

Meadow's settings

Awb_r = 0.16

Awb_b = 0.20

UCG (2020); for use with Aspect LED W-18 Ultra-Bright 6000k LED Lights

Camera_Nodes: (2102@ gin-stand 1; 2104@ gin-stand 3)

Awb_r = 0.175

Awb_b = 0.15

Exposure=2000

ROI (Max Y)=400

BLOB Size=500

Gain=50

To change edit mysql database variables in camera_nodes {awb_r, awb_b}

mysql> use camera_nodes;

New Pi OS (May 2020): login: pi pswd: cotton2020 mariadb root pswd: cotton

To do; setup 2nd server and add another autossh-tunnel2.service that connects to the 2nd server as a backup

Starting from new image for RPi VIPR system;

- 1) Update firmware to get better temperature control \$ sudo rpi-update
- 2) Change static IP to DHCP edit /etc/systemd/system/autossh-tunnel.service
- 3) Need to setup display so that it doesn't collapse in VNC
 - a) **Fix VNC screen resolution:** on RPi-4, need to edit /boot/config.txt and comment out line with `dtoverlay=vc4-fkms-v3d`
- 4) Expand file system (in cmd window \$ sudo raspi-config then select advanced options then choose Expand rootfs)
- 5) Copy LUT.png and camera_nodes.sql to home directory,
- 6) Install database from Google drive In cmd window (leave off -u as no pswd)
 - a) **\$ mysql -u root -p camera_nodes < /home/pi/camera_nodes.sql**
To edit database
\$ sudo mysql -u root -p then enter pswd: cotton
> use camera_nodes;
> show tables;
> update Camera_Configuration set Awb_r=0.175;
> update Camera_Configuration set Awb_b=0.15;
> select Awb_r from Camera_Configuration;

- 7) In google drive sub-folder (final_Meadow_Node_1 under plastic_beta_src/deploy_XXX/; get last exe's for viewer and PlasticCont_Beta.exe
 - a) Copy to home directory, then right click and make executable; should now be able to run them
- 8) May need to create Pictures/Plastic directory
- 9) Run viewer and set color-temp to 6500k and other settings (like turn off picture capture
- 10) Run classifier builder (will need to capture some files either by turning on picture capture or enabling picture capture (so that it save plastic pictures) keep an eye on the command window for PlasticCont_Beta.exe to watch for notes on training/errors ect.
- 11) Check disk-space `$df -h` (Note: PlasticCont_Beta won't save files if disk is full, so if it's not saving plastic images it might be because sd-card is full; so check)
- 12) May need to turn on autossh-tunnel.service (see ReverseTunnel notes)
- 13) Add xrdp (can be run in parallel with VNC at same time)
 - a) `$ sudo apt install xrdp`
 - b) `$ systemctl show -p SubState --value xrdp` //ck that xrdp is running
 - c) `$ sudo adduser xrdp ssl-cert` //add xrdp to the ssl-cert user group
 - d) Should now be able to connect to it via RDP (Microsoft's; either on Mac or PC)
 - i) Open cmd window (bash window on PC) and connect to cloud server through reverse ssh tunnel (port 2101) that was opened by RPi:
 - (1) `ssh -L 3389:localhost:3389 -p 2101 pi@45.56.121.38`
 - (2) Then connect RDP to: localhost:3389 should now see a full screen of RPi desktop; userName: pi password: cotton2020
- 14) Ck temperature throttling, ensure not running too hot

```
$ sudo shutdown -h now
$ watch -n 1 vcgencmd measure_clock arm
$ watch -n 1 vcgencmd measure_temp
$ watch -n 1 vcgencmd get_throttled
```

```
$ifconfig //to check which IP addr it's on
$ nmap 192.168.50.* //to check which nodes have ssh port 22 open
$ nmap -sn 192.168.50.8 //to check out what's on network (should also contain MAC addresses...)
```

```
//create autostart desktop file (be sure to change permissions to allow execution
~/config/autostart/PlasticCont_Beta.desktop
```

```
[Desktop Entry]
Encoding=UTF-8
Version=1.0
```

Type=Application
Exec=/home/pi/Qt_Dev/build-PlasticCont_Beta-Desktop-Debug/PlasticCont_Beta
Icon=
Terminal=true
Name=Plastic Contamination Beta
Categories=Application;Programming
MimeType=application/VIPR-Plastic-Detection

To enable air-knives: (in \$ mysql -u root camera_nodes)
UPDATE `camera_nodes`.`Camera_Configuration` SET `Use_Air_Knives` = '1' WHERE (`Id` = '7');

Troubleshooting notes:

- Ran into issue where camera was periodically retrieving a bad frame that had noise lines that were sometimes pink (bad connection?). We swapped out camera, clue was in detected camera images [resolved by changing algorithm for saving from separate thread to trickle saving via FIFO queue]

To test solenoid from command line:
\$ gpio mode 26 out #set wiringPi pin 26 as output
\$ gpio blink 26

