

Николай Хлюпин (RA4NAL)
г. Киров

Если у кого-то, после появления Raspberry Pi 2 и 3, первая модель не находит применения, то описанный ниже приемник – неплохой вариант ее использования.

Интернет радиоприемник на Raspberry Pi

Часть 1. Hardware

Проживая в крупном городе и имея даже простенький радиоприемник, можно слушать несколько десятков радиостанций в УКВ диапазоне с весьма неплохим качеством. Казалось бы, этого вполне достаточно...

Однако у эфирных радиостанций есть несколько недостатков. Во-первых, это неуверенный прием сигнала в некоторых районах; во-вторых, при кажущемся разнообразии найти радиостанцию, которая в полной мере соответствовала бы музыкальным вкусам, оказывается не всегда просто. Кроме того – назойливая реклама и довольно ограниченный репертуар большинства станций.

Неудивительно, что сейчас многие пользователи предпочитают слушать радиостанции, вещающие во Всемирной сети. В настоящее время количество интернет-радиостанций уже превысило 10 тысяч. Через Интернет можно слушать программы станций, расположенных практически в любой стране мира и вещающих на самых разных языках. Но включать компьютер для того, чтобы слушать радио, мягко говоря, неудобно. Как-то привычнее и удобнее – покрутил ручку – настроился на станцию...

Итак, буду делать интернет-радиоприемник. Такая мысль пришла мне в голову, когда я решал, для какого-бы проекта использовать микрокомпьютер Raspberry Pi model B, который остался не у дел после приобретения гораздо более мощного Raspberry Pi 2.

Существует много проектов интернет радио на Raspberry Pi. Лучшее, что мне удалось найти – это Pi Radio [1]. Автор – **Bob Rathbone** – работает над этим проектом на протяжении более 2 лет. Особо следует отметить открытость и доступность всей информации, а также очень подробное описание аппаратной части и методики настройки программного обеспечения.

Именно эту разработку я и решил повторить. Разумеется, я не просто слепо скопировал конструкцию, а подошел к процессу творчески, добавил кое-что свое.

Итак, что же собой представляет Pi Radio. Это интернет-приемник, который позволяет слушать on-line радиостанции, кроме того, он может проигрывать музыкальные файлы с флэшки или сетевого диска в домашней сети. В программу заложено много всевозможных сервисных функций – таймер, будильник, RSS новости, подкасты...

Подключение к сети Интернет может быть как проводным, так и по WiFi. Основа приемника – популярный микрокомпьютер Raspberry Pi, причем можно использовать любую модель: B, B+, 2 или Zero.

Bob Rathbone разработал несколько вариантов схемы и программного обеспечения. Управление возможно как с помощью энкодера, так и кнопок, отображение информации на 2-х или 4-х строчном символьном LCD дисплее. Управлять приемником можно с пульта ДУ.

Дисплей можно подключить как непосредственно к портам ввода-вывода Raspberry Pi, так и с использованием платы на основе PCF8574 по шине I2C. Автор предусмотрел возможность использования еще двух типов дисплеев – AdaFruit LCD с интерфейсом I2C и PiFace CAD с интерфейсом SPI.

Я повторил вариант с непосредственным подключением двухстрочного дисплея к портам ввода-вывода и управлением энкодерами. Результаты меня вполне удовлетворили, поэтому, когда возникла необходимость еще в одном приемнике, я изготовил второй вариант – все то же самое, но управление кнопками. В дальнейшем я буду описывать именно эти две модификации.

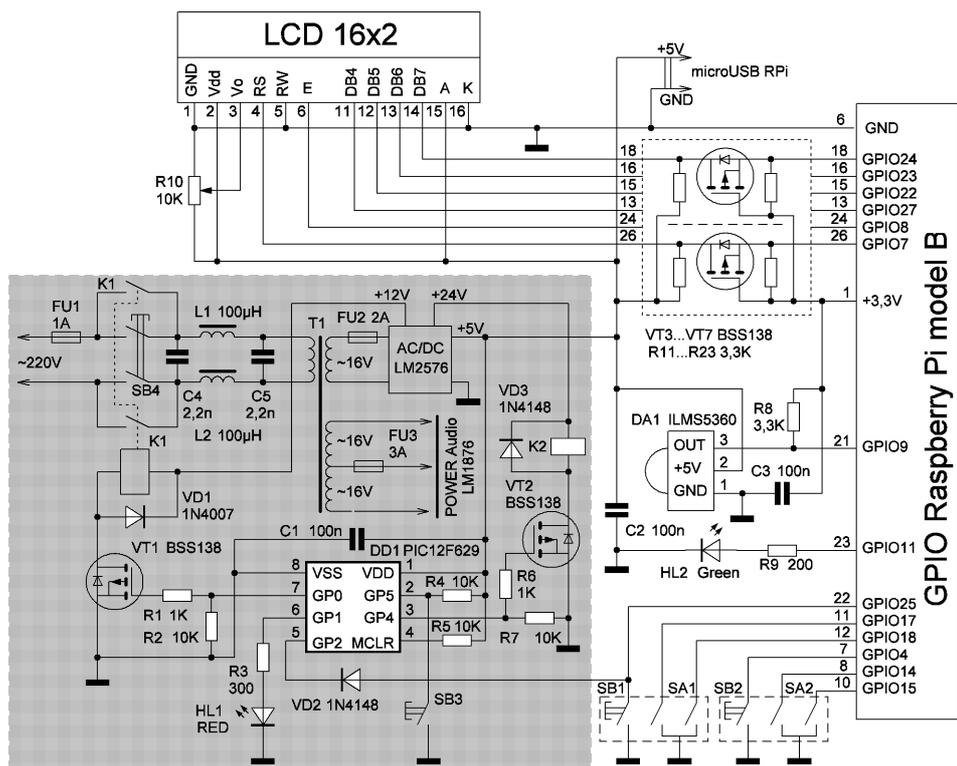
Замечу, что сначала я попытался подключить дисплей по шине I2C с использованием платы на PCF8574. Потратил массу времени на наладку, но, в конце концов, пришлось отказаться от этого варианта. Программное обеспечение довольно сложное и объемное, очевидно, у автора просто нет сил и возможности провести тщательное тестирование всех вариантов своего радио.

Программа постоянно зависала, особенно при отправке команды с пульта ДУ. В чем тут дело, разобраться сложно. А вот варианты с непосредственным подключением дисплея к Raspberry Pi работают устойчиво. Поэтому, я не рекомендую использовать шину I2C или SPI в этом приемнике.

На мой взгляд, автор уделил слишком много внимания всевозможным вспомогательным функциям в ущерб основным. Ведь возможности LCD индикатора очень ограничены и читать на нем, например, новости, не очень комфортно. Использовать приемник в качестве будильника весьма рискованно, если по какой-то причине не будет интернета – не будет и радио. Да и электричество могут ночью отключить...

Однако, в общем и целом все получилось, приемником можно пользоваться.

По сути, радиоприемник представляет собой плату Raspberry Pi, к портам ввода-вывода которой подключен дисплей и 2 энкодера. С помощью одного из них осуществляется выбор станции, с помощью второго регулируется громкость. Дисплей использован



SB1 - Menu; SA1 - Tune; SB2 - Mute; SA2 - Volume; SB3 - Power OFF; SB4 - Power ON

Рис. 1. Схема приемника. Вариант 1, с управлением энкодерами

самый распространенный, на основе контроллера HD44780 на 2 строки по 16 символов. Вместо энкодера можно использовать для навигации 5 кнопок.

Для любителей качественного звука предусмотрена возможность использования вместо AUDIO выхода Raspberry Pi внешней USB звуковой карты.

Разумеется, для приемника нужен блок питания. Тут много вариантов выбора – подойдет любой стабилизированный блок питания на 5 В с током не менее 1,5 А. Имейте в виду, что зарядник от телефона не подойдет, нужен именно стабилизированный блок питания, напряжение под нагрузкой не должно падать ниже 4,8 В, а без нагрузки подниматься выше 5,2 В.

Ну и музыку надо как-то слушать, т.е. необходим стерео усилитель или активные колонки. Тут тоже масса вариантов на любой вкус (слух) и кошелек.

Я решил для этого проекта использовать по максимуму свой радиолюбительский “ящик с хламом”. В нем нашелся прекрасный сетевой трансформатор на нужный ток и напряжение, микросхема импульсного стабилизатора LM2576-ADJ и микросхема неплохого и довольно мощного стерео усилителя НЧ LM1876.

Автор предлагает включать и выключать приемник просто сетевым выключателем. Хотя это и не причинит вреда Raspberry Pi, но при неудачном стечении обстоятельств может повредить файловую систему на SD карточке. Поэтому я решил дополнить устройство схемой корректного закрытия системы и выключения питания. Тем более, что такой режим автор предусмотрел – нужно нажать и удерживать кнопку “Меню” в течение 3 сек, после чего подождать еще примерно

30 сек и выключить питание. Согласитесь – это весьма неудобно.

В результате получилась вот такая схема (рис. 1). Это мой первый вариант с управлением энкодерами. Серым фоном на рисунке выделены мои дополнения.

Управление настройкой и громкостью осуществляется с помощью энкодеров с кнопками SA1, SB1 и SA2, SB2 соответственно. Кнопка SB1 – вход в “Меню”, SB2 – режим “Mute”. Энкодеры должны выдавать последовательность сдвинутых на 90 градусов импульсов. Я использовал тип ES110501S-NA2 5 pin. Под такой маркировкой они значились в прайс-листе магазина.

Транзисторы VT3...VT7 и резисторы R11...R23 – это преобразователи логических уровней 3,3...5 В для индикатора. Эти элементы устанавли-

вать необязательно. Как показала практика, 5-и вольтовый индикатор нормально работает с логическими уровнями 3,3 В. Однако, если возникнут проблемы, преобразователь уровней можно смонтировать на небольшой плате непосредственно на индикаторе. Еще лучше использовать индикатор с питанием от 3,3 В, но такие индикаторы довольно дефицитны и стоят существенно дороже.

DA1 – приемник от ДУ для телевизоров, на выходе в нем стоит транзистор по схеме с общим эмиттером и нагрузочным резистором около 30 кОм в цепи коллектора, так что проблемы согласования уровней нет. HL2 – индикатор активности ДУ.

