

Setup & Run – Sdcam-Raspberry

1 Precondition

This documentation shows how to setup the ready to go Sdcam-Raspberry as a sundial webcam supporting the project EarthLAT1200.org.

The aim is to setup the headless standalone system using SSH-access via WLAN (e.g. via a smartphone hotspot) without a need for attaching a HDMI-monitor and USB-keyboard/mouse. The setup reconnects to the WLAN used by the partner station and if necessary modifies the configuration files operating for the project. Later on modifications to the configuration files (e.g. change of the cropping window position, text to the website, etc.) can be done the same way via a headless SSH-access.

After a proper setup (see Fig 1), on power up the Sdcam-Raspberry automatically enters the command line mode, starts the sundial application, connects to the internet, takes images periodically, transfers them to the EarthLAT1200-server, and enters a defined HALT-status after sunset. During the night, the system is powered off and powered up again before sunrise by an electro mechanic clock timer.

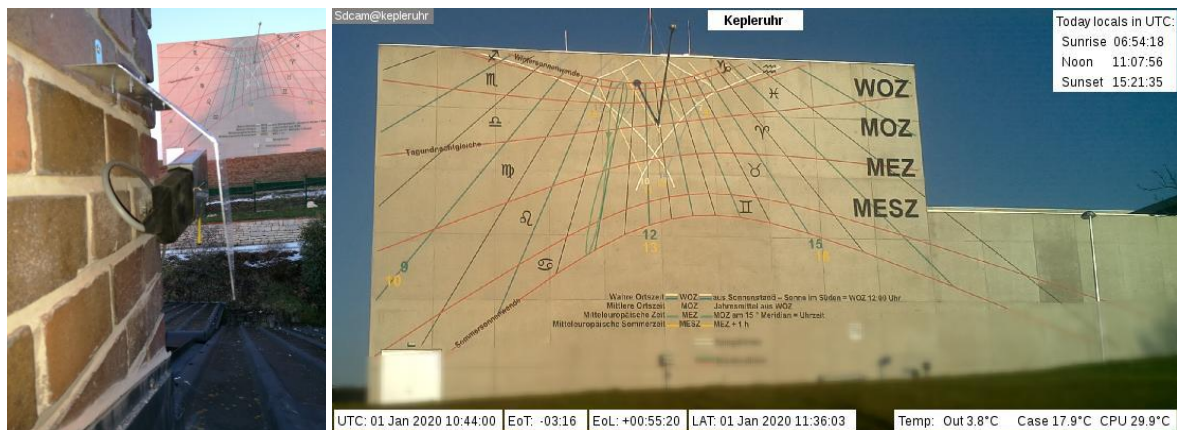


Fig 1 Webcam at Kepleruhr, AT, delivering periodic images of the sundial

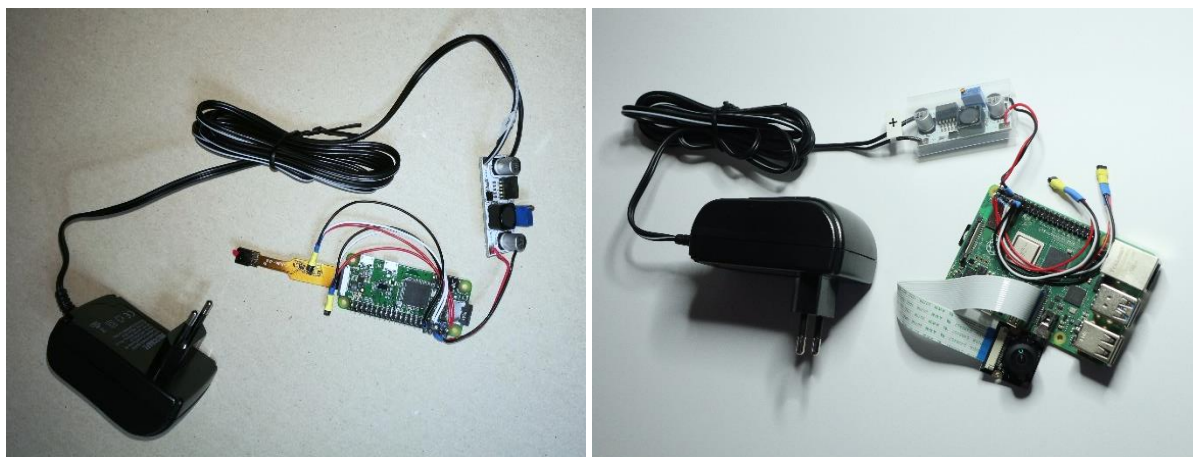


Fig 2 Sdcam delivery with a) Raspberry Zero or b) Raspberry Pi 4 with wide angle lens

2 Delivery Status

The ready to go Sdcam-Raspberry is fully connected/wired – see Fig 2. It operates headless, which means no monitor, no keyboard, and no mouse are needed. On power up by connecting to the power source the system enters the command line modus, tries to connect the internet via WLAN, starts the sundial application, takes and transfers periodically images. The delivery status awaits the given temporary hotspot-WLAN. Using this the configuration for the WLAN on site can be done headless too.

