

Solar Installations on my Balcony and Tcl

Since 1991 I use solar panels to operate lamps, radio, tablet PC, TV etc. Recently I also started to measure voltage and current and to visualize the solar-data collected.

Of course I use Tcl for this!

contact: uli@esperanto-muenchen.de

Warning!

Though I carefully prepared the following slides
and though the Tcl-scripts run without problems
in my computers at home

there can still be some errors!

Be aware of that, if you use my files in your
projects!

3 x 12V-solar-panels

~1m² total 3rd floor directed south-east

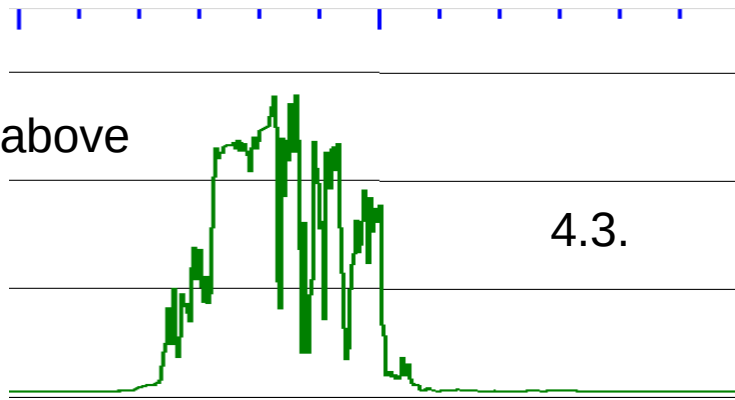
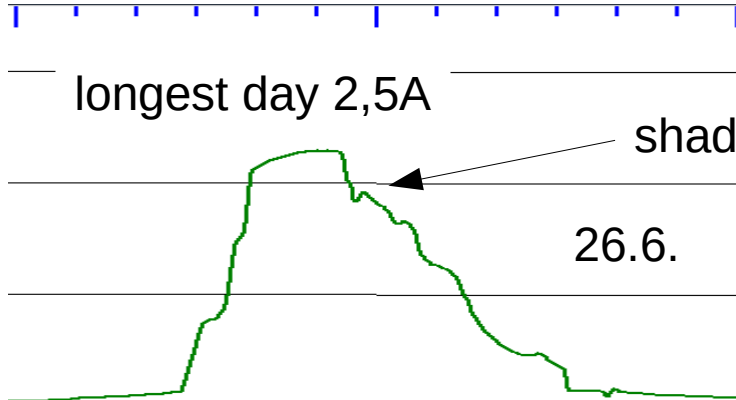
Nominally 60W max

But only sometimes 40W max reached

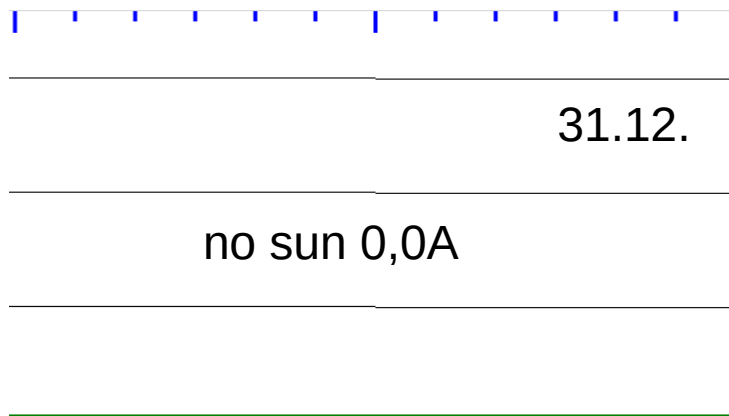
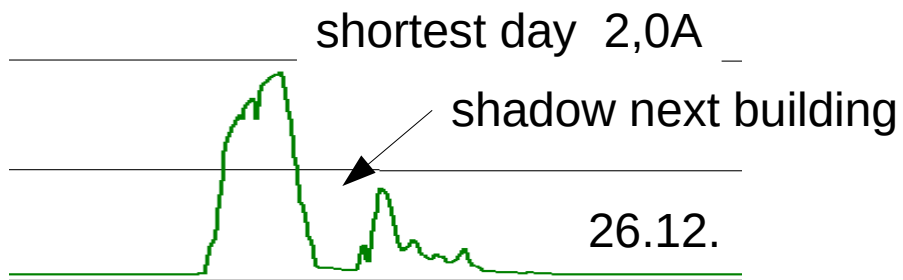
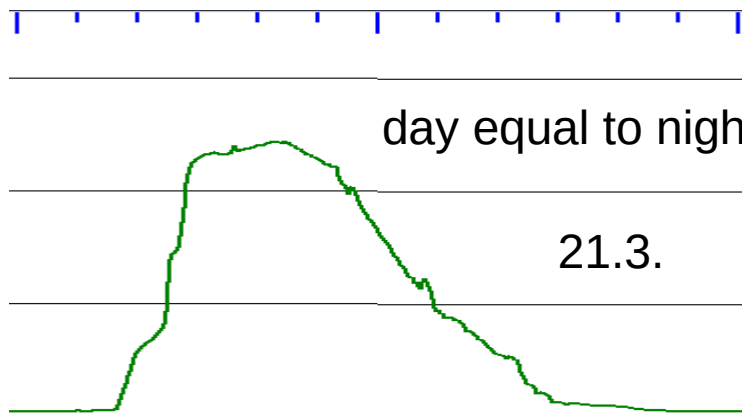