Вот, собственно, и весь приемник, а выделенная слева часть схемы – это цепи питания. Я использовал обычный трансформаторный блок питания. В выключенном состоянии все цепи обесточены, никаких дежурных режимов. При нажатии на кнопку SB4 на трансформатор подается питание. Обращаю внимание – кнопка без фиксации и она должна быть рассчитана на коммутацию 220 В. C4, C5, L1, L2 – сетевой фильтр помех, номиналы L1, L2 не критичны.

С выхода стабилизатора напряжение 5 В подается на разъем microUSB Raspberry Pi и на PIC контроллер DD1. На выводе 7 (GP0) контроллера устанавливается высокий логический уровень, транзистор VT1 открывается, срабатывает реле K1 и своими контактами блокирует кнопку включения питания.

На выводе 3 (GP4) контроллера устанавливается нулевой уровень, он используется для закорачивания входа УНЧ в процессе загрузки и закрытия системы.

Дело в том, что процесс активации USB звуковой карты сопровождается неприятными, довольно громкими звуками. Если предполагается использовать встроенный AUDIO выход Raspberry Pi, реле K2 можно не устанавливать, вывод GP4 контроллера в этом случае никуда не подключается.

После подачи питания светодиод HL1 мигает в течение примерно 50 сек, сигнализируя, что система загружается. По истечении этого времени приемник готов к работе, на выводе 3 (GP4) устанавливается высокий логический уровень, вход УНЧ разблокируется, HL1 светится постоянно. В качестве HL1 и HL2 я использовал один двухцветный светодиод с общим катодом.

Для того, чтобы выключить приемник, нужно кратковременно нажать кнопку SB3. При этом на выводе контроллера 5 (GP2) устанавливается низкий логический уровень, имитирующий нажатие кнопки SB1 "Меню". Через 3 сек начинается закрытие системы, вход УНЧ блокируется, HL1 опять мигает, сигнализируя о закрытии. Примерно через 40 сек система закрыта, на выводе 7 (GP0) устанавливается нулевой уровень, реле K1 отключается и система обесточивается.

Контроллер можно использовать типа PIC12F629 или PIC12F675, а с незначительной доработкой программы – любой PIC.

Напряжения на вторичных обмотках трансформатора несколько великоваты, но я исходил из наличия. Пришлось использовать импульсный стабилизатор напряжения на 5 В. Он собран на LM2576-ADJ, включенной по типовой схеме (рис. 2). Номинал резистора R1 рассчитывается по закону Ома исходя из номинального тока и напряжения срабатывания реле. L2, C3 – дополнительный фильтр импульсных помех. Номинал L1 выбирается исходя из входного напряжения и рабочего тока согласно рекомендациям, приведенным в datasheet на LM2576. Номинал L2 не критичен.

Единственное ноу-хау схемы – стабилизатор VD3 типа Д815Б с напряжением стабилизации 6,8 В. В работе стабилизатора он не участвует, установлен для защиты нагрузки от повышенного напряжения в случае неисправности стабилизатора. Согласно схеме – если вместо 5 В подать на схему 24 В, последствия будут весьма печальны.

Здесь нужно использовать старый стабилизатор в металлическом корпусе с гайкой, установленный на плате без радиатора.

Если по каким-либо причинам напряжение на выходе стабилизатора поднимется выше 7 В, через стабилизатор потечет большой ток, что приведет к перегоранию предохранителя в цепи обмотки трансформатора.

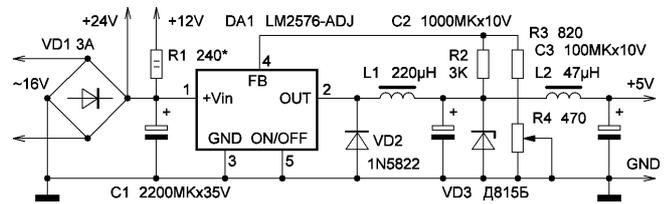


Рис. 2. Схема стабилизатора

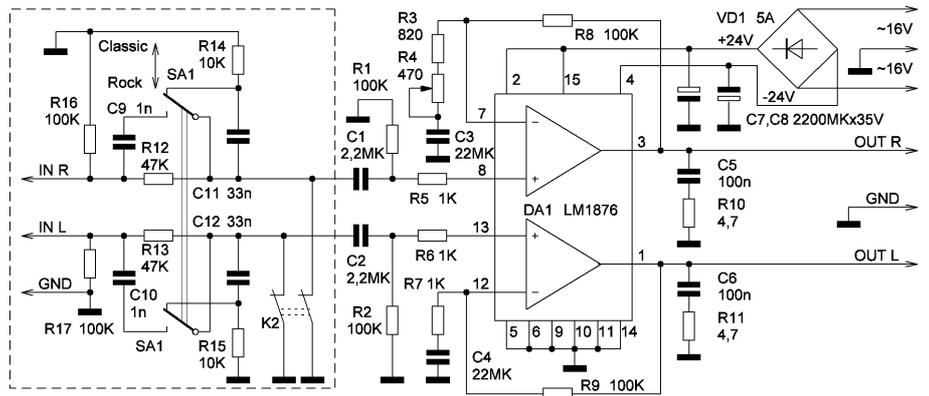
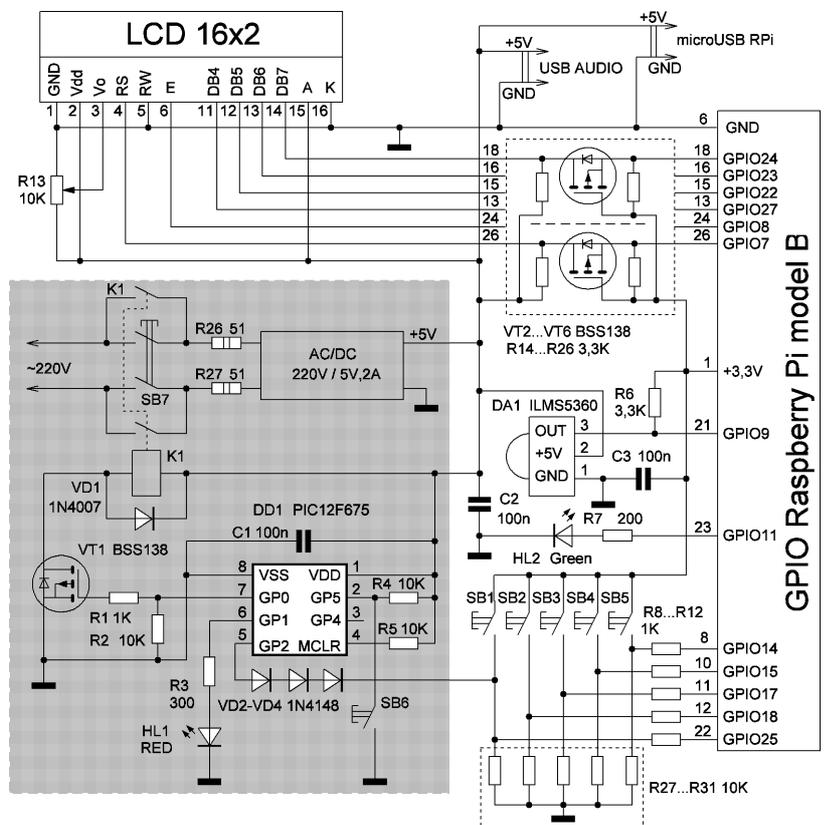


Рис. 3. Схема усилителя низкой частоты



SB1 - Menu; SB2 - Channel Down; SB3 - Channel Up; SB4 - Volume Up
SB5 - Volume Down; SB6 - Power OFF; SB7 - Power ON

Рис. 4. Схема приемника. Вариант 2, с управлением кнопками

Ну а если вместо предохранителя установлен гвоздь, стабилитрон перегреется, кристалл в нем расплавится и замкнет выход накоротко. Raspberry Pi будет спасена. Если использовать слаботочный стабилитрон в стеклянном корпусе, при перегрузке он просто взорвется и не защитит нагрузку.

Усилитель низкой частоты особенностей не имеет, он собран по типовой схеме на LM1876 (рис. 3). Элементы C9...C12 и R12...R17 – регулятор тембра на два фиксированных положения Classic и Rock. В первом случае частотная характеристика линейна, во втором – осуществляется подъем нижних и верхних частот. Если использовать тумблер SA1 на три положения, то в среднем положении, когда все контакты разомкнуты, будет подъем только нижних частот. Регулятор тембра смонтирован навесным монтажом на выводах SA1.

Нормально замкнутые контакты реле K2 закорачивают вход усилителя на время загрузки и закрытия системы. На схеме показано подключение реле с обмоткой на 24 В, но можно использовать и на 12 или 5 В, в зависимости от используемого блока питания. Как я уже писал, это реле нужно только для USB звуковой карты и качественной акустики.

С помощью подстроечного резистора R4 выравнивается усиление по каналам (стерео баланс). Выводить этот резистор для оперативной регулировки нет необходимости. Коэффициент усиления УНЧ определяется соотношением номиналов резисторов R8/(R3+R4) и R9/R7.

Усилитель питается от отдельной обмотки трансформатора. Если использовать общий источник питания для аналоговой и цифровой частей приемника, будет сложнее избавиться от помех, создаваемых Raspberry Pi – всевозможных “журчалок” и “скрипелок”. Аналоговая и цифровая “земля” должны соединяться только в одной точке – на входном разъеме УНЧ. Еще лучше использовать отдельный внешний усилитель НЧ или активные колонки.

Схема второго варианта приемника с управлением кнопками приведена на рис. 4.

Режим “Mute” включается при одновременном нажатии кнопок “Volume Up” и “Volume Down”. Резисторы R27...R31 нужны только для самых старых моделей Raspberry Pi model B rev.1. Для rev. 2, а также моделей B+, 2 и Zero их устанавливать не нужно.

