International Journal of Advance Research in Science and Engineering
Volume No.06, Issue No. 12, December 2017
www.ijarse.com
IJARSE
ISSN: 2319-8354

WORKING PRINCIPLES OF RASPBERRY PI AND ARDUINO UNO IN WINDOWS

S.Govardhini[1], G.Sumalatha[2]

[1,2]Department of computer technology
Sri Krishna arts and Science College
Coimbatore

ABSTRACT

The Raspberry Pi is a credit-card sized computer that plugs into our TV and a keyboard, which can be used for many of the things that our average desktop does - spreadsheets, word-processing, games and it also plays high-definition video. Raspberry pi has two models .Model A and Model B.By using the raspberry pi and arduino uno to monitoring the framework(sensors,alarm).The Pi has GPIOs that are great for simple Boolean tasks (On or Off) and for reading a cheap temperature sensor. For more complex systems, we will want to use a microcontroller to do the heavy work. It has an ADC with multiple channels (Analog to Digital converter), PWM (Pulse Width Modulation) channels, and very accurate timing. For example, if we want to measure the electric energy consumption of our monitoring framework; we need a current transformer and a basic circuit that will output a voltage that we can measure using the ADC. We want to output something in between 1 and 0, let's say to fade an LED, we would use the PWM outputs. Finally, we need a microcontroller if we need something with very precise timing like a PID system, multiplexing an LED array, or controlling a stepper motor. In our case, we will be using the Arduino Uno as the microcontroller.

Keywords: Raspian, arduino, raspberry, IDE

I. INTRODUCTION

The Raspberry Pi is a credit-card sized computer that plugs into our TV and a keyboard, which can be used for many of the things that our average desktop does - spreadsheets, word-processing, games and it also plays high-definition video. The Raspberry Pi charitable foundation wants to see the device being used for monitoring framework.

Arduino is an open source computer hardware and software company, project, and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world

International Journal of Advance Research in Science and Engineering
Volume No.06, Issue No. 12, December 2017
www.ijarse.com
IJARSE
ISSN: 2319-8354

