Raspberry Pi : The Master Key

Vinay Vaishnav[1], Amit Tiwari[2], Musharraf Shaikh[3], S.Raghuvenkatesh[4],Prof. Ashwini Jayawant[5]
Department of Electronics & Telecommunication,
KCCEMSR, Thane, India
[1]jiggy1106@ gmail.com, [2]tamit484@ gmail.com,[3]memush007in@ gmail .com,[4]raghu.15angels@ gmail.com,
[5]ashwinisj@yahoo.co.in

***Abstract*— The Raspberry Pi developed in this paper is providing a unique idea of making wireless internet radio alongwith interfacing electronic device like stepper motor to it .The Raspberry Pi is said the master key because it uses a simple and portable hardware which can perform all computer operations and simultaneously interface electronic devices.**

Keywords—Internet radio, Master key, stepper motor.

I. Introduction

The Raspberry Pi is a credit-card-sized single-board computer developed in the UK by the Raspberry Pi Foundation with the intention of promoting the teaching of basic computer science in schools and colleges. It's tempting to think of the Raspberry Pi as a microcontroller development board like Arduino, or as a laptop replacement. In fact it is more like the exposed innards of a mobile device, with lots of maker-friendly headers for the various ports and functions. Our project is a flexible low-cost computer. It provides an ideal environment for experimenting with programming and electronics. The computer has a credit-card sized footprint and consumes at most a few watts of power.

IJERT

The Raspberry Pi is a computer that plugs into TV and a keyboard. It's a capable little PC which can be used for many of the things that your desktop PC does, like spreadsheets, word-processing and games. It also plays high-definition video. A Raspberry -Pi leads, a power supply or SD cards that are not included but can be purchased later. One can buy preloaded SD cards too. The Raspberry Pi measures 85.60mm x 53.98mm x 17mm, with a little overlap for the SD card and connectors which project over the edges. It weighs around 45g. Overall real world performance is something like a 300MHz Pentium 2. Raspberry-Pi cannot boot without an SD card.

There are a number of operating systems running, ported or in the process of being ported to Raspberry-Pi. Like, AROS, Android 4.0,Arch Linux ARM, Debian Squeeze, Firefox OS etc.The Raspberry Pi uses Linux kernel-based operating systems. Raspbian, a Debian-based free operating system optimized for the Raspberry Pi hardware, is the current recommended system.

II. Features

This project describes how to create an Internet Radio using the Raspberry PI educational computer. It contains instructions for building the radio using either the HDD44780 LCD directly wired to the
Raspberry PI GPIO pins or using an Adafruit RGB-backlit LCD plate for Raspberry PI.

The features of the radio are:
1. Raspberry PI running standard Music Player Daemon (MPD)
2. Three different LCDs supported
3. 2 x 16 character LCD with HD44780 controller
4. 4 x 20 character LCD with HD44780 controller
5. Adafruit LCD plate with 5 push buttons (I2C interface)
6. Clock display or IP address display (for web interface)
7. Five push button operation (Menu, Volume Up, Down, Channel Up, Down)
8. Artist and track scrolling search function
9. Plays music from a USB stick or from a Network drive (NAS)
10. Menu option to display a single RSS news feed
11. Web interface using snoopy and others
12. Plays Radio streams or MP3 and WMA tracks
13. Controlled by an Object Orientated Python application
14. Support for European character sets (Limited by LCD capabilities)

III. Block Diagram

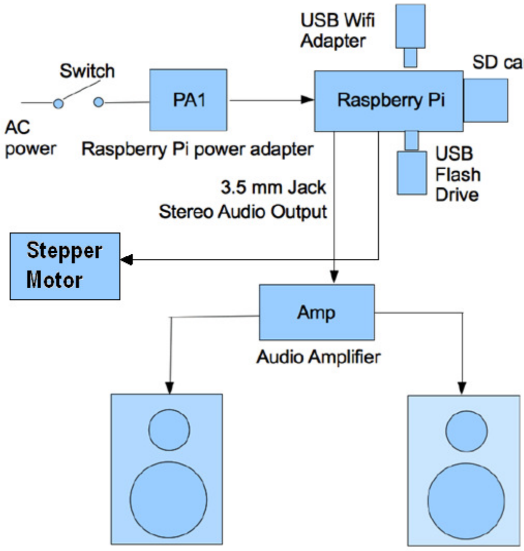

Fig.1 System block diagram used in our design

IV. Hardware

The principle hardware required to build the radio consists of the following components:

1. A Raspberry PI computer
2. An HD44780 LCD display or an Adafruit RGB-backlit LCD plate for the Raspberry PI
3. LCD and switches interface board

(a) RASPBERRY PI

The Raspberry Pi is a credit-card-sized single-board computer developed in the United Kingdom by the Raspberry Pi Foundation with the intention of promoting the teaching of basic computer science in schools.

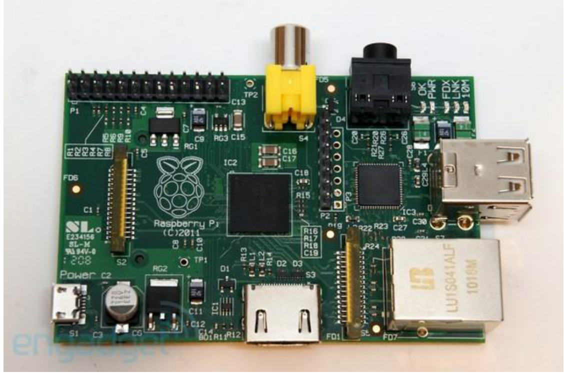

Fig.2 Raspberry Pi hardware

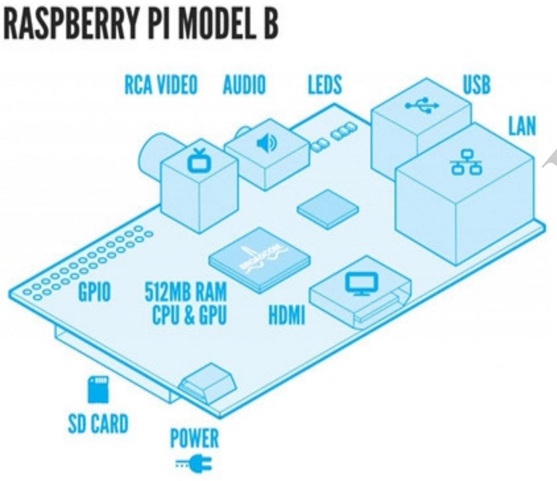

Fig.3 Raspberry Pi Model-B

The Raspberry Pi board contains a processor and graphics chip, program memory (RAM) and various interfaces and connectors for external devices. Some of these devices are essential, others are optional. RPi operates in the same way as a standard PC, requiring a keyboard for commandentry, a display unit and a power supply. It also requires 'mass-storage', but a hard disk drive of the type found in a typical PC is not really in keeping with the miniature size of RPi. Instead we will use an SD Flash memory card normally used in digital cameras, configured in such a way to 'look like' a hard drive to RPi's processor. RPi will 'boot' (load the Operating System into RAM) from this card in the same way as a PC 'boots up' into Windows from its hard disk.

The following are essential to get started:

1. SD card containing Linux Operating system
2. USB keyboard
3. TV or monitor (with HDMI, DVI, Composite or SCART input)
4. Power supply
5. Video cable to suit the TV or monitor used

(b) THE HD44780 LCD DISPLAY

The HDD44780 LCD interface is an industry standard interface for a variety of LCD displays.These can come in various sizes but the two lines by 16 character display is the most popular.

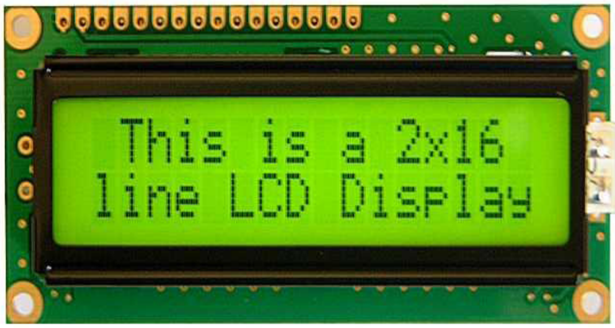

Fig.4 The HD44780 LCD display used as indicator/monitor

IJERT

(c) SD CARD

As the RPi has no internal mass storage or built-in operating system it requires an SD cardpreloaded with a version of the Linux Operating System.

