Chapter 1

Introduction

Raspberry Pi is a credit-card sized computer manufactured and designed in the United Kingdom by the Raspberry Pi foundation with the intention of teaching basic computer science to school students and every other person interested in computer hardware, programming and DIY-Do-it Yourself projects.

The Raspberry Pi is manufactured in three board configurations through licensed manufacturing deals with Newark element14 (Premier Farnell), RS Components and Egoman. These companies sell the Raspberry Pi online. Egoman produces a version for distribution solely in China and Taiwan, which can be distinguished from other Pis by their red coloring and lack of FCC/CE marks. The hardware is the same across all manufacturers.

The Raspberry Pi has a Broadcom BCM2835 system on a chip (SoC), which includes an ARM1176JZF-S 700 MHz processor, VideoCore IV GPU and was originally shipped with 256 megabytes of RAM, later upgraded (Model B & Model B+) to 512 MB. It does not include a built-in hard disk or solid-state drive, but it uses an SD card for booting and persistent storage, with the Model B+ using a MicroSD.

The Foundation provides Debian and Arch Linux ARM distributions for download. Tools are available for Python as the main programming language, with support for BBC BASIC (via the RISC OS image or the Brandy Basic clone for Linux), C, Java and Perl.

As of February 2014, about 2.5 million boards had been sold.

The board is available online in India at a price of Rs. 3000.

Chapter 2

History of Computers

2.1 A brief history of ultra-cheap and small computers:

Computers were very expensive during the 1950's. Computers of that time were used in weather forecasting, plotting values of logarithmic functions and other complex calculations. They were huge machines with little or no operating systems. They needed dedicated air-conditioned rooms and special trained operators. Examples of these include the ENIAC, the ZUSE Z3 etc. Then vacuum tubes were replaced by bipolar transistors, which made those huge machines a bit smaller. The invention of Integrated Circuit(IC) gave computers a huge leap in terms of computing power and a basis for personal computers.

In those days portable computers were available which were essentially payroll machines or had other dedicated applications. Most of them were used for single purpose such as printing bills, as calculators, digital diaries etc.

Apple released Lisa in 1983 and marked a new point in the history of computers. For the first time a Graphical User Interface (GUI) was introduced in a computer that was cheap and most small businesses could afford. Microsoft tried the same thing and released DOS which was a huge success for the IBM PC. Now people could afford their own computer for their home. IBM PC sparked a massive explosion of personal computers. People started buying computers for their homes and offices like never before. Now kids could play games on them, adults could do their spreadsheets etc.

During the decade of 80's, the advent of laptops or notebook computers started. Many companies and vendors released the then portable computers which were the basis for the modern laptops. The first mass-produced microprocessor-based portable computer was the Osborne 1 in 1981, which used the CP/M operating system. Then within a few years, in 1986, Toshiba released the T1100, which they described as the "world's first mass-market laptop computer".

Figure 1: Osborne 1 Computer

Figure 2: Toshiba T1100

As technology improved during the 1990s, the usefulness and popularity of laptops increased. Correspondingly prices went down. Several developments specific to laptops like improved battery technology, power saving processors, improved Liquid Crystal Displays, improved storage etc. were quickly implemented, improving usability and performance.

2.2 History of Embedded Devices

In the earliest years of computers in the 1930–40s, computers were sometimes dedicated to a single task, but were far too large and expensive for most kinds of tasks performed by embedded computers of today. Over time however, the concept of programmable controllers evolved from traditional electromechanical sequencers, via solid state devices, to the use of computer technology.

One of the first recognizably modern embedded systems was the Apollo Guidance Computer, developed by Charles Stark Draper at the MIT Instrumentation Laboratory. At the project's inception, the Apollo guidance computer was considered the riskiest item in the Apollo project as it employed the then newly developed monolithic integrated circuits to reduce the size and weight. An early mass-produced embedded system was the Autonetics D-17 guidance computer for the Minuteman missile, released in 1961. It was built from transistor logic and had a hard disk for main memory. When the Minuteman II went into production in 1966, the D-17 was replaced with a new computer that was the first high-volume use of integrated circuits. This program alone reduced prices on quad NAND gate ICs from $1000/each to $3/each, permitting their use in commercial products.

Since these early applications in the 1960s, embedded systems have come down in price and there has been a dramatic rise in processing power and functionality. The first microprocessor for example, the Intel 4004, was designed for calculators and other small systems but still required many external memory and support chips. In 1978 National Engineering Manufacturers Association released a "standard" for programmable microcontrollers, including almost any computer-based controllers, such as single board computers, numerical, and event-based controllers.

As the cost of microprocessors and microcontrollers fell it became feasible to replace expensive knob-based analogue components such as potentiometers and variable capacitors with up/down buttons or knobs read out by a microprocessor even in some consumer products. By the mid-1980s, most of the common previously external system components had been integrated into the same chip as the processor and this modern form of the microcontroller allowed an even more widespread use, which by the end of the decade were the norm rather than the exception for almost all electronics devices.

The integration of microcontrollers has further increased the applications for which embedded systems are used into areas where traditionally a computer would not have been considered. A general purpose and comparatively low-cost microcontroller may often be programmed to fulfil the same role as a large number of separate components. Although in this context an embedded system is usually more complex than a traditional solution, most of the complexity is contained within the microcontroller itself. Very few additional components may be needed and most of the design effort is in the software. The intangible nature of software makes it much easier to prototype and test new revisions compared with the design and construction of a new circuit not using an embedded processor.

Chapter 3

Inception of Raspberry Pi

3.1 The Idea to create the Raspberry Pi

The idea behind a tiny and affordable computer for kids came in 2006, when Eben Upton, Rob Mullins, Jack Lang and Alan Mycroft, based at the University of Cambridge's Computer Laboratory, became concerned about the year-on-year decline in the numbers and skills levels of the A Level students applying to read Computer Science. From a situation in the 1990s where most of the kids applying were coming to interview as experienced hobbyist programmers, the landscape in the 2000s was very different; a typical applicant might only have done a little web design.

Something had changed the way kids were interacting with computers. A number of problems were identified: majority of curriculums with lessons on using Word and Excel, or writing webpages; the end of the dot-com boom; and the rise of the home PC and games console to replace the Amigas, BBC Micros, Spectrum ZX and Commodore 64 machines that people of an earlier generation learned to program on.

