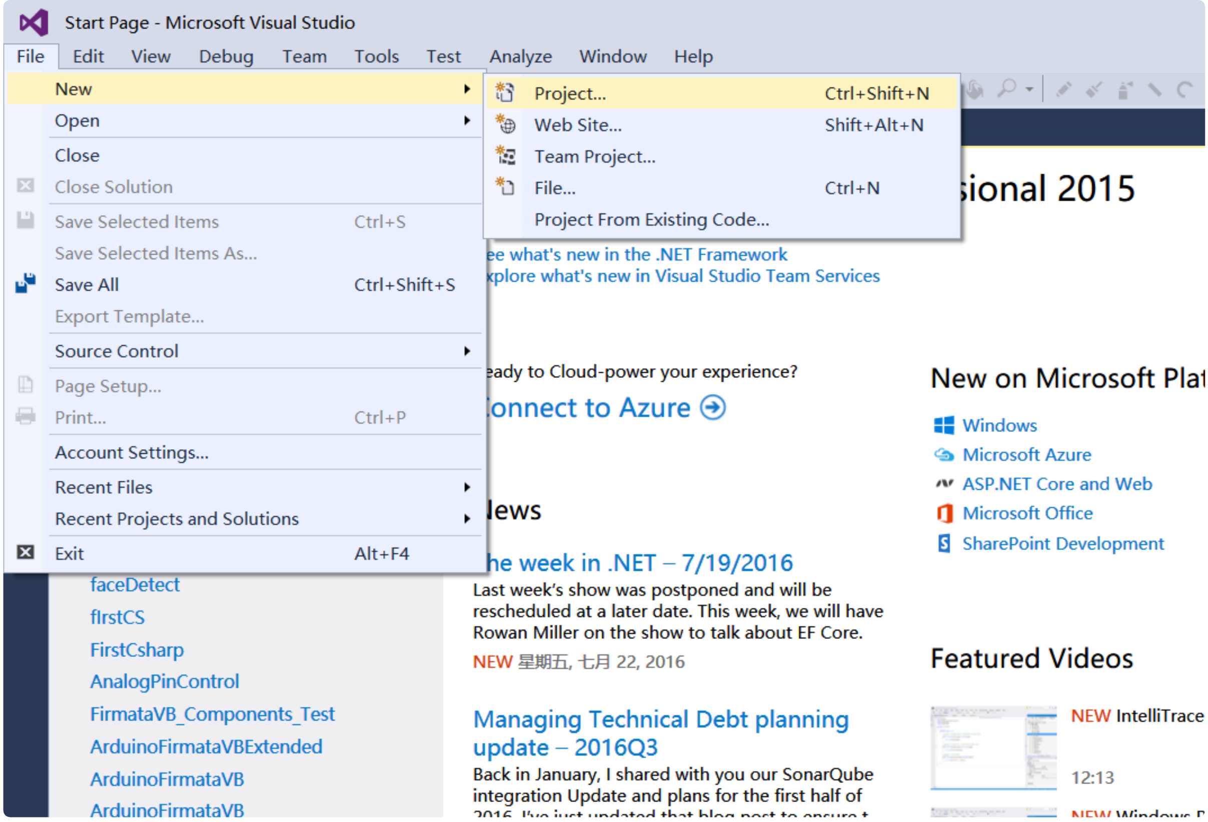


Edit environment variable

F:\Python\
F:\Python\Scripts
C:\ProgramData\Oracle\Java\javapath
C:\Program Files (x86)\Windows Kits\10\Windows Performance ...
C:\Program Files\Microsoft SQL Server\130\Tools\Binn\
F:\opencv\build\x64\vc14\bin

STEP 3: CREATE A NEW PROJECT IN VISUAL STUDIO 2015

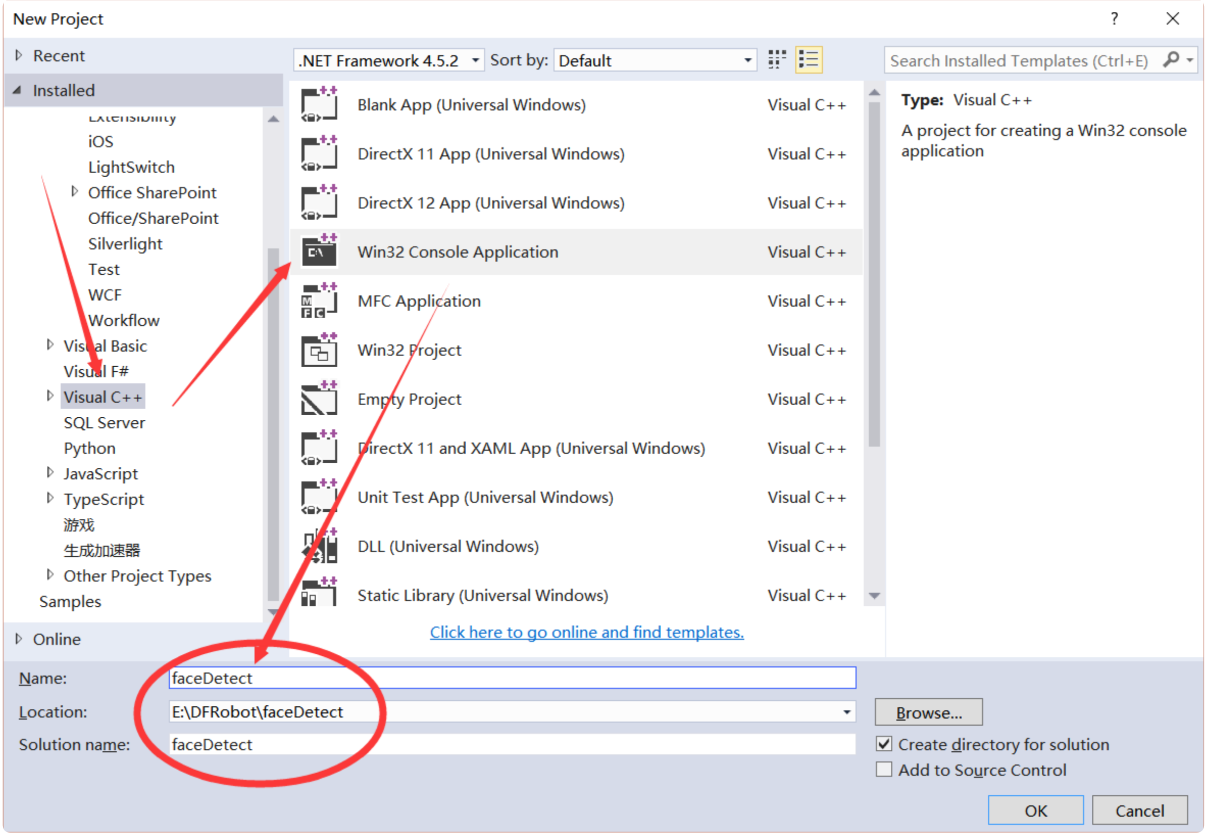
1. In Visual Studio 2015, create a new project to follow the steps in turns as the following figure shows



Last week's show was postponed and will be rescheduled at a later date. This week, we will have Rowan Miller on the show to talk about EF Core.