Fig.5 SD card used for raspberry pi

SD card setup Steps:

1. Formatting the SD card via the mkcard.txt script
2. Formatting the SD card via fdisk "Expert mode"
3. Setting up the boot partition
4. Writing the image to the SD card and booting GNU/Linux
5. SD Card Cloning/Backup
6. Reading an image from the SD Card
7. Copying an image to the SD Card
8. Software Development/Proving

(d) POWER SUPPLY

The unit is powered via the micro USB connector (only the power pins are connected, so it will not transfer data over this connection).

(e) STEPPER MOTOR

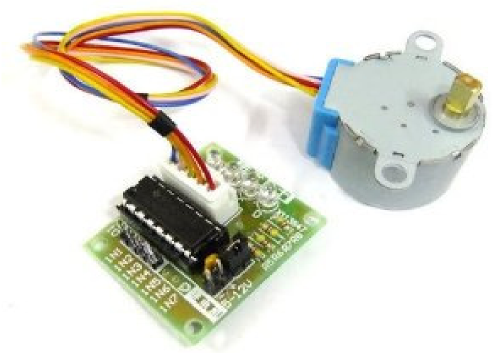

Fig.6 Stepper motor used for interfacing

The motor connects to the controller board with a pre-supplied connector.The controller board has 4+2 pins that need to be connected to the Pi header (P1).
1.5V (P1-02)
2.GND (P1-06)
3.Inp1 (P1-18)
4.Inp2 (P1-22)
5.Inp3 (P1-24)
6.Inp4 (P1-26)

(f) KEYBOARD AND MOUSE
Most standard USB keyboards and mice will work with the RPi. Wireless keyboard/mice should also function.

IJERT

(g) CABLES
You will need one or more cables to connect up your RPi system.
1. Video cable alternatives:
a) HDMI-A cable
b) HDMI-A cable + DVI adapter
c) Composite video cable
d) Composite video cable + SCART adaptor
2.Audio cable (not needed if you use the HDMI video connection to a TV)
3.Ethernet/LAN cable.

V. ADDITIONAL PERIPHERALS

You may decide you want to use various other devices with your RPi, such as Flash Drives/Portable Hard Drives, Speakers etc.

(a) INTERNET CONNECTIVITY
This may be via an Ethernet/LAN cable (standard RJ45 connector) or a USB WiFi adaptor. The RPi Model B Ethernet port is auto-sensing which means that it may be connected to a router or directly to another computer (without the need for a crossover cable).

(b) USB HUB
In order to connect additional devices to the RPi, you may want to obtain a USB hub, which will allow multiple devices to be used. It is recommended that a powered hub is used - this will provide any additional power to the devices without affecting the RPi itself.

(c) CASE
Since the RPi is supplied without a case, it will be important to ensure that you do not use it in places where it will come into contact with conductive metal or liquids, unless suitably protected.

(d) EXPANSION AND LOW-LEVEL PERIPHERALS
If you plan on making use of the low-level interfaces available on the RPi, then ensure you have a suitable plug for the GPIO header pins. Also if you have a particular low-level project in mind, then ensure you design-in suitable protection circuits to keep your RPi safe.