Figure 3: A complete Commodore 64 System

There isn't much any small group of people can do to address problems like an inadequate school curriculum or the end of a financial bubble. But those students felt that they could try to do something about the situation where computers had become so expensive and arcane that programming experimentation on them had to be forbidden by parents; and to find a platform that, like those old home computers, could boot into a programming environment. Thus came the idea of creating the device which kids could buy and learn programming or hardware on – The Raspberry Pi.

3.2 Initial Design Considerations

From 2006 to 2008 they created many designs and prototypes of what we now know as the Raspberry Pi. One of the earliest prototypes is shown below:

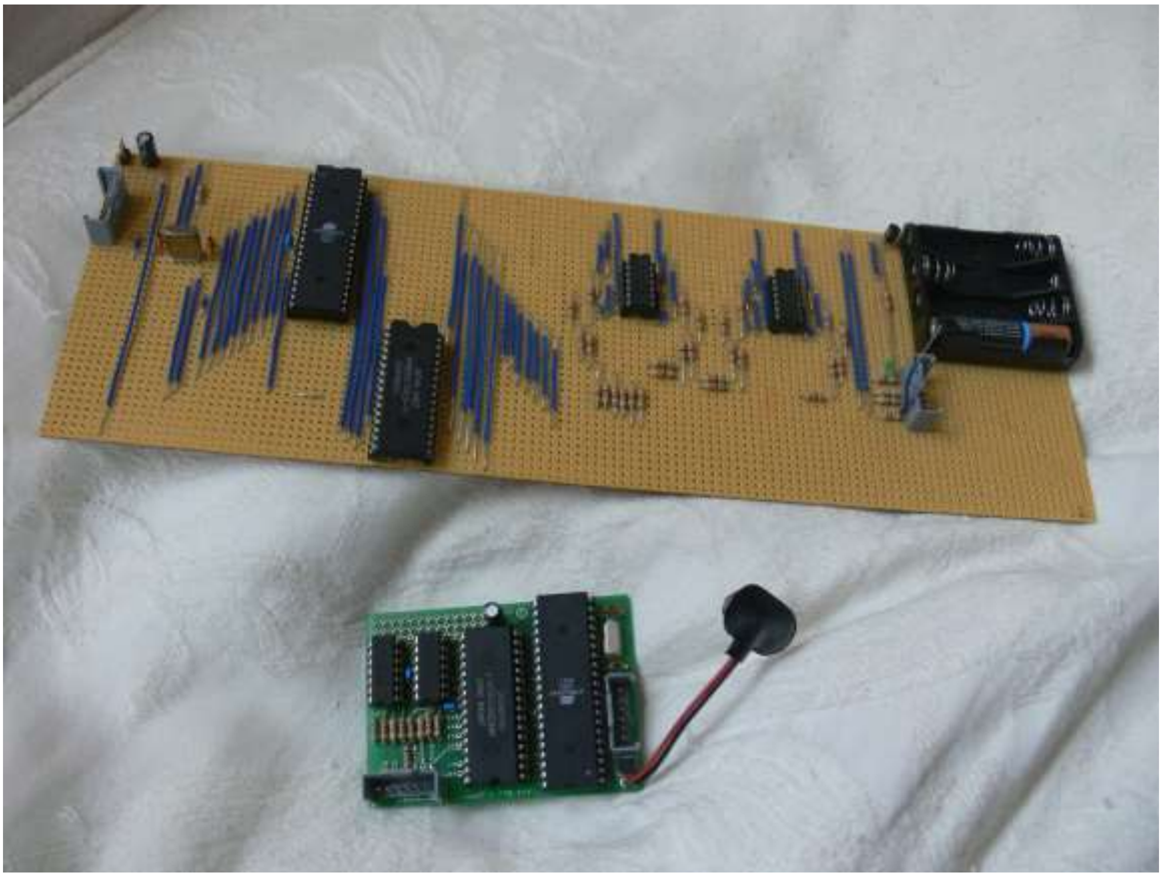

Figure 4: One of the earliest prototype of the Pi

These boards use an Atmel ATmega644 microcontroller clocked at 22.1MHz, and a 512K SRAM for data and frame buffer storage.

By 2008, processors designed for mobile devices were becoming more affordable, and powerful enough to provide excellent multimedia, a feature which would make the board desirable to kids who wouldn't initially be interested in a purely programming-oriented device. The project started to look very realisable and feasible. Eben (now a chip architect at Broadcom), Rob, Jack and Alan, teamed up with Pete Lomas, MD of hardware design and manufacture company Norcott Technologies, and David Braben, co-author of the BBC Micro game Elite, to form the Raspberry Pi Foundation to make it a reality. Three years later, the Raspberry Pi Model B entered mass production through licensed manufacture deals with Element 14/Premier Farnell and RS Electronics, and within two years it had sold over two million units!

