**User Manual**

* **Install NOOBs**
  + Use SD Card formatter 4
  + Use win32diskmanager to burn iso
  + Use FAT32 for SDHC under 32GB
* **Update RasPi**

|  |
| --- |
| 1. sudo apt-get update 2. sudo apt-get upgrade |

* **Install Teamviewer**

https://download.teamviewer.com/download/linux/teamviewer-host\_armhf.deb

* **Install synergy**

|  |
| --- |
| 1. Install dependencies (Debian 8 Raspbian) 2. sudo apt-get update 3. sudo apt-get install build-essential cmake libavahi-compat-libdnssd-dev libcurl4-openssl-dev libssl-dev python qt4-dev-tools xorg-dev 4. https://pkgs.org/download/synergy 5. Download Latest Debian i386.deb 6. sudo apt-get install synergy 7. start>accessories>synergy 9. Auto-Login user pi 10. sudo raspi-config 11. (Select "Boot Options", then "Desktop Autologin") 13. Create Synergy client autostart file (Must be as user pi, NOT root!) 14. mkdir -p ~/.config/autostart 15. nano ~/.config/autostart/synergy.desktop 17. Add lines to autostart file (configure your ip or host here) 18. [Desktop Entry] 19. Name=Synergy Client 20. Exec=/usr/bin/synergyc <synergy server ip or host name> 21. Type=application |

* **Install apache2 server**

|  |
| --- |
| 1. sudo apt-get install apache2 -y 2. Test by going to your http://localhost 3. sudo systemctl restart apache2 5. Change ownership to access directory 6. sudo chgrp -R pi /var/www 7. sudo chmod 775 /var/www 8. sudo chown -R pi/var/www/\* 10. Change listening ports 11. sudo nano /etc/apache2/ports.conf 12. Restart server, look commands below. 14. stop it: sudo /etc/init.d/apache2 stop 15. start it with: sudo /etc/init.d/apache2 start 16. reload the config with: sudo /etc/init.d/apache2 reload 17. restart (stop followed by start): sudo /etc/init.d/apache2 restart |

* **Install php5 server**

|  |
| --- |
| 1. sudo apt-get install php5 libapache2-mod-php5 -y 2. php -v 3. sudo chgrp pi/var/www 4. sudo chmod 775 /var/www 5. sudo chown -R pi/var/www/\* 6. http:// localhost/index.php |

|  |
| --- |
| 1. Create a PHP Test File 2. <html> 3. <head> 4. <title>PHP Test</title> 5. </head> 6. <body> 7. <?php echo '<p>Hello World</p>'; ?> 8. </body> 9. </html> |

* **Install mySQL server**

|  |
| --- |
| 1. sudo apt-get install mysql-server python-mysqldb 2. sudo apt-get install php5-mysqlnd 3. Add these to the php.ini file: 4. extension=msqli.so 5. extension=msql.so 6. set root password  “gforce” 7. Log into mySQL and setup a **new** user 8. mysql -uroot -hlocalhost -p 9. CREATE DATABASE temp\_database; 10. CREATE USER 'zikmir'@'localhost' IDENTIFIED BY 'gforce'; 11. GRANT ALL PRIVILEGES ON temp\_database.\* TO 'zikmir'@'localhost'; 12. FLUSH PRIVILEGES; 13. CTRL + C to exit MySQL Client 14. To log in from the **new** user created: 15. mysql -uzikmir -hlocalhost temp\_database -p 16. use temp\_database 17. CREATE TABLE time\_temp (Id **INT** PRIMARY KEY AUTO\_INCREMENT, time TIMESTAMP DEFAULT CURRENT\_TIMESTAMP, time DOUBLE); 18. insert into TIME\_TEMP values(0,NOW(),25.0); 19. SELECT \*FROM time\_temp; 20. +----+---------------------+----------+ 21. | Id | time                | temp     | 22. +----+---------------------+----------+ 23. |  1 | 0000-00-00 00:00:00 |       25 | 24. |  2 | 0000-00-00 00:00:00 |       26 | 25. |  3 | 2017-10-20 04:03:42 |   22.687 | 26. |  4 | 2017-10-20 04:04:14 |    22.75 | 27. |  5 | 2017-10-20 04:04:47 |   22.687 | 28. |  6 | 2017-10-20 04:05:28 |   22.687 | 29. |  7 | 2017-10-20 04:09:40 |       26 | 30. |  8 | 2017-10-20 04:11:26 |       26 | 31. +----+---------------------+----------+ 32. To **delete** any row, type 33. DELETE FROM time\_temp WHERE Id=1 |