Back in January, I shared with you our SonarQube integration Update and plans for the first half of

2. Select Win32 Console Application in Visual C++, then name your project and select a directory to store it


New Project
?
Recent
Installed
Extensibility
iOS
LightSwitch
Office SharePoint
Office/SharePoint
Silverlight
Test
WCF
Workflow
Visual Basic
Visual F#
Visual C++
SQL Server
Python
JavaScript
TypeScript
游戏
生成加速器
Other Project Types
Samples
Online
.NET Framework 4.5.2
Sort by: Default
Search Installed Templates (Ctrl+E)
Blank App (Universal Windows) Visual C++
DirectX 11 App (Universal Windows) Visual C++
DirectX 12 App (Universal Windows) Visual C++
Win32 Console Application Visual C++
MFC Application Visual C++
Win32 Project Visual C++
Empty Project Visual C++
DirectX 11 and XAML App (Universal Windows) Visual C++
Unit Test App (Universal Windows) Visual C++
DLL (Universal Windows) Visual C++
Static Library (Universal Windows) Visual C++
Click here to go online and find templates.
Browse...
OK
Cancel

Type: Visual C++

A project for creating a Win32 console application

Name: faceDetect

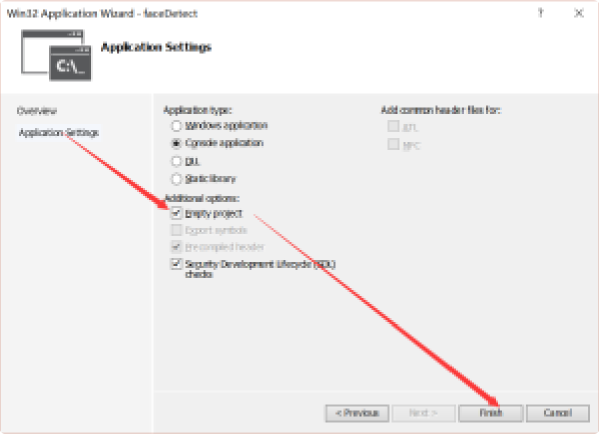
Location: E:\DFRobot\faceDetect

Solution name: faceDetect

☑ Create directory for solution

☐ Add to Source Control

3. Choose the empty project and click finish



Win32 Application Wizard - faceDetect

Application Settings

Overview

Application Settings

Application type:

○ Windows application

● Console application

○ DLL

○ Static library

Additional options:

☑ Empty project

□ Export symbols

☑ Precompiled header

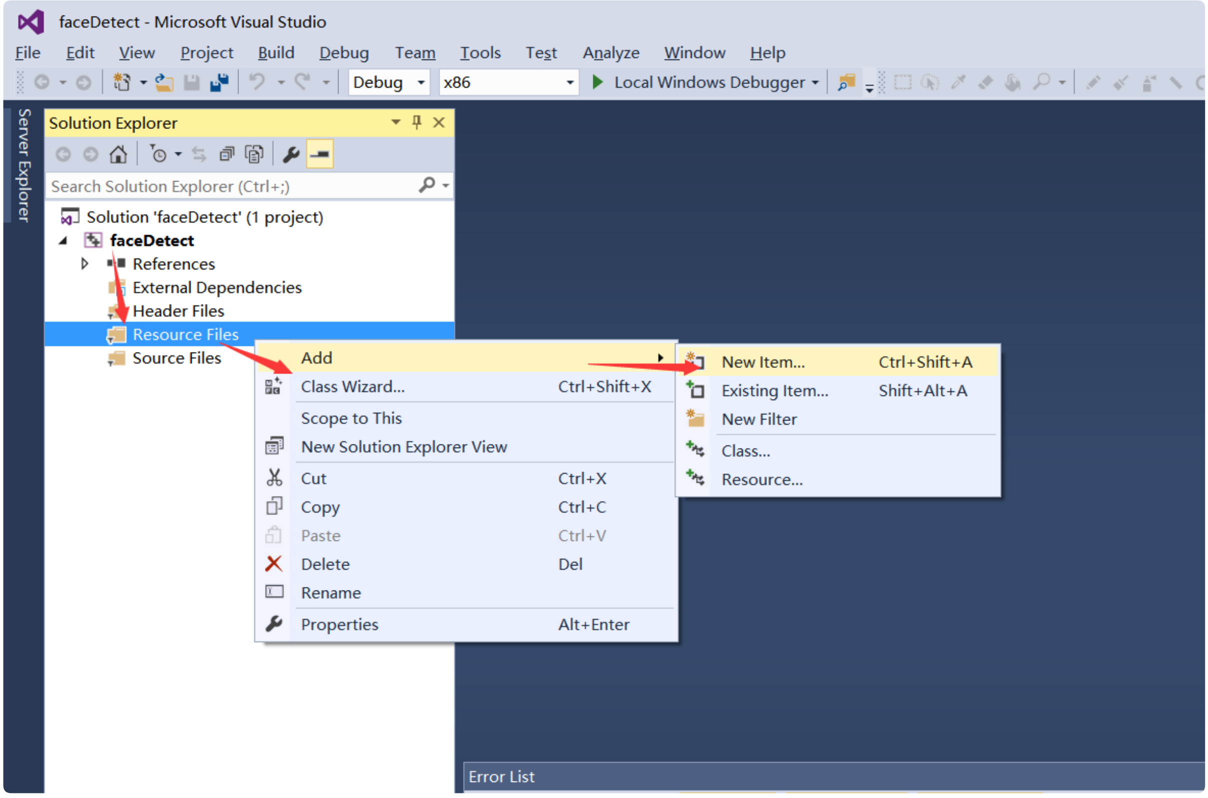
☑ Security Development Lifecycle (SDL) checks

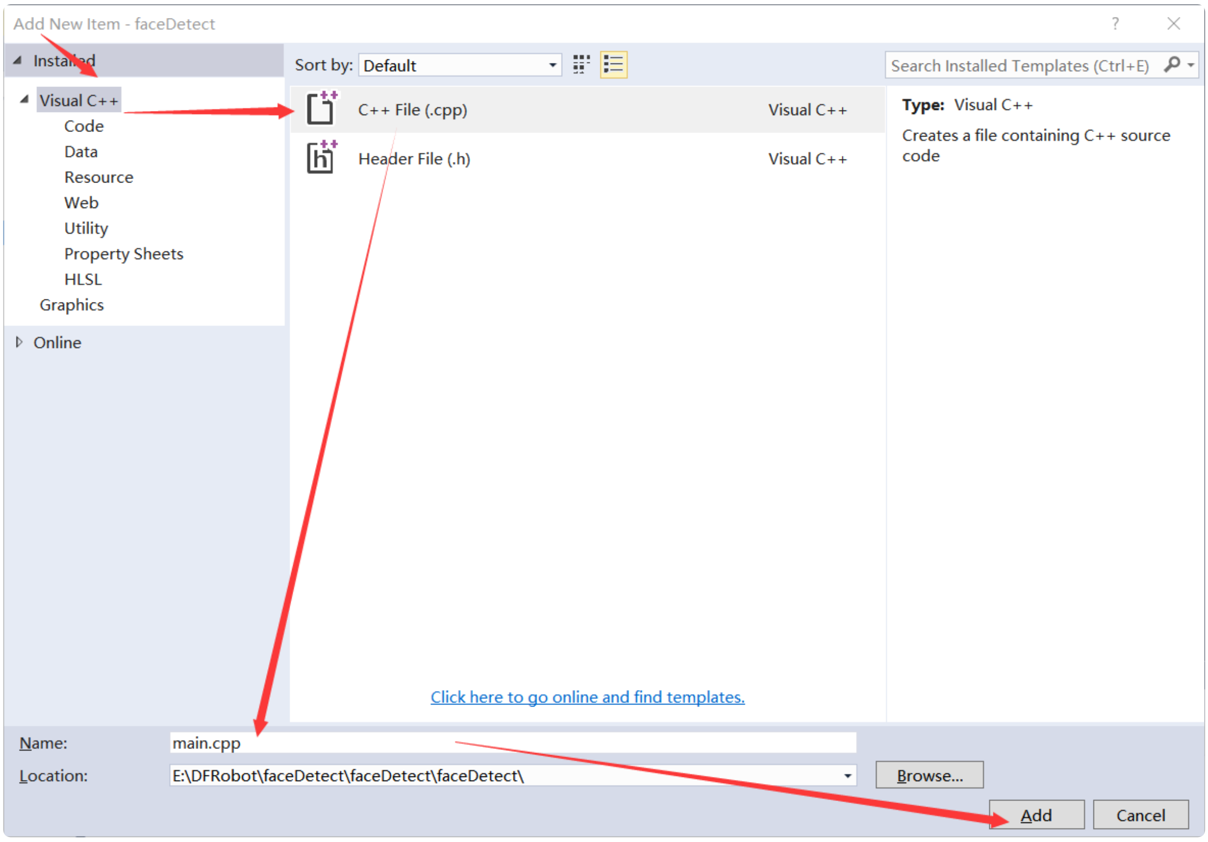
Add common header files for:

□ ATL

□ MFC

4. Add a new cpp file



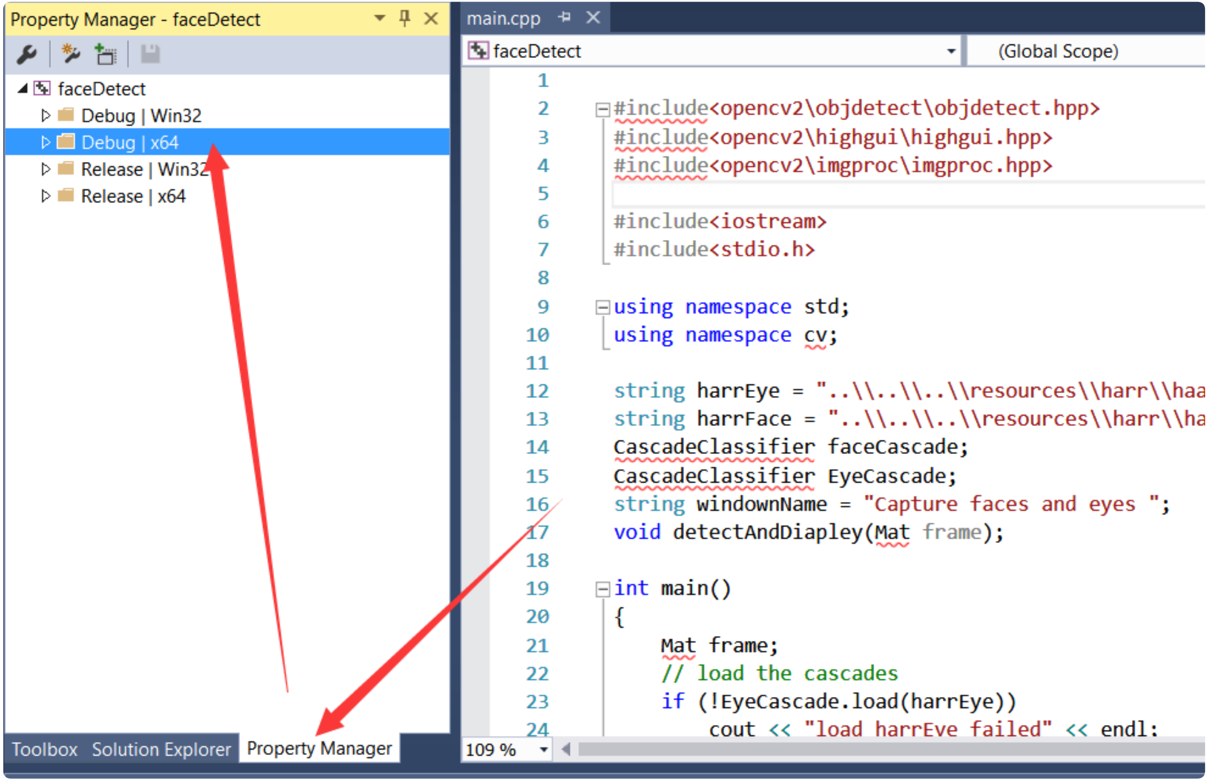

Add New Item - faceDetect
Installed
Visual C++
Code
Data
Resource
Web
Utility
Property Sheets
HLSL
Graphics
Online
Sort by: Default
Search Installed Templates (Ctrl+E)
C++ File (.cpp) Visual C++
Header File (.h) Visual C++
Type: Visual C++
Creates a file containing C++ source code
Click here to go online and find templates.
Browse...
Add
Cancel

Name: main.cpp

Location: E:\DFRobot\faceDetect\faceDetect\faceDetect\

STEP 4: CONFIGURE OPENCV IN VISUAL STUDIO 2015

1. Open the Property Manager and double click Debug|Win64



```

#include<opencv2\objdetect\objdetect.hpp>
#include<opencv2\highgui\highgui.hpp>
#include<opencv2\imgproc\imgproc.hpp>

#include<iostream>
#include<stdio.h>

using namespace std;
using namespace cv;

string harrEye = "..\\..\\..\\resources\\harr\\haa
string harrFace = "..\\..\\..\\resources\\harr\\ha
CascadeClassifier faceCascade;
CascadeClassifier EyeCascade;
string windownName = "Capture faces and eyes ";
void detectAndDiapley(Mat frame);

int main()
{
    Mat frame;
    // load the cascades
    if (!EyeCascade.load(harrEye))
        cout << "load harrEye failed" << endl;
```

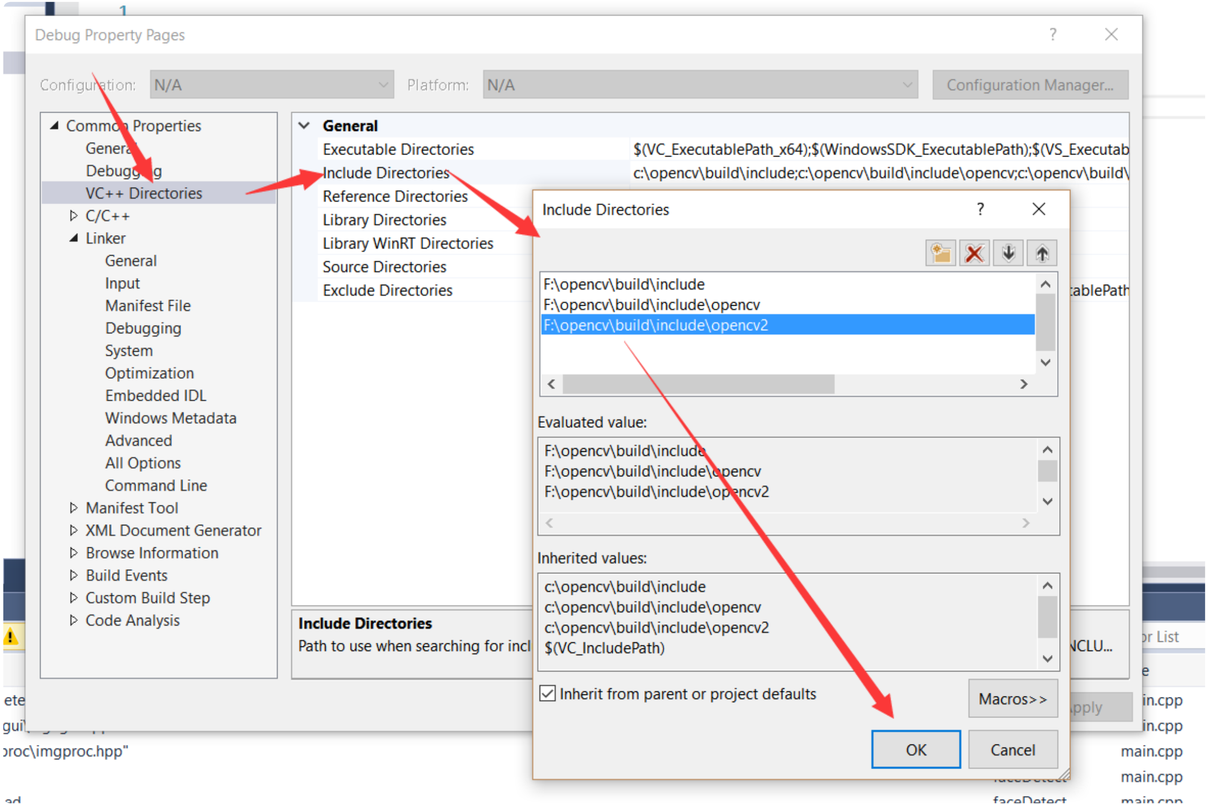
2. Select "Include Directories", and give it the following values:

`F:\opencv\build\include`

`F:\opencv\build\include\opencv`

`F:\opencv\build\include\opencv2`

Remember that change the value depending on the path you have extracted your OpenCV files to in step 2.



