

DLØSHF station remote control with BeagleBone, Python and ZMQ

Per Dudek
Martin Sufke

2018-09-16 EUCARA am Stockert

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

www.dl0shf.de
per@per-dudek.de
eucara.2018.masuefke@recursor.net

DLØSHF at a glance

- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
 - Python
 - ZMQ
 - Antenna Control
 - Amplifier Control
 - Device Controls
- Live Demo
- End.



The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

2/28

DLØSHF at a glance



1 GHz dish Ø 9m
1296 MHz Rx/Tx
1420 MHz Rx
all transistor PA



The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

2/28



2 GHz dish Ø 6m
DSN 2.1–2.2 GHz Rx
2.3 GHz Amateur radio
all transistor PA

DLØSHF at a glance



The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

2/28



8.4 GHz dish Ø 7.2m
Deep Space Network
Rx only

DLØSHF at a glance



The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

2/28



10 GHz dish Ø 7.2m
10 GHz Amateur Radio
Moon beacon
50 W transistor PA
600 W TWTA

DLØSHF at a glance



The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

DLØSHF at a glance



24 GHz dish Ø 3.7m
24 GHz Amateur Radio
30 W transistor PA



The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

2/28



32 GHz dish Ø 2m
32.1 GHz Deep Space
Network
Rx only

DLØSHF at a glance



The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

2/28



60m tower

lower tier: commercial

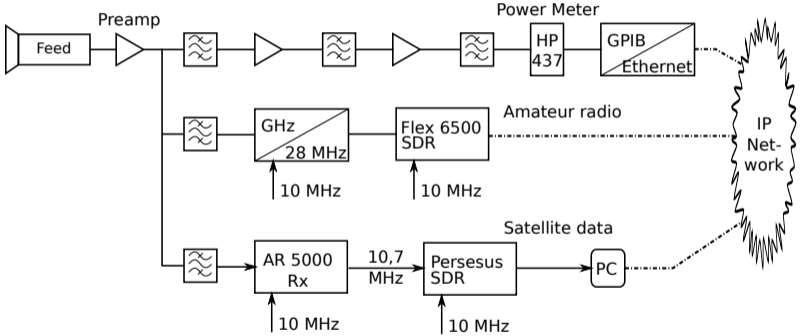
upper tier:

- 10/24/47 GHz dish Ø 2m
- 1296/2320 MHz dish Ø 2m
- 50/70/432 MHz yagi

DLØSHF at a glance

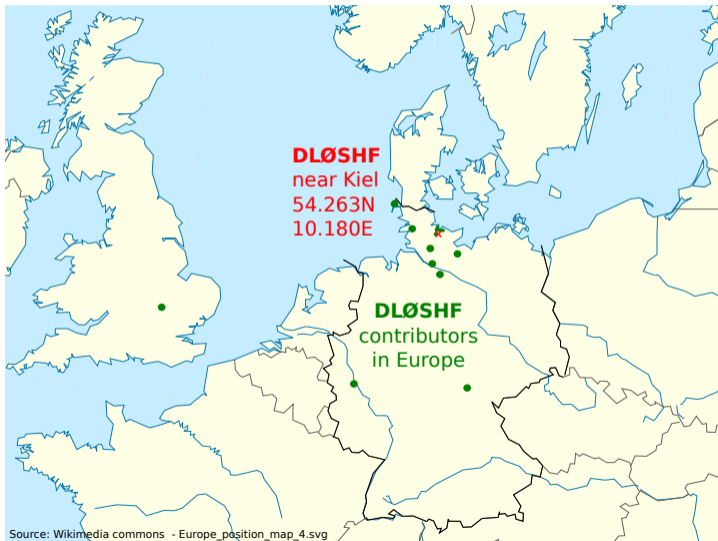


DLØSHF default Rx path



- The Station
- From above
- Usage**
- Users
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
 - Python
 - ZMQ
 - Antenna Control
 - Amplifier Control
 - Device Controls
- Live Demo
- End.

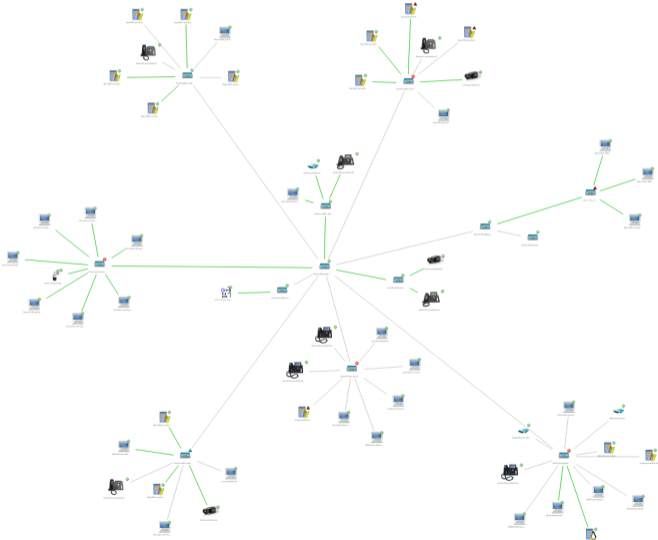
- The Station
- From above
- Usage
- Users**
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
 - Python
 - ZMQ
 - Antenna Control
 - Amplifier Control
 - Device Controls
- Live Demo
- End.



DLØSHF Ethernet (data/user)

- The Station
- From above
- Usage
- Users
- Network

- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
 - Python
 - ZMQ
 - Antenna Control
 - Amplifier Control
 - Device Controls
- Live Demo
- End.