Chapter 4

Hardware

4.1 Hardware Layout

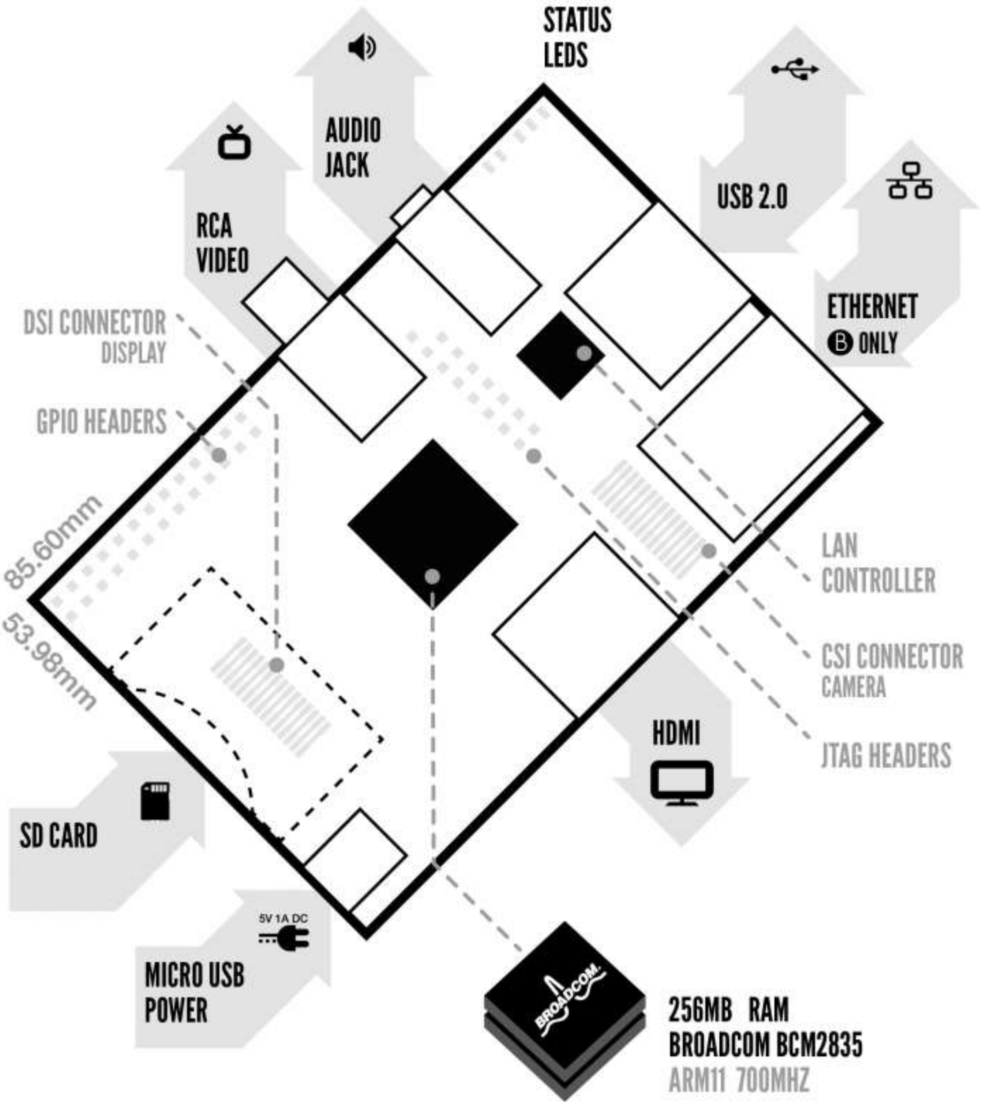

Figure 5: Block Diagram of Raspberry Pi

4.2 A brief description of the components on the Pi.

1) Processor / SoC (System on Chip)

   The Raspberry Pi has a Broadcom BCM2835 System on Chip module. It has a ARM1176JZF-S processor

   The Broadcom SoC used in the Raspberry Pi is equivalent to a chip used in an old smartphone (Android or iPhone). While operating at 700 MHz by default, the Raspberry Pi provides a real world performance roughly equivalent to the 0.041 GFLOPS. On the CPU level the performance is similar to a 300 MHz Pentium II of 1997-1999, but the GPU, however, provides 1 Gpixel/s, 1.5 Gtexel/s or 24 GFLOPS of general purpose compute and the graphics capabilities of the Raspberry Pi are roughly equivalent to the level of performance of the Xbox of 2001. The Raspberry Pi chip operating at 700 MHz by default, will not become hot enough to need a heatsink or special cooling.

2) Power source

   The Pi is a device which consumes 700mA or 3W or power. It is powered by a MicroUSB charger or the GPIO header. Any good smartphone charger will do the work of powering the Pi.

3) SD Card

   The Raspberry Pi does not have any onboard storage available. The operating system is loaded on a SD card which is inserted on the SD card slot on the Raspberry Pi. The operating system can be loaded on the card using a card reader on any computer.

4) GPIO

   GPIO – General Purpose Input Output

   General-purpose input/output (GPIO) is a generic pin on an integrated circuit whose behaviour, including whether it is an input or output pin, can be controlled by the user at run time.

GPIO pins have no special purpose defined, and go unused by default. The idea is that sometimes the system designer building a full system that uses the chip might find it useful to have a handful of additional digital control lines, and having these available from the chip can save the hassle of having to arrange additional circuitry to provide them.

GPIO capabilities may include:

- GPIO pins can be configured to be input or output
- GPIO pins can be enabled/disabled
- Input values are readable (typically high=1, low=0)
- Output values are writable/readable
- Input values can often be used as IRQs (typically for wakeup events)

The production Raspberry Pi board has a 26-pin 2.54 mm (100 mil) expansion header, marked as P1, arranged in a 2x13 strip. They provide 8 GPIO pins plus access to I²C, SPI, UART), as well as +3.3 V, +5 V and GND supply lines. Pin one is the pin in the first column and on the bottom row.

Figure 6: GPIO connector on RPi

5) DSI Connector

The Display Serial Interface (DSI) is a specification by the Mobile Industry Processor Interface (MIPI) Alliance aimed at reducing the cost of display controllers in a mobile device. It is commonly targeted at LCD and similar display technologies. It defines a serial bus and a communication protocol between the host (source of the image data) and the device (destination of the image data).

A DSI compatible LCD screen can be connected through the DSI connector, although it may require additional drivers to drive the display.

6) RCA Video

RCA Video outputs (PAL and NTSC) are available on all models of Raspberry Pi. Any television or screen with a RCA jack can be connected with the RPi.

Figure 7: RCA Video Connector

7) Audio Jack

A standard 3.5 mm TRS connector is available on the RPi for stereo audio output. Any headphone or 3.5mm audio cable can be connected directly. Although this jack cannot be used for taking audio input, USB mics or USB sound cards can be used.

8) Status LEDs

There are 5 status LEDs on the RPi that show the status of various activities as follows:

“OK” - SDCard Access (via GPIO16) - labelled as "OK" on Model B Rev1.0 boards and "ACT" on Model B Rev2.0 and Model A boards

“POWER” - 3.3 V Power - labelled as "PWR" on all boards

“FDX” - Full Duplex (LAN) **(Model B)** - labelled as "FDX" on all boards

“LNK” - Link/Activity (LAN) **(Model B)** - labelled as "LNK" on all boards

“10M/100” - 10/100Mbit (LAN) **(Model B)** - labelled (incorrectly) as "10M" on Model B Rev1.0 boards and "100" on Model B Rev2.0 and Model A boards

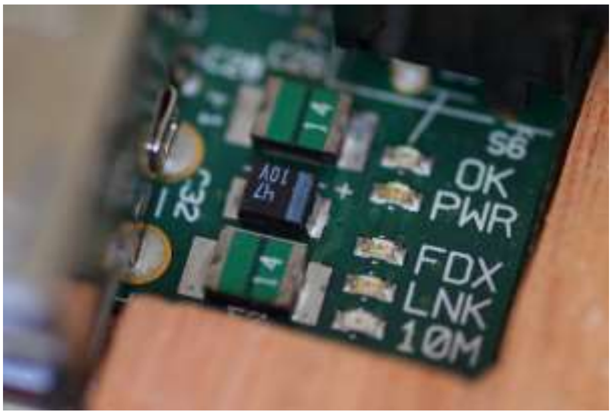

Figure 8: Status LEDs

9) USB 2.0 Port

USB 2.0 ports are the means to connect accessories such as mouse or keyboard to the Raspberry Pi. There is 1 port on Model A, 2 on Model B and 4 on Model B+. The number of ports can be increased by using an external powered USB hub which is available as a standard Pi accessory.