Solar energy stored in a
12V-lead-accumulator

75 Ah capacity



max current 2,7A, if module cool



Electrical parts that use solar-energy in my home

Lamps



1-7 W

Tablet PCs

Laptop PC

Smartphone

Television

Radio

Soldering iron

Gluestick pistol

DC-DC converters

DC-220V-AC-converter 30 W 50 W



12-15 W



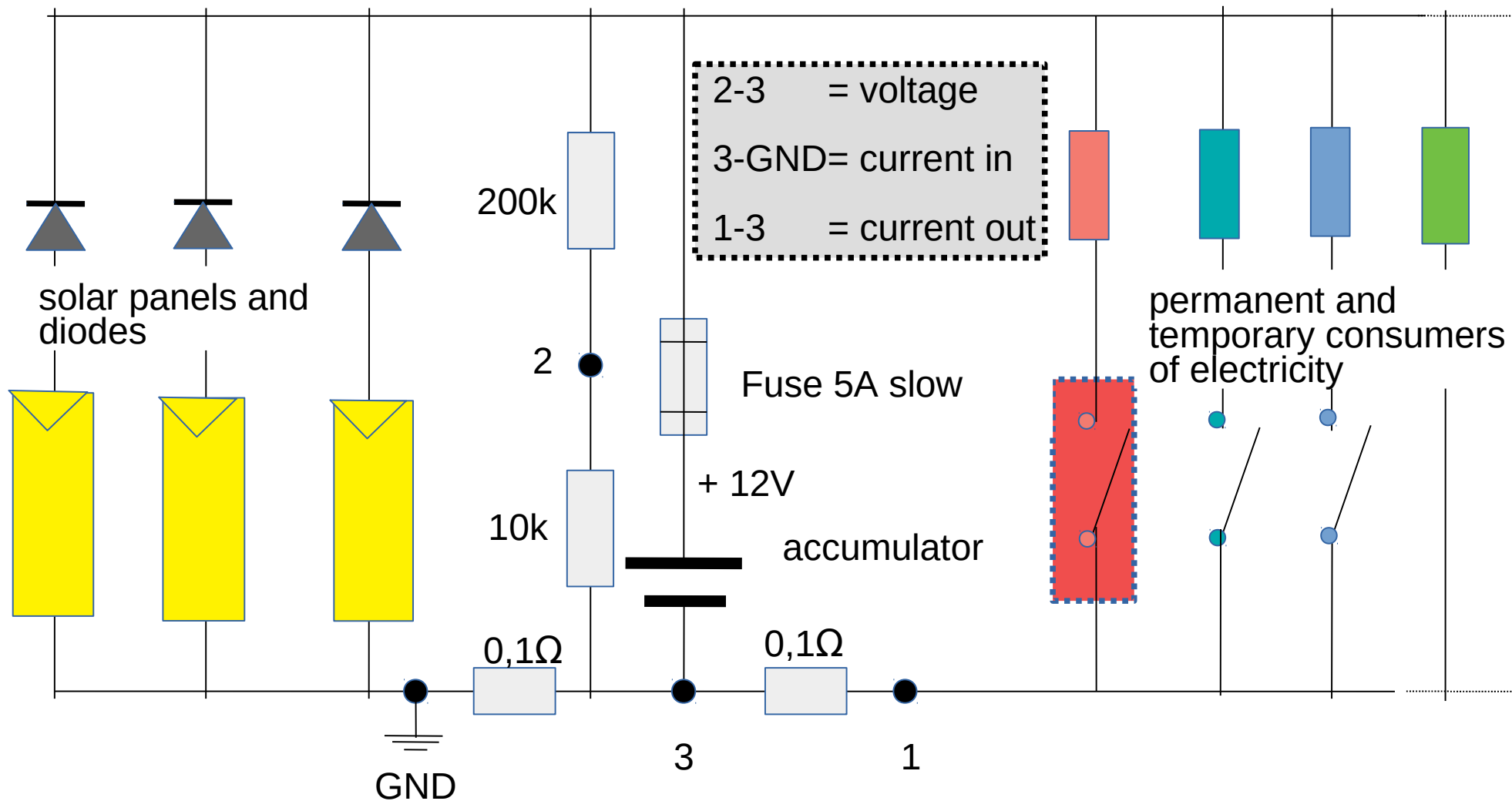
2 - 7 W



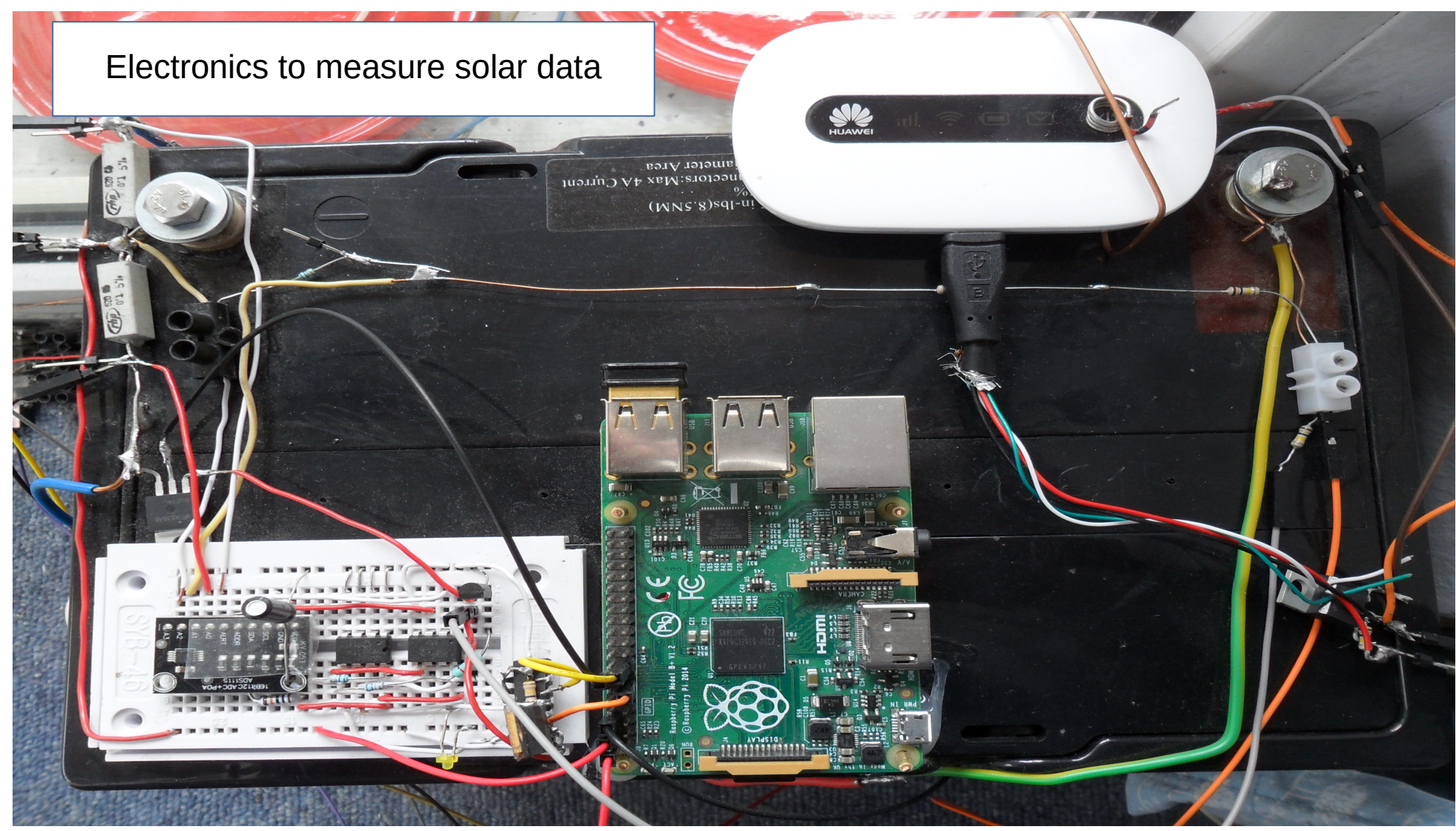
25-30 W

General Circuit

shunt-electronic
(active when >14,4V)