Блок питания импульсный 5 В, 2 А. Резисторы R26, R27 ограничивают бросок тока при включении сети, а также выполняют роль предохранителей. Диоды VD2...VD4 служат для согласования уровней 5...3,3 В. Реле K1 с обмоткой на 5 В. Если не удастся приобрести такое реле с двумя парами контактов, можно включить параллельно два реле с одной парой в

каждом. Возможно, в этом случае потребуется использовать более мощный транзистор VT1. Вместо УНЧ я использовал активные колонки с питанием от 5 В (USB). Они подключаются к AUDIO выходу Raspberry Pi. Остальные элементы схемы аналогичны варианту с энкодерами.

Чертежи всех плат в Sprint Layout (рис. 5-10) приведены только для того, чтобы их можно было “взять за

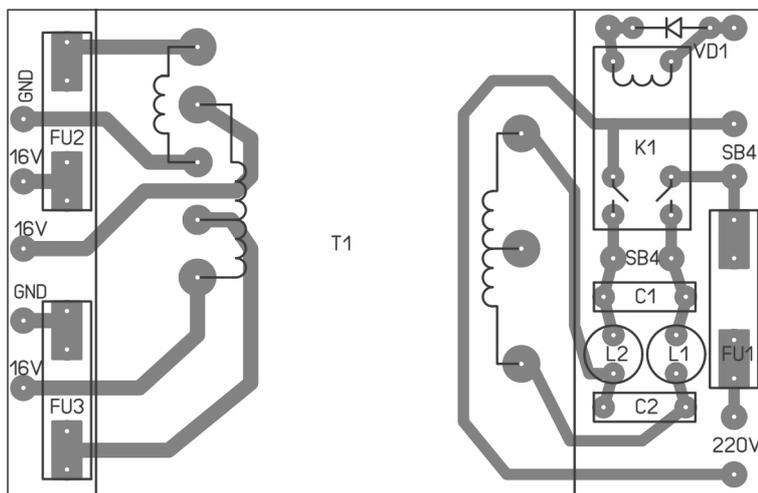


Рис. 5. Плата БП (100x65 мм)

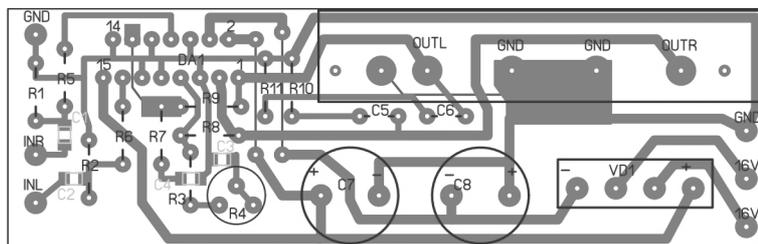


Рис. 6. Плата УНЧ (100x32 мм)

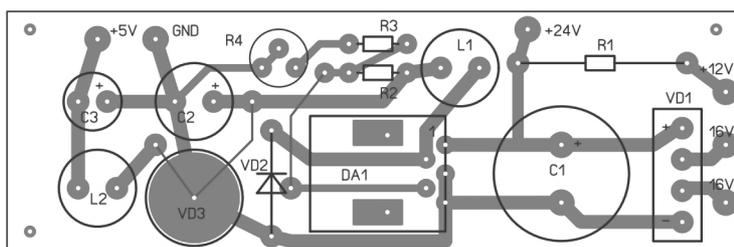


Рис. 7. Плата стабилизатора (97x32 мм)

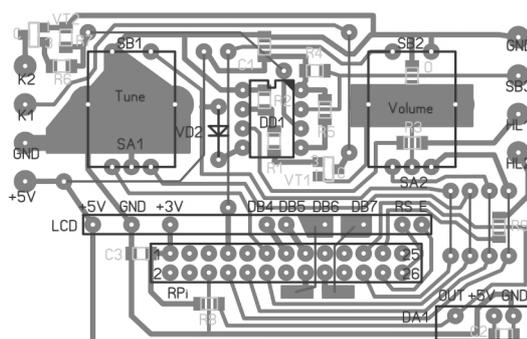


Рис. 8. Плата приемника. Вариант 1 (70x45 мм)

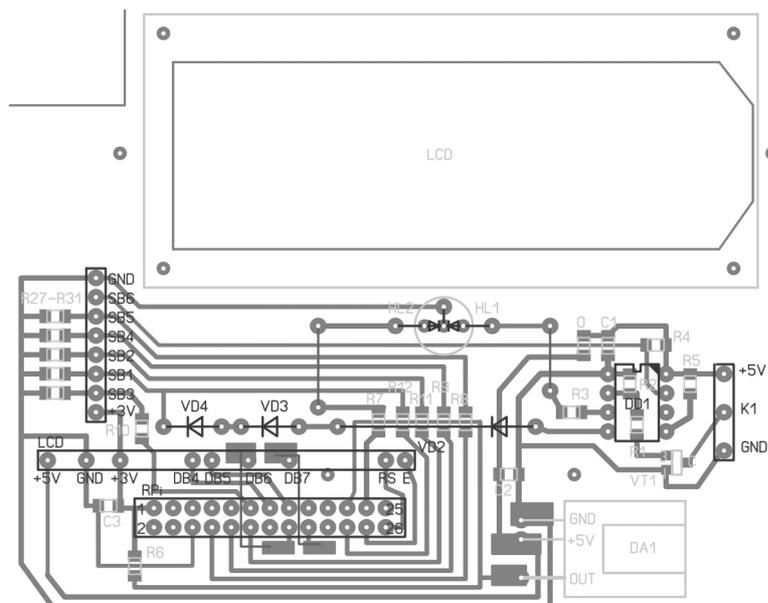


Рис. 9. Плата приемника. Вариант 2 (115x80 мм)

основу". Повторять 1:1 не рекомендую, размеры и конфигурация плат определяются, в первую очередь, корпусом.

Внешний вид устройства приведен на фото 1-2.

Прошивка PIC для двух вариантов приемника разная, для варианта с энкодерами это файл **picrdo_enc.hex**, для варианта с кнопками – **picrdo_but.hex**. Прошивки подойдут как для PIC12F629, так и для PIC12F675. В архиве с прошивками есть и файл исходного текста программы. При необходимости в нем можно легко изменить длительность задержки на включение и выключение приемника. Автор регулярно обновляет программу. С момента запуска приемника, а прошло около года, появились четыре обновленные версии.

Где приобрести комплектующие? Raspberry Pi пока еще дешевле заказать в Китае, на Aliexpress ее стоимость в зависимости от модели находится в пределах 30...40 USD (конец 2015 г.). Там же при необходимости можно подобрать блок питания и USB звуковую карту по вполне разумной цене.

Использовать USB звуковую карту имеет смысл только при достаточно хорошей акустике. Если ваши колонки стоят менее 50 USD, вполне можно обойтись встроенным в Raspberry Pi AUDIO выходом. Выбор колонок достаточно большой в любом магазине, торгующем электроникой. Разумеется, Hi End звук от интернет радио не ждите, но и "пищалки" размером с теннисный шарик – не лучший выбор.

Индикатор лучше приобретать в СНГ, если необходима поддержка русского языка. Предпочтение следует отдать моделям с темными символами на светлом фоне. Популярные индикаторы с белыми символами и синей подсветкой красиво смотрятся, но они очень "инерционные". Трудно будет следить за бегущей строкой.

Если ваш индикатор вместо русских букв показывает иероглифы, ничего страшного, его можно использовать в этом приемнике. Только русский текст будет отображаться

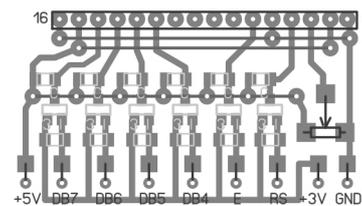


Рис. 10. Плата 3,3...5 В (47x26 мм)



Фото 1



Фото 2

латиницей. Энкодеры тоже лучше приобрести в розничном магазине, по крайней мере, будет известна марка, значит можно уточнить параметры по datasheet.

Ресурсы проекта – описание оригинальной версии, платы, прошивки, исходный текст программы (файл **piradio_1.zip**) вы можете загрузить с сайта нашего журнала: <http://www.radioliga.com> (раздел "Программы"); с сайта автора по адресам: <http://ra4nal.qrz.ru>, <http://ra4nal.lanstek.ru>

Ссылки

1. Raspberry PI Internet Radio - http://www.bobrathbone.com/raspberrypi_radio.htm



Продолжение в №10/2016

Интернет радиоприемник на Raspberry Pi

Николай Хлюпин
(RA4NAL)
г. Киров



Продолжение. Начало в №9/2016

Часть 2. Устанавливаем программное обеспечение

Займемся установкой программного обеспечения. Итак, первое, что нужно сделать, это зайти на страничку DOWNLOADS сайта сообщества raspbian.org – [2]. Здесь нужно скачать дистрибутив официальной операционной системы для Raspberry Pi – RASPBIAN WHEEZY. Я использовал версию 2015-05-05. Новую версию RASPBIAN JESSIE устанавливать не рекомендую, т.к. с ней есть проблемы при установке программного обеспечения PiRadio. Это проверено на Raspberry Pi model B.

Bob Rathbone – автор Pi Radio [1] – анонсировал выпуск 5-й версии своего радио для RASPBIAN JESSIE в начале 2016 г. Но на момент написания статьи существовала только версия 4.7, поэтому все дальнейшее описание относится именно к ней*.

Зайдите на сайт автора PiRadio и скачайте необходимое программное обеспечение – пакет **radiod_4.7_armhf.deb**.

Для того, чтобы записать систему на SD карту, потребуется программа Win32DiskImager [3]. Учтите, что в данном случае Raspberry Pi работает без дисплея и клавиатуры, настраивать все придется, используя дистанционный доступ. Пользователям Windows для этого потребуются две программы: PuTTY [4] и WinSCP [5].

Скачайте и установите также SD Card Formatter [6] и Bootice [7]. Для редактирования файлов потребуется текстовый редактор с поддержкой кодировки UTF-8, например, AkelPad [8]. Наконец, рекомендую установить CrystalDiskMark [9].

Все эти программы бесплатные, не имеют никаких ограничений и не требуют регистрации.