VI. GENERAL PURPOSE INPUT OUTPUT PINS (GPIO)

The figure shows the available pins for external access to the Raspberry Pi. Grayed out pins N/C are not for external connections and are used internally. GPIO pins 17, 18, 21, 22, 23, 24, 25 are available for external I/O connections. The pins are numbered from 1 to 26, left to right and top to bottom. The 5V pin is marked in red as a caution since the Raspberry Pi operates at the 3.3V level only. Incoming 5V signals must be voltage divided to yield only up to 3.3V before entering the Raspberry Pi. The unit is powered via the micro USB connector (only the power pins are connected, so it will not transfer data over this connection). A standard modern phone charger

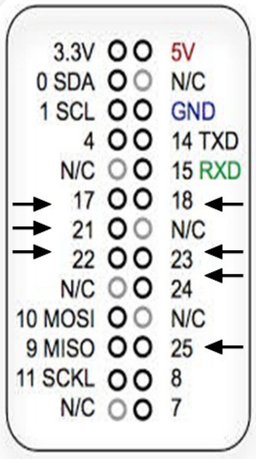

Fig.7 GPIO pins (arrowed) used in our design

VII. SPECIAL FUNCTION PINS

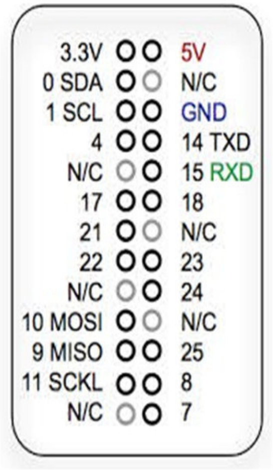

Fig.8 Special function pins

(a) LEFT COLUMN OF PINS:

0	SDA Serial Data Line 0
1	SCL Serial Clock Line 1
4	General Purpose Clock
10	MOSI Master Output Slave Input
9	MISO Master Input Slave Output
11	SCKL Serial Clock

(b) RIGHT COLUMN OF PINS:

14	TXD UART Transmit (TXD)
15	RXD UART Receive (RXD)
8	Chip Select 0 (1st Slave)
7	Chip Select 1 (2nd Slave)

VII. SOFTWARE

(a) LINUX

We choose Linux and not windows or other operating system because its open source and we have its special version dedicated for Raspberry Pi called as Debian PI.Linux understanding is very important as entire configuration of Raspberry Pi will be done on Linux console (command prompt).

(b) PYTHON

Python is high level programming (Scripting) language specially designed for Linux platform. It’s needed for developing any kind of applications on raspberry pi.

(c) MUSIC PLAYER DAEMON

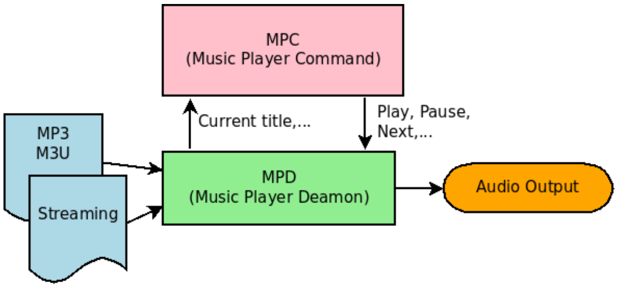

Fig.9 Music player daemon design

Music Player Daemon (MPD) is a free and open source music player server. It plays audio files, organizes playlists and maintains a music database.In order to interact with it, a separate client is needed.

(d) DESIGN

MPD simply runs in the background playing music from its playlist. Client programs communicate with MPD to manipulate playback, the playlist, and the database. It is not a full-featured music player program such as Amarok, but its clients can serve such role. MPD uses a flat file database to maintain the basic music file information when it is not running. Once the daemon is started, the database is kept completely in-memory and no hard disk access is necessary to look up or search for local audio files.Usually, music files must be below the music root directory and are only added to the database when the update command is sent to the server. Playback of arbitrary files is only allowed for local clients which are connected to the server via Unix Domain Sockets. MPD does not provide a built-in tag editor; this functionality is handled by clients or external programs, though 3rd party patches do exist to add this functionality to the server. The client–server model provides several advantages over all-inclusive music players. Clients may communicate with the server remotely over an intranet or over the Internet. The server can be aheadless computer located anywhere on the network. Music playback can continue seamlessly when not using X or restarting X. Different clients can be used for different purposes – a lightweight client left open all the time for controlling playback with a more fully featured client used for intensive database searches. Several clients can use the same database, even running simultaneously – side-by-side, remotely or under different user accounts.