2.1 HW-components delivered

- ☐ Raspberry Zero including WLAN or
- ☐ Raspberry Pi 4 including WLAN
- ☐ PI camera – resolution 5MP, 8MP, ____MP – field of view H 53°, 62°, 122°, ____°
- ☐ SD card 16GB including operating system, configuration, running SW Sdcam
- ☐ Two one-wire-temperature sensors attached to the PIO – one for inside, one for outside the case
- ☐ 5VDC power supply attached to the GPIO
- ☐ Step down converter input 6-30VDC output 5.1VDC (controllable)
- ☐ Power supply input 100-240VAC output 12VDC/1500mA.

2.2 Components which have to be added by the partner station

- Camera case – a plastic case; open hole Ø3-5mm for the camera view
- Take care to shield electrical contacts of the components within the case (see Fig 3 a)) – not just to loosen them inside
- A weather shield against direct rain/sun/snow to the camera case (see Fig 3 b))
- If the cable length between the power plug and the webcam is longer than 1.5m the 12VDC line between the power plug and the step down converter can be extended: up to 10m using 0.25mm² wires or up to 30m using 0.5mm² wires
take care not to mix up the polarity of the power supply, check by a multimeter:
PLUS: Black&White, marked '+', In+, Out+, Red; **MINUS:** Black, In-, Out-, Black
- WLAN connecting the Sdcam-Raspberry to the internet
- Electro mechanic clock timer switching power off during night (earliest 30min after sunset until sunrise) (see Fig 3 c)).

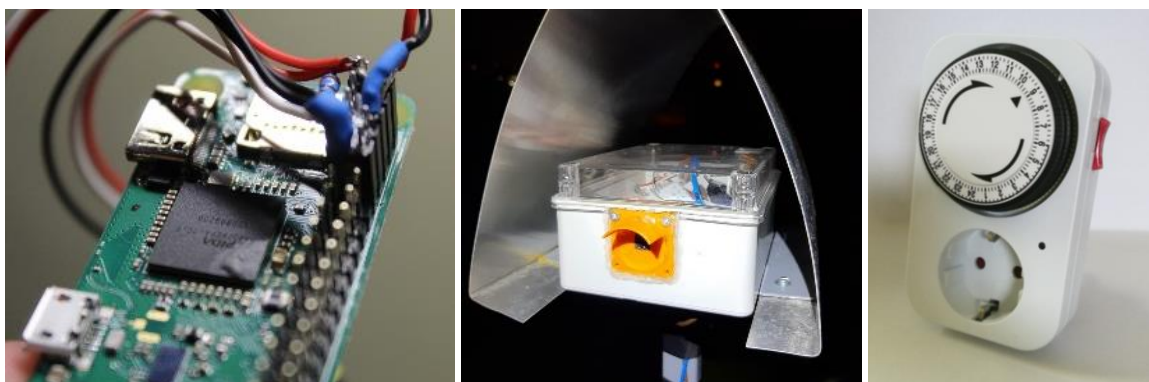


Fig 3 a) damage caused by contacting b) e.g. case, weather shield c) e.g. clock timer