Electronics to measure solar data

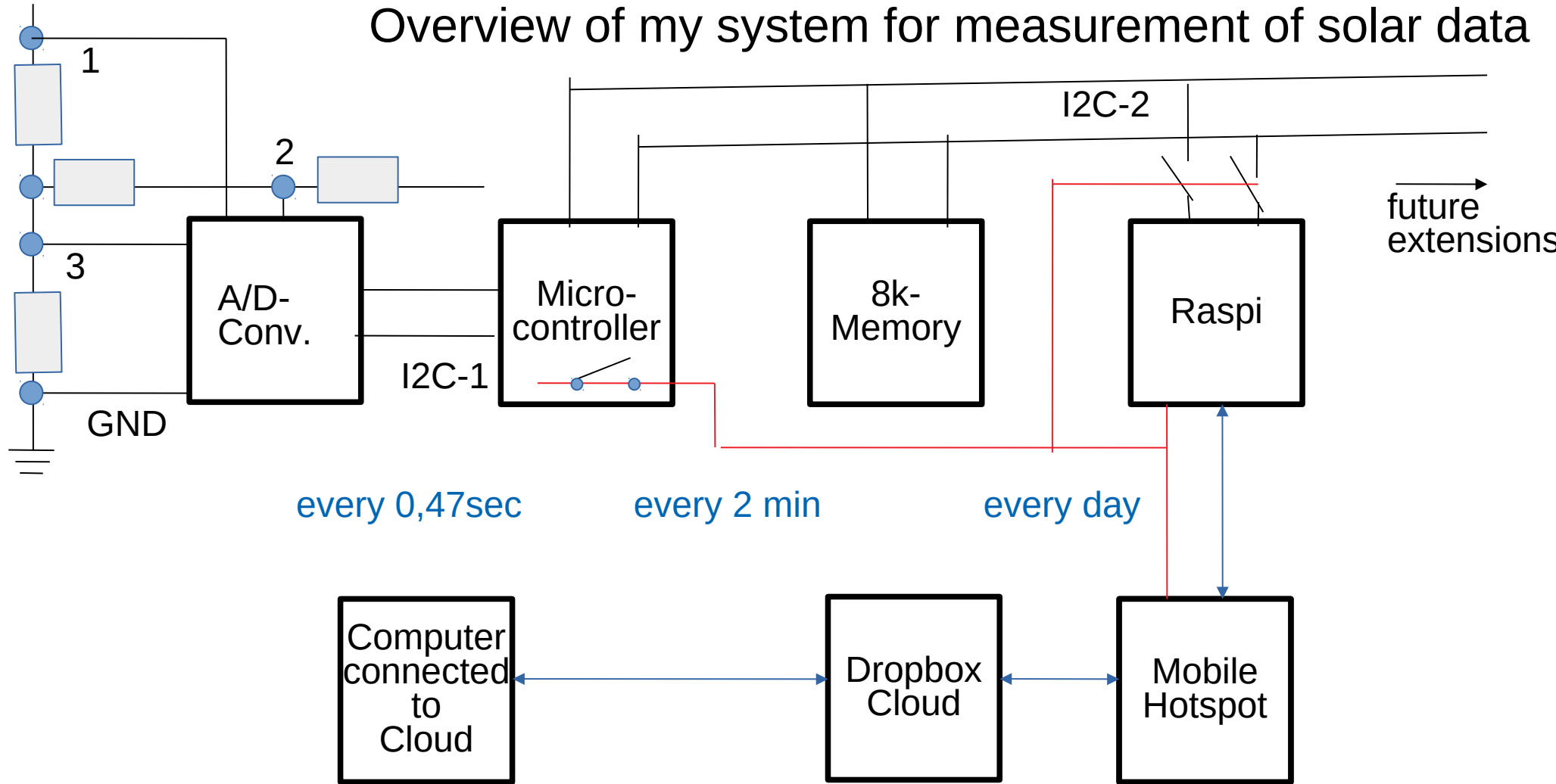


Why a microcontroller system for measurement of solar parameters is necessary

	Power	Energy per year

electricity obtained	0-40 W	20-25 kWh
electricity consumed		
Raspberry Pi + Huawei mobile wifi		
always on	3,0 W	26 kWh
2min per day on	0,004 W	0,036 kWh
ATtiny85 (128kHz) + A/D-converter + Memorychip + 5V-converter		
Always on	0,019 W	0,17 kWh

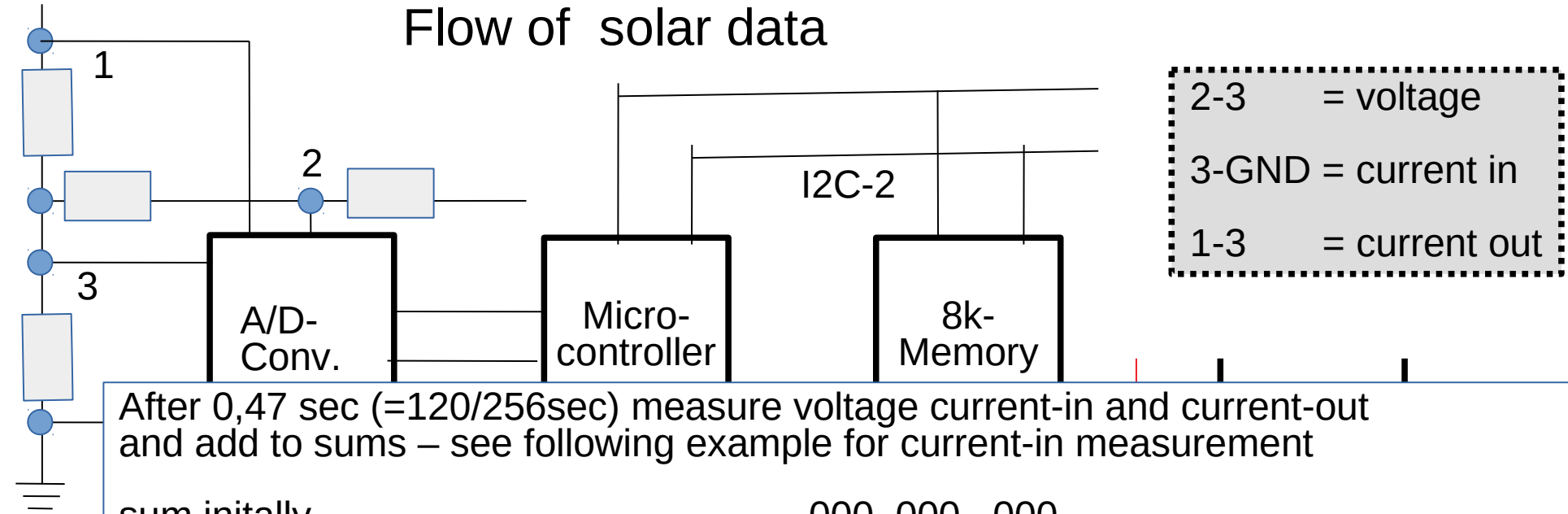
Overview of my system for measurement of solar data



TCL is involved in many steps of measuring and data processing of solar data

- writing an assembler for Attiny85
- writing a sda-interface for the raspberry pi
- programming an Attiny85 via spi-interface with the raspberry pi
- loading data from a memory chip to a raspberry pi via i2c-bus
- sending and receiving data to/from the dropbox-cloud via
- dropbox-uploader (Andrea Fabrizi www.andreafabrizi.it)
- writing a tiny IDE for the scripts used
- processing and visualising data

Flow of solar data



GND

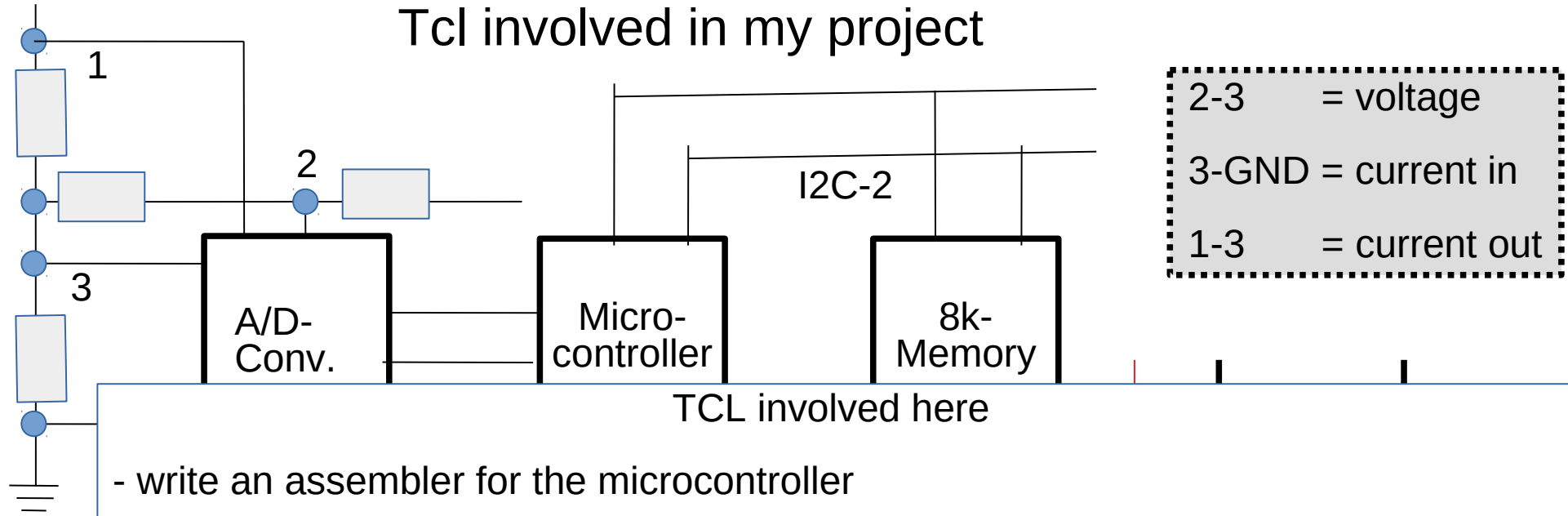
After 0,47 sec (=120/256sec) measure voltage current-in and current-out and add to sums – see following example for current-in measurement

sum initally	000	000	000
sum after 255 0,47sec-steps	123	045	001
actual current-in		020	212
sum after 256 0,47sec-steps	123	065	213
mean value of 256 measurements!	123	065	