10) Ethernet

Ethernet port is available on Model B and B+. It can be connected to a network or internet using a standard LAN cable on the Ethernet port. The Ethernet ports are controlled by Microchip LAN9512 LAN controller chip.

11) CSI connector

CSI – Camera Serial Interface is a serial interface designed by MIPI (Mobile Industry Processor Interface) alliance aimed at interfacing digital cameras with a mobile processor.

The RPi foundation provides a camera specially made for the Pi which can be connected with the Pi using the CSI connector.

12) JTAG headers

JTAG is an acronym for ‘Joint Test Action Group', an organisation that started back in the mid 1980's to address test point access issues on PCB with surface mount devices. The organisation devised a method of access to device pins via a serial port that became known as the TAP (Test Access Port). In 1990 the method became a recognised international standard (IEEE Std 1149.1). Many thousands of devices now include this standardised port as a feature to allow test and design engineers to access pins.

13) HDMI

HDMI – High Definition Multimedia Interface

HDMI 1.3 a type A port is provided on the RPi to connect with HDMI screens.

4.3 Specifications

	Model A	Model B	Model B+
Target price:	US$25	US$35	
SoC:	Broadcom BCM2835 (CPU, GPU, DSP, SDRAM, and single USB port)		
CPU:	700 MHz ARM1176JZF-S core (ARM11 family, ARMv6 instruction set)		
GPU:	Broadcom VideoCore IV @ 250 MHz		
Memory (SDRAM):	256 MB (shared with GPU)	512 MB (shared with GPU) as of 15 October 2012	
USB 2.0 ports:	1 (direct from BCM2835 chip)	2 (via the on-board 3-port USB hub)	4 (via the on-board 5-port USB hub)
Video input:	15-pin MIPI camera interface (CSI) connector, used with the Raspberry Pi Camera Addon.		
Video outputs:	Composite RCA (PAL and NTSC) –in model B+ via 4-pole 3.5 mm jack, HDMI (rev 1.3 & 1.4), raw LCD Panels via DS		
Audio outputs:	3.5 mm jack, HDMI, and, as of revision 2 boards, I²S audio (also potentially for audio input)		
Onboard storage:	SD / MMC / SDIO card slot (3.3 V card power support only)		MicroSD

Onboard network:	None	10/100 Mbit/s Ethernet (8P8C) USB adapter on the third/fifth port of the USB hub	
Low-level peripherals:	8× GPIO, UART, I^2C bus, SPI bus with two chip selects, I^2S audio +3.3 V, +5 V, ground		17× GPIO
Power ratings:	300 mA (1.5 W)	700 mA (3.5 W)	600 mA (3.0 W)
Power source:	5 V via MicroUSB or GPIO header		
Size:	85.60 mm × 56 mm (3.370 in × 2.205 in) – not including protruding connectors		
Weight:	45 g (1.6 oz)		

Table 1 Specifications

4.4 Brief description of System on Chip (SoC)

Since smartphones and tablets are basically smaller computers, they require pretty much the same components we see in desktops and laptops in order to offer us all the amazing things they can do (apps, music and video playing, 3D gaming support, advanced wireless features, etc).

But smartphones and tablets do not offer the same amount of internal space as desktops and laptops for the various components needed such as the logic board, the processor, the RAM, the graphics card, and others. That means these internal parts need to be as small as possible, so that device manufacturers can use the remaining space to fit the device with a long-lasting battery life.

Thanks to the wonders of miniaturization, SoC manufacturers, like Qualcomm, Nvidia or Texas Instruments, can place some of those components on a single chip, the System on a Chip that powers smartphones.

A system on a chip or system on chip (SoC or SOC) is an integrated circuit (IC) that integrates all components of a computer or other electronic system into a single chip. It may contain digital, analog, mixed-signal, and often radio-frequency functions—all on a single chip substrate. SoCs are very common in the mobile electronics market because of their low power consumption. A typical application is in the area of embedded systems.

The contrast with a microcontroller is one of degree. Microcontrollers typically have under 100 kB of RAM (often just a few kilobytes) and often really are single-chip-systems, whereas the term SoC is typically used for more powerful processors, capable of running software such as the desktop versions of Windows and Linux, which need external memory chips (flash, RAM) to be useful, and which are used with various external peripherals. In short, for larger systems, the term system on a chip is a hyperbole, indicating technical direction more than reality: increasing chip integration to reduce manufacturing costs and to enable smaller systems. Many interesting systems are too complex to fit on just one chip built with a process optimized for just one of the system's tasks.

A typical SoC consists of:

- A microcontroller, microprocessor or DSP core(s). Some SoCs—called *multiprocessor system on chip* (MPSoC)—include more than one processor core.
- memory blocks including a selection of ROM, RAM, EEPROM and flash memory
- timing sources including oscillators and phase-locked loops
- peripherals including counter-timers, real-time timers and power-on reset generators
- external interfaces, including industry standards such as USB, FireWire, Ethernet, USART, SPI
- analog interfaces including ADCs and DACs
- voltage regulators and power management circuits
- A bus - either proprietary or industry-standard such as the AMBA bus from ARM Holdings - connects these blocks. DMA controllers route data directly between external interfaces and memory, bypassing the processor core and thereby increasing the data throughput of the SoC.

