PC Engines APU platform

User training

(c) 2017 Stanislav Sinyagin
ssinyagin@k-open.com

PC Engines GmbH product line

- **Founded in 1995, moved to Switzerland in 2002**
- **WRAP: 2002-2006**
  - 233 MHz AMD Geode SC1100 CPU, 64/128 MB RAM
- **ALIX: 2005-2011**
  - AMD Geode LX800, 256 MB RAM

PC Engines GmbH product line

- **APU (2014)**
  - AMD G series T40E APU, 1 GHz dual core
  - 2 or 4 GB RAM
  - 3x 1GE NIC (Realtek RTL8111E)
  - MSATA slot
  - Serial console

PC Engines GmbH product line

- **APU2 (2015)**
  - AMD GX-412TC, 1 GHz quad core, 2 or 4 GB RAM
  - MSATA slot
  - 2x Mini-PCIe slots
  - 3x 1GE NICs (Intel i211AT)
  - 1 SIM card slot
  - 1 SATA interface
  - 2x USB 3.0 interfaces
  - Serial console

PC Engines GmbH product line

- **APU3 (2017)**
  - Same active components as in APU2
  - MSATA slot can hold a second LTE/3G modem
  - 2 SIM card slots

APU2/3 CPU

- **Quad-core CPU**
  - Operating speeds: 600, 800, 1000MHz
  - x86_64 architecture
  - AES-NI support
  - 32K data + 32K instruction cache per core
  - shared 2MB L2 cache

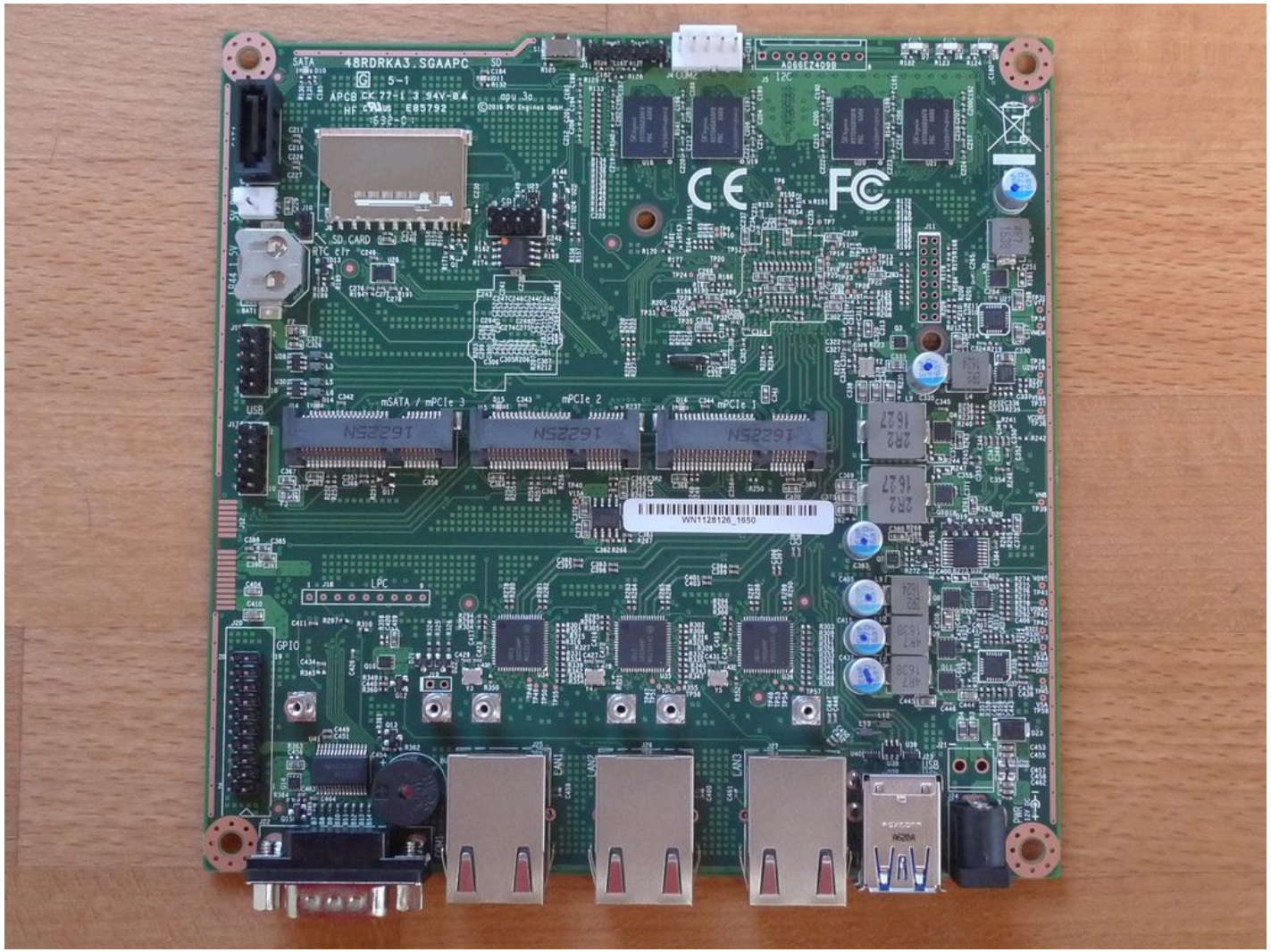
APU2/3 Network ports

- **3x 1-Gigabit NICs (Intel i211AT)**
  - 10/100/1000BASE-T with MDIX
  - VLAN support
  - Jumbo frame support
  - IP checksum offload