To autostart; create a folder, "autostart", under /home/pi/.config/
So path is /home/pi/.config/autostart
Put desktop file in that folder pointing to exe you want to run after boot-up
Permissions on desktop file is normal (doesn't have to be exe permissions), user pi:pi

Current Settings on R-Pi3 nodes at Meadow

Gain,5,
Exposure,0.5,
awb_r,0.16,
Awb_b,0.2,

Setting on Pi-4

To change in mySQL
UPDATE `camera_nodes`.`Camera_Configuration` SET `awb_r` = '0.16', `awb_b` = '0.2'
WHERE (`Id` = '7');

To get screen size;

```
$ xdpinfo | awk '/dimensions:/ { print $2; exit }'
```

<https://www.raspberrypi.org/documentation/configuration/config-txt/video.md>

```
hdmi_force_hotplug=1
```

```
hdmi_force_mode=1
```

```
hdmi_drive=2
```

```
hdmi_group=1
```

```
hdmi_mode=16
```

forum notes on config

Gidday. Just wanted to confirm and thank Beta_tester for the code he submitted which worked on my RPi 3.

I was using Teamviewer Server and knowing that it was based upon VNC I thought I would try the code suggested and it worked.

```
hdmi_force_hotplug=1
```

```
hdmi_force_mode=1 THIS ONE I COULDN'T FIND AND  
DIDN'T USE.
```

```
hdmi_drive=2
```

```
hdmi_group=1
```

```
hdmi_mode=16
```

It had cost me lots of hours searching internet and stuffing around with a DVI monitor switching back and forth between MacOSx & the Pi.

Can confirm #hdmi_safe=1 did NOT help and
hdmi_force_hotplug=1 by itself did NOT cure the problem.

Thanks once again.

JimJtron

OMR

Posts: 2

Joined: Tue Jul 16, 2019 9:46 pm

Re: Set VNC resolution?

- [Quote](#)

Tue Jul 16, 2019 9:49 pm

This is still true for the Pi 4, just comment out the line in the
[pi4] section:

Code: **Select all**

```
[pi4]
```

```
# Enable DRM VC4 V3D driver on top of the dispmanx display stack
```

```
#dtoverlay=vc4-fkms-v3d
```

```
max_framebuffers=2
```

To measure temperature under stress on RPi:

- 1) `Install stress`

```
$ [sudo] apt install stress
```

2) Run stress test and get temps

```
$ while true; do vcgencmd measure_clock arm; vcgencmd measure_temp; sleep 10; done & stress -c 4 -t 100s
```

To execute mySQL commands in shell script see (so should be able to write simple shell script to log cpu temp into dBase (perhaps another table so we have a running history along with Ram usage and clock frequency and time-stamp)?

<https://hoststud.com/resources/how-to-execute-mysql-queries-from-command-line-bash-shell.136/>

Other options (don't seem to work...)

<https://github.com/nschloe/stressberry>

```
$ [sudo] apt install stress
```

```
$ pip3 install stressberry --user
```

To run:

```
$ stressberry-run out.dat
```

```
$ stressberry-plot out.dat -o out.png
```

Python script to read cpu temp and store into mySQL table

SQL to create table we'll use:

```
CREATE DATABASE temp_database;
```

```
USE temp_database;
```

```
CREATE TABLE tempLog (
```

```
id MEDIUMINT NOT NULL AUTO_INCREMENT,
```

```
datetime DATETIME NOT NULL,
```

```
temperature FLOAT(5,2) NOT NULL,
```

```
freeRam MEDIUMINT NOT NULL,
```

```
PRIMARY KEY (id)
```

```
);
```

Examine: `mysql> describe tempLog;`

After running python to insert vals; to look at entries

```
mysql> select * from tempLog;
```

To get last entry

```
mysql> select * from tempLog order by id desc limit 1;
```

Includes how to setup mySQL table to take cpu temp data input; full example

<https://wingoodharry.wordpress.com/2015/01/05/raspberry-pi-temperature-sensor-web-server-part-2-setting-up-and-writing-to-a-mysql-database/>

```
import os
import time
import datetime
import glob
import MySQLdb
import re, commands
```

```
from time import strftime
```

```
# Variables for MySQL
```

```
db = MySQLdb.connect(host="localhost", user="ginuser",passwd="Hello2018",
db="temp_database")
cur = db.cursor()
```

```

def check_CPU_temp():
    temp = None
    err, msg = commands.getstatusoutput('vcgencmd measure_temp')
    if not err:
        m = re.search(r'-?\d\.\d*', msg) # https://stackoverflow.com/a/49563120/3904031
        try:
            temp = float(m.group())
        except:
            pass
    return temp, msg

while True:
    #temp = tempRead()
    temp, msg = check_CPU_temp()
    print temp
    datetimeWrite = (time.strftime("%Y-%m-%d ") + time.strftime("%H:%M:%S"))
    print datetimeWrite
    sql = ("INSERT INTO tempLog (datetime,temperature) VALUES
    (%s,%s)",(datetimeWrite,temp))
    try:
        print "Writing to database..."
        # Execute the SQL command
        cur.execute(*sql)
        # Commit your changes in the database
        db.commit()
        print "Write Complete"

    except:
        # Rollback in case there is any error
        db.rollback()
        print "Failed writing to database"

    #sleep for 10 secs
    time.sleep(10)
    #cur.close()
    #db.close()
    #break

# *****

