The universal PCI Express Device Driver for MTCA.4

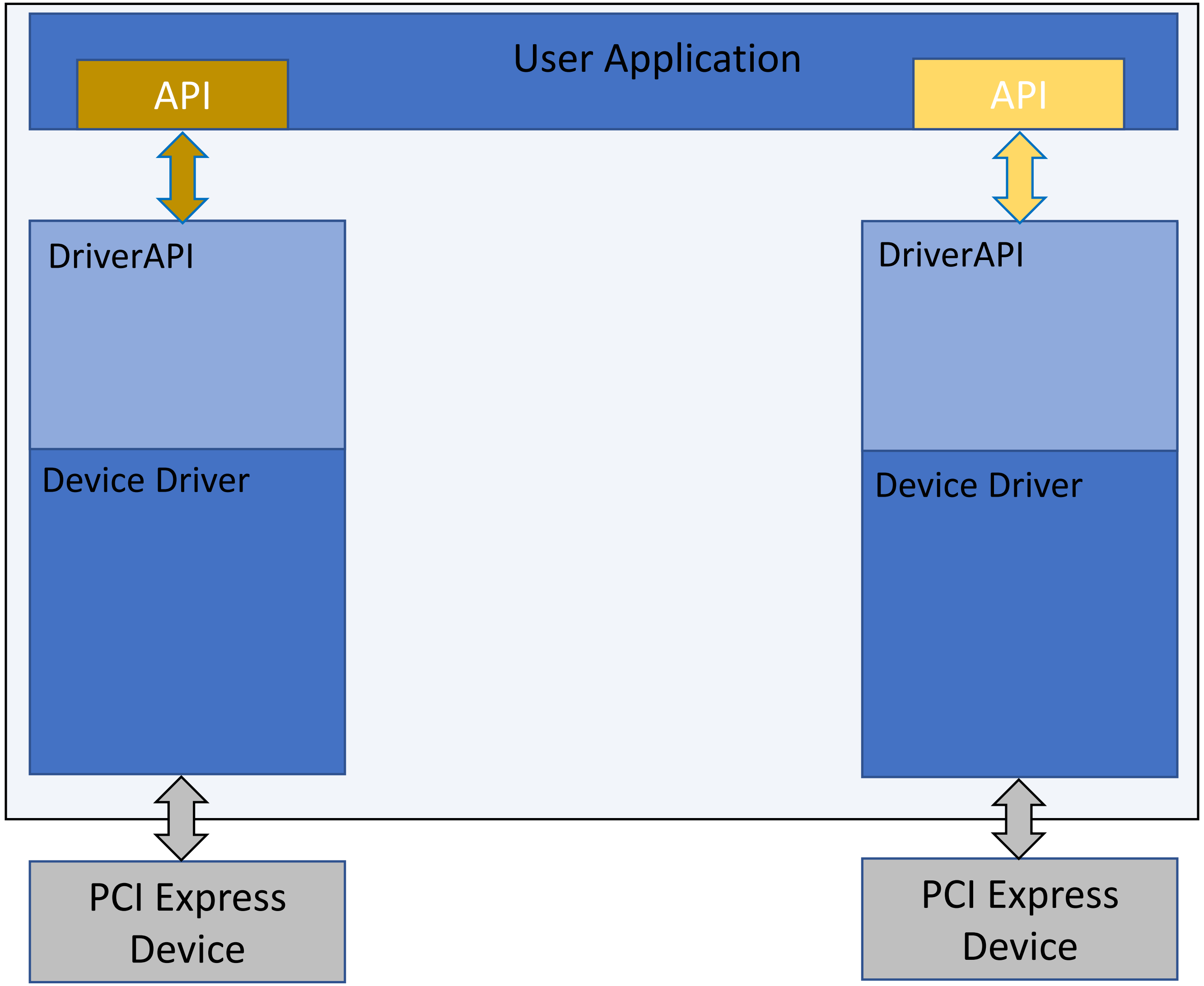
L.Petrosyan

The Universal PCI Express Device Driver

- The PCI Express standard is currently the most widely used architecture.
- The MTCA uses the PCI Express as a central bus of data transmissions.
- In order to take full advantage of PCI Express enhanced features,
  - Hot Plug
  - High Transfer Rate
  - Well defined Interrupts Handling
  - DMA
  - Point to Point transactions
  - Non Transparent Bridging
    - Have a redundant CPU
    - Add second CPU in to existing PCE System
  - …
- More robust device drivers are required.