Для установки программного обеспечения нужна SD (или microSD для Raspberry Pi B+, 2 и Zero) карта. Минимально необходимый объем 4 ГБ, но чтобы было место для хранения музыкальных файлов, лучше использовать на 8 ГБ. Использовать карты большего объема особого

* На официальном сайте сообщества raspbian.org [2] в настоящее время нет ссылки на скачивание системы RASPBIAN WHEEZY. Можно скачать образ только новой RASPBIAN JESSIE. В основном все шаги по установке ПО для этой системы аналогичны описанным автором для RASPBIAN WHEEZY. Но есть некоторые отличия, и устанавливать следует последнюю авторскую версию ПО для PiRadio – пакет **radiod_5.4_armhf.deb**. В случае возникновения проблем – читайте авторское описание (на английском языке), или ставьте старую версию RASPBIAN WHEEZY, которую можно найти здесь: <http://ftp.jaist.ac.jp/pub/raspberrypi/raspbian/images/>. Выбрать следует raspbian-2015-05-07/ – это последняя версия WHEEZY.

На сайте автора PiRadio [1] тоже нельзя скачать версию ПО для WHEEZY – там сейчас нет ссылки на файл **radiod_4.7_armhf.deb**. Этот файл можно скачать с сайта автора или сайта журнала. Новая версия **radiod_5.4_armhf.deb** не подойдет для старой системы.

смысла нет – возможности навигации в файловой системе весьма ограничены, трудно будет найти нужный трек.

Карточку придется выбирать всего по двум параметрам – производитель и класс. Класс лучше брать по максимуму 10-й. Хорошо, если будет маркировка по стандарту UHS - U1.

Такая карта гарантированно подойдет для любого фотоаппарата или видеокамеры. К сожалению, ничего нельзя гарантировать в случае с Raspberry Pi. На вопрос “Почему?” вы услышите от опытных программистов туманные разъяснения о какой-то несовместимости. Попробуем разобраться, в чем тут дело. Я протестировал две карточки. Первая, SunDisk, приобретена в России. Вторая, с этикеткой Transcend, доставлена из Китая.

Установим карточку в кардридер и запустим на компьютере программу CrystalDiskMark. Выбираем в ней тестируемый диск – нашу карточку, для удобства можно даже включить русский язык. Все остальные настройки по умолчанию. Нажимаем кнопку “All” и ждем несколько минут результатов теста. На **рис. 1** – SunDisk, на **рис. 2** – Transcend (Китай) .



Рис. 1

Рис. 2

Первая и третья строчки – скорость чтения и записи больших блоков данных. Как видите, здесь все нормально, скорости даже несколько превосходят заявленный 10-й класс. А вот скорость записи небольших файлов объемом 4 KB по случайным адресам – вторая и четвертая строчки – гораздо ниже. В принципе, так и должно быть, но не в тысячи же раз, как для китайской карты! Сначала я даже подумал, что тут какая-то ошибка.

Никакой ошибки. С картой SunDisk результаты вполне ожидаемые. Получается, что быстродействие карточек одного класса может отличаться почти в 1000 раз! Причем этот параметр не документирован и его невозможно узнать при покупке.

Дело в том, что основное назначение SD карточек – запись фотографий и видео. Там данные записываются последовательно, большими блоками. Соответственно, скорость записи небольших блоков информации по случайным адресам не имеет значения.

Совсем другое дело, если карточка используется в качестве системного диска, как в Raspberry Pi. Там этот

параметр очень важен. Недостаточное быстродействие приводит к системным ошибкам или зависанию.

В общем, китайская карточка “несовместима” с Raspberry Pi. Минимальное значение скорости записи во второй и четвертой строчках, при которой карточка еще пригодна для установки операционной системы Raspberry Pi, около 0,02 МВ/сек. Результат теста может несколько отличаться для разных версий программы и разных компьютеров. Но точное значение не важно, важно количество нулей после запятой.

Эта же проблема может возникнуть и в других устройствах, например в смартфонах, в которых SD карточка может использоваться для хранения исполняемых файлов. Видеолюбителям тоже следует иметь в виду эту особенность и не допускать фрагментирования файлов на карте памяти. Т.е. в процессе видеосъемки не следует удалять или редактировать записанные видеофайлы. Начинать запись следует с чистого листа, т.е. карточки. Пусть информация пишется последовательными блоками.

Ну и как же выбрать нужную карту? Самое надежное – это поискать информацию на форумах, может быть кто-то уже приобрел и протестировал карту данного производителя и данного типа. Если ничего не удастся найти, нужно ориентироваться на известную марку и не покупать самые дешевые экземпляры. Быстродействующая память всегда дороже стоит. Хорошо, если кроме 10-го

класса будет указана маркировка U1. Почему все так сложно? Ну какой же уважающий себя производитель будет афишировать цифру 0,01 МВ/сек, если можно написать 10 МВ/сек. Реклама...

Как записать образ диска на карточку, я не буду подробно объяснять. Делается это с помощью программы Win32DiskImager (**рис. 3**). Если что-то непонятно, Google поможет.

Обратим внимание на другой момент. А что, если после записи образа RASPBIAN вам срочно понадобилась эта карточка для фотоаппарата? Форматирование не поможет, в Windows будет виден только диск объемом около 100 МВ.

Вот для этого и нужен Bootice (**рис. 4**). Выбираем “Utilities -- Select and Fill”. Затем выбираем “Physical disk”. “Sectors” ставим, например, 1000 и жмем “Start Filing” (**рис. 5**). Файловая система разрушена, теперь можно форматировать карточку. Только не ошибитесь и случайно не разрушите жесткий диск компьютера! Форматировать карточку лучше с помощью программы SD Card Formatter (**рис. 6**).

Вставляем карту с записанным образом RASPBIAN WHEEZY в слот Raspberry Pi, подключаем сетевой кабель

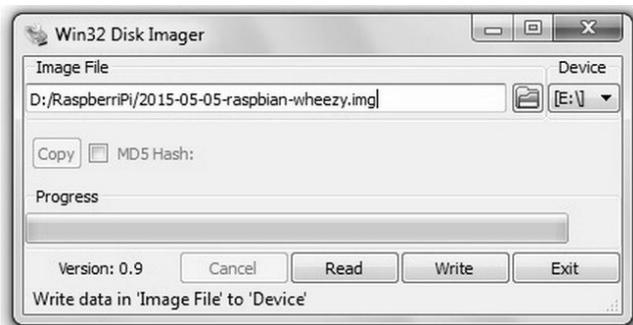


Рис. 3

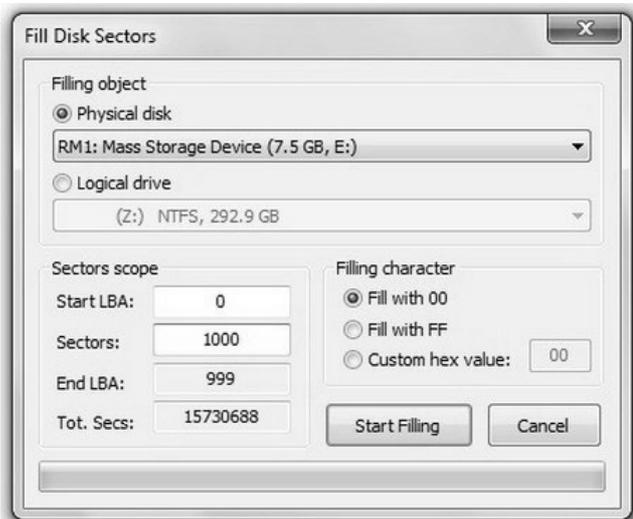


Рис. 5



Рис. 4

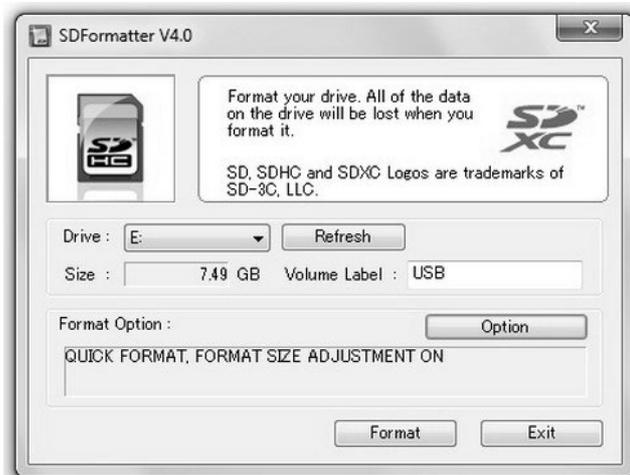


Рис. 6

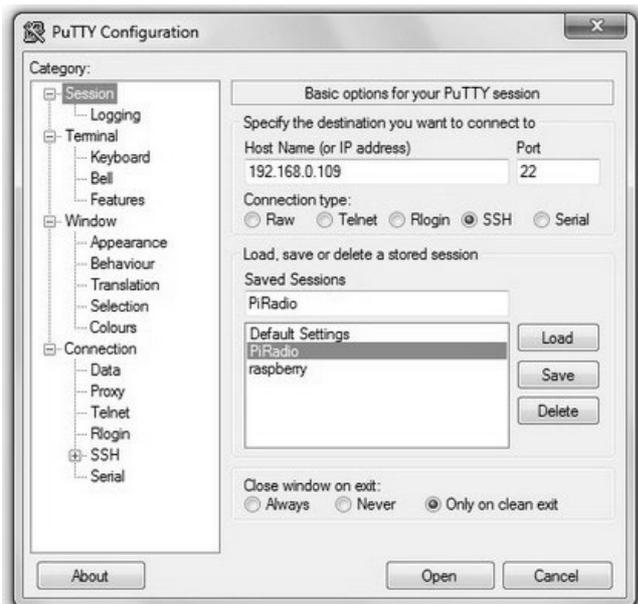


Рис. 7

от роутера и включаем питание. В дальнейшем можно будет использовать беспроводное Wi-Fi соединение, но на время наладки проводная связь предпочтительнее. Монитора и клавиатуры нет, поэтому установку и настройку программного обеспечения будем проводить дистанционно.

В первую очередь нужно выяснить, какой IP адрес роутер присвоил нашему интернет радиоприемнику. Заходим через WEB интерфейс в настройки роутера и выясняем, какой адрес присвоен устройству “raspberrypi”. Как это сделать – читайте руководство пользователя для вашей модели роутера и (или) спрашивайте у Google.