Executable Directories: $(VC_ExecutablePath_x64);$(WindowsSDK_ExecutablePath);$(VS_Executab

Include Directories: c:\opencv\build\include;c:\opencv\build\include\opencv;c:\opencv\build\

Include Directories
Path to use when searching for incl

Include Directories:

```
F:\opencv\build\include
F:\opencv\build\include\opencv
F:\opencv\build\include\opencv2
```

Evaluated value:

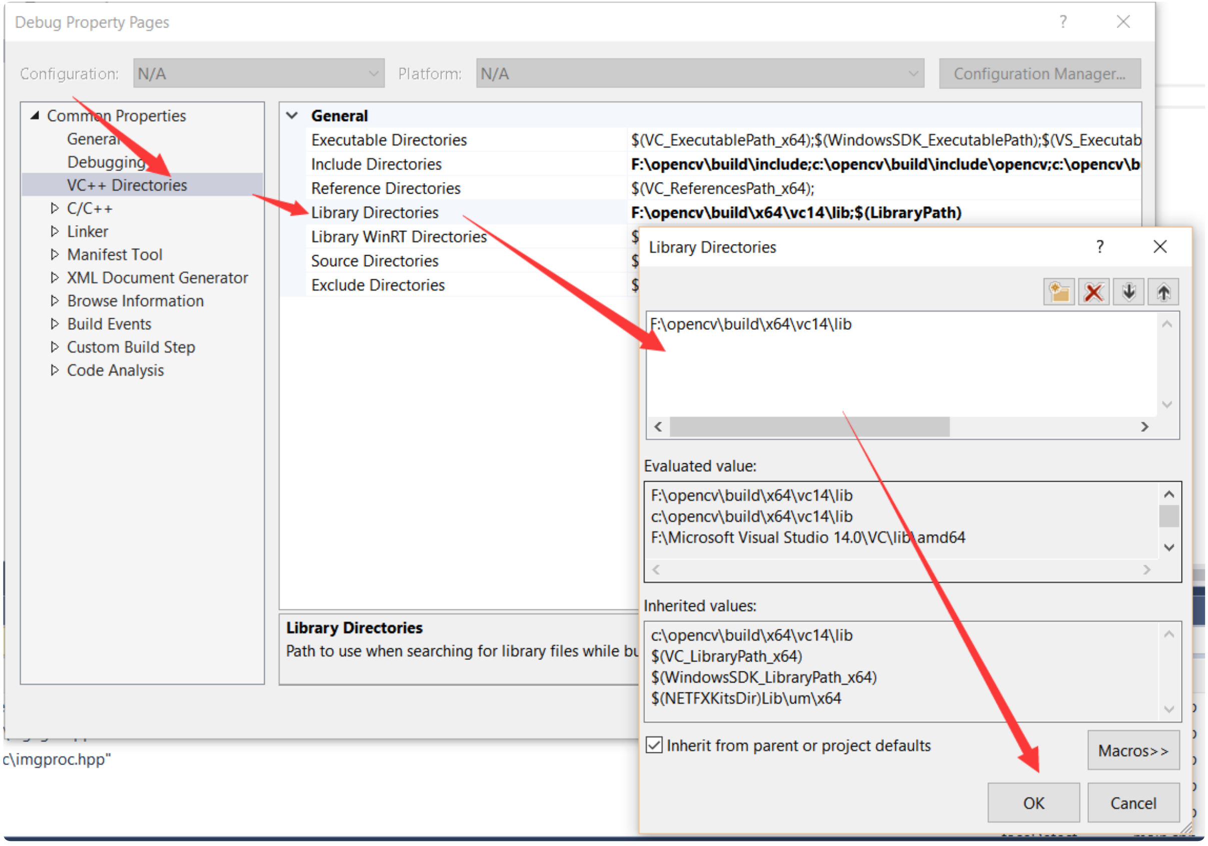
```
F:\opencv\build\include
F:\opencv\build\include\opencv
F:\opencv\build\include\opencv2
```

Inherited values:

```
c:\opencv\build\include
c:\opencv\build\include\opencv
c:\opencv\build\include\opencv2
$(VC_IncludePath)
```

☑ Inherit from parent or project defaults

3. Add Library Directories, give the value of `F:\opencv\build\x64\vc14\lib`. Remember that changing the values depends on the path where you have extracted your OpenCV in step 2



Executable Directories: $(VC_ExecutablePath_x64);$(WindowsSDK_ExecutablePath);$(VS_Executab

Include Directories: **F:\opencv\build\include;c:\opencv\build\include\opencv;c:\opencv\b**

Reference Directories: $(VC_ReferencesPath_x64);

Library Directories: **F:\opencv\build\x64\vc14\lib;$(LibraryPath)**

Library WinRT Directories

Source Directories

Exclude Directories

Library Directories
Path to use when searching for library files while bu

Library Directories:

```
F:\opencv\build\x64\vc14\lib
```

Evaluated value:

```
F:\opencv\build\x64\vc14\lib
c:\opencv\build\x64\vc14\lib
F:\Microsoft Visual Studio 14.0\VC\lib\amd64
```

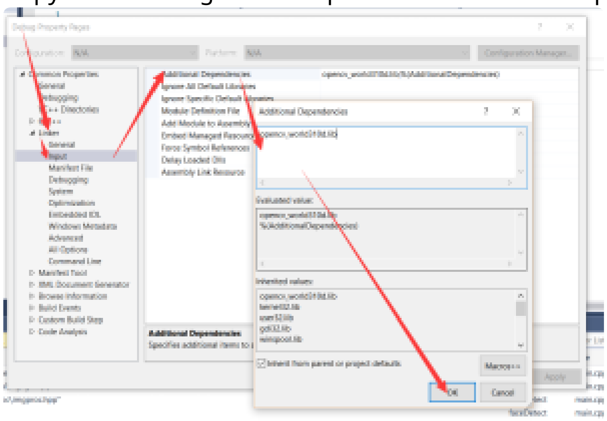
Inherited values:

```
c:\opencv\build\x64\vc14\lib
$(VC_LibraryPath_x64)
$(WindowsSDK_LibraryPath_x64)
$(NETFXKitsDir)Lib\um\x64
```

☑ Inherit from parent or project defaults

1.Add additional dependences

Copy the following item and paste it in additional Dependences blank `opencv_world310d.lib`



Additional Dependencies: opencv_world310d.lib;%(AdditionalDependencies)

Additional Dependencies

opencv_world310d.lib

Evaluated value:

opencv_world310d.lib
%(AdditionalDependencies)