After 2 min append data record to 8k-memory

solar data-record	number	voltage	current in	current out
	017 168	077 151	123 065	002 062
	...			

Tcl involved in my project

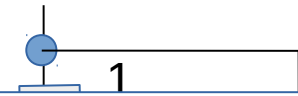


TCL involved here

GND

- write an assembler for the microcontroller
- write software for sda-interface for the raspberry pi
- write server / client to communicate between laptop and raspberry-pi
- program the microcontroller with the raspberry-pi

Flow of measured solar data

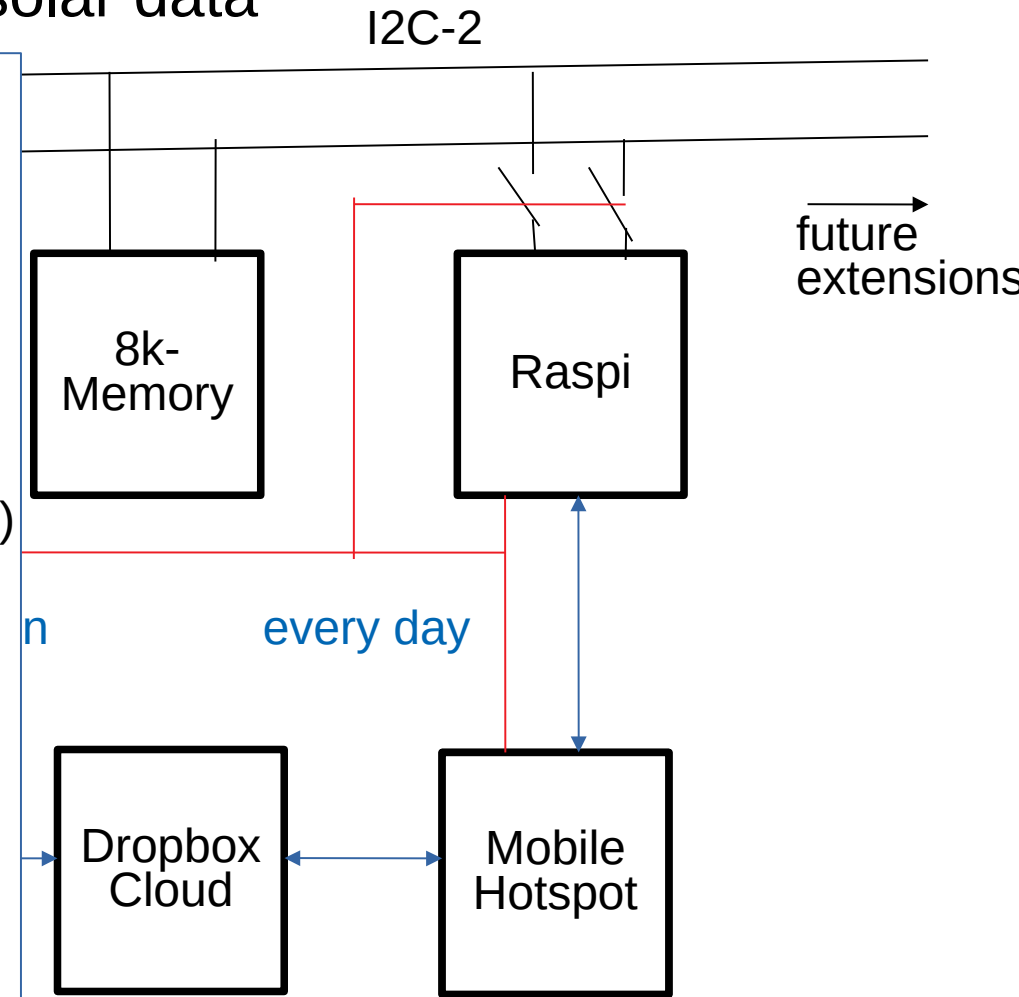


Every day at midnight

- controller switches Raspi & Hotspot on
- raspberry pi runs script „einmal“ (=once)

- initialize I2C-Bus of Raspi
- copy content of 8k-memory to file „Speicherdump.txt“
- increment number in “zaehler.txt“ (=counter)
- get time from internet
- store time + solar-data in file with the new File-number in SD-card of Raspi
- calculate and store new parameter-record In memory locations 0-7
- send data-file (e.g. datei-389.txt) to dropbox via dropbox-uploader

- controller switches Raspi off after 2 minutes
- controller reads and processes new parameter-record



Dropbox uploader

I2C-2

(bash-script of Andrea Fabrizi www.andreafabrizi.it)

used to download additional script:

```
exec /home/pi/Dropbox-Uploader/dropbox_uploader.sh download  
skript /home/pi/skript
```

used to upload daily data-file e.g.:

```
exec /home/pi/Dropbox-Uploader/dropbox_uploader.sh upload  
/home/pi/datei-394.txt datei-394.txt
```