1.1Raspberry pi and Arduino circuit

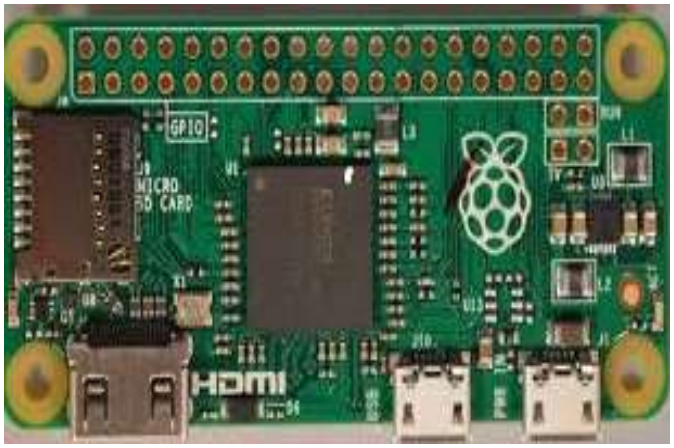

Fig. 1 Raspberry Pi circuit

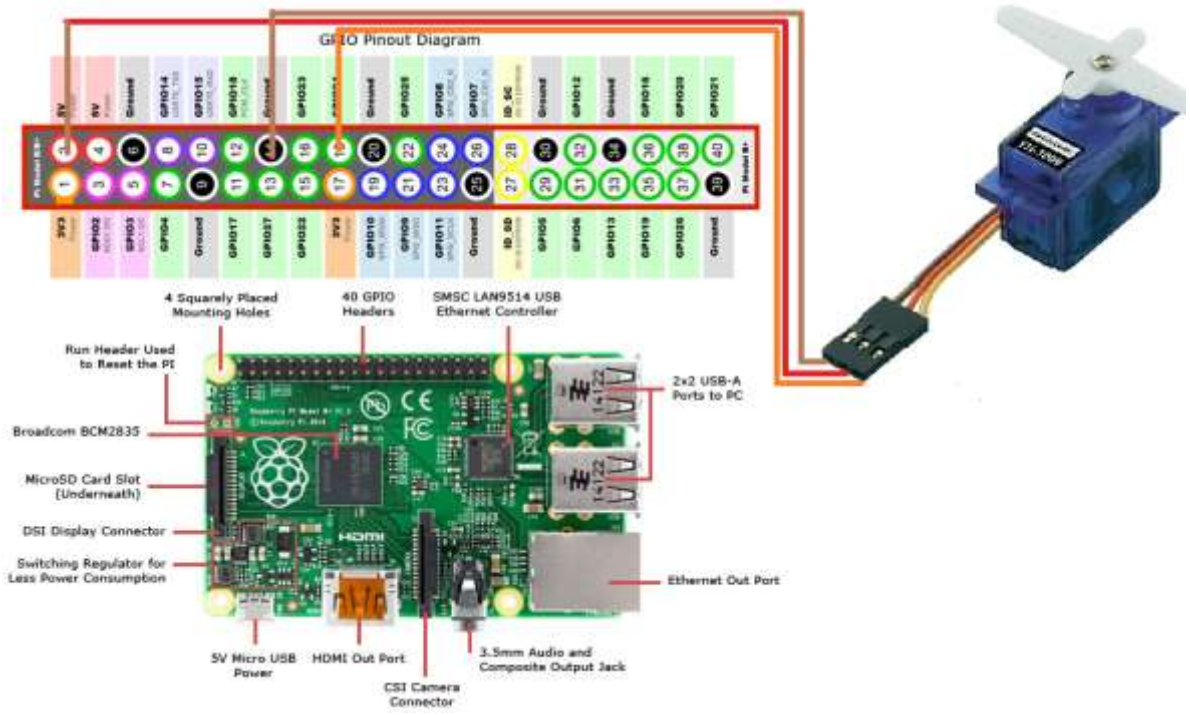

Fig. 2 Raspberry Pi Pin diagram

Fig.3 Arduino circuit

Installation of Raspberry pi in Raspberry kit

Step 1:go to the page https://www.raspberrypi.org/downloads/

Step 2: Inthat site click the Downloads

Step 3: In that downloads select the raspian

Step 4: click the raspian we have raspian stretch lite or raspian stretch with desktop

Step 5: Pick any one of that.

Step 6: in that I have pick the option raspian stetch with desktop it automatically save in zip file

Mac OS X

On Mac OS X, we will use a utility called dd. This is a Unix tool for flashing disk images, among other things.Once we've plugged in the SD card to our Mac, Open the "Disk Utility" application from the Utilities folder in Applications. We'll see a screen similar to the one below, listing our Mac's hard drive, any external hard drives plugged in, and our SD card. We can see my Raspberry Pi SD card is a 4GB SD card

Fig. 4 Mac OS Raspian

1.2Format the SD Card

If we haven't already, format the drive to FAT-32. This can be done from Disk Utility by clicking the "4.01 GB APPLE SD Card Reader Media" item in the list on the left. On our computer, it may be called something different if our SD card model or capacity is different than mine. *Make sure we select the SD card and not our hard drive*.

Once we've highlighted the SD card in the list, click the "Erase" tab that appeared on the right hand pane. We'll see a "Format" option and a "Name" field. Choose "MS-DOS (FAT)" as the format and enter a name. For the FAT-32 format, the name *must* be uppercase and we will not be able to enter lowercase letters. I chose "RASPBIAN."

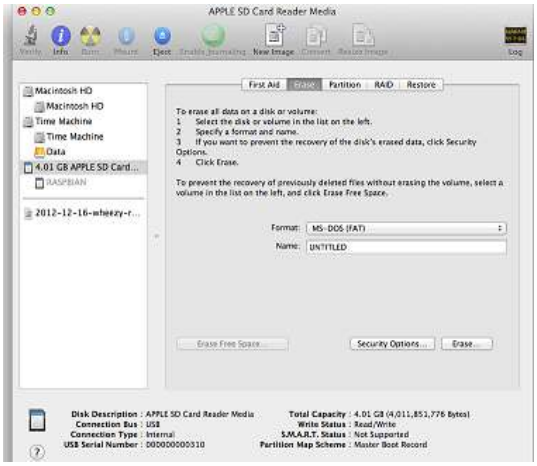

Fig. 5 Mac Os Raspian Apple Sd Card

After we've verified the selected item is actually our SD card and we have entered a name and selected "MS-DOS (FAT)," click the erase button to begin the format process. This should be fairly quick as long as we do not use a Secure Erase.