2.3 SW-configuration at the Sdcam-Raspberry

- Operating system Raspberian capable for command-line-modus (terminal) or desktop-modus, SSH-enabled, VNC-enabled
- WLAN connecting to defined SSID
- All components/files for operating the Sdcam
- Python-files (SW-code):
 - `sundialcam.py`
 - `scdfun.py`
 - `scdfun2.py`
- Configuration-files:
 - `station.cfg` – parameters for the FTP-connection, window positioning
 - `stationinfo.txt` – additional public text info at EarthLAT1200.org
- Configuration-file stored at EarthLAT1200-server:
 - `remotecmd.cfg` – several working parameters for remote control by the server administrator.

2.4 SW needed at your notebook to configure the Sdcam-Raspberry

- PuTTY – a ssh-connection via a terminal window to login and operate the Sdcam-Raspberry; free downloads from the Internet available, e.g. <https://www.putty.org>
- WinSCP – optional; allows for an easy file transfer from the Sdcam-Raspberry to your notebook; free downloads from the Internet available, e.g. <https://winscp.net>.

3 Setup for the appropriate WLAN

Take care: The Sdcam-Raspberry will enter a HALT modus 15min after the local sunset and a dark scene in front of the camera. The last image sent is an image indicating “No live image – cloudy and/or night yet”. It will re-enter normal operation (periodically uploading a live image) after a power up.

3.1 Connect Sdcam-Raspberry via hotspot to a PC/notebook

1. Establish a WLAN with SSID “HotspotRaspi” / password: “earth4LAT” – e.g. via a smartphone or your notebook nearby the Sdcam-Raspberry
2. Power up the Sdcam-Raspberry (will take 1-2min until it started and connected to the WLAN)
3. Connect the PC/notebook to the same WLAN
4. Start a SSH-connection terminal to the Sdcam-Raspberry (e.g. via PuTTY or WinSCP) using it’s hostname (e.g. “sdcamearth002” or similar)
5. Login to the Sdcam-Raspberry as user “pi” and password “sdc4EL”.

3.2 Modify the WLAN configuration via SSH-terminal at Sdcam-Raspberry

6. Enter the command within the terminal window `sudo raspi-config`
7. Go to “2 Network Options” - arrow down ones, than enter-key
8. Go to “N2 Wi-fi” - enter-key, than enter the SSID for your partner station’s WLAN, than enter-key
9. Enter the passphrase/password for your partner station’s WLAN, than enter-key
10. Go to <Finish> (arrow right), than enter-key.

While the connection has been changed from the hotspot to the partner station’s WLAN the PC/notebook lost the connection to the Sdcam-Raspberry via the hotspot.

11. Reconnect the PC/notebook to the same partner station’s WLAN using it’s SSID and password

12. Start again a SSH-connection terminal to the Sdcam-Raspberry (e.g. via PuTTY or WinSCP) using it's hostname (e.g. "sdcamearth002" or similar)
13. Login to the Sdcam-Raspberry as user "pi" and password "sdc4EL"
14. Enter the command '`sudo reboot now`'.

4 Check operation with reboot

- See what happens at least at EarthLAT1200.org using the registered access (very left menu).
- If it does not work properly, check the crontab log file and/or the Sdcam log file (e.g. "sdc2019Nov.log") at the Raspberry.

5 Follow SW reversions via GitHub

There is a GitHub repository with the revisions of the Sdcam-SW. You may follow this hub and/or feel free to do your own modifications:

<https://github.com/kepleruhr/Sdcam>

The actual version can be seen at the Sdcam log file.