future
extensions

Raspi

every 0,47 sec

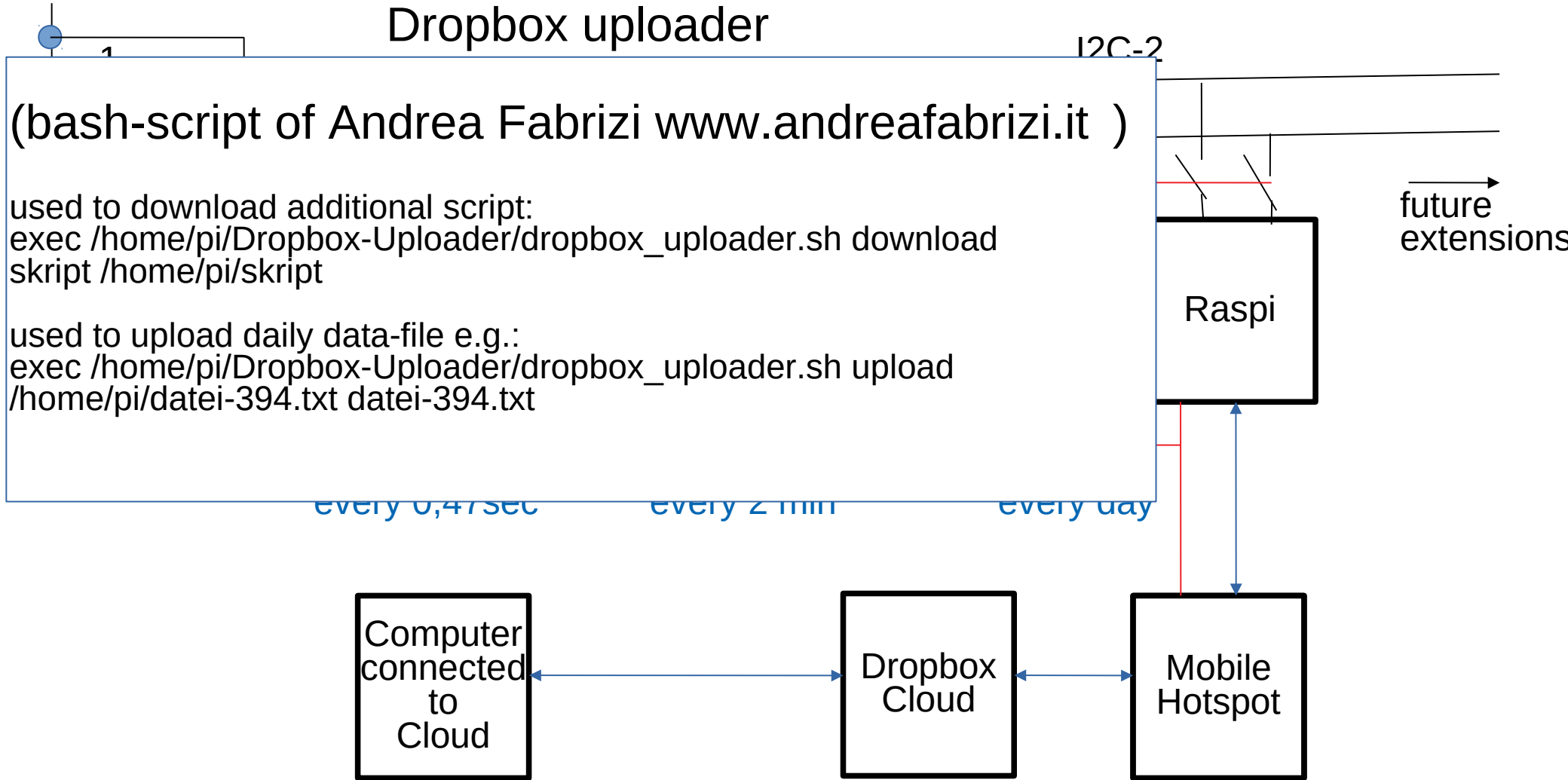
every 2 min

every day

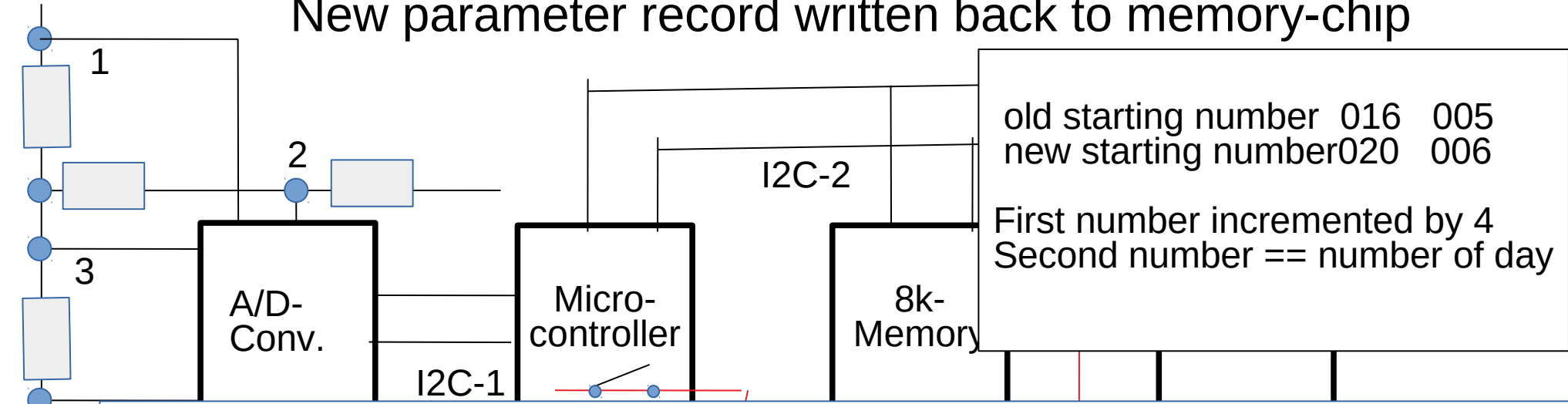
Computer
connected
to
Cloud

Dropbox
Cloud

Mobile
Hotspot



New parameter record written back to memory-chip



GND	Time of last record	Thu Jun 6 22:04:09 UTC 2019					
	Parameter-record	adjust timer	starting number		no 2min steps		future ext
	Start of memory	021 192	016	005	002	207	000 000
	Solar data-record	number	voltage		current in		current out
	First data-record	014 168	077	151	002	057	002 062
	First data of day	016 005	077	028	255	246	000 015
Last record of day		018 212	076	157	255	247	000 121
End of memory		014 167	077	169	002	190	002 062

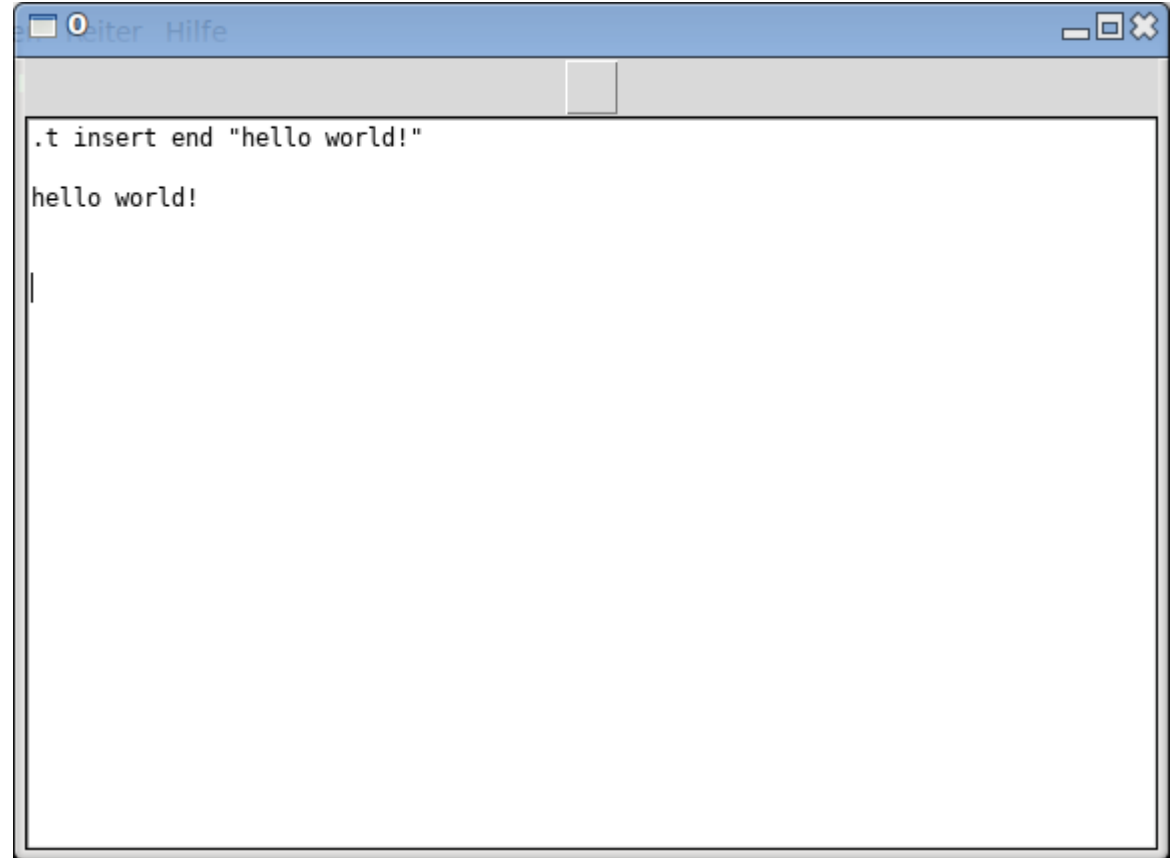
Tcl involved in my solar project

- writing a little IDE in Tcl
- writing scripts to convert solar data from binary form to useful data for visualisation