Inherited values:

opencv_world310d.lib
kernel32.lib
user32.lib
gdi32.lib
winspool.lib

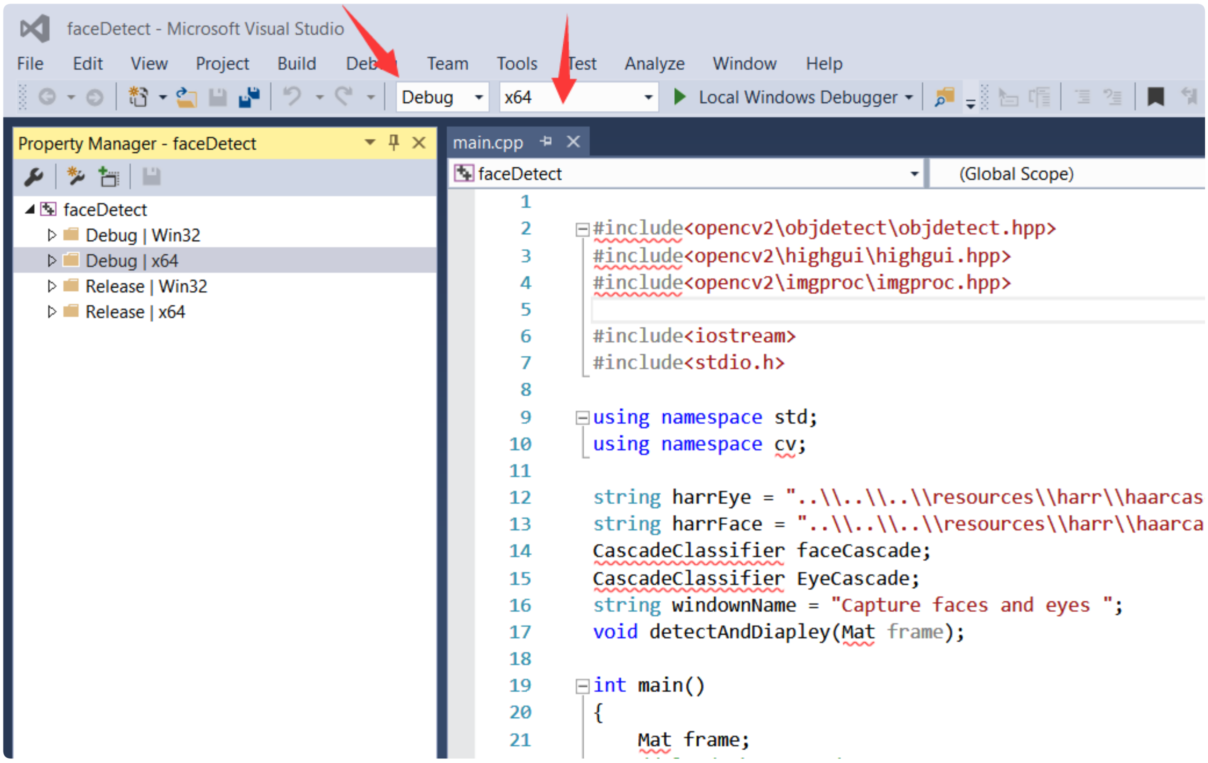
☑ Inherit from parent or project defaults

STEP 5: PASTE THE FOLLOWING CODE TO THE .CPP FILE YOUR ADDED IN STEP 4.

```
#include<opencv2\objdetect\objdetect.hpp>
#include<opencv2\highgui\highgui.hpp>
#include<opencv2\imgproc\imgproc.hpp>
#include<iostream>
#include<stdio.h>

using namespace std;
using namespace cv;

string harrEye = "..\\..\\..\\resources\\harr\\haarcascade_eye_tree_eyeglasses.xml";
string harrFace = "..\\..\\..\\resources\\harr\\haarcascade_frontalface_alt.xml";
CascadeClassifier faceCascade;
CascadeClassifier EyeCascade;
string windownName = "Capture faces and eyes ";
void detectAndDiapley(Mat frame);

int main()
{
    Mat frame;

    // load the cascades
    if (!EyeCascade.load(harrEye))
        cout << "load harrEye failed" << endl;
    if (!faceCascade.load(harrFace))
        cout << "load harrFace failed" << endl;

    // read the video stream
    VideoCapture capture(0);
    if (capture.isOpened())
    {
        while (true)
        {
            capture >> frame;

            // apply the cascaders to the frame
            if (!frame.empty())
            {
                detectAndDiapley(frame);
            }
            else
            {
                cout << "input video frame is empty" << endl;
            }
            if (waitKey(30) >= 0)break;
        }
    }
    return 0;
}

void detectAndDiapley(Mat frame)
{
    vector<Rect> faces;
    Mat frameGray;
    cvtColor(frame, frameGray, CV_BGR2GRAY);
    equalizeHist(frameGray, frameGray);

    //Detect faces
    faceCascade.detectMultiScale(frameGray, faces, 1.1, 2, 0 | CV_HAAR_SCALE_IMAGE, Size(30, 30));
    for (int i = 0; i < faces.size(); i++)
    {
        Point Vertex1(faces[i].x, faces[i].y);
        Point Vertex2(faces[i].x + faces[i].width, faces[i].y + faces[i].height);
        rectangle(frame, Vertex1, Vertex2, Scalar(0, 0, 255), 2, 8, 0);
        Mat faceROI = frameGray(faces[i]);
        vector<Rect> eyes;

        // detect eyes in each face
        EyeCascade.detectMultiScale(faceROI, eyes, 1.1, 2, 0 | CV_HAAR_SCALE_IMAGE, Size(30, 30));
        for (int j = 0; j < eyes.size(); j++)
        {
            Point center(faces[i].x + eyes[j].x + eyes[j].width / 2, faces[i].y + eyes[j].y + eyes[j].height / 2);
      int radius = cvRound((eyes[j].width + eyes[j].height)*0.25);
            Size axes(eyes[j].width / 2, 13);
            ellipse(frame, center, axes, 0, 0, 360, Scalar(255, 255, 0), 2, 8, 0);
        }
    }
    // show the faces and eyes detected
    imshow(windownName, frame);
}
```

STEP 6: DEBUG YOUR PROJECT

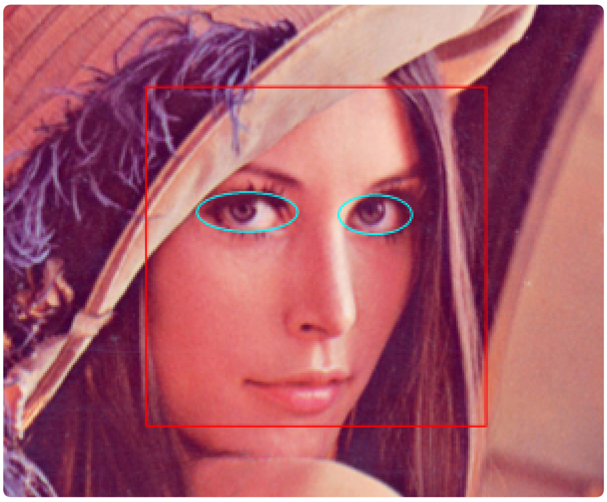
Set two options as following figure shows:



Debug x64 ▶ Local Windows Debugger

```
#include<opencv2\objdetect\objdetect.hpp>
#include<opencv2\highgui\highgui.hpp>
#include<opencv2\imgproc\imgproc.hpp>

#include<iostream>
#include<stdio.h>

using namespace std;
using namespace cv;

string harrEye = "..\\..\\..\\resources\\harr\\haarcas
string harrFace = "..\\..\\..\\resources\\harr\\haarca
CascadeClassifier faceCascade;
CascadeClassifier EyeCascade;
string windownName = "Capture faces and eyes ";
void detectAndDiapley(Mat frame);

int main()
{
    Mat frame;
```

Press F5 to execute the face detection project, your PC camera will turn on and your face and eyes will be highlighted like so:



REFERENCES:

1. http://www.michaelpsyllakis.com/install-opencv-on-visual-studio-2015-community-tutorial/
2. http://docs.opencv.org/2.4/opencv_tutorials.pdf
3. http://docs.opencv.org/2.4/opencv2refman.pdf

Node-RED

Node-RED (http://nodered.org) is a visual programming tool for Internet of Things. IBM is actively bringing its latest functions, including Waston and IOT service, into the toolbox ever since its acquisition.
Here we are taking the advantage of the Node-RED tool and the email service to build a simple but awesome IOT control unit.

