

mount-rpi-image.sh

fa0x0998e58 [0,0]

Contents: [Dobrica PavlinuÄiÄ 's random unstructured stuff]

- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(Interestring projects\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(development\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(OV7670\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(DHT11/DHT22\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(BlinkM + i2c\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(CEC - control your TV\)](#)
 - ♦ [Dobrica PavlinuÄiÄ 's random unstructured stuff \(TV on\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(Hints from forum\)](#)
 - ♦ [Dobrica PavlinuÄiÄ 's random unstructured stuff \(screen saver\)](#)
 - ♦ [Dobrica PavlinuÄiÄ 's random unstructured stuff \(YouTube transcoding in VLC\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(Education materials\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(Raspberry Pi Accessories Starter Kit\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(GPU\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(Arduino\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(JTAG\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(Arduino ProMini 3.3V 8MHz\)](#)
 - ♦ [Dobrica PavlinuÄiÄ 's random unstructured stuff \(Disable serial port console\)](#)
 - ♦ [Dobrica PavlinuÄiÄ 's random unstructured stuff \(connect Arduino serial\)](#)
 - ♦ [Dobrica PavlinuÄiÄ 's random unstructured stuff \(flasing\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(DS1307 Tiny RTC\)](#)
 - ♦ [Dobrica PavlinuÄiÄ 's random unstructured stuff \(DS18B20\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(avrdude\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(logic analyzer\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(rotary encoder\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(SWD\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(PIC\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(esp8089\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(PS/2 GPIO\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(GPIO header annotations\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(Signal generator\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(LCD panels\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(Pi Zero USB OTG\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(PATA IDE emulation\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(DPI LCD screen\)](#)
- [Dobrica PavlinuÄiÄ 's random unstructured stuff \(i2c clock stretching\)](#)

fa0x099a361b8 [0,0]

Recent Changes in Tag raspberrypi

- [raspios](#)
- [PCF8591](#)
- [OV7670](#)

Interesting projects

TODO: test them out and write full description on this wiki :-)

- FreeLSS is a laser scanning program for the Raspberry Pi. It allows a Raspberry Pi to function as the core to a complete turn table laser scanning system.
 - ♦ <https://github.com/hairu/freelss>
- PWM on the Raspberry pi - done properly (in hardware, stable)
 - ♦ <https://github.com/sarfata/pi-blaster>
- <https://github.com/ali1234/raspi-teletext>
- https://github.com/jgarff/rpi_ws281x
- u-boot on sdcard for nfs boot: http://elinux.org/RPi_U-Boot
- RF transmitter for Raspberry Pi <https://github.com/F5OEO/rpitx>

development

- distcc setup <http://openframeworks.cc/setup/raspberrypi/Raspberry-Pi-DISTCC-guide.html>
- RPi (Raspbian arm) chroot on x86 Linux
http://dev.xff.lt/b/archives/2015/01/11/rpi_raspbian_arm_chroot_on_x86_linux/
 - ♦ <https://wiki.debian.org/QemuUserEmulation>
 - ♦ <https://wiki.debian.org/RaspberryPi/qemu-user-static>

OV7670

- <http://www.raspberrypi.org/phpBB3/viewtopic.php?uid=25856&f=44&t=11839&start=0>

DHT11/DHT22

- <http://www.tortosaforum.com/raspberrypi/dht11driver.htm>
- <http://learn.adafruit.com/dht-humidity-sensing-on-raspberry-pi-with-gdocs-logging/overview>

BlinkM + i2c

- <http://www.raspberrypi.org/forums/viewtopic.php?t=11969>

CEC - control your TV

<http://www.raspberrypi.org/forum/viewtopic.php?f=35&t=15749>

```
root@raspberrypi:/home/pi# git clone https://github.com/Pulse-Eight/libcec.git
root@raspberrypi:/home/pi# cd libcec/
root@raspberrypi:/home/pi/libcec# apt-get install autoconf automake libtool liblockdev1-dev libud
root@raspberrypi:/home/pi/libcec# sh -x bootstrap
root@raspberrypi:/home/pi/libcec# ./configure --with-rpi-include-path=/opt/vc/include/ --with-rpi
```

```
#####
```

```
libCEC version 2:1:0 configured
```