4.5 Accessories

Raspberry Pi being a very cheap computer has attracted millions of users around the world. Thus it has a large user base. Many enthusiasts have created accessories and peripherals for the Raspberry Pi. This range from USB hubs, motor controllers to temperature sensors. There are some official accessories for the RPi as follows:

Camera – On 14 May 2013, the foundation and the distributors RS Components & Premier Farnell/Element 14 launched the Raspberry Pi camera board with a firmware update to support it. The Raspberry Pi camera board contains a 5 MPixel sensor, and connects via a ribbon cable to the CSI connector on the Raspberry Pi. In Raspbian support can be enabled by the installing or upgrading to the latest version of the OS and then running Raspi-config and selecting the camera option. The cost of the camera module is 20 EUR in Europe (9 September 2013). and supports 1080p, 720p, 640x480p video. The footprint dimensions are 25 mm x 20 mm x 9 mm.

Gertboard – A Raspberry Pi Foundation sanctioned device designed for educational purposes, and expands the Raspberry Pi's GPIO pins to allow interface with and control of LEDs, switches, analog signals, sensors and other devices. It also includes an optional Arduino compatible controller to interface with the Pi. The Gertboard can be used to control motors, switches etc. for robotic projects.

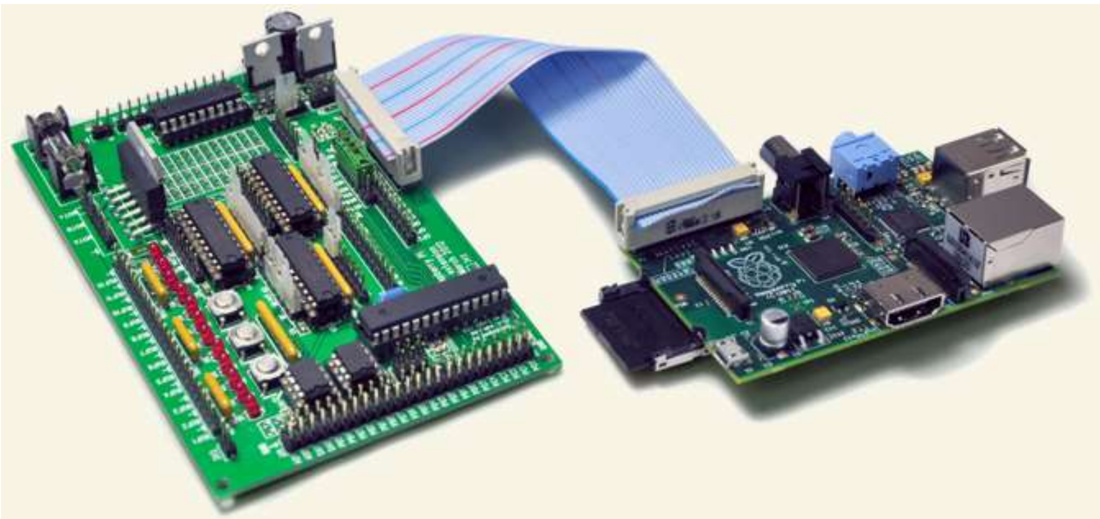

Figure 9: Gertboard (left) & Raspberry Pi(Right)

USB Hub – Although not an official accessory, it is a highly recommended accessory for the Pi. A powered USB Hub with 7 extra ports is available at almost all online stores. It is compulsory to use a USB Hub to connect external hard disks or other accessories that draw power from the USB ports, as the Pi cannot give power to them.

Chapter 5

Software

5.1 Operating System

The Raspberry Pi primarily uses Linux kernel-based operating systems. The ARM11 is based on version 6 of the ARM which is no longer supported by several popular versions of Linux, including Ubuntu. The install manager for Raspberry Pi is NOOBS. The OSs included with NOOBS are:

- Archlinux ARM
- OpenELEC
- Pidora (Fedora Remix)
- Raspbmc and the XBMC open source digital media center
- RISC OS – The operating system of the first ARM-based computer
- Raspbian (recommended) – Maintained independently of the Foundation; based on ARM hard-float (armhf)-Debian 7 'Wheezy' architecture port, that was designed for a newer ARMv7 processor whose binaries would not work on the Rapberry Pi, but Raspbian is compiled for the ARMv6 instruction set of the Raspberry Pi making it work but with slower performance. It provides some available deb software packages, pre-compiled software bundles. A minimum size of 2 GB SD card is required, but a 4 GB SD card or above is recommended. There is a Pi Store for exchanging programs. The 'Raspbian Server Edition (RSEv2.4)', is a stripped version with other software packages bundled as compared to the usual desktop computer oriented Raspbian.

5.2 Boot Process

The Raspberry Pi does not boot as a traditional computer. The VideoCore i.e. the Graphics processor actually boots before the ARM CPU.
The boot process of the Raspberry Pi can be explained as follows:

- When the power is turned on, the first bits of code to run is stored in a ROM chip in the SoC and is built into the Pi during manufacture. This is the called the **first-stage bootloader**.

- The SoC is hardwired to run this code on startup on a small RISC Core (**R**educed **I**nstruction **S**et **C**omputer). It is used to mount the FAT32 boot partition in the SDCard so that the **second-stage bootloader** can be accessed. So what is this 'second-stage bootloader' stored in the SD Card? It's **'bootcode.bin'.** This file can be seen while mount process of an operating system on the SD Card in windows.

- Now here's something tricky. The first-stage bootloader has not yet initialized the ARM CPU (meaning CPU is in **reset**) or the RAM. So, the second-stage bootloader also has to run on the GPU. The bootloader.bin file is loaded into the 128K 4 way set associative L2 cache of the GPU and then executed. This enables the RAM and loads **start.elf** which is also in the SD Card. This is the **third-stage bootloader** and is also the most important. It is the firmware for the GPU, meaning it contains the settings or in our case, has instructions to load the settings from **config.txt** which is also in the SD Card. We can think of the config.txt as the 'BIOS settings'.

- The **start.elf** also splits the RAM between the GPU and the ARM CPU. The ARM only has access the to the address space left over by the GPU address space. For example, if the GPU was allocated addresses from 0x000F000 – 0x0000FFFF, the ARM has access to addresses from 0x00000000 – 0x0000EFFF.

- The physical addresses perceived by the ARM core is actually mapped to another address in the VideoCore (0xC0000000 and beyond) by the MMU (Memory Management Unit) of the VideoCore**.**

- The config.txt is loaded after the split is done so the splitting amounts cannot be specified in the config.txt. However, different .elf files having different splits exist in the SD Card. So, depending on the requirement, the file can be renamed to start.elf and boot the Pi. In the Pi, the GPU is King!