You can remotely control your Arduino board by sending an email to your own account. The process roughly takes half an hour, with minimum coding required.

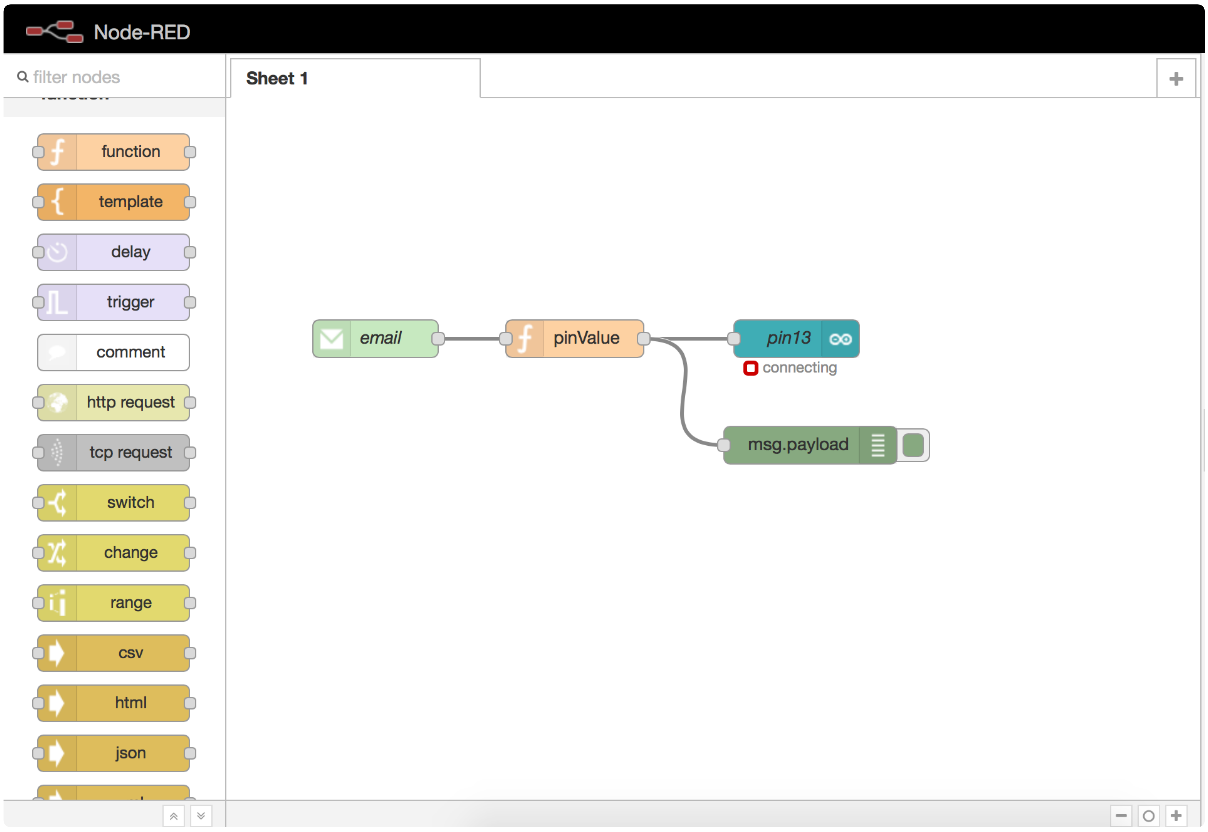
HARDWARE LIST

- LattePanda

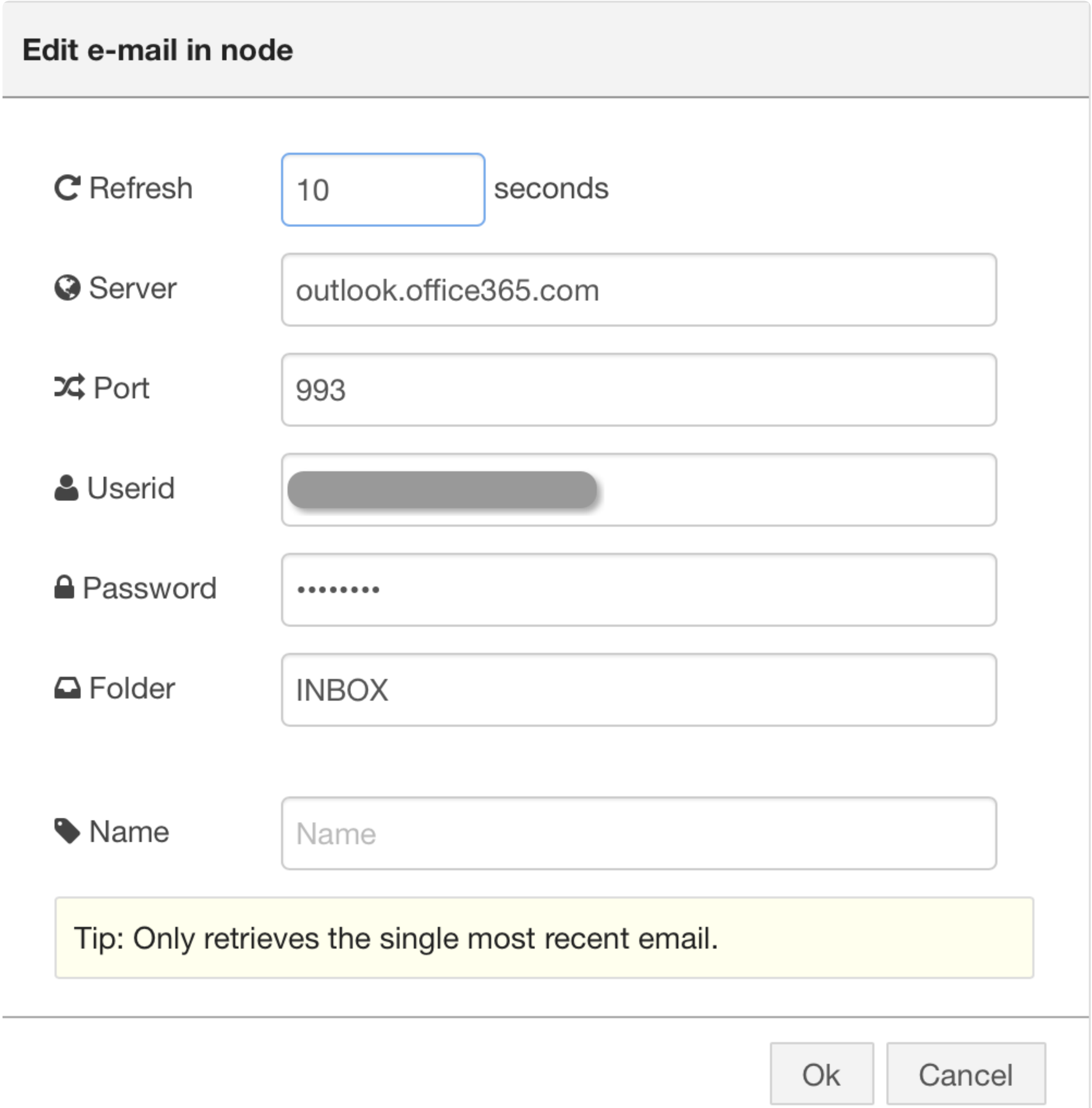
SOFTWARE SETUP

1. Flash Firmata into Your Uno Board
2. Install Node-RED (http://nodered.org/docs/getting-started/installation).
3. Please follow the official Arduino documentation from Node-RED (http://nodered.org/docs/hardware/arduino). As soon as you get past the first blinking example, you can start to go to the IOT part.