Compilation flags:

```
CXXFLAGS :      -g -O2 -fPIC -Wall -Wextra -Wno-missing-field-initializers -Wno-psabi
libCEC LDFLAGS : -lbcm_host -llockdev -lrt -ldl -lpthread -L/opt/vc/lib/ -lvcos -lvchiq_arm -l
client LDFLAGS : -lrt -ldl -lpthread
```

Configured features:

```
Pulse-Eight CEC Adapter :      yes
Pulse-Eight CEC Adapter detection :  yes
Raspberry Pi support :      yes
TDA995x support :      no
```

You can now build libCEC by running:
make

#####

make install

TV on

```
pi@raspberrypi ~ $ echo "on 0" | cec-client -s
```

Hints from forum

screen saver

<http://www.raspberrypi.org/phpBB3/viewtopic.php?p=84831#p84831>

```
# prevent screensaver from powering down display
setterm -blank 0 -powerdown 0 > /dev/tty0

# clear the display and turn off the flashing cursor
clear > /dev/tty0
setterm -cursor 0 > /dev/tty0

# turn the cursor back on when done with omxplayer
setterm -cursor 1 > /dev/tty0
```

YouTube transcoding in VLC

<http://www.raspberrypi.org/phpBB3/viewtopic.php?p=163658#p163658>

```
cvlc 'youtube.com/watch?v=XXXXXXXXXX' :sout='#transcode{vcodec=h264,vb=0,fps=0,scale=0,height=720
```

Education materials

- http://pi.cs.man.ac.uk/download/Raspberry_Pi_Education_Manual.pdf
- <http://www.ocr.org.uk/qualifications/by-subject/computing/raspberry-pi/>

- <http://www.cl.cam.ac.uk/projects/raspberrypi/>

Raspberry Pi Accessories Starter Kit

<http://www.conrad.com/ce/en/product/409208/Raspberry-Pi-Accessories-Starter-Kit?ref=searchDetail>

- 1x Rapberry sticker
- Power supply for the Raspberry
- Sturdy acrylic housing, which includes all possible connections for this Raspberry and perfectly protects
- 4 GB SD card to store your programs
 - ♦ <https://learn.adafruit.com/adafruit-raspberry-pi-lesson-1-preparing-and-sd-card-for-your-raspb>
- USB Micro Card Reader
- USB cable (for power supply or the power supply of a PC)
- 3 m Ethernet cable
 - ♦ <https://learn.adafruit.com/adafruits-raspberry-pi-lesson-6-using-ssh>
 - ♦ <https://learn.adafruit.com/adafruit-raspberry-pi-lesson-7-remote-control-with-vnc>
- TTL USB cable
 - ♦ <https://learn.adafruit.com/adafruits-raspberry-pi-lesson-5-using-a-console-cable/test-and-conf>
- PI Cobbler (KIT hereby connect the RP with other circuit boards)
- Large breadboard circuit
- Colored board cable connector
- 5x 560 ohm resistors (for LEDS)
- 3x LED's (red, green, blue)
 - ♦ <https://learn.adafruit.com/raspberry-pi-e-mail-notifier-using-leds>
 - ♦ <https://learn.adafruit.com/debugging-with-the-raspberry-pi-webide/debug-a-blinking-led>
- 5x 10K resistors (for buttons)
- 3x pushbutton
 - ♦ <https://learn.adafruit.com/playing-sounds-and-using-buttons-with-raspberry-pi/bread-board-se>
- 1x photocell
 - ♦ <https://learn.adafruit.com/basic-resistor-sensor-reading-on-raspberry-pi/basic-photocell-readin>
- 1x 1uf capacitor (for photocell)

GPU

- FFT
 - ♦ <http://www.raspberrypi.org/accelerating-fourier-transforms-using-the-gpu/>
 - ♦ https://github.com/raspberrypi/userland/tree/master/host_applications/linux/apps/hello_pi/hell
- SHA-256
 - ♦ <http://rpiplayground.wordpress.com/>
 - ♦ <https://github.com/elorimer/rpi-playground/tree/master/QPU>
- Machine learning - the Deep Belief image recognition SDK
 - ♦ <http://petewarden.com/2014/06/09/deep-learning-on-the-raspberry-pi/>
 - ♦ <http://scientistnabee.wordpress.com/2014/06/20/machine-learning-with-raspberry-pi/>