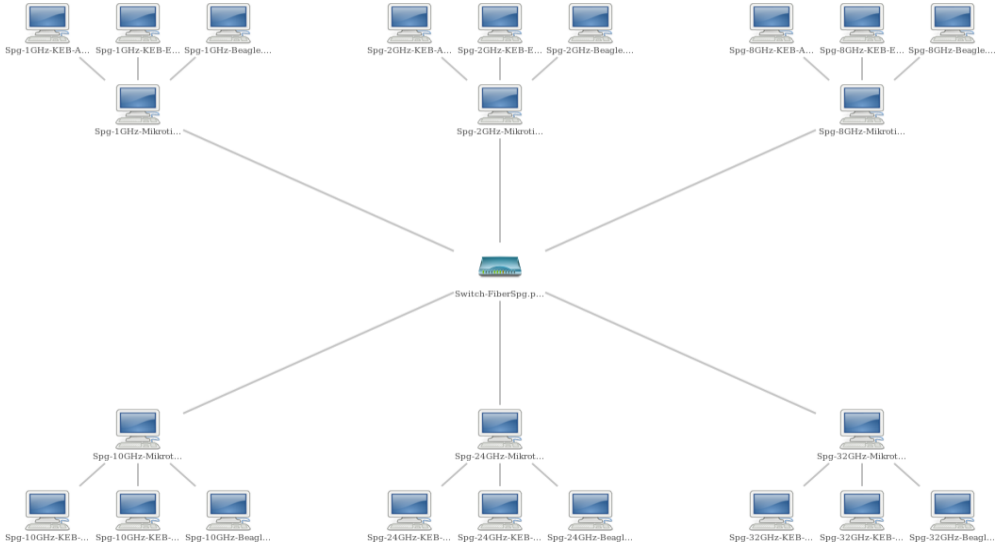
DLØSHF Ethernet – 130 nodes active



Active node Free IP on the network 192.168.9.0/24 at 2018-09-14.

- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
 - Python
 - ZMQ
 - Antenna Control
 - Amplifier Control
 - Device Controls
- Live Demo

DLØSHF Ethernet (antennas)



- The Station
- From above
- Usage
- Users
- Network

- Hardware
- Antennas
- BeagleBone
- PRU
- GPIO

- Software
- Python
- ZMQ
- Antenna Control
- Amplifier Control
- Device Controls

- Live Demo
- End.

Antenna control (one antenna)

- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas**
- BeagleBone
 - PRU
 - GPIO
- Software
- Python
- ZMQ
- Antenna Control
- Amplifier Control
- Device Controls
- Live Demo
- End.

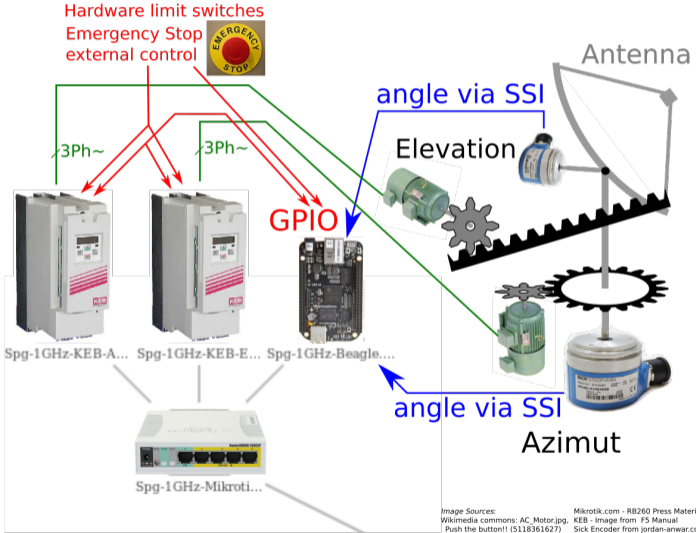
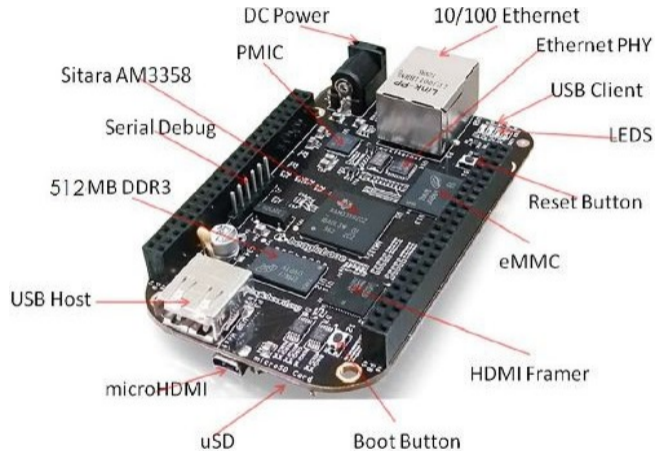


Image Sources:
 Wikimedia commons: AC_Motor.jpg
 Push the button!! (5118361627)
 Mikrotik.com - RB260 Press Material
 KEB - Image from P3 Manual
 Sick Encoder from jordan-anwar.com

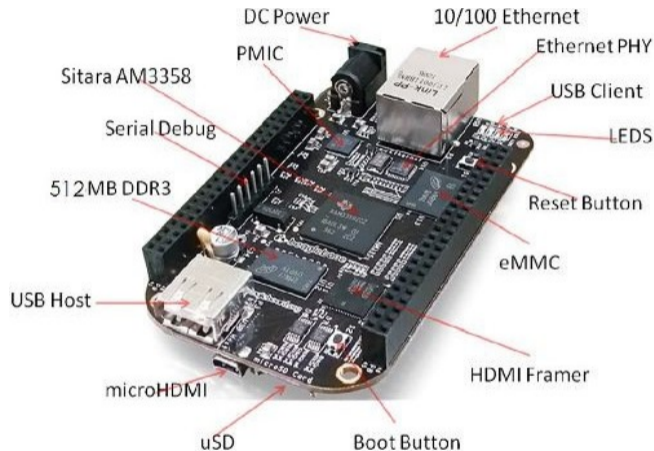
- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas
- BeagleBone
- PRU
- GPIO
- Software
- Python
- ZMQ
- Antenna Control
- Amplifier Control
- Device Controls
- Live Demo
- End.