At this point, make sure we have a little bit of time to keep our computer on or awake. This process will likely take ten to twenty minutes or more.

1.3 Find the Disk Image

Open the Terminal application from the Utilities folder in Applications, and navigate to the directory where we extracted the .img file earlier. For those that used Safari, it will likely be the ~/Downloads folder. If we are unfamiliar with the Terminal and used Safari or another web browser that placed the file in our Downloads folder, we can use the following command: cd ~/Downloads.

Now, type ls in the Terminal and hit enter to list all of the files in the directory, If we see the Raspbian image file (*not* the .zip file– this one will end in .img), we're in the right place. If not, try the step above again and make sure we cd into the right directory. Remember or write down the exact file name of the .img file.

We're going to list all of the attached disks to find the SD card's mount point. We do this with the command df -h. We'll see a list like the one below. Notice how I have a ton of different disks, but only one has a cell with the value /Volumes/RASPBIAN. If we named our drive something else, look for that instead.

The item I am looking for is the bottom row. Make a note of the first column value, which is /dev/disk1s4 for me. Now, because we want to write to the raw disk, we need to change the value we're going to use. Essentially, we want to add an "r" before "disk" and remove the "s4" at the end. So, while we see the value /dev/disk1s4, we'll need to remember the value /dev/rdisk1.

Once we've made a note of this value, we can now unmount the partition so that dd can flash the disk image.

1.4 Unmount the Volume

Open Disk Utility again and we'll see the SD card in the list to the left with the name we chose earlier. Right click the *name* of the SD card this time. In my case, it is "RASPBIAN." Do *not* click the "4.01 GB APPLE SD Card Reader Media" list item, because we are selecting the partition in this case. In the menu that pops up, click "Unmount."

1.5 Flash the Disk Image

Go back to the Terminal and type the following command, ensuring we replace the "[FILESYSTEM]" value with the one we noted earlier and the "[DISK IMAGE NAME]" with the proper file name obtained above.

sudo dd bs=1m if=[DISK IMAGE NAME] of=[FILESYSTEM]

For me, the command would look something like this:

sudo dd bs=1m if=2012-12-16-wheezy-raspbian.img of=/dev/rdisk4

Hit enter, and wait until the command completes. Once dd flashes the disk image, we can remove it from our Mac and plug it into our Raspberry Pi. The default username is pi and the default password is raspberry.

1.6 Windows

The recommended method for flashing an SD for use in a Raspberry Pi is a program called Win32DiskImager. The latest version can be found on the official website. Personally I recommend version 0.5, since the latest version (0.6) has a bug that may cause our entire hard drive to be flashed instead of the SD card.

Once we've downloaded the Win32DiskImager application and extracted the ZIP file, download the Raspbian distribution. This can be found on the Raspberry Pi website under the heading "Raspbian 'wheezy"'. Once the ZIP file downloads, extract the .img from the .zip.

In Win32DiskImager, ensure we select the correct drive letter for our SD card. In my case, the SD card was drive F:/. Ours may be different, so check in Windows Explorer to make sure we have the correct letter. Do *not* choose C:\, since that is our main hard drive.

Also, select the .img file we extracted from the Raspbian distribution above using the file picker. Once we have made sure we have the correct .img file and drive letter for our SD card, click "Write" (*not* read) to flash the SD card. This will take less than five minutes on average and we can see the current progress in the Win32DiskImager window. Once the flash completes, we can exit the program.