* **Configure MHPG-Streamer**

|  |
| --- |
| 1. sudo apt-get install libjpeg8-dev imagemagick libv4l-dev 2. sudo ln -s /usr/include/linux/videodev2.h /usr/include/linux/videodev.h 3. wget http://sourceforge.net/code-snapshots/svn/m/mj/mjpg-streamer/code/mjpg-streamer-code-182.zip 4. unzip mjpg-streamer-code-182.zip 5. cd mjpg-streamer-code-182/mjpg-streamer 6. make mjpg\_streamer input\_file.so output\_http.so 7. sudo cp mjpg\_streamer /usr/local/bin 8. sudo cp output\_http.so input\_file.so /usr/local/lib/ 9. sudo cp -R www /usr/local/www |

|  |
| --- |
| 1. Start the Camera: 2. mkdir /tmp/stream 3. raspistill --nopreview -w 1280 -h 720 -q 30 -o /tmp/stream/pic.jpg -tl 100 -t 9999999 -th 0:0:0 & 4. LD\_LIBRARY\_PATH=/usr/local/lib mjpg\_streamer -i "input\_file.so -f /tmp/stream -n pic.jpg" -o "output\_http.so -w /usr/local/www" 5. cd ../../ 6. rm -rf mjpg-streamer-182 8. Apply Patch by downloading input\_uvc\_patch.txt 9. cd mjpg-streamer-code/mjpg-streamer 10. patch -p0 < input\_uvc\_patch.txt 11. make USE\_LIBV4L2=**true** clean all 12. sudo make DESTDIR=/usr/local install 14. Run the script: 15. /usr/local/bin/mjpg\_streamer -i "/usr/local/lib/input\_uvc.so" -o "/usr/local/lib/output\_http.so -w /usr/local/www" 17. Stop the script 18. Ctrl+C |

Make it a service and place it in /etc/init.d:

|  |
| --- |
| 1. #!/bin/sh 2. # /etc/init.d/livestream.sh 3. ### BEGIN INIT INFO 4. # Provides:          livestream.sh 5. # Required-Start:    $network 6. # Required-Stop:     $network 7. # Default-Start:     2 3 4 5 8. # Default-Stop:      0 1 6 9. # Short-Description: mjpg\_streamer for webcam 10. # Description:       Streams /dev/video0 to http://IP/?action=stream 11. ### END INIT INFO 12. f\_message(){ 13. echo "[+] $1" 14. } 16. # Carry out specific functions when asked to by the system 17. **case** "$1" in 18. start) 19. f\_message "Starting mjpg\_streamer" 20. /var/www/html/MJPG/mjpg-streamer-code-182/mjpg-streamer/mjpg\_streamer -b -i "/var/www/html/MJPG/mjpg-streamer-code-182/mjpg-streamer/input\_uvc.so -d /dev/video0 -f 15 -r 320x240" -o "/var/www/html/MJPG/mjpg-streamer-code-182/mjpg-streamer/output\_http.so -p 8080 -w /var/www/html -n -c zikmir:1990" 21. sleep 2 22. f\_message "mjpg\_streamer started" 23. ;; 24. stop) 25. f\_message "Stopping mjpg\_streamer…" 26. killall mjpg\_streamer 27. f\_message "mjpg\_streamer stopped" 28. ;; 29. restart) 30. f\_message "Restarting daemon: mjpg\_streamer" 31. killall mjpg\_streamer 32. /usr/local/bin/mjpg\_streamer -b -i "/usr/local/lib/input\_uvc.so" -o "/usr/local/lib/output\_http.so -w /usr/local/www -c zikmir:1990" 33. sleep 2 34. f\_message "Restarted daemon: mjpg\_streamer" 35. ;; 36. status) 37. pid=`ps -A | grep mjpg\_streamer | grep -v "grep" | grep -v mjpg\_streamer. | awk ‘{print $1}’ | head -n 1` 38. **if** [ -n "$pid" ]; 39. then 40. f\_message "mjpg\_streamer is running with pid ${pid}" 41. f\_message "mjpg\_streamer was started with the following command line" 42. cat /proc/${pid}/cmdline ; echo "" 43. **else** 44. f\_message "Could not find mjpg\_streamer running" 45. fi 46. ;; 47. \*) 48. f\_message "Usage: $0 {start|stop|status|restart}" 49. exit 1 50. ;; 51. esac 52. exit 0 |

|  |
| --- |
| 1. sudo chmod +x /etc/init.d/livestream.sh 2. sudo chmod +x /usr/sbin/livestream.sh 3. sudo chmod 755 /etc/init.d/livestream.sh 4. sudo chmod 755 /usr/sbin/livestream.sh 6. sudo update-rc.d livestream.sh defaults |

For streaming online, create an html file and place it in usr/local/www or the instillation folder of MJPG

|  |
| --- |
| 1. <html> 2. <head><title> Live-stream</title></head> 3. <body> 4. <center><h1>Meow</h1></center> 6. <center> 7. <img src="/?action=stream" /> 8. </center> 9. <body> 10. </html> |

Run this html file in the browser: http:localhost:8080/cam.html

* **Configure GPIO**

|  |
| --- |
| 1. sudo apt-get install rpi.gpio 2. sudo raspi-config 3. enable 1-wire 4. enable ssh |

any usb is mounted in /media/pi/root