BeagleBoneBlack

- CPU: 32 bit ARM v7

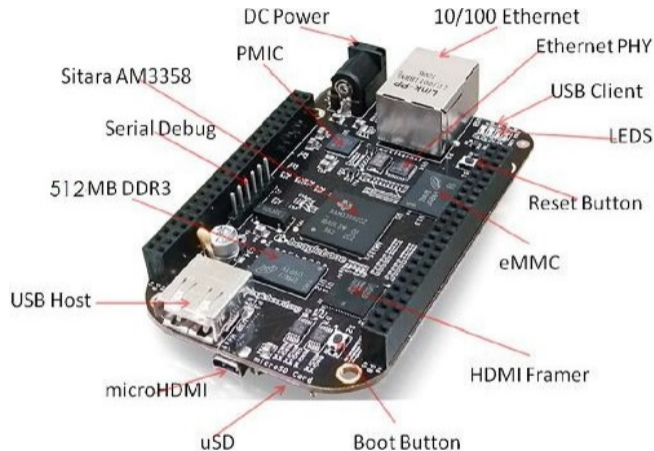


BeagleBoneBlack



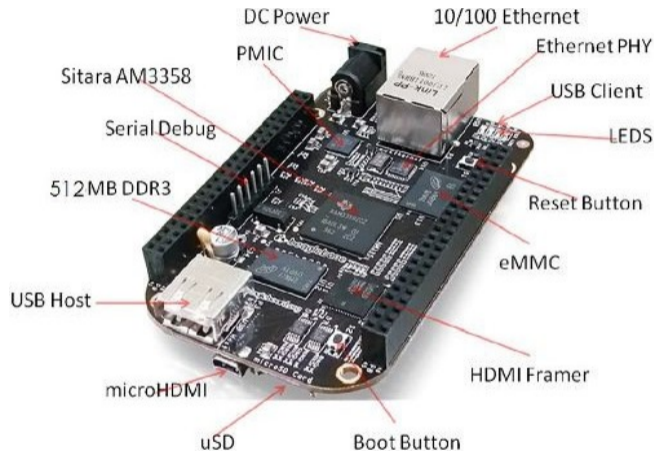
- CPU: 32 bit ARM v7
- 512 MB DDR3

BeagleBoneBlack



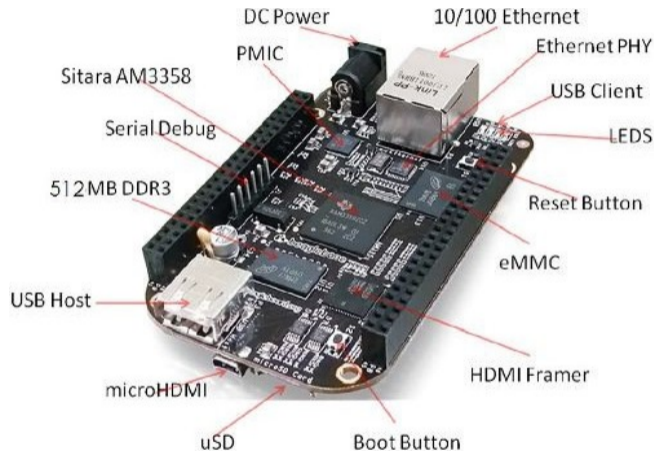
- CPU: 32 bit ARM v7
- 512 MB DDR3
- 4 GB eMMC/
any SDcard

BeagleBoneBlack



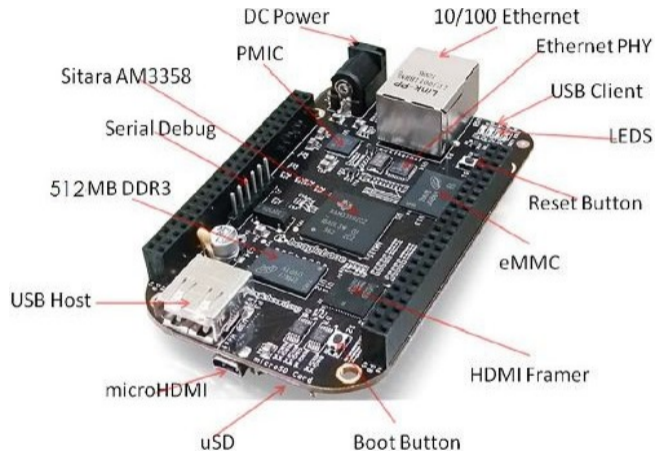
- CPU: 32 bit ARM v7
- 512 MB DDR3
- 4 GB eMMC/
any SDcard
- 65 GPIO