APU2/3 Storage options

- **MSATA slot for standard SSD module**
  - The only needed storage in most cases
- **SD card slot**
  - In APU3, the MSATA slot can be occupied by a second 3G/LTE modem
  - Some designs require a read-only partition for OS or hypervisor
- **SATA port and 5VDC for additional HDD or SSD**

APU3 board


SATA
SD
SD CARD
USB
mSATA / mPCIe 3
mPCIe 2
mPCIe 1
LPC
GPIO
LAN1
LAN2
LAN3
USB

APU2/3 standard enclosure

- **Color options: red, black, blue**
  - also silver, but not recommended because of heat radiation efficiency)
- **Two SMA antenna mount holes**
  - option with 6 mount holes available on request
- **Heat dissipation through enclosure (do not stack!)**

Calexium CXM-CASE1BLK

- One or two 2.5” disks
- Holes for additional ports
- No rackmount option
- External power supply

Calexium RackMatrix

- One or two APU boards
- Rack mount
- Internal power supply
- Customizable front panel
- 2.5” disks mount

Extension modules

- Mini-PCIe WiFi (+Bluetooth) adapter, 2 SMA pigtails, 2 SMA antennas
- 3G or LTE Mini-PCIe modem, SMA pigtail, SMA antenna
- MSATA to SATA adapter
- Mini-PCIe to USB3.0 adapter
- Mini-PCIe to SATA adapter

Debian Installer

- github.com/ssinyagin/pcengines-apu-debian-cd
- Build host of the same Debian release as the target ISO
- Internet connection during ISO build
- Physical USB port
- USB thumb drive
- Internet connection during Debian installation

Debian Installer

- **“apu64” profile**
  - The only user input is the host name
  - As much as possible predefined and automated
  - fstrim cronjob for SSD maintenance
- **“manual” profile**
  - Almost nothing is predefined
- **Your own profile**
  - Useful for mass installation

Debian Installer profiles

- **PROFILE.conf**
  - Defines profile name
  - Sets Debian mirror URL for installer
  - Sets country and locale for installer
- **PROFILE.packages**
  - Defines additional Debian packages to be installed

Debian Installer profiles

- **PROFILE.preseed**
  - Defines answers to installer questions
  - Sets network, disk layout, root password
  - Sets country, locale and mirrors for the target system

Debian Installer profiles

- **PROFILE.postinst**
  - Shell script to be launched after the installation
  - Sysctl parameters
  - Cron jobs
  - SSH keys
  - Custom scripts

Preparing installer media

```
./build apu64
dd if=images/debian-9.1-amd64-CD-1.iso of=/dev/sdc bs=16M
```

Installing Debian

- Null-modem cable for serial console
- 115200 baud
- Standard terminal program
- Support for F10 key
- Default boot from USB stick, but F10 and boot intervention might be needed
- Internet and DHCP needed on NIC1
- Installer halts when complete

Alternative ways of installing OS

- Voyage Linux on SD card (http://linux.voyage.hk/)
- Debian on external SATA disk
- Dual-disk software mirror
- Other distributions (RHEL, SmartOS, ...)

3G/LTE modems

- Mini-PCIE full-size
- USB pins for data and control
- SIM card pins
- PCI pins not used
- U.FL(IPEX/IPX) antenna connector(s)

Frequency bands

- **6 regions**
  - North America, Latin America, Europe, Asia, Africa, Oceania
- **UMTS frequency bands (2100 and 900 for Europe, 1900 and 850 for Americas)**
- **LTE frequency bands**
- **Per-country list: www.frequencycheck.com**

Frequency bands (Switzerland)

- GSM: 900, 1800
- UMTS: B1 (2100), B8 (900 GSM)
- LTE: B3 (1800+), B7 (2600), B20 (800 DD)

LTE modem vendors and models

- **Huawei ME909s-120**
- **Sierra Wireless MC7304**
- **SimCom SIM7100E**
  - USB voice interface available

Udev rules

- Problem: USB device numbers assigned randomly
- A modem is represented by 4 or 5 ttyUSB devices
- There might be USB serial adapters which are also ttyUSB devices
- There might be two modems
- Udev rules can be adapted to assign numbers in deterministic manner