На компьютере запускаем PuTTY (рис. 7). В настройках в разделе “Translation” нужно выставить кодировку UTF-8, затем, вернувшись в раздел “Session”, в поле “Host Name (or IP Address)” ввести IP адрес. Поле “Port” оставляем по умолчанию – 22. Отмечаем SSH, а в поле “Saved sessions” вводим, например, PiRadio. Чтобы не делать это при каждом запуске, сохраняем настройки – “Save”. Затем выбираем в списке подключений “PiRadio” и жмем “Open”.

В открывшемся окне вводим имя пользователя “pi” и пароль “raspberrypi” (рис. 8). Если при первом запуске связь не устанавливается, выключите Raspberry Pi и через 5-10 сек вновь включите питание.

В первую очередь нужно сделать основные системные настройки. Вводим команду:

```
$ sudo raspi-config
```

Ввод любой команды в консоли всегда завершается нажатием “Enter”. Символ “\$” в начале строки означает, что команда вводится от имени пользователя “pi”. Как и какие настройки нужно сделать в программе конфигурации, я рассказывал в [10, 11]. Сделайте все в точности, как там описано. Единственное отличие – клавиатура не подключена, поэтому настроить ее раскладку не удастся. Новый пункт Device Tree оставляем по умолчанию – Enable.

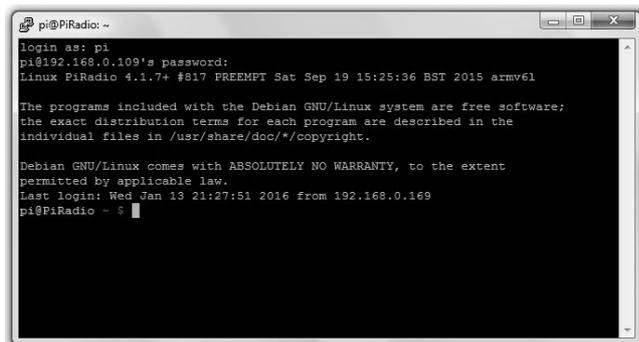


Рис. 8

После нажатия “Finish” система предложит перезагрузку. Соглашайтесь. После каждой перезагрузки необходимо перезапускать PuTTY и вновь вводить логин и пароль.

Теперь нужно сделать upgrade. Автор в своем описании не рекомендует делать upgrade, только update. Однако, у меня с этим никаких проблем не было. Поэтому я все-таки рекомендую сделать upgrade по максимуму, чтобы при установке не было конфликтов. Это справедливо для Raspberry Pi model B, как ведут себя B+ и 2, я не проверял. В крайнем случае, если возникнут сложности с установкой, придется повторить все сначала, сделав только update. Но, повторяю, у меня проблем не было.

```
$ sudo rpi-update
$ sudo reboot
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get dist-upgrade
$ sudo reboot
```

Со всеми предложениями соглашаемся – “Y”. Не спешите и терпеливо ждите, пока закончит выполняться каждая команда. Если появятся какие-то ошибки, наиболее вероятная причина – “несовместимая” SD карточка. Пока все не пройдет без ошибок, дальше двигаться нельзя. При наборе команд строго соблюдайте регистр и, где нужно, пробелы.

Ну вот, система установлена и настроена. Создаем “точку возврата” – сохраняем с помощью Win32DiskImager образ карточки с настроенной системой, чтобы в случае неправильных действий в дальнейшем не пришлось повторять все эти шаги. Для корректного закрытия системы перед отключением питания наберите команду

```
$ sudo halt
```

Теперь устанавливаем плеер mpd и его окружение. На предупреждения в конце установки не обращайтесь.

```
$ sudo apt-get install mpd mpc python-mpd
```

Сейчас нужно скопировать скачанный ранее дистрибутив **radiod_4.7_armhf.deb** в каталог **/home/pi**. Копировать проще всего с помощью программы WinSCP.

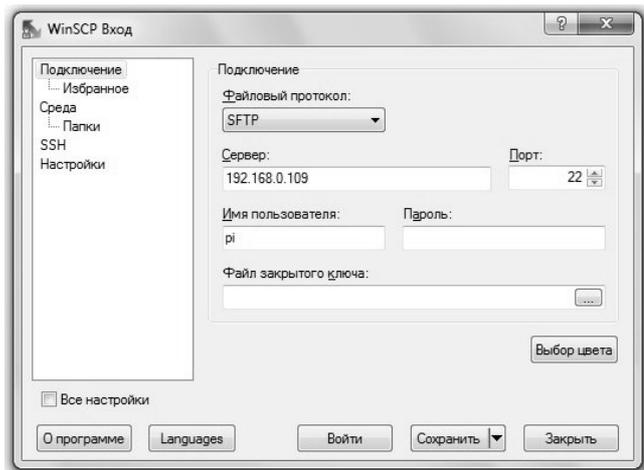


Рис. 9

Запускаем программу (рис. 9), в поле “Сервер” указываем IP адрес нашей Raspberry Pi. Порт 22, имя пользователя “pi”, пароль “raspberrу”. Нажимаем кнопку “Сохранить”. Затем выбираем в списке наше новое подключение и жмем кнопку “Войти”. В открывшемся окне соглашаемся с принятием ключа. В результате получаем доступ к файлам Raspberry Pi через удобный файловый менеджер (рис. 10).

Копируем пакет radiod_4.7_armhf.deb с компьютера в /home/pi и устанавливаем его, возвращаясь в PuTTY:

```
$ sudo dpkg -i radiod_4.7_armhf.deb
```

Программа инсталляции попросит выбрать нужную вам версию. Автор разработал 8 вариантов своего радио, однако на основе собственного опыта я рекомендую только первые четыре. Варианты с подключением индикатора по шине I2C (наверняка и по SPI) в версии 4.7 нестабильны. Я в этом убедился, пытаюсь запустить радио с платой на основе PCF8574.

Итак, выбираем нужный вариант. Я проверил:

- 1) Two line LCD with push buttons (radiod.py)
- 3) Two line LCD with rotary encoders (rradiod.py)

Подробнее о других вариантах читайте в авторском описании. Если все установилось без ошибок, можно

Ссылки

2. Сообщество raspbian.org - <http://www.raspberrypi.org/downloads/>
3. Win32DiskImager - <https://sourceforge.net/projects/win32diskimager/>
4. PuTTY - <http://www.chiark.greenend.org.uk/~sgtatham/putty/>
5. WinSCP - <http://winscp.net/eng/docs/lang:ru>
6. SD Card Formatter - https://www.sdcard.org/downloads/formatter_4/
7. Bootice - <http://www.ipauly.com/>
8. AkelPad - <http://akelpad.sourceforge.net/ru/index.php>
9. CristalDiskMark - <http://crystalmark.info/?lang=en>
10. Raspberry Pi – ваш второй компьютер - http://ra4nal.qrz.ru/rpi_comp_1.shtml
11. Н. Хлюпин (RA4NAL). Raspberry Pi – ваш второй компьютер. - Радиолобитель, 2015, №11, стр. 36-38; №12, стр. 46-48.

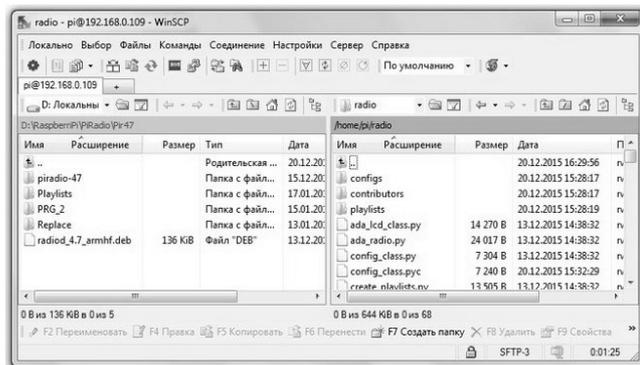


Рис. 10

перезагружаться. Если в процессе установки были какие-то ошибки, надо искать их причину. Для начала внимательно ознакомьтесь с тем, что советует автор. В его документации есть много полезных рекомендаций. У меня до этого шага все проходило без проблем.

Итак, перезагружаемся. Усилитель НЧ должен быть подключен к AUDIO выходу Raspberry Pi независимо от того, предполагаете ли вы использовать этот выход или USB звуковую карту.

```
$ sudo reboot
```

После перезагрузки вы услышите какую-нибудь станцию, а на индикаторе будет точное время и информация о станции. Вращая энкодеры или нажимая кнопки, можно регулировать громкость и выбирать станции, предварительно запрограммированные автором.

Авторский вариант Pi Radio установлен. Если что-то не работает, читайте в первую очередь описание от автора, разбирайтесь. Ищите ошибки в log файлах /var/log/radio.log и /var/log/mpd/mpd.log. Я не буду переписывать авторское описание, буду останавливаться только на проблемах, с которыми сам столкнулся.

Ресурсы проекта – программное обеспечение для Pi Radio by Bob Rathbone – radiod_4.7_armhf.deb (файл piradio_2.zip) вы можете загрузить с сайта журнала:

<http://www.radioliga.com> (раздел “Программы”);

с сайта автора по адресам:

<http://ra4nal.qrz.ru>, <http://ra4nal.lanstek.ru>



Интернет радиоприемник на Raspberry Pi

Николай Хлюпин
(RA4NAL)
г. Киров



Продолжение. Начало в №№9-10/2016

Часть 3. Настройка

Сейчас займемся настройкой и русификацией Pi Radio. Затем установим и настроим дистанционное управление.

Все файлы Pi Radio находятся в каталоге `/home/pi/radio`. Основной файл программы был выбран в процессе установки в зависимости от конфигурации аппаратной части. Для варианта с двухстрочным индикатором и энкодерами это файл `rradiod.py`, а для варианта с кнопками – `radiod.py`.

На всякий случай не поленитесь сделать сейчас точку возврата – выключите радио и создайте образ с помощью программы Win32DiskImager. Ссылки на все программы есть во второй части описания.