BeagleBoneBlack



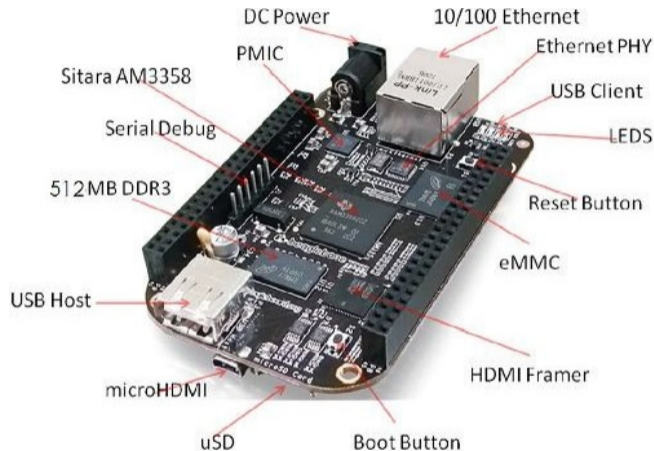
- CPU: 32 bit ARM v7
- 512 MB DDR3
- 4 GB eMMC/
any SDcard
- 65 GPIO
- 6 RS232

BeagleBoneBlack



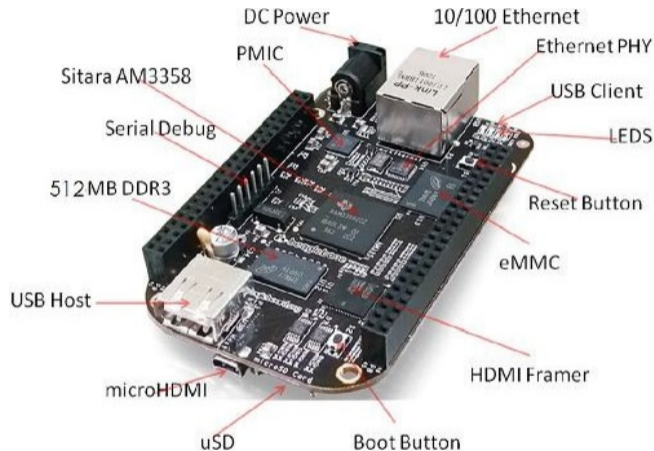
- CPU: 32 bit ARM v7
- 512 MB DDR3
- 4 GB eMMC/
any SDcard
- 65 GPIO
- 6 RS232
- 2 I²C

BeagleBoneBlack



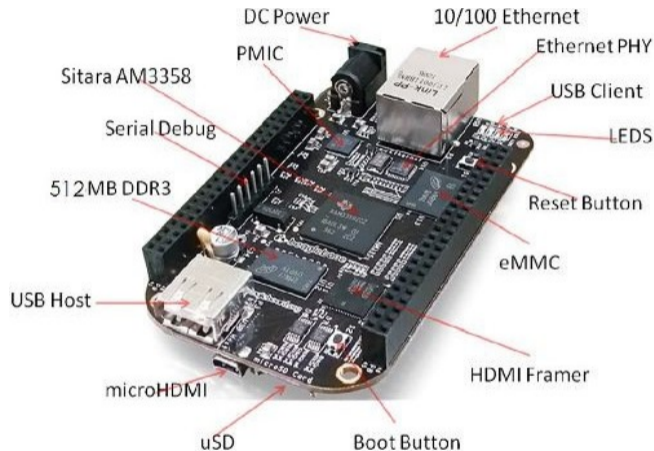
- CPU: 32 bit ARM v7
- 512 MB DDR3
- 4 GB eMMC/
any SDcard
- 65 GPIO
- 6 RS232
- 2 I²C
- 2 SPI

BeagleBoneBlack



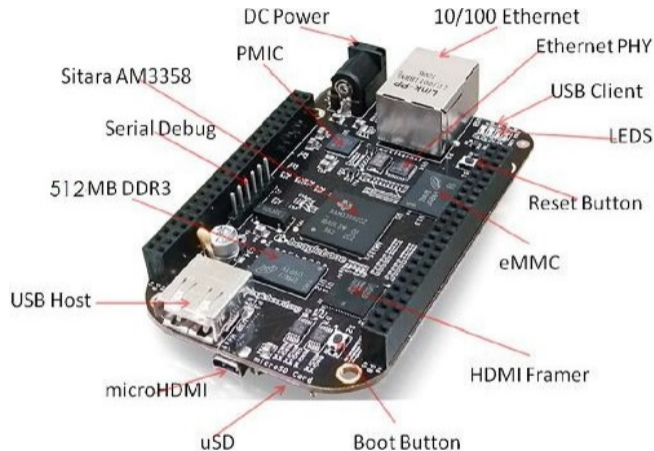
- CPU: 32 bit ARM v7
- 512 MB DDR3
- 4 GB eMMC/
any SDcard
- 65 GPIO
- 6 RS232
- 2 I²C
- 2 SPI
- 2 CAN

BeagleBoneBlack



- CPU: 32 bit ARM v7
- 512 MB DDR3
- 4 GB eMMC/
any SDcard
- 65 GPIO
- 6 RS232
- 2 I²C
- 2 SPI
- 2 CAN
- 8 analog in

BeagleBoneBlack



- CPU: 32 bit ARM v7
- 512 MB DDR3
- 4 GB eMMC/ any SDcard
- 65 GPIO
- 6 RS232
- 2 I²C
- 2 SPI
- 2 CAN
- 8 analog in
- 2 PRU

PRU: Programmable Real-time Unit

- 32 bit RISC microcontroller

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

PRU: Programmable Real-time Unit

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

- 32 bit RISC microcontroller
- running at 200 MHz

PRU: Programmable Real-time Unit

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

- 32 bit RISC microcontroller
- running at 200 MHz
- with 8 kB instructions + 8 kB RAM

PRU: Programmable Real-time Unit

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

- 32 bit RISC microcontroller
- running at 200 MHz
- with 8 kB instructions + 8 kB RAM
- Assembler + C-compiler available