Arduino

- Connect using I2C
 - ♦ <http://www.pihomeserver.fr/en/2013/08/13/raspberry-pi-home-server-arduino-lier-les-deux-via-bus-i2>

JTAG

- <https://github.com/synthetos/PiOCD/wiki/Using-a-Raspberry-Pi-as-a-JTAG-Dongle>

Arduino ProMini 3.3V 8MHz

Disable serial port console

edit `/boot/cmdline.txt` and remove all `ttyAMA0` leaving:

```
pi@raspberrypi ~ $ cat /boot/cmdline.txt
dwc_otg.lpm_enable=0 console=tty1 root=/dev/mmcblk0p2 rootfstype=ext4 elevator=deadline rootwait
```

also, comment out `ttyAMA0` in `/etc/inittab`

connect Arduino serial

RX->TX

TX->RX

https://github.com/dpavlin/avrdude-rpi/tree/fix_channel_in_use

pin definitions in `GPIO.RPi` are physical pins on board **not** BCM or wiring ip!

P1: The Main GPIO connector							
WiringPi Pin	BCM GPIO	Name	Header		Name	BCM GPIO	WiringPi Pin
		3.3v	1	2	5v		
8	Rv1:0 - Rv2:2	SDA	3	4	5v		
9	Rv1:1 - Rv2:3	SCL	5	6	0v		
7	4	GPIO7	7	8	TxD	14	15
		0v	9	10	RxD	15	16
0	17	GPIO0	11	12	GPIO1	18	1
2	Rv1:21 - Rv2:27	GPIO2	13	14	0v		
3	22	GPIO3	15	16	GPIO4	23	4
		3.3v	17	18	GPIO5	24	5
12	10	MOSI	19	20	0v		
13	9	MISO	21	22	GPIO6	25	6
14	11	SCLK	23	24	CE0	8	10
		0v	25	26	CE1	7	11
WiringPi Pin	BCM GPIO	Name	Header		Name	BCM GPIO	WiringPi Pin

flasing

```
dpavlin@blue:~/Arduino/RF433_Sockets$ scp /tmp/build*.tmp/*.hex pi@rpi.tv:/tmp/
RF433_Sockets.cpp.hex                                100% 19KB 18.6KB/s 00:00
```

```
pi@raspberrypi ~ $ cat avrdude.sh
#!/bin/sh
hex=`ls -t /tmp/*.hex | head -1`
echo "flash hex: $hex"
```

```
sudo /home/pi/avrdude-rpi/avrdude-autoreset -C /home/pi/avrdude.conf -v -v -patmega328p -carduino
```

```
pi@raspberrypi ~ $ ./avrdude.sh
flash hex: /tmp/RF433_Sockets.cpp.hex
```

```
avrdude: Version 6.1, compiled on Sep 17 2014 at 05:41:35
Copyright (c) 2000-2005 Brian Dean, http://www.bdmicro.com/
Copyright (c) 2007-2014 Joerg Wunsch
```

```
System wide configuration file is "/home/pi/avrdude.conf"
User configuration file is "/root/.avrduderc"
User configuration file does not exist or is not a regular file, skipping
```

```
Using Port                : /dev/ttyAMA0
Using Programmer          : arduino
Overriding Baud Rate      : 57600
```

```
done with autoreset
```

```
strace: | /home/pi/avrdude-rpi/autoreset: Broken pipe
strace: | /home/pi/avrdude-rpi/autoreset: Broken pipe
```

```
AVR Part                : ATmega328P
Chip Erase delay         : 9000 us
PAGEL                    : PD7
BS2                      : PC2
RESET disposition        : dedicated
RETRY pulse              : SCK
serial program mode      : yes
parallel program mode    : yes
Timeout                  : 200
StabDelay                : 100
CmdexeDelay              : 25
SyncLoops                : 32
ByteDelay                : 0
PollIndex                : 3
PollValue                : 0x53
Memory Detail            :
```