Tcl in my project : little IDE in Tcl

(filename is 0 Script has 65 bytes only)

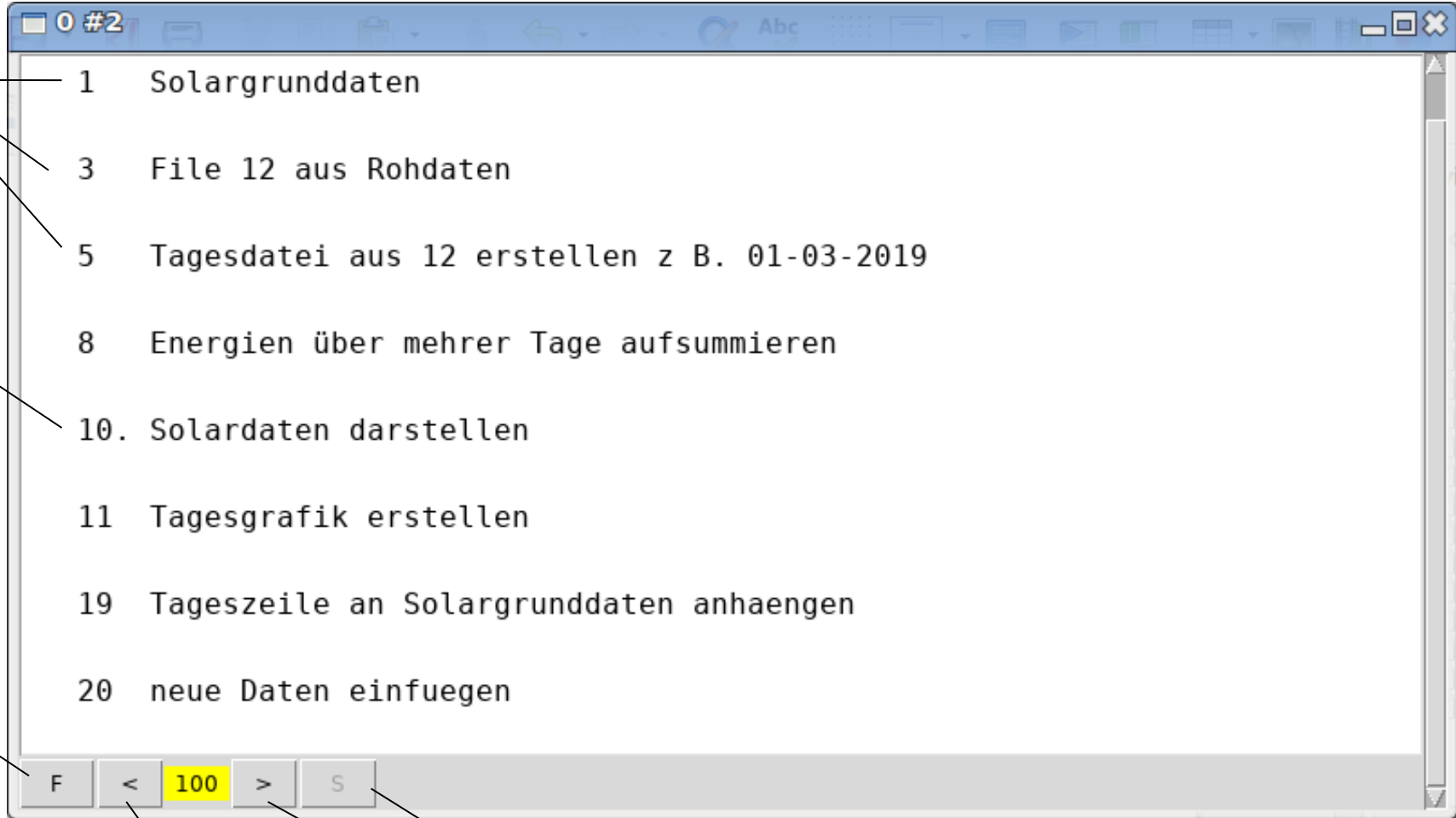
```
button .b -command {eval [.t get 0.0 end]}  
text .t  
pack .b .t
```



page 0 is the actual IDE (3.1 k) page 100 is the list of content!

If integer
move to
the marked
page

if real number
is marked
move to this
page and
source the
script there



run script

move 1 page backward / forward

save script, if button is enabled

script 3
datei-393.txt
datei-394.txt
↓
file 12

	number	voltage	in	out	date/time
	34	215	77	122	255 248 0 15 719
	34	216	77	122	255 248 0 15 720
	36	10	77	55	255 246 5 205 721
	36	11	77	63	255 247 0 15 722
	36	12	77	80	255 247 0 15 723
	36	13	77	89	255 247 0 15 724
	.	.	.		

script 5

file 12
↓
11-06-2019

time	in	out	voltage
00:00	0.408	3.437	13.157
00:02	0.408	3.437	13.157

set out [expr (int(5*256+205)/32768*1024*21.2287)]

00:04	0.12	187.417	13.113
00:06	0.264	3.437	13.118
00:08	0.264	3.437	13.129
00:10	0.264	3.437	13.135
.	.	.	