PRU: Programmable Real-time Unit

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

- 32 bit RISC microcontroller
- running at 200 MHz
- with 8 kB instructions + 8 kB RAM
- Assembler + C-compiler available
- 2 of them in **BeagleBoneBlack**

PRU: Programmable Real-time Unit

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

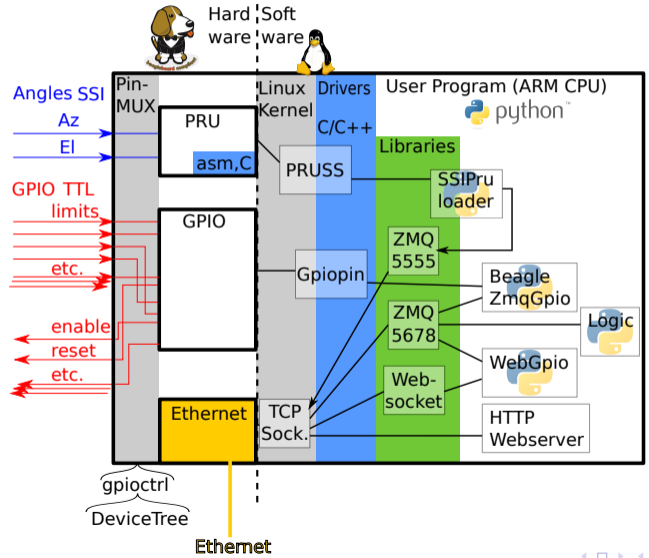
Device Controls

Live Demo

End.

- 32 bit RISC microcontroller
- running at 200 MHz
- with 8 kB instructions + 8 kB RAM
- Assembler + C-compiler available
- 2 of them in **BeagleBoneBlack**
- well-connected to the ARM CPU

PRU+GPIO in BeagleBone



- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
 - Python
 - ZMQ
 - Antenna Control
 - Amplifier Control
 - Device Controls
- Live Demo
- End.



- invented 1991 by Guido van Rossum

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- invented 1991 by Guido van Rossum
- compiled on runtime, interpreted, source-code always available

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- invented 1991 by Guido van Rossum
- compiled on runtime, interpreted, source-code always available
- current version 3.7.x (DLØSHF: 3.4.5)

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- invented 1991 by Guido van Rossum
- compiled on runtime, interpreted, source-code always available
- current version 3.7.x (DLØSHF: 3.4.5)
- extensive library

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- invented 1991 by Guido van Rossum
- compiled on runtime, interpreted, source-code always available
- current version 3.7.x (DLØSHF: 3.4.5)
- extensive library
- humongous eco-system (3rd-party projects, tutorials. . .)

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- invented 1991 by Guido van Rossum
- compiled on runtime, interpreted, source-code always available
- current version 3.7.x (DLØSHF: 3.4.5)
- extensive library
- humongous eco-system (3rd-party projects, tutorials. . .)
- integrates C/C++ code easily

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- invented 1991 by Guido van Rossum
- compiled on runtime, interpreted, source-code always available
- current version 3.7.x (DLØSHF: 3.4.5)
- extensive library
- humongous eco-system (3rd-party projects, tutorials. . .)
- integrates C/C++ code easily
- quite fast

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- invented 1991 by Guido van Rossum
- compiled on runtime, interpreted, source-code always available
- current version 3.7.x (DLØSHF: 3.4.5)
- extensive library
- humongous eco-system (3rd-party projects, tutorials. . .)
- integrates C/C++ code easily
- quite fast
- tolerable overhead

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- invented 1991 by Guido van Rossum
- compiled on runtime, interpreted, source-code always available
- current version 3.7.x (DLØSHF: 3.4.5)
- extensive library
- humongous eco-system (3rd-party projects, tutorials. . .)
- integrates C/C++ code easily
- quite fast
- tolerable overhead
- interactive development/debugging possible

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware
- DESY (Hamburg) has positive experience

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware
- DESY (Hamburg) has positive experience
- Language bindings for many languages

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware
- DESY (Hamburg) has positive experience
- Language bindings for many languages
- Message queuing without central message broker

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware
- DESY (Hamburg) has positive experience
- Language bindings for many languages
- Message queuing without central message broker
- Delivery guarantes:

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware
- DESY (Hamburg) has positive experience
- Language bindings for many languages
- Message queuing without central message broker
- Delivery guarantes:
 - whole messages

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware
- DESY (Hamburg) has positive experience
- Language bindings for many languages
- Message queuing without central message broker
- Delivery guarantes:
 - whole messages
 - ordered delivery

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware
- DESY (Hamburg) has positive experience
- Language bindings for many languages
- Message queuing without central message broker
- Delivery guarantees:
 - whole messages
 - ordered delivery
 - delivery, (limited) internal caching

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware
- DESY (Hamburg) has positive experience
- Language bindings for many languages
- Message queuing without central message broker
- Delivery guarantees:
 - whole messages
 - ordered delivery
 - delivery, (limited) internal caching
- many peers

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware
- DESY (Hamburg) has positive experience
- Language bindings for many languages
- Message queuing without central message broker
- Delivery guarantees:
 - whole messages
 - ordered delivery
 - delivery, (limited) internal caching
- many peers
- automatic (re-)connects

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
- invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
- 'gilded' 2011 by CERN study for best middleware
- DESY (Hamburg) has positive experience
- Language bindings for many languages
- Message queuing without central message broker
- Delivery guarantees:
 - whole messages
 - ordered delivery
 - delivery, (limited) internal caching
- many peers
- automatic (re-)connects

==> *a socket on steroids*

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
 - invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
 - 'gilded' 2011 by CERN study for best middleware
 - DESY (Hamburg) has positive experience
 - Language bindings for many languages
 - Message queuing without central message broker
 - Delivery guarantees:
 - whole messages
 - ordered delivery
 - delivery, (limited) internal caching
 - many peers
 - automatic (re-)connects
- ==> *a socket on steroids*
- quite fast

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.