```

To get 1st 3 columns of \$ free -m command

```
$ free -m | awk '{print $1, $2, $3; }'
```

Returns:

```
total used free
Mem: 3854 350
Swap: 99 0
```

To get just free memory column

```
$ free -m | awk '{print $3; }'
```

Returns

```
Used
3854
0
```

TO GET Just Free memory from Ram (1st row only)

```
$ /usr/bin/free -m | awk 'NR==2 {print $3;}'
```

Returns

```
350
```

#python to get freeRam

```
def check_CPU_freeRam():
```

```
    ram = None
```

```
    err, msg2 = commands.getstatusoutput("free -m | awk 'NR==2 {print $3;}'")
```

```
    if not err:
```

```
        m = re.search(r'-?\d\.\d*', msg2) # https://stackoverflow.com/a/49563120/3904031
```

```
        try:
```

```
            ram = float(m.group())
```

```
        except:
```

```
            pass
```

```
    return ram, msg2
```

```
ram, msg2 = check_CPU_freeRam()
```

```
print ram
```

Appendix: Database command variables, used to control software.

```
MariaDB Table [camera_nodes]> desc Camera_Configuration;
```

```
+-----+-----+-----+-----+-----+
| Field          | Type    | Null | Key | Default | Extra |
```

0	Id	int(11)	NO	PRI	NULL		
1	Location	text	YES		NULL		
2	Function	text	YES		NULL		
3	IP_Address	text	YES		NULL		
4	SSH_Port	int(11)	YES		NULL		
5	Status	text	YES		NULL		
6	ReBoot	int(11)	YES		0		
7	CPU_Temp	int(11)	YES		0		
8	Comm_Messages	text	YES		NULL		
9	Error_Message_Log	text	YES		NULL		
10	Open_ReverseTunnel	int(11)	YES		NULL		
11	Reset_Variables	int(11)	YES		0		
12	LastPic_AveWhite	int(11)	YES		NULL		
13	Run_Config	int(11)	YES		NULL		
14	Run_Classifier	int(11)	YES		NULL		
15	AddTo_Classifier	int(11)	YES		NULL		
16	Use_Air_Knives	int(11)	YES		NULL		
17	AirKnife_PulseTest	int(11)	YES		NULL		
18	AirKnife_PulseDelay_ms	int(11)	YES		NULL		
19	AirKnife_PulseOnTime_ms	int(11)	YES		NULL		
20	Cool_Group_ID	int(11)	YES		NULL		
21	Min_BlobSize	int(11)	YES		50		
22	Roi_StartX	int(11)	YES		NULL		
23	Roi_StopX	int(11)	YES		NULL		
24	Roi_StartY	int(11)	YES		NULL		
25	Roi_StopY	int(11)	YES		NULL		
26	Classifier_Image	blob	YES		NULL		
27	LastTime_ContFound	datetime	YES		NULL		
28	LastTime_NormalPic_Taken	datetime	YES		NULL		
29	Total_Plastic_Count	int(11)	YES		NULL		
30	Capture_Image_DataSet	int(11)	YES		NULL		
31	Gain	varchar(45)	YES		NULL		
32	Exposure	varchar(45)	YES		NULL		
33	ColorBalance	int(11)	YES		1		
34	awb_r	varchar(45)	YES		NULL		
35	awb_b	varchar(45)	YES		NULL		
36	TimeToSavePic_mins	varchar(45)	YES		NULL		
37	Number_of_Pics_to_Take	int(11)	YES		NULL		
38	Number_of_Pics_to_Skip	int(11)	YES		0		
39	Enable_Input_PicCaptures	int(11)	YES		0		
40	SoftwareStart_PicCaptures	int(11)	YES		0		
41	Enable_Diagnostic_Display	int(11)					
42	Save_Directory	varchar(45)			"/home/pi/Pictures/Capture1"		

43	Run_Analysis	int(11)	0
44	Percent_Pics_with_Objects	int(11) -1	
45	HighSpeed_CaptureMode	int(11)	
46	MaxImages_PlasticSaveQueue		
47	MaxImages_SavedPerDay		
48	NumPlasticImages_SavedToday		
49	SavePlasticImages		
50	PlasticSaveDirectory		
51	DownScaleX_PlasticImages		
52	ReduceSize_PlasticImages		
53	Progress_PicCapture		
54	Progress_BuildClassifier		
55	Current_Velocity		
+-----+-----+-----+-----+-----+			