Итак, займемся русификацией интерфейса. Перед редактированием файлов Python с расширением `.py` нужно останавливать Pi Radio:

```
$ sudo service radiod stop
```

Запускаем на компьютере программу WinSCP, заходим в директорию `/home/pi/radio` и копируем файл `rradiod.py` (для варианта с двухстрочным индикатором и энкодерами) на компьютер для редактирования. Оригинальный файл в `/home/pi/radio` переименовываем в `rradiod_org.py`. Для варианта радио с кнопками аналогичным образом работаем с файлом `radiod.py`.

Если нет желания разбираться и редактировать файлы, можете взять нужный отредактированный файл из архива, все файлы там для версии PiRadio 4.7.

Во-первых, в процессе монтажа сложно угадать направление вращения энкодеров. Если оказалось, как у меня, что направление нужно сменить, сделать это можно программно. Ищем в файле строки:

```
# Switch definitions #
Volume rotary encoder
LEFT_SWITCH = 15
RIGHT_SWITCH = 14 ...
```

При необходимости меняем местами цифры. Для варианта с кнопками это менее актуально, но, при необходимости, тоже возможно. Имейте в виду, что можно только менять цифры местами, заменить их на другие нельзя.

Очень важный момент! В языке программирования Python важно количество пробелов в начале строки. Почему-то программисты не любят пробелы, а предпочитают использовать табуляцию. Поэтому нужно редактировать файлы в редакторе, который позволяет определить, сколько пробелов или табуляций в начале строки. И не менять при редактировании их количество.



Помните, если заменить, например, одну табуляцию на четыре пробела, работа программы может полностью нарушиться, хотя визуально в тексте ничего не изменится. Я использовал для редактирования AkelPad.

Изменение громкости при вращении энкодера оказалось слишком плавным, ускорим его. Ищем строки:

```
volume = radio.getVolume()
while volAdjust > 0:
    volume -= 1
```

и чуть дальше:

```
volume = radio.getVolume()
while volAdjust > 0:
    volume += 1
```

Заменяем цифру 1 на 2 в обоих фрагментах. Т.е. получим ... `-= 2` и ... `+= 2`. Для варианта с кнопками этот шаг пропускаем, там все и так нормально. Затем ищем фрагмент кода:

```
if len(ipaddr) < 1 and radio.getSource() != radio.PLAYER:
    lcd.line2("No IP network")
```

Комментируем эти строки:

```
# if len(ipaddr) < 1 and radio.getSource() !=
radio.PLAYER:
# lcd.line2("No IP network")
```

Для варианта радио с управлением кнопками в файле `radiod.py` этот фрагмент выглядит немного иначе:

```
# if len(ipaddr) < 1:
# lcd.line2("No IP network")
```

Это нужно для того, чтобы можно было слушать музыкальные файлы на локальном источнике без подключения к сети Интернет. Очевидно, Bob Rathbone даже представить себе не может, что можно жить без Интернета.

РАДИОПРИЕМ

На этом редактирование файла для “китайского” индикатора (с отображением русских букв латиницей) закончено. Сохраняем файл. Если ваш индикатор приобретен в России и вы хотите видеть на нем русские буквы, продолжаем. Ищем в файле строку:

```
lcd.scroll2(current[0:160],interrupt)
```

Убираем из нее параметр [0:160].
Получается:

```
lcd.scroll2(current,interrupt)
```

Точно так же поступаем еще с двумя строками:

```
lcd.scroll1((" + str(index+1) + ") +  
current_artist[0:160],interrupt)
```

```
...  
lcd.scroll2(current_station[0:160],interrupt)
```

Имейте в виду, что эти строки идут не друг за другом. Будьте внимательны и не удалите что-нибудь лишнее! Этим мы снимаем ограничение на длину строки на индикаторе, что нужно для корректного отображения русских названий станций и треков.

Сохраняем отредактированный файл, затем копируем его обратно на Raspberry Pi в `/home/pi/radio/`. Напоминаю, что я выложил как русские, так и латинские варианты отредактированных файлов `rradiod.py` и `radiod.py` в архиве.

Переименовываем еще один файл – `/home/pi/radio/translate_class.py` в `translate_class_org.py`. Затем копируем русский или латинский вариант файла `translate_class.py` из моего архива в `/home/pi/radio/`. Все это легко делается в WinSCP. Изменений в этом файле гораздо больше, я не буду их описывать. При желании можете сравнить с оригиналом.

Для русского индикатора, который поддерживает отображение кириллицы, нужно заменить еще один файл – `/home/pi/radio/lcd_class.py` одноименным файлом из моего архива. Оригинальный файл на всякий случай не удаляйте, а переименуйте, как описано выше.

Таким образом, для радио с “китайским” индикатором мы заменили два файла: **rradiod.py** (или **radiod.py**) и **translate_class.py**. Для радио с индикатором, поддерживающим отображение кириллицы, дополнительно заменили еще один файл: **lcd_class.py**. Все эти файлы должны быть в `/home/pi/radio/`.

Это еще не все. В Linux гораздо больше проблем с правами доступа, чем в Windows. Установите на все скопированные или переименованные файлы права доступа **rw-rw-rw (755)** (рис. 1). Это можно сделать тоже в WinSCP нажатием кнопки “F9 Свойства”.

Запускаем радио:

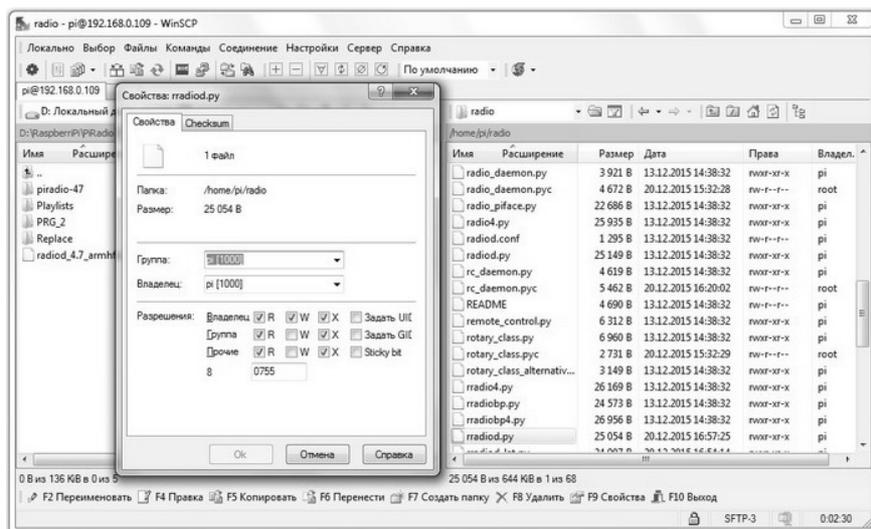


Рис. 1

```
$ sudo service radiod start
```

Если радио работает и управляется, значит вы все сделали правильно. Если нет, восстанавливаем точку возврата и начинаем все сначала, но уже с проверкой работоспособности после редактирования каждого файла. Не забывайте останавливать и запускать радио.

Теперь можно подключить USB звуковую карту. Я проверил две вот такие карты (рис. 2), обе работают с Raspberry Pi. Но в первой (той, что слева), очевидно, установлен USB разъем очень плохого качества. Малейшие прикосновения к нему вызывают отчетливо слышимый “шурш” и шорох. Поэтому, лучше вскрыть корпус и припаять кабель с USB разъемом непосредственно к плате.

Вторая карта не имеет такого недостатка, качество звучания вполне на уровне. Но при каждом 5-6 включении приемника она не активируется, в динамиках – тишина. После выключения и повторного включения радио все ОК. В чем причина – не знаю.

Чтобы включить вывод звука через USB, нужно отредактировать файл `/etc/modprobe.d/alsa-base.conf`. Методика справедлива только для RASPBIAN WHEEZY, в новой системе RASPBIAN JESSIE формат этого файла совсем другой и находится он по другому адресу.

```
$ sudo nano /etc/modprobe.d/alsa-base.conf
```



Рис. 2

Ищем и комментируем в нем строку:

```
#options snd-usb-audio index=-2
```

Вместо нее добавляем две строки:

```
options snd-usb-audio index=0 nrpacks=1
options snd-bcm2835 index=-2
```

Сохраняем отредактированный файл и перезагружаемся. Если mpd не запускается, звука нет или громкость не регулируется, открываем аналогичным образом файл **/etc/mpd.conf**, ищем там секцию **audio_output** и добавляем в нее параметр **mixer_type "software"**. Получится так:

```
audio_output {
type      "alsa"
name      "My ALSA Device"
device    "hw:0,0"
format    "44100:16:2"
mixer_device  default"
mixer_control "PCM"
mixer_index  "0"
mixer_type  "software" # Add this line for some
USB devices
}
```

Сохраняем изменения и перезагружаемся. Теперь все должно работать. Если все-таки не работает, читайте авторское описание, ищите информацию в Интернет.

Можно увеличить уровень сигнала с USB звуковой карты, что улучшит отношение сигнал/шум. Для этого наберите команду:

```
$ alsamixer
```

В открывшемся окне нажмите "F6" и выберите "USB Audio Device". Затем перейдите на "Speaker" и установите уровень выхода 60...80 (**рис. 3**).

Если все работает нормально, приступаем к установке дистанционного управления – LIRC. Сначала ставим **PiFacecad**:

```
$ sudo apt-get install python{,3}-pifacecad
```

Перезагружаемся и продолжаем установку.

```
$ sudo reboot
$ sudo bash
# cd /home/pi/radio
```

Скачиваем и запускаем **setup_pifacecad_lirc.sh**:

```
# wget https://raw.githubusercontent.com/piface/pifacecad/master/
bin/setup_pifacecad_lirc.sh
# chmod +x setup_pifacecad_lirc.sh
# ./setup_pifacecad_lirc.sh
```

На вопрос "Do you wish to continue?", отвечаем "Yes", выбрав пункт "1". Когда установка закончится, нужно отредактировать в файле **/boot/config.txt** параметр **gpio_in_pin**, т.е. указать, куда подключен ИФК приемник. В нашем случае это pin 21 (GPIO 9).

```
# nano /boot/config.txt
```

В открывшемся для редактирования файле в самом его конце ищем строку, которая начинается с "dtoverlay=" и исправляем в ней **gpio_in_pin=9**.

```
dtoverlay=lirc-rpi,gpio_in_pin=9,gpio_in_pull=high
```

Сохраняем отредактированный файл, закрываем редактор и продолжаем.

```
# cp lircrc.dist /etc/lirc/lircrc
# reboot
```

LIRC установлен, убедимся, что все работает.

```
$ sudo mode2 -d /dev/lirc0
```

Если при нажатии кнопок на пульте на экран будут выводиться примерно такие строки, значит вы все сделали правильно.

```
space 5358604
pulse 4591
space 4459
pulse 651
.....
```

Если ничего не происходит, – ошибка в монтаже, неисправен фотоприемник или пульт. Ну и почитайте рекомендации автора в описании PiRadio. У меня проблем с установкой LIRC не было. Для выхода из программы нажмите Ctrl-C.