- Other than loading config.txt and splitting RAM, the **start.elf** also loads **cmdline.txt** if it exists. It contains the command line parameters for whatever kernel that is to be loaded. This brings us to the final stage of the boot process. The **start.elf** finally loads **kernel.img** which is the binary file containing the OS kernel and releases the **reset** on the CPU. The ARM CPU then executes whatever instructions in the **kernel.img** thereby loading the operating system.

- After starting the operating system, the GPU code is not unloaded. In fact, start.elf is not just firmware for the GPU, It is a proprietary operating system called VideoCore OS (VCOS). When the normal OS (Linux) requires an element not directly accessible to it, Linux communicates with VCOS using the mailbox messaging system

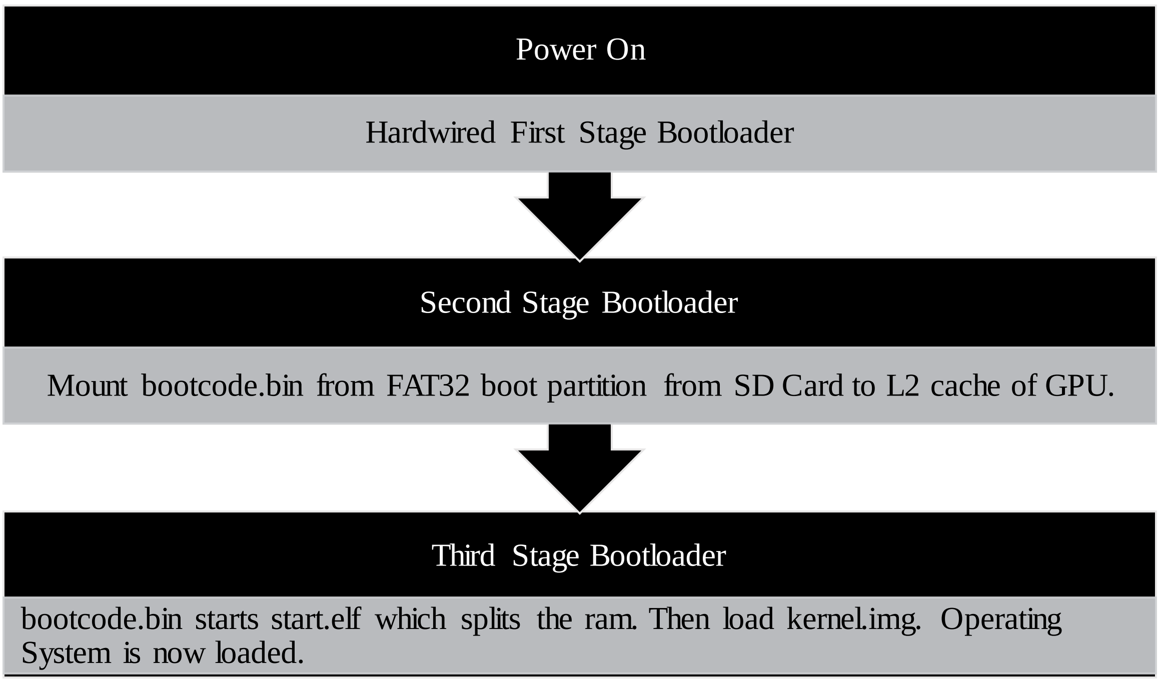

Figure 10: Boot process of Raspberry Pi

5.3 The NOOBS installer

The Raspberry Pi package only comes with the main board and nothing else. It does not come shipped with an operating system. Operating systems are loaded on a SD card from a computer and then the SD card is inserted in the Pi which becomes the primary boot device.

Installing operating system can be easy for some enthusiasts, but for some beginners working with image files of operating systems can be difficult. So the Raspberry Pi foundation made a software called NOOBS – New Out Of Box Software which eases the process of installing an operating system on the Pi.

The NOOBS installer can be downloaded from the official website. A user only needs to connect a SD card with the computer and just run the setup file to install NOOBS on the SD card. Next, insert the card on the Raspberry Pi. On booting the first time, the NOOBS interface is loaded and the user can select from a list of operating systems to install. It is much convenient to install the operating system this way. Also once the operating system is installed on the card with the NOOBS installer, every time the Pi boots, a recovery mode provided by the NOOBS can be accessed by holding the shift key during boot. It also allows editing of the config.txt file for the operating system

5.4 Raspberry Pi compatible operating systems

Distribution	Type	Memory footprint	Packages
Arch Linux ARM	Linux		8,700
BerryTerminal	Linux		
Bodhi Linux	Raspbian		35,000+ ARMHF
Debian ARM	Linux		20,000+
Fedora Remix	Linux		16,464?
Gentoo Linux	Linux	~23 MiB	
IPFire	Linux	~20 MiB	144
I2PBerry	Linux		20,000+
Meego MER + XBMC	Linux (embedded)	~34 MiB + XBMC	~320 (core)
Moebius	Raspbian	~20 MiB	(core) + Raspbian Repositories
nOS	Linux	~90 MiB	35,000+
openSUSE	Linux 3.11	28 MiB (inc. X11)	6300
OpenWRT	Linux	3,3MiB	3358
PiBang Linux	Linux_3.6.11 & SystemD		
PwnPi	Linux		20,000+
QtonPi	Linux		
VPNbian	Linux	~40 MiB w/o desktop	35,000+
Raspbian	Linux	~30 MiB w/o desktop	35,000+
OpenELEC	Linux 3.10.16 (embedded)	95 MiB (incl. XBMC)	~140 (+ 7 via xbmc)
XBian	Raspbian		35,000+

raspbmc	Raspbian		20,000+
RISC OS	RISC OS		
Aros hosted on Raspbian Limited Demo	Mixed Debian6 and Aros	<~50 MiB	
Plan9	Plan 9		
SlaXBMCRPi	Linux 3.10.36+		476 (+ Official SlackwareARM 14.1 Packages)
PiMAME	Linux		
PiBox	Linux/Buildroot		
pipaOS	Raspbian	~32 MiB	37.500
Raspberry WebKiosk	Raspbian		
Volumio	Raspbian		
Nard SDK	Embedded Linux	~40 MB	

Table 2: List of supported Operating Systems

Chapter 6

Applications

6.1 Applications

The major aim behind the Raspberry Pi was to educate people, especially children and teenagers, towards programming and basic hardware interfacing. The open body structure of the Raspberry Pi makes it a machine on which one can learn computer concepts.