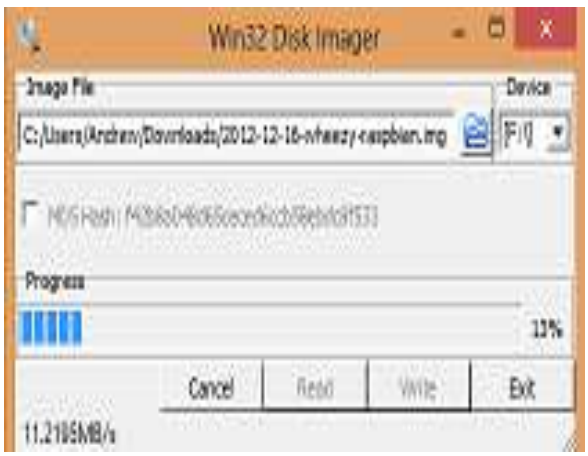

Fig. 6 Win 32 disk manager

Setting Up Raspberry Pi

Once we've flashed the disk image using the methods above, place the SD card into our Raspberry Pi, plug in the HDMI monitor, any keyboards and mice, and then the power cable. Our Raspberry Pi should begin to boot

and we should be able to see Raspbian on our screen.Complete the Raspbian setup, and our Raspberry Pi is now ready to be used!

II.INSTALLING ARDUINO IDE

2.1 Windows

This page will show us how to install and test the Arduino software with a Windows operating system (Windows 8, Windows 7, Vista, and XP).

2.1.1 Windows 8, 7, Vista, and XP

- Go to the Arduino download page and download the latest version of the Arduino software for Windows.
- When the download is finished, un-zip it and open up the Arduino folder to confirm that yes, there are indeed some files and sub-folders inside. The file structure is important so don't be moving any files around unless we really know what we're doing.
- Power up our Arduino by connecting our Arduino board to our computer with a USB cable (or FTDI connector if we're using an Arduino pro). We should see the an LED labed 'ON' light up. (this diagram shows the placement of the power LED on the UNO).
- If we're running Windows 8, we'll need to disable driver signing, so go see the Windows 8 section. If we're running Windows 7, Vista, or XP, we'll need to install some drivers, so head to the Windows 7, Vista, and XP section down below.

2.1.2 Windows 8

Windows 8 comes with a nice little security 'feature' that 'protects' we from unsigned driver installation. Some older versions of Arduino Uno come with unsigned drivers, so in order to use our Uno, we'll have to tell Windows to disable driver signing. This issue has been addressed in newer releases of the Arduino IDE, but if we run into issues, we can try this fix first.

To *temporarily* disable driver signing:

- From the Metro Start Screen, open Settings (move our mouse to the bottom-right-corner of the screen and wait for the pop-out bar to appear, then click the Gear icon)
- Click 'More PC Settings'
- Click 'General'
- Scroll down, and click 'Restart now' under 'Advanced startup'.
- Wait a bit.
- Click 'Troubleshoot'.

- Click ‘Advanced Options’
- Click ‘Windows Startup Settings’
- Click Restart.
- When our computer restarts, select ‘Disable driver signature enforcement‘ from the list.

To *permanently* disable driver signing (recommended, but has some minor security implications):

- Go to the metro start screen
- Type in “cmd”
- Right click “Command Prompt” and select “Run as Administrator” from the buttons on the bottom of our screen
- Type/paste in the following commands: bcdedit -set loadoptions DISABLE_INTEGRITY_CHECKS bcdedit -set TESTSIGNING ON
- Reboot!

2.1.3 Windows 7, Vista, and XP

Installing the Drivers for the Arduino Uno (from Arduino.cc)

- Plug in our board and wait for Windows to begin it’s driver installation process
- After a few moments, the process will fail, despite its best efforts
- Click on the Start Menu, and open up the Control Panel
- While in the Control Panel, navigate to System and Security. Next, click on System
- Once the System window is up, open the Device Manager
- Look under Ports (COM & LPT). We should see an open port named “Arduino UNO (COMxx)”. If there is no COM & LPT section, look under ‘Other Devices’ for ‘Unknown Device’