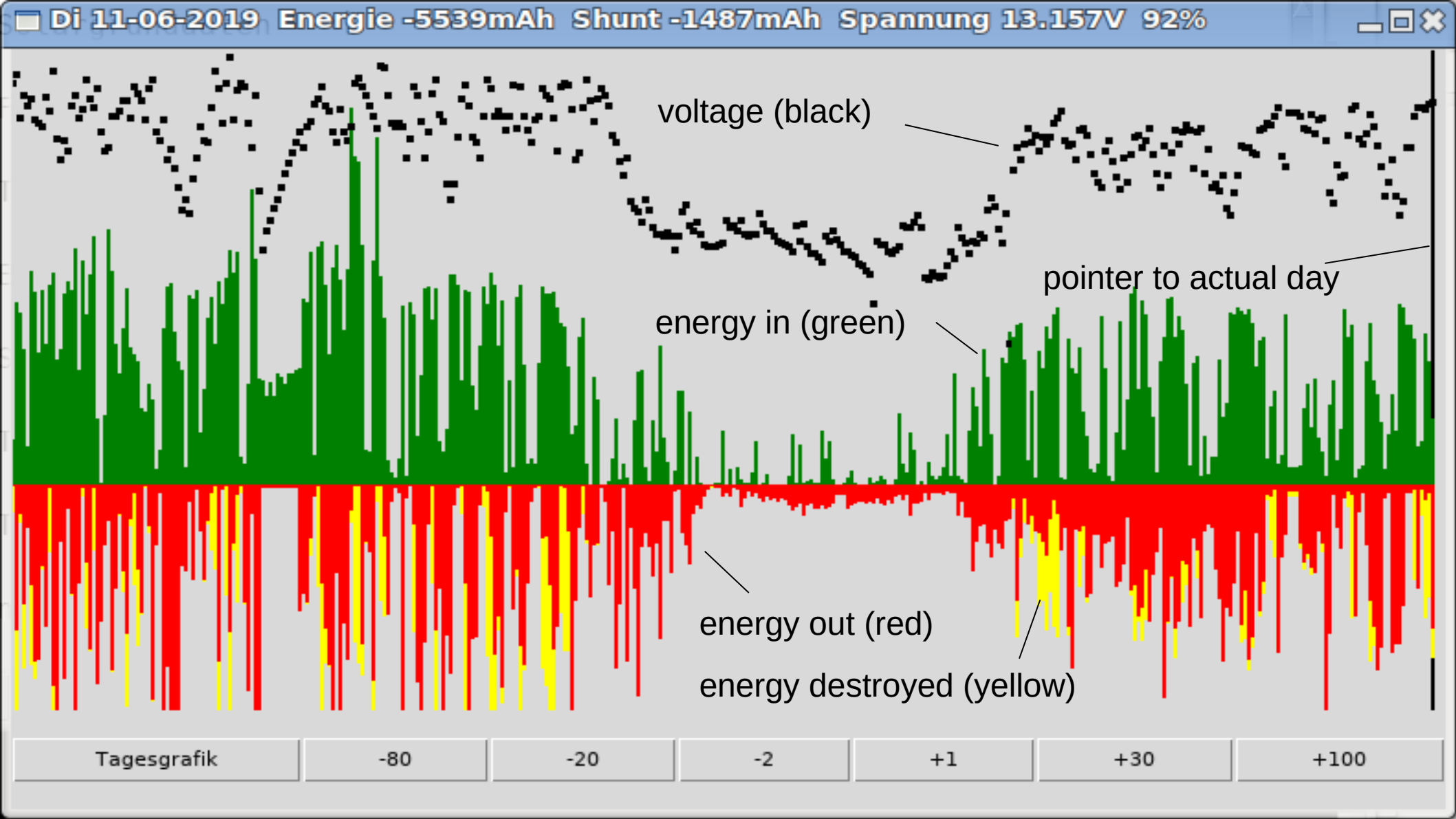
script 19
append line
in file 1

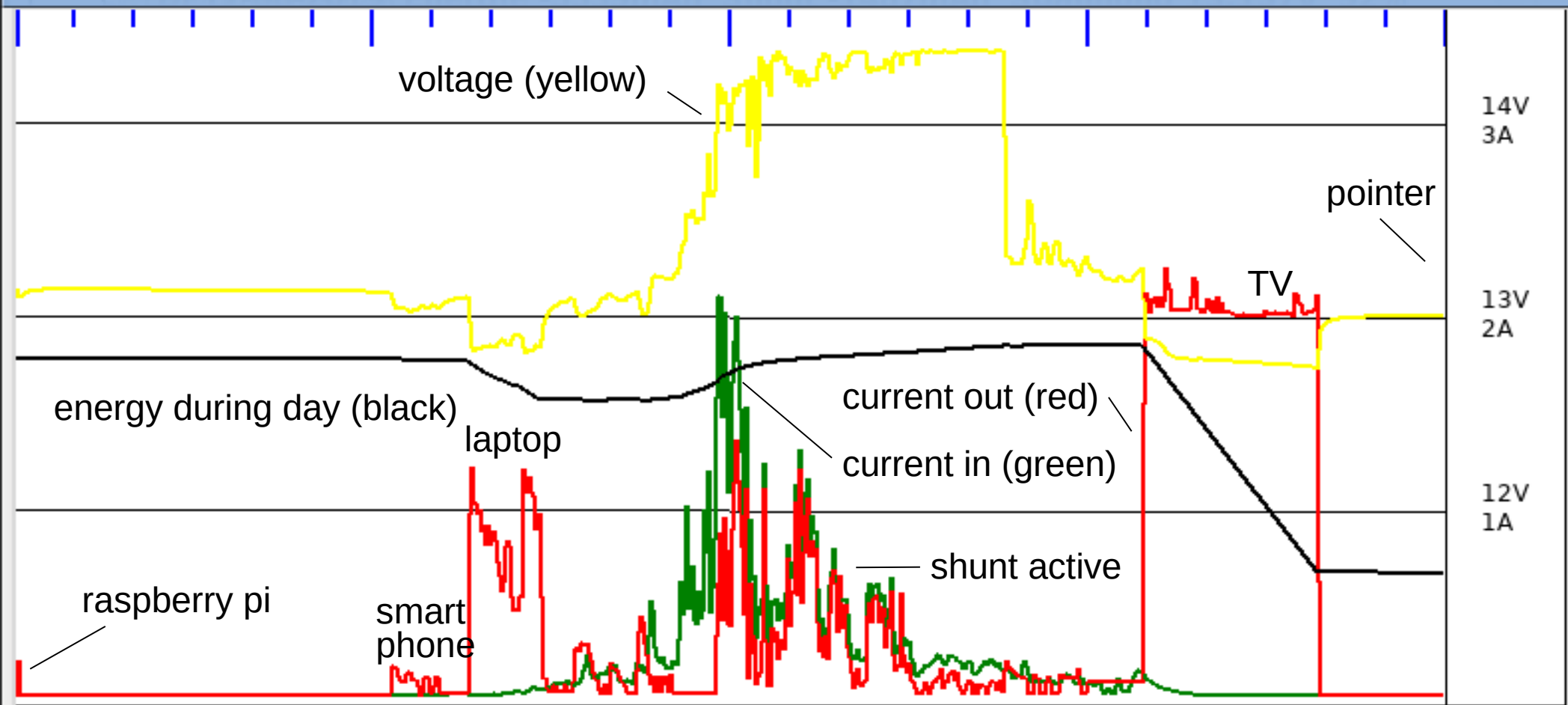
Σ ↓

date	day	delta	shunt	out	in	voltage
11-06-2019	Di	-5539	-1487	-8982	3443	13.157

Tcl involved in my solar project

- Visualisation of solar-data

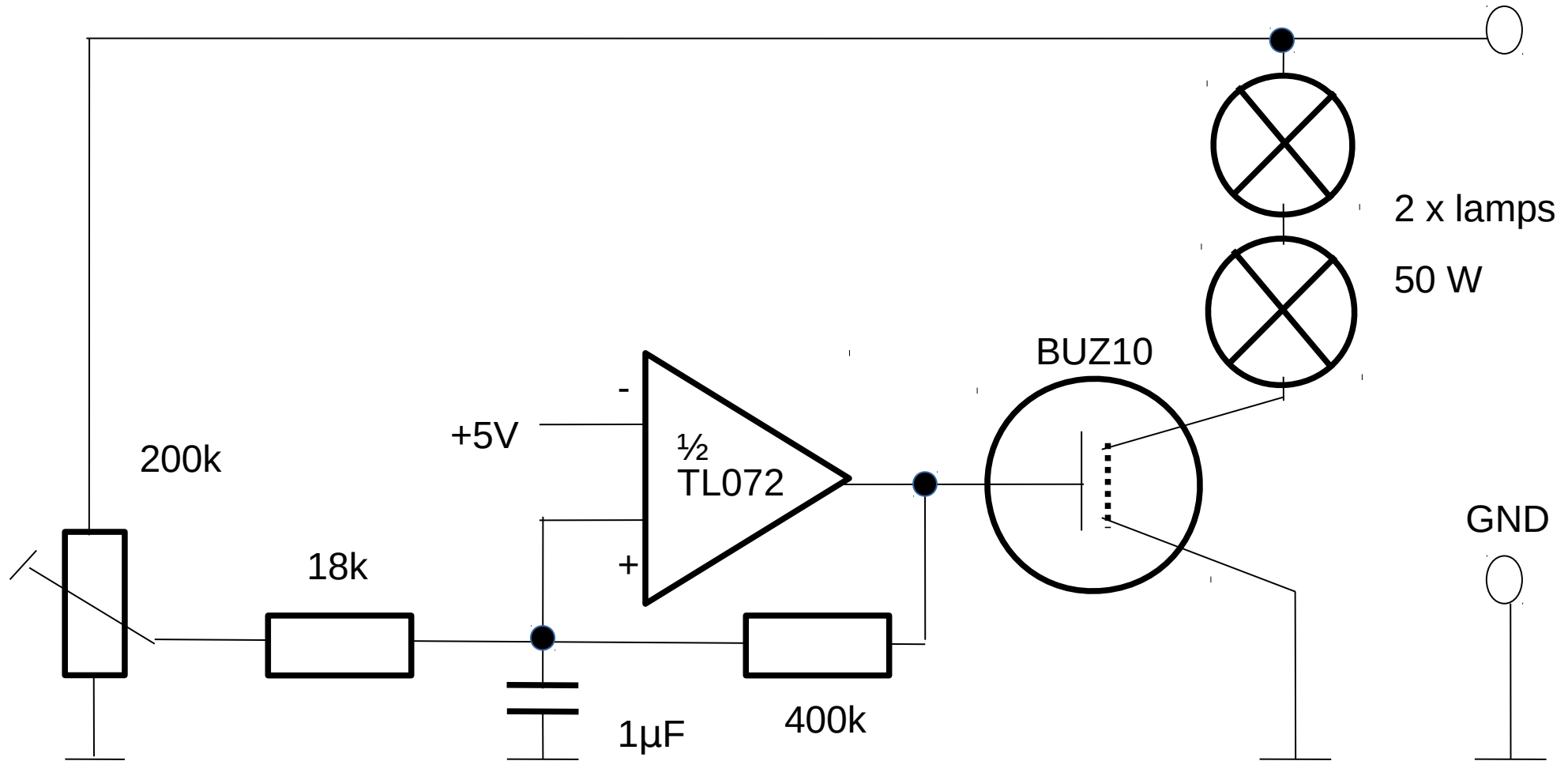


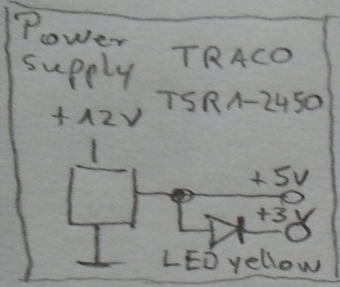


Thanks for listening!

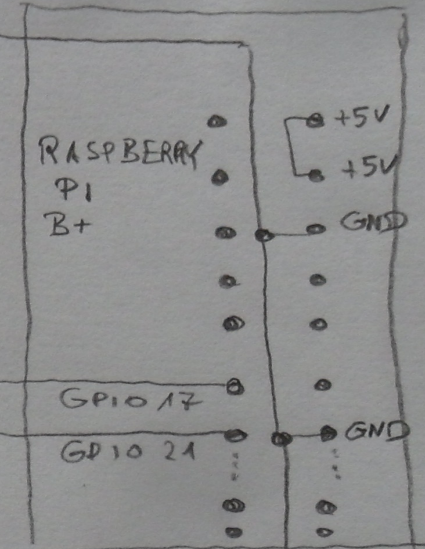
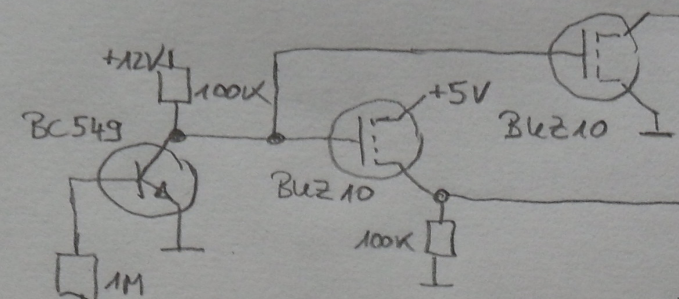
shunt electronic

10,6 – 14,4V

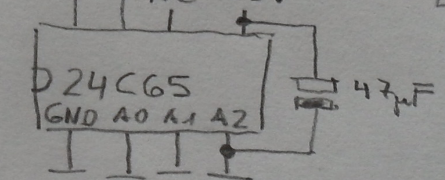
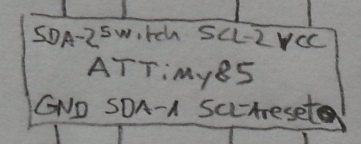
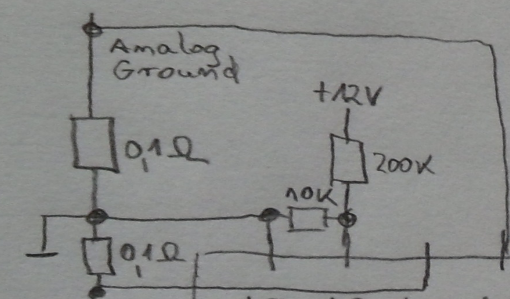




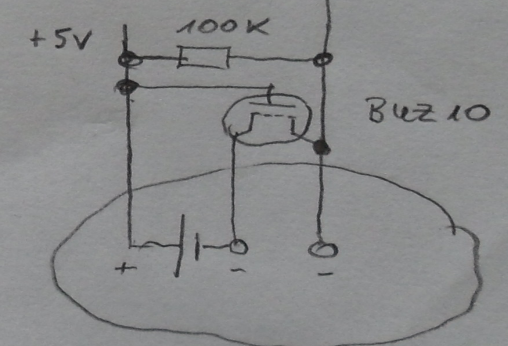
future extensions



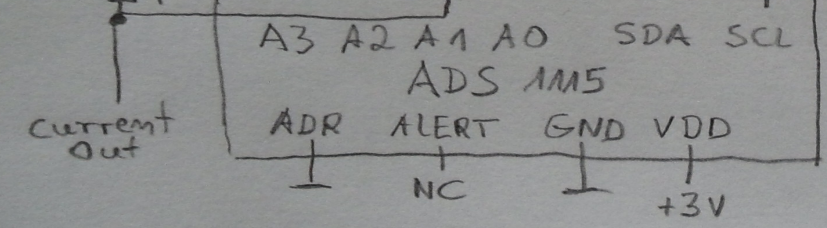
Current in



my Solar electronics



Huawei E5330 3G Mobile W:fi Button always pressed



example of source code for solar measurements with ATtiny85

# program solar installation	initialize_microcontroller	
location 1	fetch_analogue_values & calculate_mean_values	