Applications of the Raspberry Pi can be given as follows:

- Teaching programming concepts.
- Teaching hardware interfacing.
- Raspberry Pi being very cost effective can be deployed in large numbers in underdeveloped and developing countries like Africa, India, China, Brazil etc. to schools and colleges and to everyone who is interested in computers and electronics.
- It can be used in robotics for controlling motors, sensors, etc.
- It can be used as a downloading machine replacing desktop computers. It consumes very low power and also can be accessed remotely.
- It can be used as a media centre at home. Any television can be converted to a smart TV with internet capabilities with the Pi.
- It can be used for designing prototypes of DIY projects and certain embedded devices. It becomes very cheap option for testing and evaluation purpose.
- Can be used in creating and handling small servers.
- It can be used for making digital photo frames, tablets etc at home

6.2 Examples of projects using Raspberry Pi

1) Home Automation

   With a Raspberry Pi, switches, web server, enthusiasts have created home automation systems that can control fans and lights of a home from the Pi or even a smartphone.

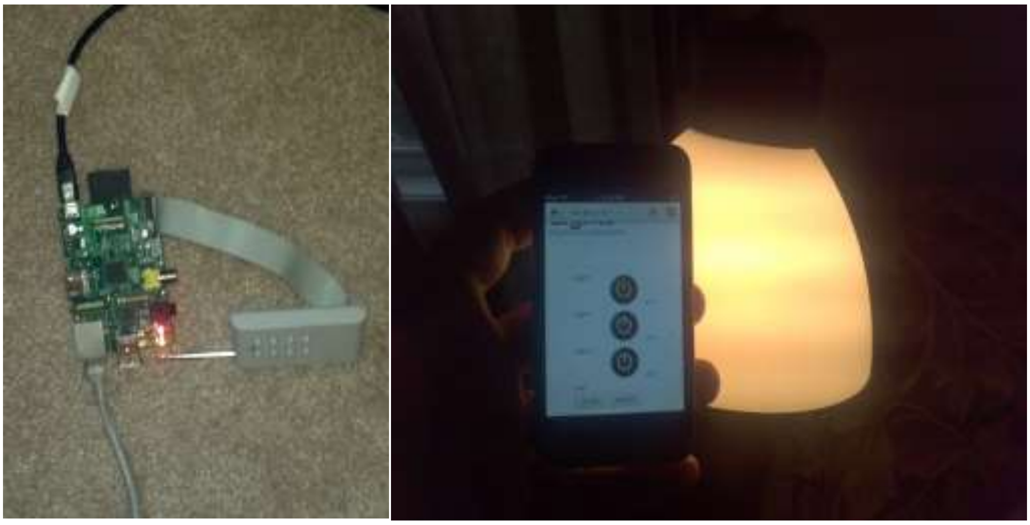

Figure 11: An example of a home automation system using RPi.

2) Arcade gaming machine

   Since the games of the 80’s and 90’s are forgotten today, people created retro style arcade gaming machines using simple switches, a screen and the RPi.

Figure 12: Arcade gaming machines using RPi

3) Robot

While robotics has a great fan base, creating a star wars robot from RPi deserves a mention here!

Figure 13: Raspberry Pi inside the robot

Figure 14: The robot with the Pi. The R2D2.

4) Landline

Why pay for your landline when you can use your internet and your old landline phone with the Pi!

Figure 15: Landline using RPi and Google Voice

5) Diwali lights / Decoration lights controller

The RPi can be used to make a remote control for all the decoration lights in the house!

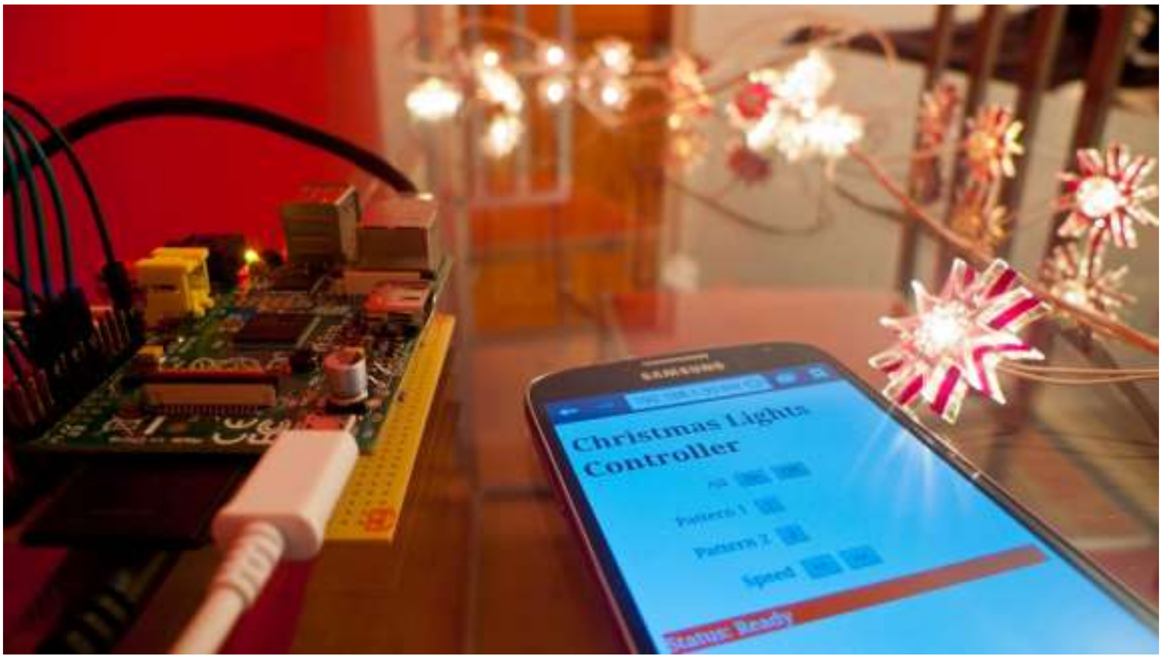

Figure 16: RPi based decoration lights controller

6.3 Comparison of Raspberry with the competitors

The chief competitors of the Raspberry Pi are the Arduino and the Beagleboard. Both are single board computers and have applications similar to the Raspberry Pi. A brief comparison of the three of them is shown below:

Name	Arduino Uno	Raspberry Pi	BeagleBone
Model Tested	R3	Model B	Rev A5
Price	$29.95	$35	$89
Size	2.95"x2.10"	3.37"x2.125"	3.4"x2.1"
Processor	ATMega 328	ARM11	ARM Cortex-A8
Clock Speed	16MHz	700MHz	700MHz
RAM	2KB	256MB	256MB
Flash	32KB	(SD Card)	4GB(microSD)
EEPROM	1KB		
Input Voltage	7-12v	5v	5v
Min Power	42mA (.3W)	700mA (3.5W)	170mA (.85W)
Digital GPIO	14	8	66
Analog Input	6 10-bit	N/A	7 12-bit
PWM	6		8
TWI/I2C	2	1	2
SPI	1	1	1
UART	1	1	5
Dev IDE	Arduino Tool	IDLE, Scratch, Squeak/Linux	Python, Scratch, Squeak, Cloud9/Linux
Ethernet	N/A	10/100	10/100
USB Master	N/A	2 USB 2.0	1 USB 2.0
Video Out	N/A	HDMI, Composite	N/A
Audio Output	N/A	HDMI, Analog	Analog

Figure 17: Comparison of RPi with chief competitors

6.4 Advantages and disadvantages

Advantages of the Raspberry Pi:

It is important for customers and business owners that want to get the Raspberry Pi to consider whether it fits with their business strategy and are willing to go through the process of putting it together and tailoring the product to their own needs. The benefits that this products offers beside the low price point are:

- This microcomputer is useful for small or home based businesses that run on a smaller budget than bigger companies for you are not required to purchase any special licenses from the Raspberry Pi Foundation to use their product or if you invent new technology that embeds the product. Small business owners can use it to automate any small task, such as using the Pi to run a website (as long as it does not have a lot of traffic), or use it as a small database and media server... pretty much anything that doesn't require the Windows operating system or other systems that does not support Linux and lots of traffic). Businesses can also save money on buying cooling systems that are required to cool servers.
- The product does not require the user to have extensive programming experience since it is aimed for the younger generation to learn about programming. Python, the programming language that the Pi uses, is less complex than other languages available. For example, it has better code readability and allows the user to type concepts using fewer amount of lines. Python also has an automatic memory management function.
- The product also gives you a lot of room to experiment and turn it into something else that is entirely different. The SD cards on the board can be easily switched, which allows you to change the functions of the device without spending a lot of time re-installing the software.
- The Raspberry Pi is perfect for adaptive technology: it is able to display images or play videos at 1080p high definition resolution to building systems such as digital jukeboxes or prototyping embedded systems. This product makes it possible to build complex and effective products at a cheaper price.
- The product is energy efficient and provides a greener ethical alternative to small businesses. This small credit card sized product makes it easy to recycle and does

not release as much carbon dioxide emissions into the environment, unlike big servers that require lots of energy and extensive cooling systems.

Disadvantages:

- It does not replace your computer, since the Ethernet is only a 10/100 and the processor is not as fast, it is time consuming to download and install software and is unable to do any complex multitasking.
- Not compatible with other operating systems such as Windows (There are currently 1.3 billion Windows users around the world.)
- To use the Raspberry Pi,it will take more than just 35 dollars to get it to do what you need through buying extra accessories such as the SD card, USB power supply, keyboard..etc and if you take into account the acquisition cost of the product. This is only fit for those who want a gadget that they can tailor to their own needs and tastes, not for those who just wants to get a job done fast. Business owners need to consider if the extra hassle is worth it.
- This product will not be useful for bigger businesses that already have big servers, which would already do everything that the Raspberry Pi does, so it would not be worth it to take the time to get someone to put it together.

Conclusion

Raspberry Pi is an innovative product. The sheer number of users and fan base support the fact that the device can see a great future ahead. The device can surely help anyone who really wants to lean electronics and computers.

Increasing the processing power can surely help the product in the future. Also supplying a case and a proper instruction manual will improve the product. Also currently Windows operating systems are not compatible because of the ARM processor. If the processor is improved or any workaround is found to run Windows directly on the Raspberry Pi, then it can be a great step for the Pi.

The Raspberry Pi is an amazing piece of hardware because of the combination of the features of a traditional computer and an embedded device. Supporting computer operating systems like Linux and providing simple input/output lines i.e. the GPIO makes it perfect for controlling almost anything. Programming the GPIO is much easy and intuitive then an traditional FPGA or microprocessor.

Finally it can be said that Raspberry Pi can be effectively used if its processing power is kept in mind. It can work as a personal computer but cannot replace it.

References

[1] History of embedded devices, http://www.discussionsworld.com/forum_posts.asp?TID=42603

[2] About Raspberry Pi, http://www.raspberrypi.org/about/

[3] Early prototype of Raspberry Pi, http://www.raspberrypi.org/raspberry-pi-2006-edition/

[4] Raspberry Pi Hardware information, http://elinux.org/RPi_Hardware

[5] How SoC works, http://www.androidauthority.com/how-it-works-systems-on-a-chip-soc-93587/

[6] Boot process, http://thekandyancode.wordpress.com/2013/09/21/how-the-raspberry-pi-boots-up/

[7] NOOBS, http://www.raspberrypi.org/introducing-noobs/

[8] List of available OS, http://elinux.org/RPi_Distributions

[9] Home automation project, http://www.instructables.com/id/Raspberry-Pi-GPIO-home-automation

[10] Arcade gaming machine project, http://www.cnet.com/how-to/25-fun-things-to-do-with-a-raspberry-pi/

[11] Robot project, http://www.wired.com/2013/01/raspberry-pi-r2d2/

[12] Landline telephone project, https://www.twilio.com/blog/2013/03/build-a-twilio-hard-phone-with-sip-from-twilio-raspberry-pi-asterisk-freepbx-and-the-obihai-obi100.html

[13] Decoration lights project, http://www.element14.com/community/community/raspberry-pi/raspberrypi_projects/blog/2013/12/12/raspberry-pi-enabled-christmas-lights

[14] Advantages-Disadvantages, https://sites.google.com/site/mis237groupprojectraspberrypi/home/what-is-raspberry-pi/pros-and-cons-of-the-raspberry-pi