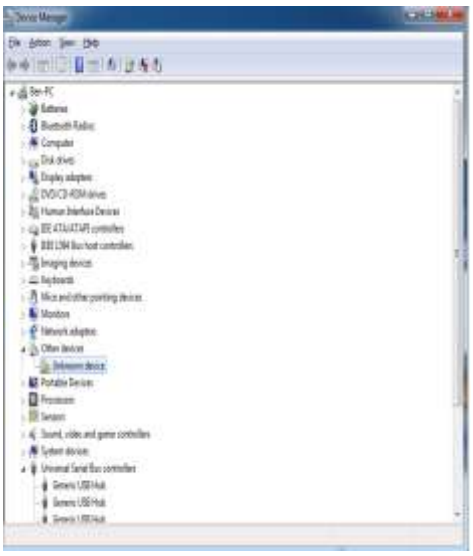

Fig. 7 Device manager

- Right click on the "Arduino UNO (COMxx)" or "Unknown Device" port and choose the "Update Driver Software" option
- Next, choose the "Browse my computer for Driver software" option

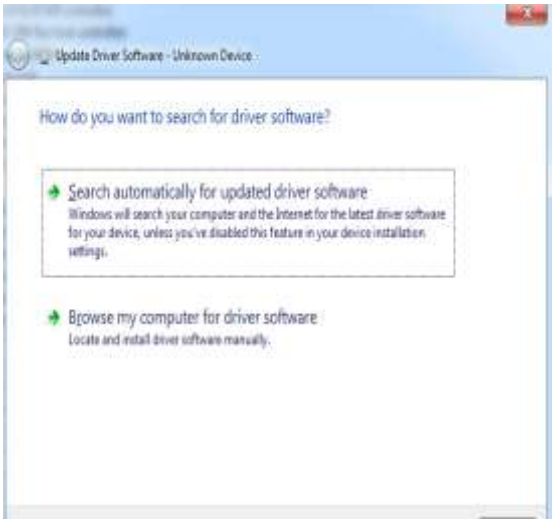

Fig.8 Update drive

- Finally, navigate to and select the Uno's driver file, named "ArduinoUNO.inf", located in the "Drivers" folder of the Arduino Software download (not the "FTDI USB Drivers" sub-directory). If we cannot see the .inf file, it is probably just hidden. We can select the 'drivers' folder with the 'search sub-folders' option selected instead.
- Windows will finish up the driver installation from there

For earlier versions of the Arduino boards (e.g.Arduino Duemilanove, Nano, or Diecimila) check out this page for specific directions.

2.1.4 Launch and Blink!

After following the appropriate steps for our software install, we are now ready to test our first program with our Arduino board!

- Launch the Arduino application
- If we disconnected our board, plug it back in
- Open the Blink example sketch by going to: File > Examples > 1.Basics > Blink

International Journal of Advance Research in Science and Engineering
Volume No.06, Issue No. 12, December 2017
www.ijarse.com
IJARSE
ISSN: 2319-8354