location 2	if_047sec_are_not_over_(R3=0)_goto	address 2
	0->R3	
	if_2minutes_are_not_over_(R4=0)_goto	address 1
	0->R4	
	fetch_memory_address_7_from_memory_chip	
	if_memory_address_7_≡0_goto	address 3
location 6	switch_raspi_on	
	1->R2	
location 4	store_8_data_bytes_&_memory-pointer_+_8	
	goto	address 1
location 3	switch_raspi_off	
	if_R2=1_goto	address 5
	compare_data_pointer_with_end-value	
	if_end-value_reached_goto	address 6
	goto	address 4
location 5	0->R2	
	move_parameter-record_to_microcontroller	
	calculate_new_end-value	
	goto	address 4
	subprograms	

main steps of assembling

macro expansion
eventually several steps

location 2 if_047sec_are_not_over_(R3=0)_goto adress 2

↓
loko 2 se_R3=0 adr 2

↓
loko 2 a< - R3
a+ 0
ir0 adr 2

↓
loko 2 ldi 27 0
ldi 26 3
ld 16
ldi 17 0
add 16 17
breq adr 2

calculation relative addresses
from labels

↓
ldi 27 0
ldi 26 3
ld 16
ldi 17 0
add 16 17
breq -6

creating HEX-file for Atmel-controllers

....
40 01 7A B0
48 01 7A E0
40 01 7B A3
48 01 7B E0
40 01 7C 0C
48 01 7C 91
....