Осталось самое главное – настроить систему ДУ под конкретный пульт. LIRC позиционируется, как универсальная система, способная работать с любым

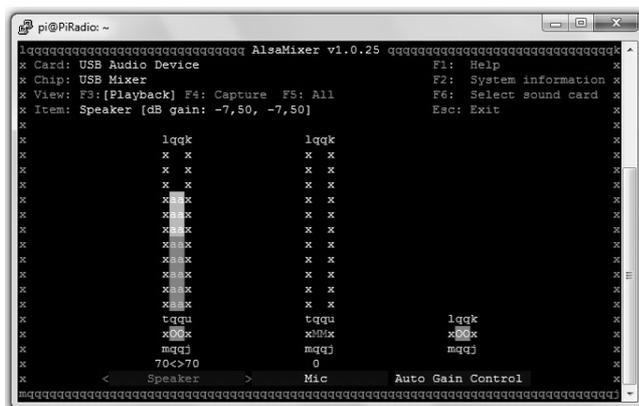


Рис. 3

пультом ДУ. В некоторых источниках, правда, добавляется “почти”. Можно найти массу информации о том, как изготовить и настроить систему. Особенно меня впечатлили подробнейшие инструкции и высококачественные фотографии, разъясняющие, как припаять три провода к фотоприемнику...

А вот о том, как конкретно работает система, что и каким образом там измеряется, информации практически нет. Даже на сайте разработчиков я не нашел информации о том, каким требованиям должен удовлетворять сигнал, формируемый пультом ДУ. Сколько максимум, минимум может быть импульсов в команде, какова их допустимая длительность. С какой точностью и периодичностью измеряются параметры импульсов и т.п.

Вместо нескольких строк конкретной информации ссылка на огромную базу данных протестированных на совместимость пультов. Увы, но эта база практически бесполезна, т.к. точная марка пульта в большинстве случаев неизвестна, да и никто не запрещает производителям радиоаппаратуры “вносить в конструкцию незначительные изменения, не влияющие на качество”.

Настройка заключается в создании специального файла конфигурации **/etc/lirc/lircd.conf**. Сначала переименуем существующий файл.

```
$ sudo mv /etc/lirc/lircd.conf /etc/lirc/lircd.conf.org
```

Затем запускаем программу конфигурации. Автор рекомендует запускать программу с дополнительным параметром `-f`, но, по-моему, это ничего не дает, скорее наоборот, все портит. Так что набирайте в PuTTY:

```
$ sudo irrecord -d /dev/lirc0 /etc/lirc/lircd.conf
```

Если программа `irrecord` не запускается, а выдает сообщение об ошибке, перезагрузитесь.

При запуске программа выводит на экран подробные инструкции, что нужно делать. Если появятся сообщения об ошибках, читайте авторское описание, ищите информацию в Интернет. Если все ОК, строго следуйте инструкциям. Вас попросят нажимать кнопки, затем ввести имя для каждой кнопки. Используйте только имена, рекомендуемые автором. Их назначение, думаю, понятно.

```
KEY_VOLUMEUP
KEY_VOLUMEDOWN
KEY_CHANNELUP
KEY_CHANNELDOWN
KEY_MUTE KEY_MENU
KEY_UP KEY_DOWN
KEY_LEFT
KEY_RIGHT
KEY_OK
```

Кнопки `KEY_MENU` и `KEY_OK` равноценны. Можно найти много информации о настройке LIRC с помощью программы `irrecord`, поэтому я не буду подробно описывать этот процесс. Если что-то непонятно, Google поможет.

Конфигурируем установленное программное обеспечение для автозапуска и запускаем его.

```
$ sudo update-rc.d -f pifacercd defaults
$ sudo service pifacercd start
```

Теперь можете проверить, управляется ли приемник с пульта ДУ. Если все работает, вам повезло. Если какие-то кнопки не распознаются, попробуйте повторить настройку, предварительно удалив файл `/etc/lirc/lircd.conf` и перезагрузившись.

```
$ sudo rm /etc/lirc/lircd.conf
$ sudo reboot
```

Если это не поможет, попробуйте настроить другой пульт ДУ.

Наконец, система ДУ настроена и команды уверенно распознаются. Если светодиод активности пульта ДУ не вспыхивает при нажатии кнопок на пульте, откройте для редактирования файл `/etc/radiod.conf`.

```
$ sudo nano /etc/radiod.conf
```

Убедитесь, что значение параметра `remote_led=11`:

```
# Output LED for remote control, default GPIO 11 (pin
23) or
# GPIO 13 (pin 33) for AdaFruit plate or PiFace CAD
(40 pin RPi needed)
# remote_led=0 is no output LED
remote_led=11
```

При необходимости, исправьте. Изменения вступят в силу после перезагрузки или перезапуска PiRadio.

Ресурсы проекта – файлы для русификации Pi Radio by RA4NAL – `piradio_replace.zip` (файл `piradio_3.zip`) вы можете загрузить с сайта нашего журнала:

<http://www.radioliga.com> (раздел “Программы”);
с сайта автора по адресам:
<http://ra4nal.qrz.ru>, <http://ra4nal.lanstek.ru>



Окончание в №12/2016

**МИР ЭЛЕКТРОНИКИ:
радиолюбительские
конструкции RA4NAL
<http://ra4nal.qrz.ru>,
<http://ra4nal.lanstek.ru>**

Интернет радиоприемник на Raspberry Pi

Николай Хлюпин
(RA4NAL)
г. Киров



Окончание. Начало в №№9-11/2016

Часть 4. Руководство пользователя

Интернет радиоприемник изготовлен, необходимое программное обеспечение установлено и настроено. Здесь я расскажу, как пользоваться приемником.

В отличие от эфирного УКВ приемника интернет радио не может само настраиваться на станции. Предварительно нужно создать playlist, в котором будут указаны URL радиостанций.

Т.е. вначале нужно узнать адреса нужных вам станций. Выбор большой – порядка 10000 станций. Но если вы читаете эти строки, значит вас интересуют в первую очередь русскоязычные станции. Выбор тоже большой. Вопрос, где взять адреса.

Дело в том, что для станции выгоднее, когда вы ее слушаете с официального сайта, а не через плеер. На сайте ведь можно разместить много дополнительной рекламы. Поэтому, в большинстве случаев станции не дают прямой ссылки на URL потока. Правда, бывают и исключения. Кроме того, часто можно найти ссылку, просматривая исходный html код странички. Как искать, объяснять не буду. Если вы дошли до этого места в настройке приемника, при желании сможете разобраться и с html кодом. Кроме того, есть сайты, с которых можно слушать множество станций, например:

<http://www.glaz.tv/online-radio/> – Глаз.TB
<http://radiopotok.ru/> – Радио Поток
http://guzei.com/online_radio/ – Guzei.com
<http://rusradist.narod.ru/> – ТВ и Радио г.Москва
<http://www.windowsmedia.com/radioui/home.aspx> – Windows Media Guide

Ссылка на станцию должна начинаться с `http://`. Чтобы убедиться, что ссылка работает, скопируйте ее и проверьте в каком-нибудь плеере, например, Winamp. Выберите в меню Winamp “Файл” — “Открыть ссылку” и вставьте скопированную ссылку. Если услышите то, что хотели – ссылка работает. Можно составлять playlist.

Существует несколько форматов плейлистов – **m3u**, **pls**, **asx**. Подробно о каждом из них можно прочитать в авторском описании Pi Radio. Формат **ram**, к сожалению, не поддерживается. Самый простой и удобный для редактирования, на мой взгляд – **m3u**. Это обычный текстовый файл с любым именем и расширением **m3u**. Его формат такой:

```
#EXTM3U
#EXTINF: -1, Radio 10 Gold NL
http://icecast.streaming.castor.nl:80/radio10
.....
```



Рис. 1

В первой строчке файла должен быть заголовок – **#EXTM3U** (рис. 1). Затем идет список радиостанций, на каждую станцию по две строки. В первой, после символов **#EXTINF: -1**, название станции. Оно может быть любым, на ваше усмотрение, но обязательно в кодировке UTF-8, иначе не будет корректно отображаться на индикаторе. Поэтому составляйте и редактируйте плейлист в редакторе AkeiPad, выбрав в нем кодировку UTF-8. Следующая строчка – URL радиостанции.

И так по 2 строки на каждую станцию. Последняя строка в файле обязательно должна заканчиваться нажатием “Enter”, иначе плеер зависнет. Станции будут проигрываться в том порядке, в каком они занесены в плейлист. Плейлистов может быть несколько, они будут отсортированы плеером в алфавитном порядке. Все они должны находиться в директории **/var/lib/mpd/playlists**.

Автор разработал специальную программу для составления плейлистов. Но, по-моему, проще и удобнее сделать это вручную. Если хотите попробовать авторскую методику – читайте его описание [1].

При установке программного обеспечения несколько плейлистов будут созданы автоматически. Можно их отредактировать или удалить и составить свои. Можете также использовать мои плейлисты, см. прикрепленный в архиве файл.

Только не судите строго – вкусы у всех разные, я выбирал в соответствии со своим. Не нравится – выбирайте то, что нравится именно вам.