- spelled zero-em-queue, ØMQ, ZeroMQ, short ZMQ
 - invented 2007 by Martin Sustrik, Pieter Hintjens (iMatix corp.)
 - 'gilded' 2011 by CERN study for best middleware
 - DESY (Hamburg) has positive experience
 - Language bindings for many languages
 - Message queuing without central message broker
 - Delivery guarantees:
 - whole messages
 - ordered delivery
 - delivery, (limited) internal caching
 - many peers
 - automatic (re-)connects
- ==> *a socket on steroids*
- quite fast
 - tolerable overhead

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control


Amplifier Control

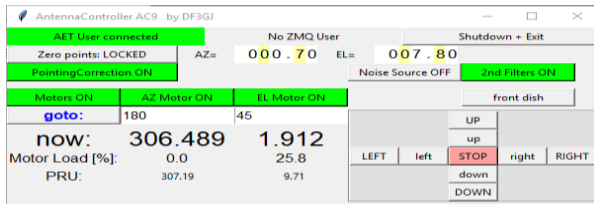
Device Controls

Live Demo

End.

AntennaController AC9

Written in  python™ by Joachim Köppen DF3GJ



AntennaController AC9 by DF3GJ

AET User connected

No ZMQ User

Shutdown + Exit

Zero points: LOCKED

AZ= 000.70 EL= 007.80

PointingCorrection ON

Noise Source OFF

2nd Filters ON

Motors ON

AZ Motor ON

EL Motor ON

goto: 180 45

front dish

UP

up

LEFT left STOP right RIGHT

down

DOWN

now: 306.489 1.912

Motor Load [%]: 0.0 25.8

PRU: 307.19 9.71

In: Current position via ZMQ from Beaglebone, Target pos.

Antenna pointing control

Antenna pointing correction¹

Out: Motor drive via ZMQ-KEB-Gateway to motor inverters

¹J. Köppen 2014-11, *Pointing Correction for the DL0SHF 24 GHz Antenna* in www.dl0shf.de → Technische Berichte

<https://sat-sh.lernnetz.de/pdf/PointingCorrection24GHz.pdf>

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

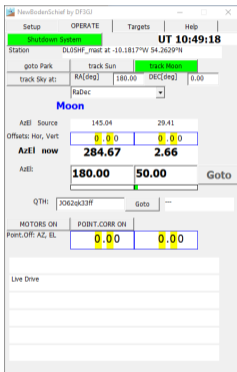
Device Controls

Live Demo

End.

Data source 'NewBodenSchief'

Written in C++ by Joachim Köppen DF3GJ



Radio astronomy targeting

Amateur radio pointing (astro objects, beacons, QTH-locators)

sends position-to-track to "AntennaControl AC9"

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

Amplifier control panel for 1 GHz PA

Hardware solution:

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

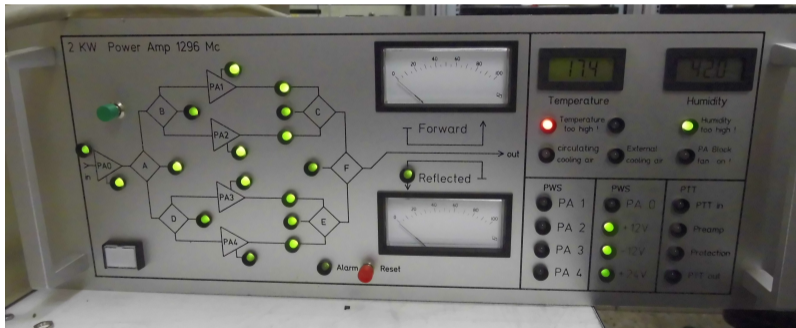
Device Controls

Live Demo

End.

Amplifier control panel for 1 GHz PA

Hardware solution:



- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
 - Python
 - ZMQ
 - Antenna Control
 - Amplifier Control**
 - Device Controls
- Live Demo
- End.

1 GHz Amplifier GPIOs

Beaglebone GPIO data (via ZMQ):

```

{"Action":
  {"Alarm_Reset": {
    "_active": 1, "_duration": 0.5, "_initial": 0,
    "_type": "Pulse", "val": 0},
    "Flex_6500_on": 0,
    "PA_ein": 0,
    "PTT_Test": {"_active": 1, "_addend": 820265503, "_initial": 0,
      "_key": 1128659581, "_modulus": 1327218127, "_timeout":
      0.5, "_type": "Momentary", "val": 0},
    "Spare_1": 0,
    "Spare_2": 0,
    "Spare_3": 0,
    "Spare_4": 0},
    "Analog_Input": {
      "Feuchte": {"_precision": 4, "_unit": "V", "val": "0.9499"},
      "PA_FWD": {"_precision": 4, "_unit": "V", "val": "1.594"},
      "PA_RWD": {"_precision": 4, "_unit": "V", "val": "1.716"},
      "Temperatur": {"_precision": 4, "_unit": "V", "val": "1.236"},
      "Treiber_FWD": {"_precision": 4, "_unit": "V", "val": "1.661"}
    }
  },

```

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

18/28

- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
- Python
- ZMQ
- Antenna Control
- Amplifier Control
- Device Controls
- Live Demo
- End.
- 19/28