	Memory	Type	Mode	Delay	Block Size	Poll Indx	Paged	Size	Page Size	#Pages	MinW	MaxW	Polled ReadBack
	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
	eeeprom		65	20	4	0	no	1024	4	0	3600	3600	0xff 0xff
	flash		65	6	128	0	yes	32768	128	256	4500	4500	0xff 0xff
	lfuse		0	0	0	0	no	1	0	0	4500	4500	0x00 0x00
	hfuse		0	0	0	0	no	1	0	0	4500	4500	0x00 0x00
	efuse		0	0	0	0	no	1	0	0	4500	4500	0x00 0x00
	lock		0	0	0	0	no	1	0	0	4500	4500	0x00 0x00
	calibration		0	0	0	0	no	1	0	0	0	0	0x00 0x00
	signature		0	0	0	0	no	3	0	0	0	0	0x00 0x00

```
Programmer Type : Arduino
Description      : Arduino
Hardware Version: 2
Firmware Version: 1.16
Vtarget         : 0.0 V
Varef           : 0.0 V
Oscillator      : Off
SCK period      : 0.1 us
```

```
avrdude: AVR device initialized and ready to accept instructions
```

```
Reading | ##### | 100% 0.01s
```

```
avrdude: Device signature = 0x1e950f
avrdude: safemode: lfuse reads as 0
avrdude: safemode: hfuse reads as 0
avrdude: safemode: efuse reads as 0
avrdude: reading input file "/tmp/RF433_Sockets.cpp.hex"
```

```

avrdude: writing flash (6770 bytes):

Writing | ##### | 100% 1.98s

avrdude: 6770 bytes of flash written
avrdude: verifying flash memory against /tmp/RF433_Sockets.cpp.hex:
avrdude: load data flash data from input file /tmp/RF433_Sockets.cpp.hex:
avrdude: input file /tmp/RF433_Sockets.cpp.hex contains 6770 bytes
avrdude: reading on-chip flash data:

Reading | ##### | 100% 1.49s

avrdude: verifying ...
avrdude: 6770 bytes of flash verified

avrdude: safemode: lfuse reads as 0
avrdude: safemode: hfuse reads as 0
avrdude: safemode: efuse reads as 0
avrdude: safemode: Fuses OK (E:00, H:00, L:00)
strace: |/home/pi/avrdude-rpi/autoreset: Broken pipe
strace: |/home/pi/avrdude-rpi/autoreset: Broken pipe
strace: |/home/pi/avrdude-rpi/autoreset: Broken pipe
strace: |/home/pi/avrdude-rpi/autoreset: Broken pipe
strace: |/home/pi/avrdude-rpi/autoreset: Broken pipe

avrdude done. Thank you.

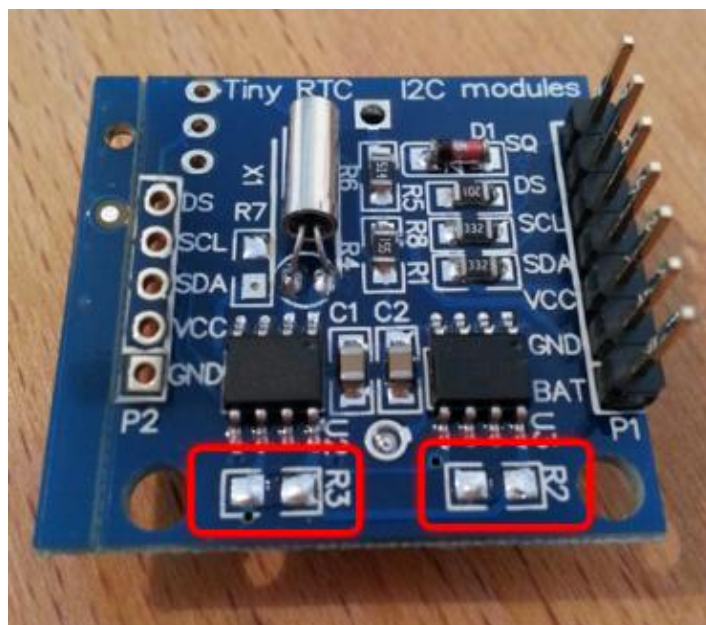
strace: |/home/pi/avrdude-rpi/autoreset: Broken pipe

```

DS1307 Tiny RTC

- <http://www.instructables.com/id/Set-up-Real-Time-Clock-RTC-on-Raspberry-Pi/>
- <http://electronics.stackexchange.com/questions/98361/how-to-modify-ds1307-rtc-to-use-3-3v-for-ras>