4. Build Control flow
   Here is the simple flow setup.



5. Configure Email Node
   Now we can configure the email node. I am using the Microsoft office email server. If you are using other email service providers, please check your server settings for setting up an `imap` client.



Edit e-mail in node

Refresh: 10 seconds

Server: outlook.office365.com

Port: 993

Userid: [redacted]

Password: •••••••••

Folder: INBOX

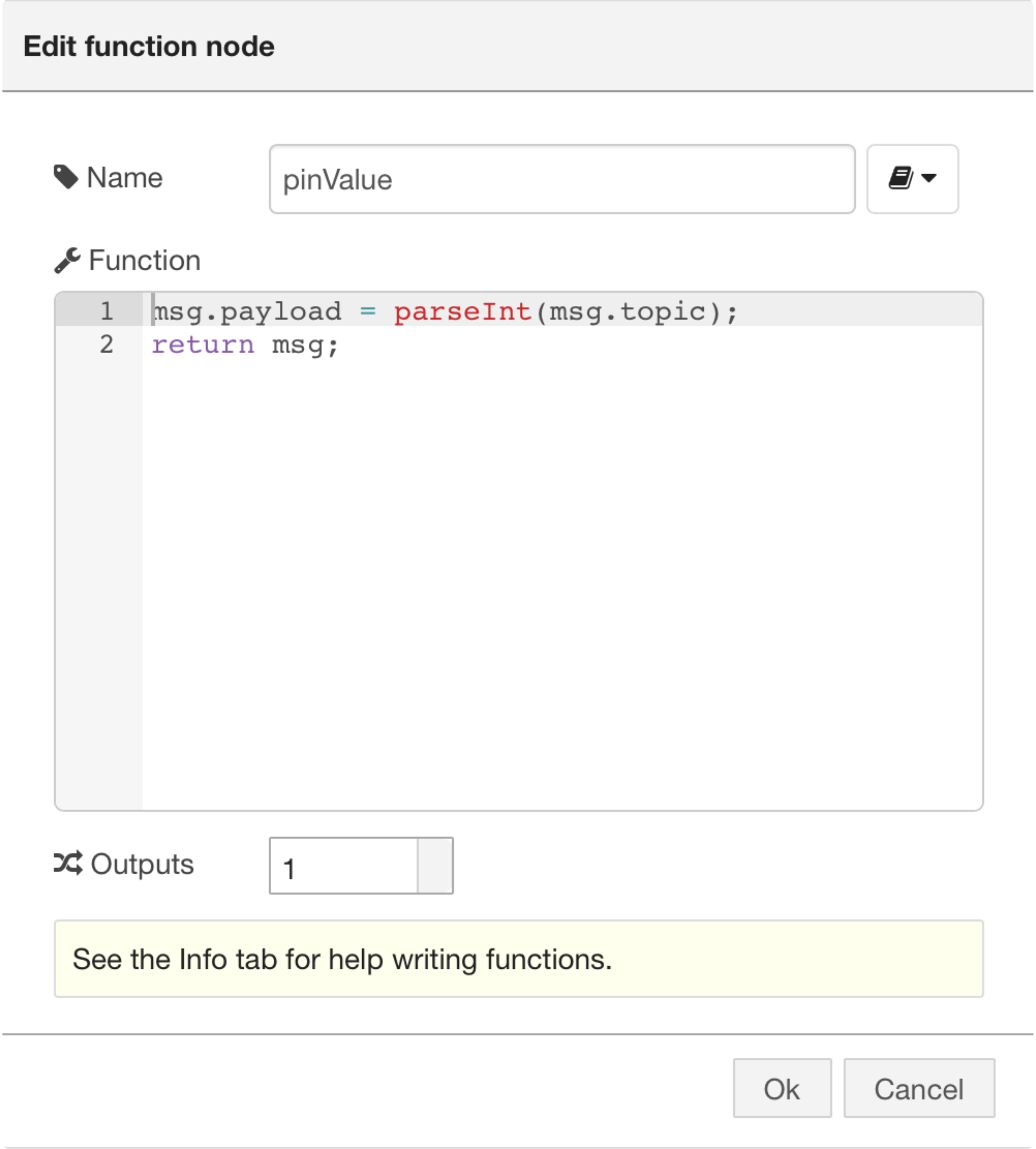
Name: Name

Tip: Only retrieves the single most recent email.

And also if you are using *Gmail*, you need to create an application specific password for third-party application authentication.

6. Configure Function Node We need a tiny bit of code to interpret the email messages before relay that on to your Arduino board. Specifically, we are using the email's **subject** to control our Arduino board. Because some server may automatically attach or transform the email **body** content or format for better user experience (in its own opinion), it would be easier to write working code for our example using only the **email subject**.
Here is the code
`javascript`

```
msg.payload = parseInt(msg.topic);
return msg;
```



Edit function node

Name: pinValue

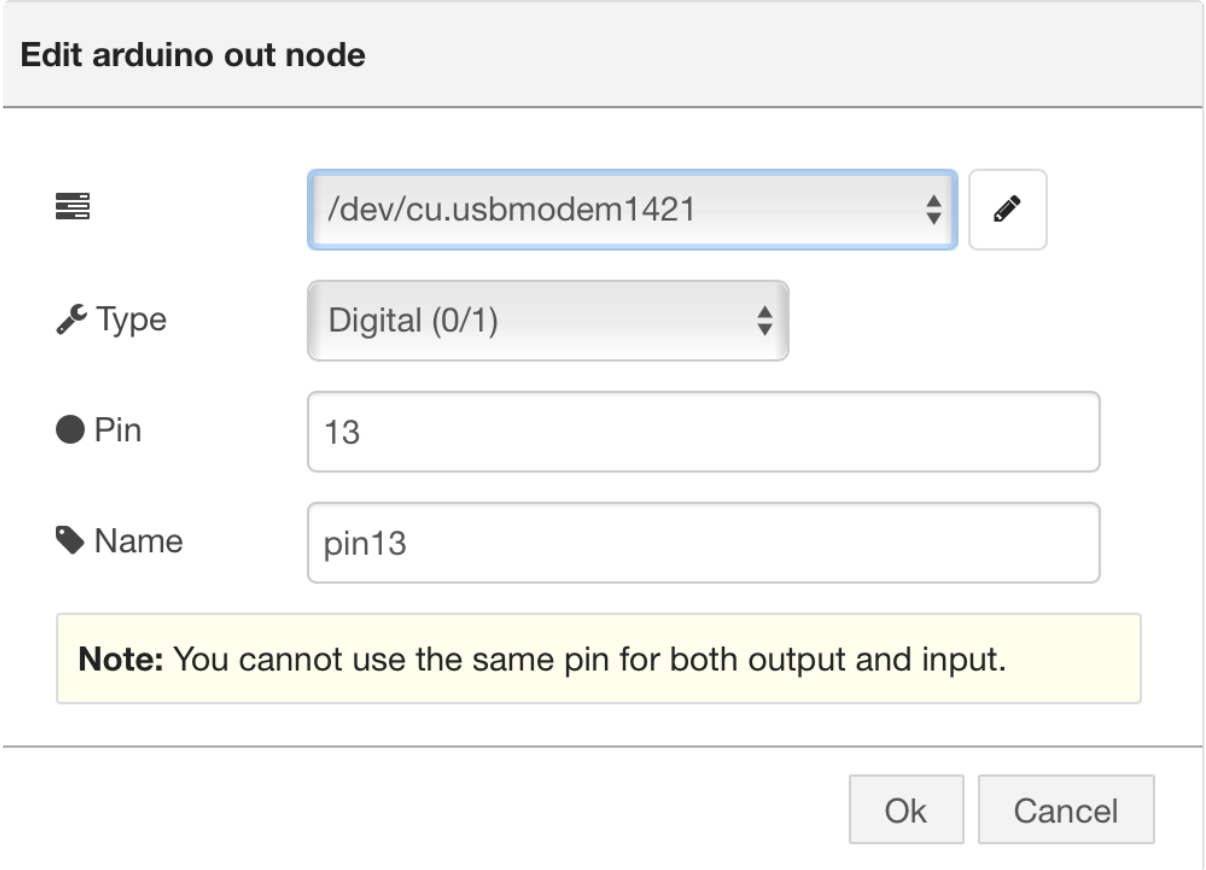
Function:

```
msg.payload = parseInt(msg.topic);
return msg;
```