Simple webbrowser interface:

pa-1ghz-beagle.per-dudek.de/gpio.html
120% ☆ 🏠 ↻ 🔍 Search

GPIOs of PA-1GHz-Beagle

Sensors:

0	Combiner A Alarm	1	PWS +12V OK
0	Combiner B Alarm	1	PWS +24V OK
0	Combiner C Alarm	1	PWS -12V OK
0	Combiner D Alarm	1	PWS PA0 OK
0	Combiner E Alarm	1	PWS PA1 OK
0	Combiner F Alarm	1	PWS PA2 OK
0	Feuchte Alarm	1	PWS PA3 OK
0	Haupt Alarm	1	PWS PA4 OK
0	Input PA0 Alarm	0	SWR Alarm
0	Kühlung Stufe 1 (Pum	0	Spare 1
0	Kühlung Stufe 2	0	Spare 2
0	Kühlung Stufe 3	0	Spare 3
0	Kühlung Stufe 4	0	Temperatur Alarm
1	PA an	0	Zirkulator PA1 Alarm
0	PTT hinter Preamp	0	Zirkulator PA2 Alarm
0	PTT hinter Protection	0	Zirkulator PA3 Alarm
0	PTT in	0	Zirkulator PA4 Alarm
0	PTT out	0	Überstrom PA0 Alarm
		0	Überstrom PA1 Alarm
		0	Überstrom PA2 Alarm
		0	Überstrom PA3 Alarm
		0	Überstrom PA4 Alarm

Actions:

0	Pulse	Alarm Reset
0	Toggle	Flex 6500 on
1	Toggle	PA ein
0	Push	PTT Test
0	Toggle	Spare 1
0	Toggle	Spare 2
0	Toggle	Spare 3
0	Toggle	Spare 4

Analog Inputs:

0.9815 V	Feuchte
1.555 V	PA FWD
1.695 V	PA RWD
0.9393 V	Spare 1
0.9156 V	Spare 2
1.262 V	Temperatur
1.615 V	Treiber FWD

- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
- Python
- ZMQ
- Antenna Control
- Amplifier Control**
- Device Controls
- Live Demo
- End.

Intermediate webbrowser interface:

pa-1ghz-beagle.per-dudek.de/gpio2.html

GPIOs of PA-1GHz-Beagle

Power on/off

1 PA ein

1 PA ist eingeschaltet

0 Flex 6500 ein

Gehäuse

Temperatur

Feuchte

Alarmer

	Combiner	Zirkulator	Überstrom
<input type="checkbox"/> Hauptalarm	A	PA1	PA0
<input type="checkbox"/> SWR	B	PA2	PA1
<input type="checkbox"/> Temperatur	C	PA3	PA2
<input type="checkbox"/> Feuchte	D	PA4	PA3
<input type="checkbox"/> Input PA0	E		PA4
	F		

Kühlung Stufe

0 1 (Pumpe)

0 2

0 3

0 4

Powersupply OK

1	+12V	1	PA0
1	+24V	1	PA1
1	-12V	1	PA2
1		1	PA3
1		1	PA4

PTT

0 PTT Test

0 in

0 hinter Protection

0 hinter Preamp

0 out

Hf Power

<input type="text" value="1.513 V"/>	Treiber FWD
<input type="text" value="1.477 V"/>	PA FWD
<input type="text" value="1.634 V"/>	PA RWD

Spares

0 Input 1	0 <input type="button" value="Toggle"/>	Action 1	<input type="text" value="0.8523 V"/>	AIN_5
0 Input 2	0 <input type="button" value="Toggle"/>	Action 2	<input type="text" value="0.8532 V"/>	AIN_6
0 Input 3	0 <input type="button" value="Toggle"/>	Action 3		
0	0 <input type="button" value="Toggle"/>	Action 4		

Advanced webbrowser interface:



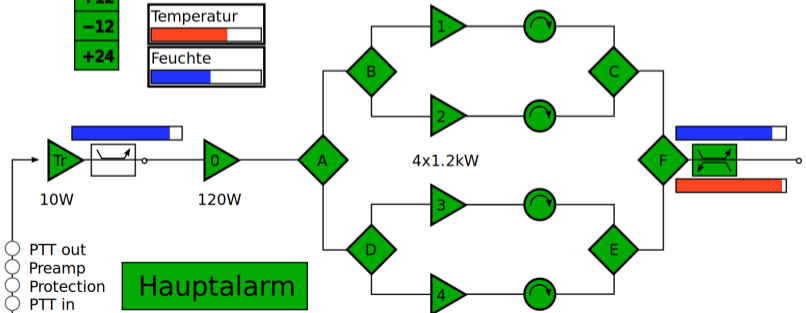
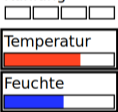
PA control at PA-1GHz-Beagle



Netzteile



Kühlung



- The Station
- From above
- Usage
- Users
- Network

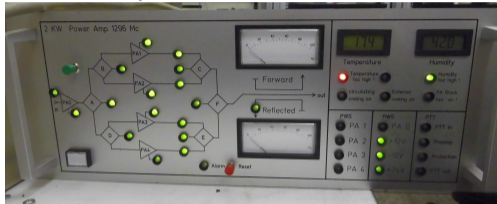
- Hardware
- Antennas
- BeagleBone
- PRU
- GPIO

- Software
- Python
- ZMQ
- Antenna Control
- Amplifier Control**
- Device Controls

- Live Demo
- End.