Check if module has 5V pull-up I2C resistors! If it does (like Tiny RTC has R2 and R3) you will have to remove them.



```

root@raspberrypi:/home/pi# apt-get install i2c-tools

root@raspberrypi:/home/pi# modprobe i2c-dev

root@raspberrypi:/home/pi# i2cdetect -y 1
    0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50: 50  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  68  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --

root@raspberrypi:/home/pi# modprobe rtc-ds1307

root@raspberrypi:/home/pi# echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-1/new_device

root@raspberrypi:/home/pi# hwclock --systohc -D --noadjfile --utc
hwclock from util-linux 2.20.1
Using /dev interface to clock.
Assuming hardware clock is kept in UTC time.
Time elapsed since reference time has been 0.722720 seconds.
Delaying further to reach the new time.
Setting Hardware Clock to 11:53:41 = 1429876421 seconds since 1969
ioctl(RTC_SET_TIME) was successful.

```

DS18B20

```

root@raspberrypi:/home/pi# grep wl /boot/config.txt
dtoverlay=wl-gpio

# reboot

root@raspberrypi:/home/pi# cat /sys/bus/w1/devices/28-0000043c5507/w1_slave
50 05 4b 46 7f ff 0c 10 1c : crc=1c YES
50 05 4b 46 7f ff 0c 10 1c t=85000

```

Hmm, you will also want to remove R1 pullup to make it work?

avrdude

Using Raspberry Pi as Arduino ISP programmer

- <https://github.com/onandoffables/avrdude-linuxgpio>

```

root@raspberrypi:/home/pi# avrdude -c linuxgpio -p atmega328p

avrdude: AVR device initialized and ready to accept instructions

Reading | ##### | 100% 0.00s

avrdude: Device signature = 0x1e950f

avrdude: safemode: Fuses OK (E:05, H:DA, L:FF)

```


avrdude done. Thank you.

logic analyzer

- bare metal: <https://github.com/tuxyme/metal-pi>
- Panalyzer - a RaspberryPi based Logic Analyzer <https://github.com/richardghirst/Panalyzer>

rotary encoder

http://www.bobrathbone.com/raspberrypi_rotary.htm

SWD

<http://www.disk91.com/2015/technology/programming/swd-programming-using-a-raspberrypi/>
<https://github.com/disk91/PySWD>

```
pi@raspberrypi ~ $ git clone https://github.com/disk91/PySWD
pi@raspberrypi ~ $ cd PySWD/
```

PIC

- rpp - PIC Programmer using GPIO <http://holdenc.altervista.org/rpp/>
- <http://mujweb.cz/tlc/rpp/>
- <https://www.raspberrypi.org/forums/viewtopic.php?f=44&t=14689&start=75>
- <https://github.com/oh7bf/PiPIC>
- <https://github.com/WallaceIT/picberry>

esp8089

- <https://hackaday.io/project/8678-rpi-wifi-hat>
- <https://github.com/al177/esp8089>

PS/2 GPIO

- <http://www.deater.net/weave/vmwprod/hardware/pi-ps2/>
 - ♦ <https://github.com/deater/vmw-meter/blob/master/pi-ps2/linux-kernel/pi-ps2gpio.c>

GPIO header annotations

- <https://github.com/splitbrain/rpiplusleaf/tree/master>

Signal generator

- <https://github.com/JamesP6000/WsprryPi>

```
pi@rpi2 ~/WsprryPi $ sudo ./wspr --test-tone 1000000  
A test tone will be generated at frequency 1.000000 MHz
```

```
Using local mbox device file with major 249.  
Transmitting test tone on frequency 1.000000 MHz  
Press CTRL-C to exit!
```

- http://github.com/gryrmln/RPi_Wobbulator

LCD panels

<http://blog.reasonablycorrect.com/raw-dpi-raspberry-pi/>

Pi Zero USB OTG

libcomposite

- <http://isticktoit.net/?p=1383>

PATA IDE emulation

- <https://www.retrotronics.org/home-page/netpi-ide/>

DPI LCD screen

- <http://blog.reasonablycorrect.com/raw-dpi-raspberry-pi/>
- <https://www.raspberrypi.org/forums/viewtopic.php?p=1023780#p1023780>

i2c clock stretching

- <https://github.com/raspberrypi/linux/issues/254#issuecomment-217047171>