Outputs: 1

See the Info tab for help writing functions.

7. Configure the Arduino LED Node You should have configured the LED node on pin 13 already if you have passed the example. But here is a screenshot just in case.



Edit arduino out node

/dev/cu.usbmodem1421

Type: Digital (0/1)

Pin: 13

Name: pin13

Note: You cannot use the same pin for both output and input.

TEST

Now you can send an email to your account, with the subject of a single digit: 1(or 0 if your pin 13 LED is already turned on). You should see the LED on the board change after about 10 or 20 seconds after you have sent the email.

You can leave the email body empty, we are not going to use that part for now. However you can choose to try more tricks with the email body once you feel more confident.

HOW IT WORKS

According to Node-RED documentation, the email node will try to get your latest email from the given mail folder between the given time interval. It will notice any new email, and pass on email information to our logical function. Then our function tries to interpret the subject to an integer and fill in the data our Arduino output pin needs. Finally the Arduino pin node controls the board using firmata protocol and changes the LED status.

Troubleshooting & FAQs

How do I know LattePanda is on?

There is one LED under LattePanda. When LP is on, you should see the PWR LED light up continuous and bright.

How do I re-install LattePanda Windows system?

A simple solution is useful for the **following situation**:

1. You can't startup your Windows system properly

2. Windows system crashes
3. You need to recover the default system after hacking Android or Linux.

Device required: Only a USB flash drive.
Please follow the simple steps below to re-install your Windows 10.

1. Format your USB disk to FAT32 and named it as "WINPE"
2. Unzip image.zip to root of USB flash disk (Download the official iso image from here):

   x86 version for 2G/32GB (https://drive.google.com/file/d/0B5YJ8UIGswVqeXFsbnRnWi1xXzg/view)
   x64 version for 4G/64GB (https://drive.google.com/file/d/0B5YJ8UIGswVqSm5qWF91MG1NWnc/view)

   Make sure that you extract the TAR to a FAT32 flash drive, labelled WINPE again.
3. Plug the USB disk in USB host
4. Power on the LattePanda
5. The system will install the image automatically. Just wait until the recovery is finished.
6. Unplug the USB disk. Enjoy hacking again.

What Boot Method Does LattePanda Use?

1. LattePanda BIOS uses UEFI boot
2. As we use different Windows images for two LP versions, the BIOS's bit depth should match the image. So LattePanda 2G/32GB uses a 32bit UEFI boot and 4G/64GB uses a 64bit UEFI boot.

If you are trying to install a different OS such as Linux, we have had some luck using multiarch images. We are still experimenting with this. We recommend looking on the forum (http://www.lattepanda.com/forum/viewtopic.php?f=5&t=275) for more information.

Common Adapters and USB cable recommendations:

Good power supply can ensure stable work of lattepanda. If you can’t start it, experience automatic shutdown and other problems in the process of using lattepanda, please check your adapter and USB cable so that whether they reach the specified requirements.
We tested some common adapter and USB cable so that it is convenient for your selection. The results are shown in the table below.

ADAPTER:

Model: AUKEY PA-T9 (American Standard)
Test Results: available
Peak current: 3.0A
link: Amazon (https://www.amazon.com/Charge-Charger-Galaxy-Qualcomm-Certified/dp/B018RR30TK/ref=sr_1_1?ie=UTF8&qid=1466745738&sr=8-1&keywords=AUKEY+PA-T9)

Model: AUKEY PA-T9-DE (European Standard)
Test Results: available
Peak current: 3.0A
link: Amazon (https://www.amazon.de/AUKEY-Ladeger%C3%A4t-Port-schwarz-schnell/dp/B01AHWOH54/ref=sr_1_1?ie=UTF8&qid=1469606331&sr=8-1&keywords=PA-T9)

Model: NILLKIN 5V@2A USB Adapter
Test Results: available
Peak current: 2.5A
link: DFRobot (http://www.dfrobot.com/index.php?route=product/product&product_id=933&search=lattepanda&description=true#.V1UDL2OxhVo), Amazon (http://www.amazon.com/Adapter-Nillkin®US-Standard-Charger-Samsung/dp/B015SL0Y3I/ref=sr_1_1?ie=UTF8&qid=1465189045&sr=8-1&keywords=NILLKIN+5V+2A+USB)

Model: APPLE ipad 10W usb power adapter
Test Results: available
Peak current: 2.8A
link: Amazon (http://www.amazon.com/Apple-A1357-Power-Adapter-iPhone/dp/B008QYIMO8/ref=sr_1_1?ie=UTF8&qid=1465189264&sr=8-1&keywords=APPLE+10W++adapter)

Model: JDB 5500
Test Results: available
Peak current: 2.8A
link: Amazon (http://www.amazon.com/Charger-JDB-Charge-Adapter-Samsung/dp/B00R26VRHA/ref=sr_1_1?ie=UTF8&qid=1465189309&sr=8-1&keywords=JDB+5500)

Model: HUAWEI HW-050200C3W
Test Results: available
Peak current: 2.5A
link: Amazon (http://www.amazon.com/Huawei-Original-Travel-Charger-Micro/dp/B017M0LNYC/ref=sr_1_1?ie=UTF8&qid=1465189465&sr=8-1&keywords=HW-050200C3W)

Model: HUAWEI HW-059200CHQ
Test Results: unavailable

USB CABLE:

Model: LattePanda Double Sided Micro USB Cable
Test Results: available
Wire Diameter: 0.9mm
link: DFRobot (http://www.dfrobot.com/index.php?route=product/product&product_id=1430&search=lattepanda&description=true#.V1UEk2OxhVo)

Model: REMAX RC-025T
Test Results: available
Wire Diameter: 0.8mm
link: Amazon (http://www.amazon.com/REMAX-RC-025t-Charger-Charging-Smartphone/dp/B01CNG4UHW/ref=sr_1_1?ie=UTF8&qid=1465189542&sr=8-1&keywords=REMAX+RC-025T)

Model: JDB 5500
Test Results: available
Wire Diameter: 0.7mm
link: Amazon (http://www.amazon.com/Charger-JDB-Charge-Adapter-Samsung/dp/B00R26VRHA/ref=sr_1_1?ie=UTF8&qid=1465189309&sr=8-1&keywords=JDB+5500)

OTHER

Specific to embedded and robot applications scenarios, lattepanda also provides two pairs of external power supply pin of 5v and Gnd, check pinout diagram to find the external power connector beside the MicroUSB connector.