6 Modify parameter files according the partner stations needs

There are two parameter files containing working data for the partner station:

<code>station.cfg</code>	FTP connection set, total and detail window size and position and working parameters
<code>stationinfo.txt</code>	Text snippets shown at the website including partner stations Long/Lat position for calculating local apparent time LAT.

Both are located in the same folder as the python script files. On startup of the python script `sundialcam.py` (which is done after a reboot) the content of these files is read and used for operation. They can be modified locally or via SSH with the command '`sudo nano 'filename'`'. The nano-editor stores the modified file by `ctrl-o` and is terminated by `ctrl-x`.

A third parameter file is stored at the EarthLAT1200-server and retrieved by Sdcam. The server administrator may change some parameters remotely for the partner station:

<code>remotecmd.cfg</code>	Several working parameters and zoom window positioning.
----------------------------	---

The following tables 5.1-3 and drawing 5.4 shows the meaning of all parameters. Within the parameter files, they can appear unsorted within a section, but this does not border their proper operation.

6.1 Detail content of `station.cfg`

<i>variable</i>	<i>e.g. value</i>	<i>remark</i>	<i>min</i>	<i>max</i>
[Header]				
idno =	AT001	future use (partner station id for server)		
idname =	sdcamearth001	future use (Raspi name for ssh connection)		
[FTP]				
host =	xxx	login data		
port =	22	login data		
user =	xxx	login data		
password =	xxx	login data		
[Total]				
cropper =	72	crop width percentage of camera image	25	100
croppaspratioper =	50	crop high percentage of cropped width	40	150
cropcentpercx =	0	crop center X +/- percentage at camera image	-30	30
cropcentpercy =	0	crop center Y +/- percentage at camera image	-30	30
cropwebwidth =	800	reduced crop width of total view transferred to host	400	800
zoomperc =	22	zoom width percentage of cropped image	15	50
zoomaspratioper =	75	zoom high percentage of zoomed width	50	200
zoomwebwidth =	400	zoomed width of detail view transferred to host	300	500
[Detail]				
ftpupload =	1	no/yes FTP transfer to host		
remotecmd =	1	no/yes accept remote parameter set of *		
camoffline =	0	* no/yes indicate offline status		
periodm =	2	* period of image capture [min]		
series =	0	* no/yes store live images locally		
stream =	0	future use		
zoomdrawrect =	0	* no/yes draw a rectangle for the detail view		
zoomcentpercx =	0	* zoom center X +/- percentage at cropped image	-40	40
zoomcentpercy =	0	* zoom center Y +/- percentage at cropped image	-40	40
zoommove =	0	future use (move zoom automatically)		
[Dial]				
dials12x =	1276	future use (move zoom automatically)		
dials12y =	937	future use (move zoom automatically)		
diale09x =	957	future use (move zoom automatically)		
diale09y =	752	future use (move zoom automatically)		
diale15x =	1458	future use (move zoom automatically)		
diale15y =	700	future use (move zoom automatically)		
dialw12x =	1268	future use (move zoom automatically)		
dialw12y =	627	future use (move zoom automatically)		

6.2 Detail content of stationinfo.txt

<i>variable</i>	<i>e.g. value</i>	<i>remark</i>
[Header]		
idname =	sdcamearth001	future use (Raspi name for ssh connection)
idno =	AT101	future use (partner station id for server)
[Info]		
name =	Kepleruhr	display at server site inserted to total view
location =	Grieskirchen, Austria	display at server site
latitude =	48.235	display at server site taken for calculations of LAT, sun positions
longitude =	13.834	display at server site taken for calculations of LAT, sun positions
typewebcam =	RaspberryPI + PiCam	display at server site
typetransfer =	RaspberryPI + PiCam, Python	display at server site
text =	This sundial has been erected in 2014, takes 240 sqm, and is starting point of a series of public astronomical events.	display at server site
website =	https://kepleruhr.eu	display at server site
team =	Kurt Niel	display at server site
nearbypublicinst =	schulzentrum grieskirchen	yet not displayed
organization =	non profit private	yet not displayed