1 GHz Amplifier Panel

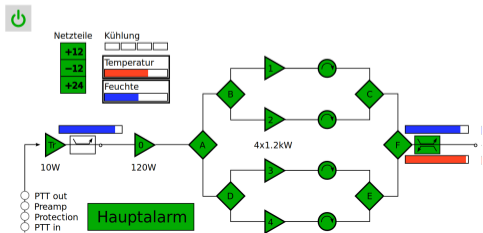
Hardware panel:



Software panel:



PA control at PA-1GHz-Beagle



- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
 - Python
 - ZMQ
 - Antenna Control
 - Amplifier Control
 - Device Controls
- Live Demo
- End.
- 22/28

Controlling other devices

- HF switches on 60m tower (9 bands)

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

Controlling other devices

- HF switches on 60m tower (9 bands)
- 2 masts (28m high) for short wave

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

Controlling other devices

- HF switches on 60m tower (9 bands)
- 2 masts (28m high) for short wave
- 6 cameras (az,el,zoom,focus,iris)

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

Controlling other devices

- HF switches on 60m tower (9 bands)
- 2 masts (28m high) for short wave
- 6 cameras (az,el,zoom,focus,iris)
- Moon beacon <http://moonbeacon.dl0shf.de>

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

Controlling other devices

- HF switches on 60m tower (9 bands)
- 2 masts (28m high) for short wave
- 6 cameras (az,el,zoom,focus,iris)
- Moon beacon <http://moonbeacon.dl0shf.de>
- PA 10 GHz: 50W Transistor + 600W TWT

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

Controlling other devices

- HF switches on 60m tower (9 bands)
- 2 masts (28m high) for short wave
- 6 cameras (az,el,zoom,focus,iris)
- Moon beacon <http://moonbeacon.dl0shf.de>
- PA 10 GHz: 50W Transistor + 600W TWT

Conclusion:

+ WebSocket + JavaScript = versatile

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

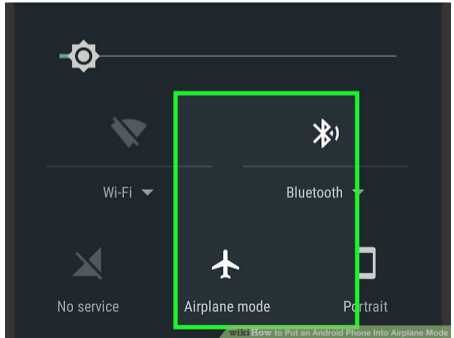
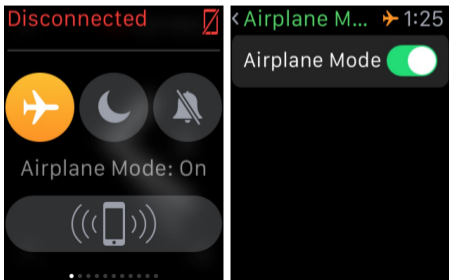
Live Demo

End.

- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
 - Python
 - ZMQ
 - Antenna Control
 - Amplifier Control
 - Device Controls
- Live Demo
- End.

Please turn off all WiFi and mobile data. We need all bandwidth to run our demo. Thank you.

Please turn on "Airplane Mode". Bitte „Flugmodus“ aktivieren.



Thank you for listening.

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

Thank you for listening.

We are willing to share techniques and software.
Please ask.

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

Cpu usage for antenna control

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

26/28

```
root@Spg-1GHz-Beagle:~# uptime
19:54:53 up 314 days, 1:04, 1 user, load average: 0,10, 0,13, 0,10
```

10 months without reboot and counting.

```
root@Spg-1GHz-Beagle:~# cat /sys/devices/system/cpu/cpu0/cpufreq/stats/
time_in_state
300000 2707389289
600000 5774576
720000 11004
800000 6622
1000000 161727
```

Using only 10% of system running at 300 (max: 1000) MHz for > 99.8% of the time

PRU process statistics

How much data to process ?

```
root@Spg-1GHz-Beagle:~# cat /proc/interrupts | grep pruss_evt0
201: 540847970          INTC    20 Level          pruss_evt0
```

540847970 PRU interrupts \approx 20 int/sec for 314 days $\hat{=}$ 10 Hz SSI data rate.

```
root@Spg-1GHz-Beagle:~# ps wuax
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           568  5.3  2.3  35560 11620 ?        S<s1  2017  24395:05
               python3.4 /opt/SsiPru/loader.py -q
```

SsiPru loader/network data pump running for 24395 minutes of 452160 minutes uptime (314 days) == 5.4% cpu time.

==> Python is efficient!

The Station

From above

Usage

Users

Network

Hardware

Antennas

BeagleBone

PRU

GPIO

Software

Python

ZMQ

Antenna Control

Amplifier Control

Device Controls

Live Demo

End.

GPIO process statistics

- The Station
- From above
- Usage
- Users
- Network
- Hardware
- Antennas
- BeagleBone
 - PRU
 - GPIO
- Software
- Python
- ZMQ
- Antenna Control
- Amplifier Control
- Device Controls
- Live Demo
- End.

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	295	0.0	2.7	55012	13944	?	Ssl	2017	327:12	/usr/ bin/python3.4 /opt/LichtTimer/LichtTimer.py -q
root	296	0.1	2.9	56628	15112	?	Ssl	2017	627:50	/usr/ bin/python3.4 /opt/BeagleZmqGpio/WebGpioZmq.py -q
root	445	0.1	2.2	34728	11272	?	Ssl	2017	573:27	/usr/bin/python3.4 /opt/BeagleZmqGpio/BeagleZmqGpio.py
root	316	0.0	0.4	146656	2276	?	Ssl	2017	5:26	/usr/ bin/weborf -x -b /var/www -p 80 -u www

GPIO uses negligible cpu-time, quite a lot of RAM (RSS!)