Udev rules

- **Simple case: one modem of a known vendor**
  - Matches on subsystem, vendor ID and product ID
  - Automatically creates symlinks /dev/ttyWWANx

```
cat >/etc/udev/rules.d/99-wwan.rules <<'EOT'
# SIMCom SIM7100
SUBSYSTEM=="tty", ATTRS{idVendor}=="1e0e", ATTRS{idProduct}=="9001", SYMLINK+="ttyWWAN%E{ID_USB_INTERFACE_NUM}"
SUBSYSTEM=="net", ATTRS{idVendor}=="1e0e", ATTRS{idProduct}=="9001", NAME="lte0"
EOT
```

Udev rules

- **Difficult case: Two LTE modems in APU3**
  - Modem slots are attached to different USB controllers
  - The new symlink name contains the controller ID

```
cat >/etc/udev/devpath_to_pcislot <<'EOT'
#!/bin/sh echo ${DEVPATH} | sed -r \
    -e 's,^\/[^\/]+\/[^\/]+\/[0-9af]{4}:[0-9af]{2}:,,' \
    -e 's,\/.+,,' -e 's,\.,,g'
EOT

cat >/etc/udev/rules.d/99-wwan.rules <<'EOT'
SUBSYSTEM=="tty", ATTRS{idVendor}=="12d1", ATTRS{idProduct}=="15c1", PROGRAM="/etc/udev/devpath_to_pcislot"
SYMLINK+="ttyWWAN%c{1}_%E{ID_USB_INTERFACE_NUM}"
SUBSYSTEM=="net", ATTRS{idVendor}=="12d1", ATTRS{idProduct}=="15c1", PROGRAM="/etc/udev/devpath_to_pcislot"
NAME="lte%c{1}"
EOT
```

Modems and udev rules: more info

- More details in my blog: **https://txlab.wordpress.com/tag/3g/**
- Sets of udev rules: **https://github.com/ssinyagin/wwan_udev_rules**
- Where to buy modems: **https://techship.com/**
  - Also, aliexpress :-)

Setting up Huawei ME909s-120

- Install picocom for manual commands
- Set up udev rules for ME909s-120 and reboot

```
cat >/etc/udev/rules.d/99-huawei-wwan.rules <<'EOT'
SUBSYSTEM=="tty", ATTRS{idVendor}=="12d1", ATTRS{idProduct}=="15c1", SYMLINK+="ttyWWAN%E{ID_USB_INTERFACE_NUM}"
SUBSYSTEM=="net", ATTRS{idVendor}=="12d1", ATTRS{idProduct}=="15c1", NAME="lte0"
EOT
```

Setting up Huawei ME909s-120

- AT command interface
  `picocom -b 115200 /dev/ttyWWAN02`
- SIM card PIN disabling:
  `AT+CPIN="1111"`
  `AT^CARDMODE`
  `AT+CLCK="SC",0,"1111"`
- Status information:
  `AT^SYSINFOEX`

ME909s-120 connect script

```
cat >/etc/chatscripts/sunrise.HUAWEI <<'EOT'
ABORT BUSY
ABORT 'NO CARRIER'
ABORT ERROR
TIMEOUT 10
'' ATZ
OK 'AT+CFUN=1'
OK 'AT+CMEE=1'
OK 'AT\^NDISDUP=1,1,"internet"'
OK
EOT
```

ME909s-120 disconnect and interface init

```
cat >/etc/chatscripts/gsm_off.HUAWEI <<'EOT'
ABORT ERROR
TIMEOUT 5
'' AT+CFUN=0 OK
EOT

cat >/etc/network/interfaces.d/lte0 <<'EOT'
auto lte0
iface lte0 inet dhcp
    pre-up /usr/sbin/chat -v -f /etc/chatscripts/sunrise.HUAWEI >/dev/ttyWWAN02 </dev/ttyWWAN02
    post-down /usr/sbin/chat -v -f /etc/chatscripts/gsm_off.HUAWEI >/dev/ttyWWAN02 </dev/ttyWWAN02
EOT
```

OpenVPN optimized for mobile internet

```
port 1194
proto udp
fragment 1300
mssfix
dev tun
topology subnet
keepalive 10 120
```

LXC: light-weight containers

- Kernel namespaces
- By default, no CPU and memory restrictions
- Container processes are allowed to take high priority
- Virtual bridge can be used for inter-container communication
- Physical interface can be dedicated to a container

LXC flavors

- Debian way: do-it-yourself
- Ubuntu way: bridge and NAT masquerading preconfigured
- Docker way: predefined network, process priority restrictions

LXC example: network probe

- LTE modem and OpenVPN client in main system
- A NIC dedicated to the probe container
- No internal bridge for security reasons
- Probe has its own DHCP client and routing table
- Probe performs network tests or remote management, and saves data to the local disk
- A process in main system picks up the data from the disk and sends it to the management system

LXC example: SIP SBC

- Two independent containers, each having a dedicated NIC, and an internal bridge for communicating between containers
- Each container running a SIP and RTP proxy
- Secure separation between two SIP domains
- No direct IP communication between domains

Conclusion

- Universal appliance platform
- Extension flexibility
- Good performance
- Zero moving parts