С течением времени наверняка потребуется что-то удалить или добавить новые станции. Сделаем процедуру редактирования и загрузки плейлистов в интернет приемник максимально простой. Самое удобное и простое решение – установить пакет Samba. Запускаем PuTTY и набираем:

```
$ sudo apt-get install samba samba-common-bin
```

На предупреждения отвечаем “Y”. Ждем завершения установки. Если используется SD карта объемом 8 и более ГБ, создадим на ней в директории /home/pi папку для музыкальных файлов и свяжем ее с папкой /var/lib/mpd/music, где плеер mpd ищет музыку. Помните, строчные и прописные буквы в Linux – это разные вещи.

```
$ mkdir /home/pi/mymusic
$ cd /var/lib/mpd/music
$ sudo ln -s /home/pi/mymusic
```

Плейлисты находятся в папке /var/lib/mpd/playlists. Вот к этим двум папкам и нужно обеспечить удобный доступ по локальной сети. Выставим права полного доступа к этим директориям.

```
$ sudo chmod 777 /home/pi/mymusic
$ sudo chmod 777 /var/lib/mpd/playlists
```

Теперь нужно настроить Samba. Возьмите файл **smb.conf** из моего архива (ссылка в конце статьи) и скопируйте его с помощью программы WinSCP в /home/pi. Затем открываем PuTTY и перемещаем этот файл в /etc/samba. На всякий случай предварительно переименовываем оригинальный smb.conf.

```
$ cd /home/pi
$ sudo mv /etc/samba/smb.conf /etc/samba/smb_org.conf
$ sudo mv smb.conf /etc/samba/smb.conf
```

WinSCP не сможет этого сделать, т.к. программа запущена от имени пользователя pi. Перезагружаемся. Теперь в локальной сети будет виден новый компьютер **raspberrypi** с двумя папками **mymusic** и **playlists**, которые доступны всем пользователям сети.

Но нужно еще настроить локальную сеть на компьютере Windows, если вы не сделали этого раньше. Запустите Total Commander и выберите вместо локального диска “Сетевое окружение”. Если через непродолжительное время там появится RASPBERRYPI – вам повезло, все работает. Можно копировать музыкальные файлы и плейлисты на SD карту.

Если в сетевом окружении нет RASPBERRYPI, попробуйте выключить Pi Radio. Подождите 1-2 минуты и вновь включите питание. Если RASPBERRYPI так и не появился в сетевом окружении – нужно настраивать локальную сеть на компьютере Windows. Вопрос отдельный и достаточно сложный. Не буду на нем останавливаться, т.к. не специалист в этом. Google вам поможет. Можно подключить Raspberry Pi и как сетевой диск.

Наконец, все настроено. Копируем созданные самостоятельно или взятые из моего архива плейлисты в папку **playlists** с помощью, например Total Commander, предварительно очистив эту папку. Можно сохранить авторские плейлисты на компьютере на всякий случай.

Аналогично, музыкальные файлы в формате mp3 копируем в **mymusic**. Если использована карта на 8 ГБ,

на ней будет более 4 ГБ свободного места. Более чем достаточно для музыки, учитывая, что автор не предусмотрел никакой навигации. Все музыкальные файлы свалены в одну большую кучу. Можно только их последовательно перебирать или включить режим случайного воспроизведения.

Хотя Bob Rathbone и пишет о возможности навигации в файлах, реально ее нет. На индикаторе отображается информация только из тэгов mp3 файлов, а не их имена и расположение. Поэтому тэги должны быть в кодировке UTF-8, иначе на индикаторе будут иероглифы. Это, конечно, большой недостаток и поле деятельности для доработки программы. Но ведь нам нужен в первую очередь интернет радиоприемник, а не музыкальный плеер.

Не имеет смысла использовать карту или флэшку большого объема – трудно будет что-то на ней найти. Однако при записи музыки не заполняйте все место на карте под завязку. Оставляйте свободным хотя бы 100 МБ для работы системы.

Наконец, два заключительных штриха.

```
$ sudo nano /etc/radiod.conf
```

В открывшемся файле заменяем строку **loglevel=INFO** на **loglevel=NONE**. Этим мы отключаем запись лога при работе радио. Если все работает нормально, он не нужен. Ведь ресурс SD карточки не бесконечен. Если потребуется, запись лога всегда можно снова включить.

И второе. Запускаем **raspi-config**, выбираем **Advanced Options — Hostname** и меняем имя, под которым наше радио видно в сети. По умолчанию там стоит raspberrypi. Меняем его для наглядности на **PiRadio**. Перезагружаемся.

Pi Radio может проигрывать музыку не только с SD карты, но и с флэшки. К сожалению, эта функция тоже не доведена до ума. Чтобы воспроизвести файлы с флэшки, придется очистить папку mymusic на SD карте, т.е. удалить все ее содержимое. Если все равно вы получите сообщение о том, что USB Flash не найдена, отформатируйте флэшку с помощью именно программы SD Card Formatter в FAT32 и после этого скопируйте на нее файлы в формате mp3. Можно создать папки, но они все равно не будут видны.



Рис. 2

Теперь расскажу, как пользоваться приемником и слушать музыкальные файлы. После включения радиоприемник включается в режим "TIME". В первой строке индикатора отображается дата и время, во второй – название станции и звучащая в данный момент композиция (рис. 2). Эта информация извлекается из принимаемых данных.

Одним энкодером можно переключать станции, вторым – регулировать громкость. Кнопка первого энкодера – "Menu", второго – "Mute". Управление в варианте с кнопками аналогично, только функция "Mute" включается при одновременном нажатии кнопки увеличения и уменьшения громкости.

При нажатии кнопки "Menu" приемник переходит в режим "SEARCH". Вращая энкодер или нажимая кнопки, можно выбирать нужную станцию, названия станций из плейлиста выводятся в первой строке индикатора. Повторное нажатие "Menu" подтверждает выбор станции и включает ее прослушивание.

Если после входа в режим "SEARCH" энкодер не вращать, а еще раз нажать "Menu", приемник перейдет в режим "SOURCE". Здесь энкодером или кнопками можно выбрать режим работы – "Internet Radio" или "Music library". Но этот выбор при выключении питания не сохраняется. Источник, который будет выбираться по умолчанию, при включении приемника, – RADIO или MEDIA – можно установить в файле radiod.conf.

Следующее нажатие "Menu" включает режим "OPTIONS". С помощью энкодера можно выбрать: Random, Consume, Repeat, Reload Music, Timer, Alarm. Наиболее нужные опции это Random – воспроизведение локальной музыки в случайном порядке и Reload Music – этот пункт необходимо выбирать после загрузки музыкальных файлов или плейлистов на SD карту, а также после подключения флэшки. Опции включаются и выключаются вторым энкодером – тем, которым регулируется громкость.

Нажатие "Menu" – выход из этого режима. Следующий режим – "IP address", это просто информация. Еще раз нажав "Menu", возвращаемся в основной режим – "TIME".

Я рассказал далеко не обо всех возможностях PiRadio. Основное внимание я уделил проблемам, с которыми столкнулся при установке и настройке программного обеспечения. Также постарался максимально подробно описать некоторые важные детали, которые только вскользь упоминаются автором. Надеюсь, что элементарные вещи понятны и без разъяснения.

Кроме основных функций Pi Radio может показывать RSS новости и воспроизводить подкасты. Однако, учитывая скромные возможности дисплея, я не вижу смысла использовать эти "навороты". Если кому-то

интересно, читайте авторское описание, там объясняется, как это настроить.

Проигрывать музыкальные файлы можно не только с SD карты или флэшки, но и с диска в локальной сети. Тоже не вижу смысла, учитывая более чем скромные возможности навигации в файловой системе. Как искать нужный трек в терабайтах информации? Но, опять же, желающие могут найти все подробности по монтированию сетевого диска в авторском описании.

Более того, PiRadio может транслировать поток в локальную сеть, вы сможете его прослушивать через WEB интерфейс на любом устройстве, например смартфоне или компьютере в этой сети. Наверное, я отстал от прогресса современных технологий, т.к. не понимаю, зачем это нужно. Почему бы с компьютера не зайти напрямую на сайт радиостанции. Зачем промежуточный узел?

Можно контролировать работу приемника с мобильного устройства. С этим я тоже не разобрался из-за отсутствия необходимости в таком сервисе. Классический пульт ДУ для меня как-то удобнее.

Функция таймера... Наверное, засыпать лучше в тишине, не мешая окружающим. Кроме того, питание приемника при срабатывании таймера все равно не отключится. Будильник. Если у вас нет такого специализированного девайса, функция будильника есть в любом соевом телефоне. Это гораздо проще и надежнее.

Наконец, я ничего не написал о том, как настроить Wi-Fi. Просто я использовал проводную сеть и рассказал только о том, что протестировал. Мои аргументы в пользу проводной сети я уже описал на страничке [].

Если кому-то интересны все вышеперечисленные навороты – читайте описание от автора.

В заключение – список файлов и папок, которые могут потребоваться при настройке и эксплуатации приемника.

```
/etc/mpd.conf – файл конфигурации плеера mpd;
/etc/radiod.conf – файл конфигурации PiRadio;
/etc/samba/smb.conf – файл конфигурации Samba;
/etc/lirc/lircd.conf – файл конфигурации LIRC;
/var/log/mpd/mpd.log – лог плеера mpd;
/var/log/radio.log – лог загрузки PiRadio;
/var/lib/mpd/playlists – плейлисты для радио;
/home/pi/mymusic – папка для музыки mp3 и wma;
/home/pi/radio – все файлы программы PiRadio.
```

Видео см. по адресу: <https://youtu.be/zX3LoWTw0ng>

Ресурсы проекта – плейлисты для Pi Radio, файл конфигурации SAMBA для Pi Radio (файл *piradio_4.zip*) вы можете загрузить с сайта нашего журнала:

<http://www.radioliga.com> (раздел "Программы");

с сайта автора по адресам:

<http://ra4nal.grz.ru>, <http://ra4nal.lanstek.ru>



Ссылки

12. Медиацентр на Raspberry Pi - http://ra4nal.grz.ru/rpi_elec_1.shtml

13. Николай Хлюпин (RA4NAL). Медиацентр на Raspberry Pi. - Радиолюбитель, 2015, №9, стр. 49-51; №10, стр. 32-35.