IJERT

(e) FEATURES

1.Plays Ogg Vorbis, FLAC, WavPack, MP2, MP3, MP4/AAC, MOD, Musepack, wave files and any other files supported by FFmpeg.

2.Remotely control MPD over a network (IPv4 and IPv6 supported).

3.Play FLAC, OggFLAC, MP3 and Ogg Vorbis HTTP streams.

3.Reads and caches metadata information (ID3: ID3v1 and ID3v2), Vorbis Comments, and MP4 Metadata.

4.Metadata information can be searched.

5.Buffer support for playback (prevents skipping due to high load or network latency).

6.Gapless playback.

7.Crossfading support.

8.Seeking support.

9.Save, load, and manage playlists (in M3U format).

10.Native Zeroconf support.

11.libsamplerate and native sample rate conversion

12.Support for ALSA, PulseAudio, OSS, MVP, JACK, Windows, and Mac OS X.

13.Can be used as a source for an Icecast stream, in Ogg Vorbis and MP3. Other formats can be converted to Ogg/MP3 on the fly before output to the stream server.

14.Built-in HTTP streaming server, capable of producing Ogg Vorbis and MP3 streams of a chosen quality on-the-fly.

15.Independent of a GUI. Music will continue playing whether a front-end is open or not,and will continue playing even if the X server is killed.

(f) CLIENTS

MPD has a variety of front-ends which communicate with the server using a custom protocol over a TCP connection. Clients usually implement different types of interfaces.

1.mpc (part of the MPD project) is a simple command line interface to most of the server's functions

2.ncmpc is a more fully featured ncurses client similar in concept to MOC

3.ncmpcpp is another ncurses client that clones ncmpc's functionality, but includes new features, such as a tag editor.

4.Sonata uses GTK+ to provide a GUI for playing files and managing playlists.

5.Intelligent Music Player Client (IMPC) uses GTK+ 3.0 and supports loading related information (articles, covers, images) with actually playing song and it learns to classify related content.

6.Cantata uses Qt to manage the library or playlists, with extras like tags editor, lyrics, cover fetching and support for mass storage devices.

7.Qmobilempd provides a Qt4 client especially for mobile devices like Symbian.

8.Qmpdplasmoid provides a client directly embedded into the KDE desktop.

9.Ario uses tabs in its interface.

10.Gimmix provides a simple interface with a very small memory footprint.

11.mpdlirc interfaces MPD via an infrared remote control For a more complete list, see the Clients MPD Wiki.

Simple clients can script the mpc program to issue commands to the server. Some clients run within a web browser and can be located on the same computer as the server, requiring only a browser be installed on the client machine.There is a client implemented as a Firefox add-on, one as a Xfce panel plugin and one as a Wii application. and one as a Windows Store application for Windows 8/Windows RT.

(g) LIBRARIES

Libraries exist to interface with MPD from many programming languages, including C, Python, Ruby, Perl and Lua.

VIII. CONCLUSION

In this paper, a novel method of wireless internet radio and simultaneously using a stepper motor for any applications, using a portable mini computer system known as raspberry pi will be desingned and linux operating system is used and python programming language is used. This paper mainly describes the design and simulation of Internet Radio alongwith Stepper Motor which is based on Raspberry Pi, Linux OS, Python Programming and GPIO pins.

REFERENCE

1] Official site: www.raspberrypi.org

2] “FAQs”. Raspberry Pi. Retrieved 3 November 2011.

3] "BCM2835 Media Processor; Broadcom". Broadcom.com. 1 September 2011. Retrieved 6 May 2012.

4] Introducing turbo mode: upto 50% more performance for free. Raspberrypi.org. Retrieved 20 September 2012.

5] At first, for a short time, revision 2 boards were made that had 256 MB of RAM

6] Raspberry Pi: Cheat Sheet. Silicon.com. Retrieved 6 May 2012.

7] Price, Peter (3 June 2011). “Can a £15 computer solve the programming gap?” BBC Click. Retrieved 2 July 2011.

8] Bush, Steve (25 May 2011). “Dongle computer lets kids discover programming on a TV". Electronics Weekly. Retrieved 11 July 2011

IJERT