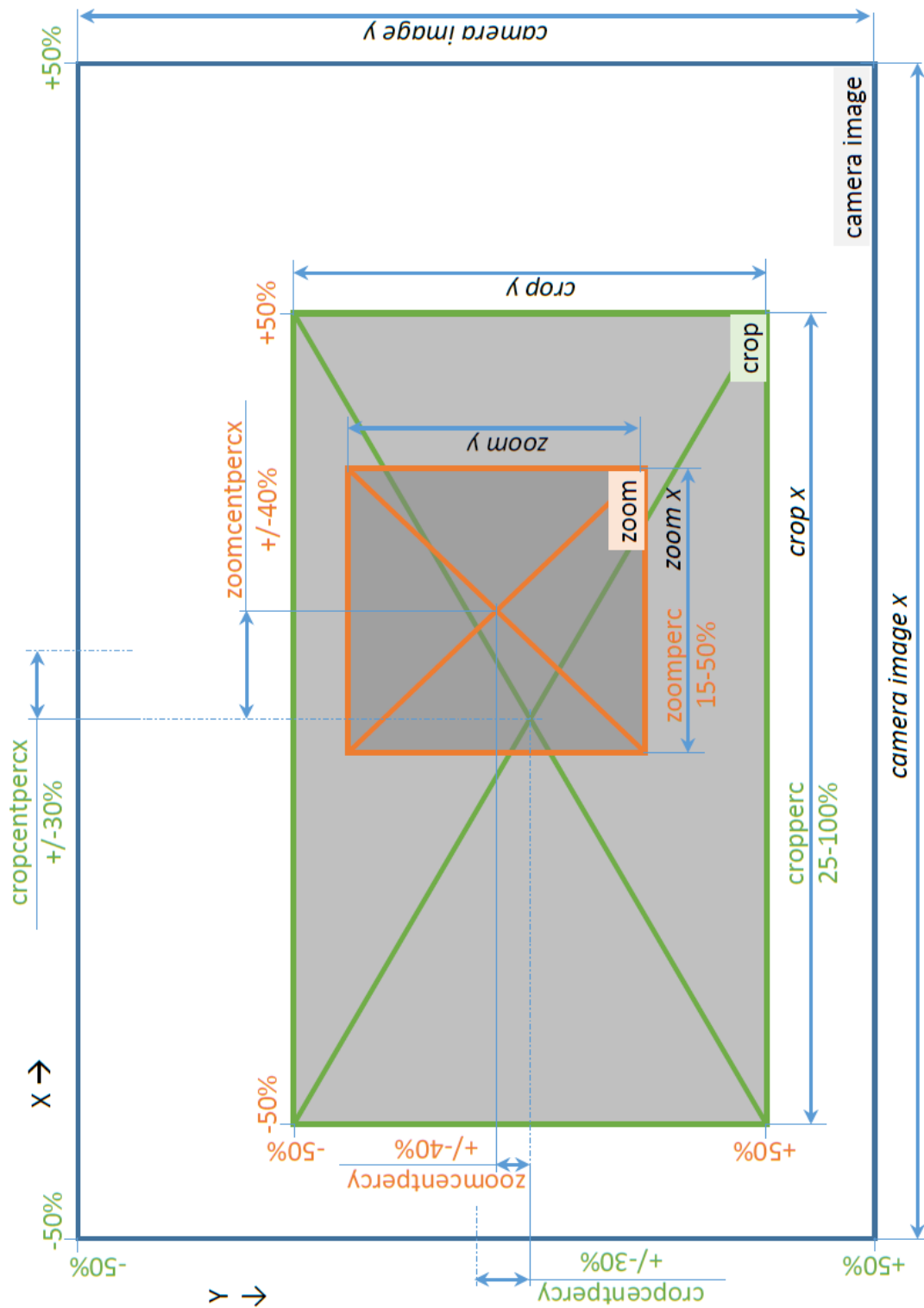
6.3 Detail content of remotecmd.cfg

<i>variable</i>	<i>e.g. value</i>	<i>remark</i>	<i>min</i>	<i>max</i>
[Command]				
camoffline =	0	no/yes indicate offline status	0	1
periodm =	0	period of image capture [min]	1	15
series =	0	no/yes store live images locally	0	1
zoommove =	0	future use (move zoom automatically)	0	1
zoomdrawrect =	0	no/yes draw a rectangle for the detail view	0	1
[Detail]				
zoomcentpercx =	0	zoom center X +/- percentage at cropped image	-40	40
zoomcentpercy =	0	zoom center Y +/- percentage at cropped image	-40	40

6.4 Drawing of images and parameters

camera image x / y = $\begin{cases} 2592 / 1944 \text{ pix} \dots \text{Raspi Cam v1} \\ 3280 / 2464 \text{ pix} \dots \text{Raspi Cam v2} \end{cases}$

Sdcam – Image Parameters