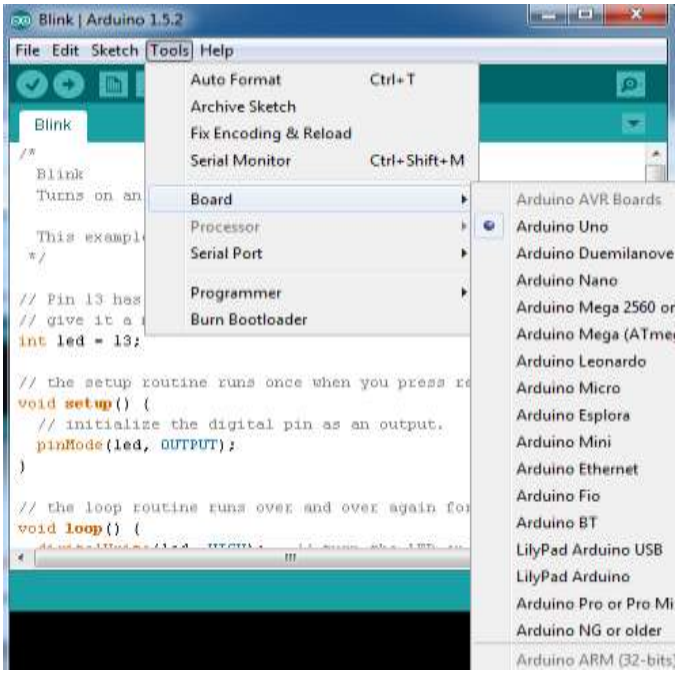

Fig.9 Blink in arduino Uno

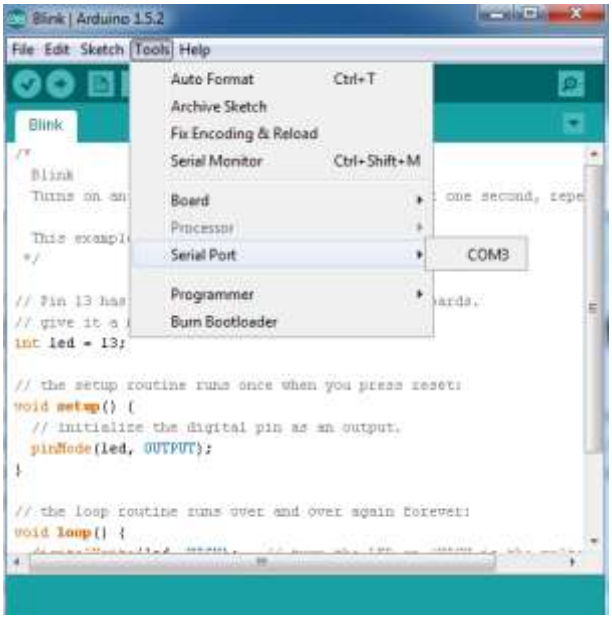

Fig.10 arduino serial port setup

 Ifwe're not sure which serial device is our Arduino, take a look at the available ports, then unplug our Arduino and look again. The one that disappeared is our Arduino.

 With our Arduino board connected, and the Blink sketch open, press the 'Upload' button

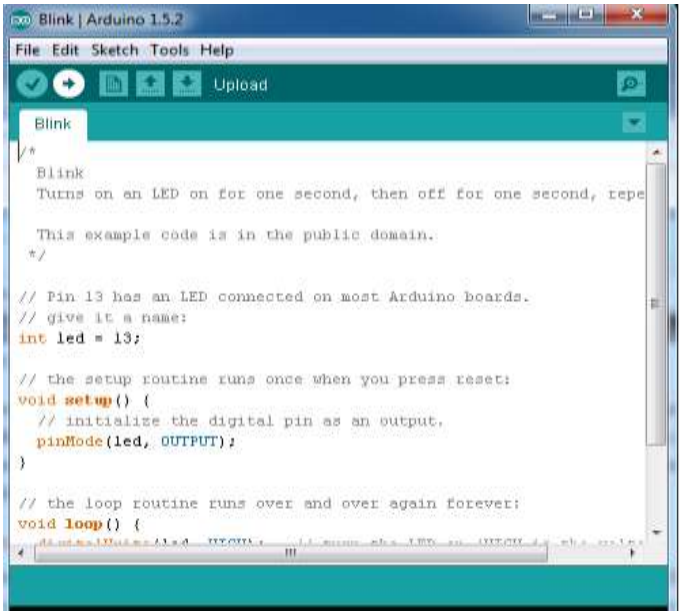

Fig. 11Blink in arduino

III.CONCLUSION

The Raspberry Pi is a powerful little beast and a great platform for building low-cost, but highly capable, embedded systems. The interfaces built into its GPIO connector make it easy to bolt on modules using simple low-cost electronics and a bit of configuration to create very functional and flexible systems. The inclusion of a dedicated camera interface and networking interfaces give you everything you could possible need for an Internet-connected home security system.

Journal Papers:

[1] A. Vichare, S. Verma, "Embedded Web Server for Home Appliances", International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622, pp. 190–193, 2012.

[2] Raspberry Pi Foundation. Raspberry pi 2 Model B. Retrieved 10 May 2015, from https://www.raspberrypi.org/products/raspberry-pi-2-model-b

[3] Raspberry Pi Foundation. Camera Module. Retrieved 10 May 2015, from https://www.raspberrypi.org/documentation/usage/camera/

[4] https://sourceforge.net/projects/win32diskimager/

[5]http://www.putty.org/download-putty

[6]https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html

[7]https://www.researchgate.net/profile/Vladimir_Vujovic/publication/280344140_ELI16_Maksimovic_Vujovic_Davidovic_Milosevic_Perisic/links/55b3368608ae9289a08594aa.pdf

[8]https://link.springer.com/article/10.1007/s00450-014-0290-8