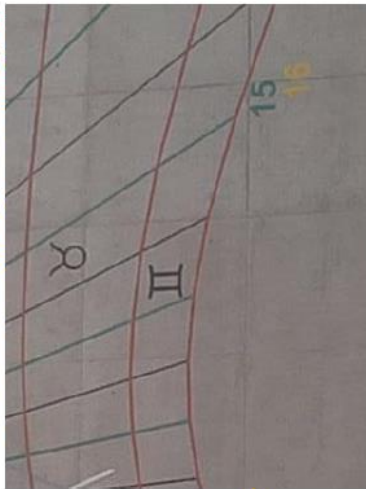


$\text{cropasratioperc} = \text{crop y x [pix]} / \text{crop x [pix]} \dots 40-150\%$
 $\text{cropwebwidth} = \text{crop x [pix]} \text{ on site} \dots 400-800 \text{ pix}$
 $\text{zoomasratioperc} = \text{zoom y x [pix]} / \text{zoom x [pix]} \dots 50-200\%$
 $\text{zoomwebwidth} = \text{zoom x [pix]} \text{ on site} \dots 300-500 \text{ pix}$

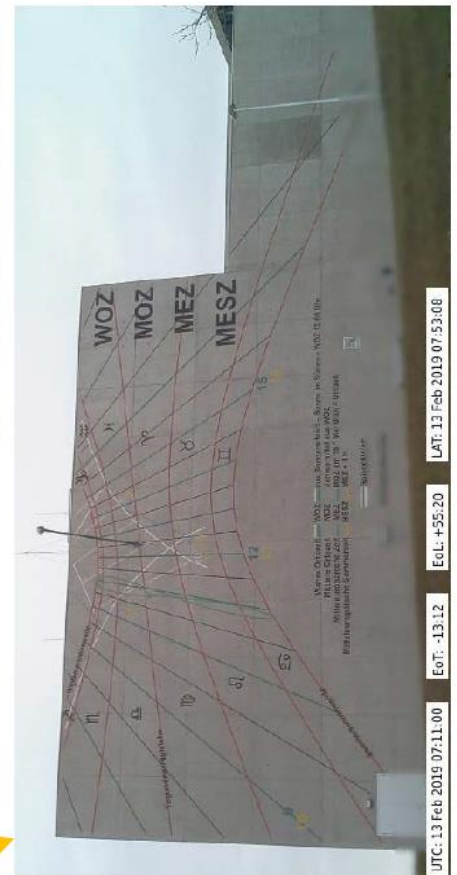
crop = imgtotal.jpg
 zoom = imgdetail.jpg

Kurt.Niel@EarthLAT1200.org

Sdcam – Image Parameters – Example KEPLERUHR



Website „imgtotal.jpg“



[Total]	
cropper =	72
cropaspratioper =	50
cropcentper =	14
cropcentper =	-3
cropwebwidth =	800
zoomper =	22
zoomaspratioper =	75
zoomwebwidth =	400
[Detail]	
...	
zoomcentper =	1
zoomcentper =	-2
...	