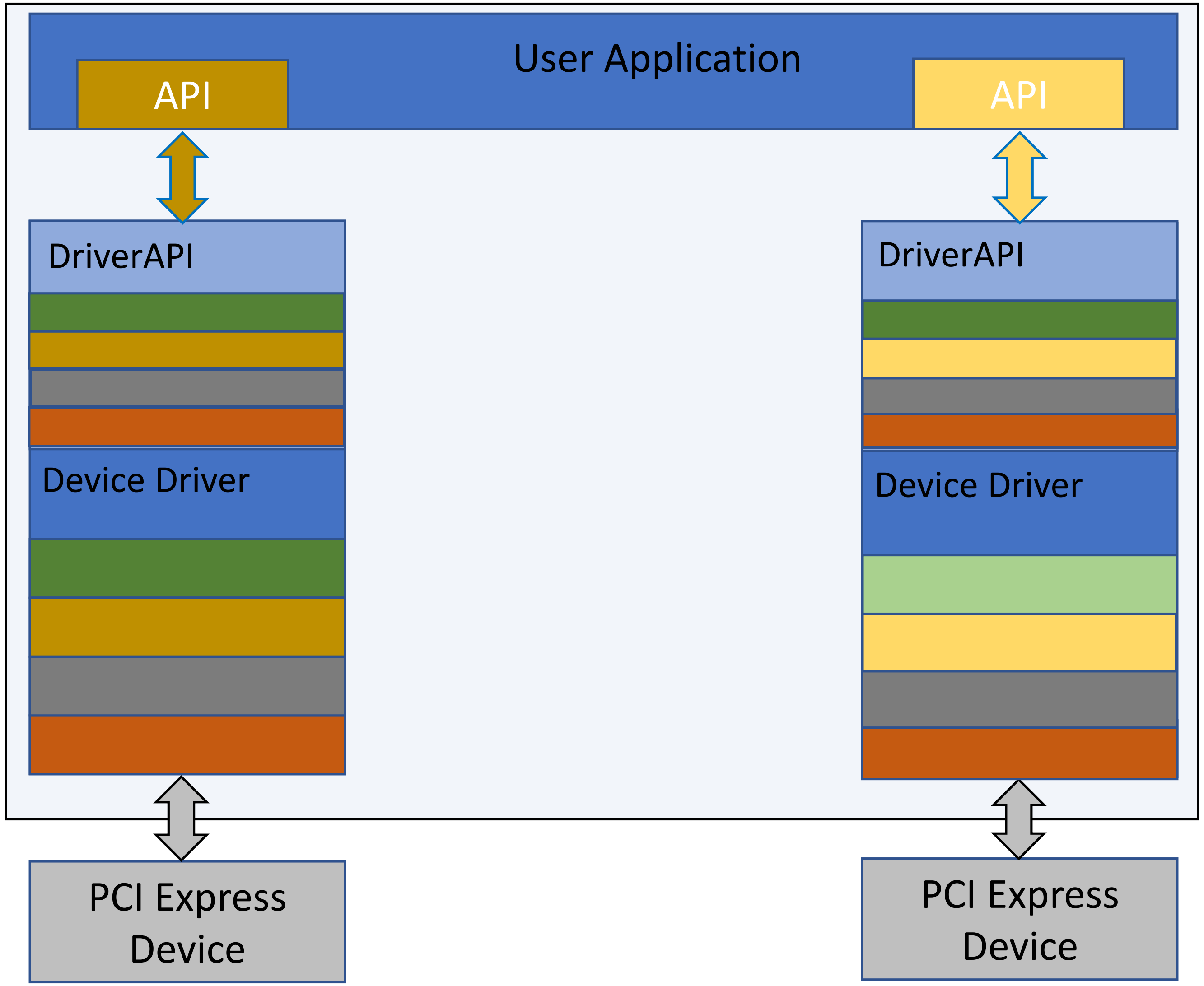
The Universal PCI Express Device Driver

- A Device Driver is loadable kernel Module that provides access to the particular device attached to a computer (PCI Express Bus)
- User Application use the Device Driver API to access the Device
- Over the time the support of the increasing number of Device Drivers from different providers is becoming increasingly difficult
  - More Devices -> More Drivers
  - Different Drivers -> different APIs



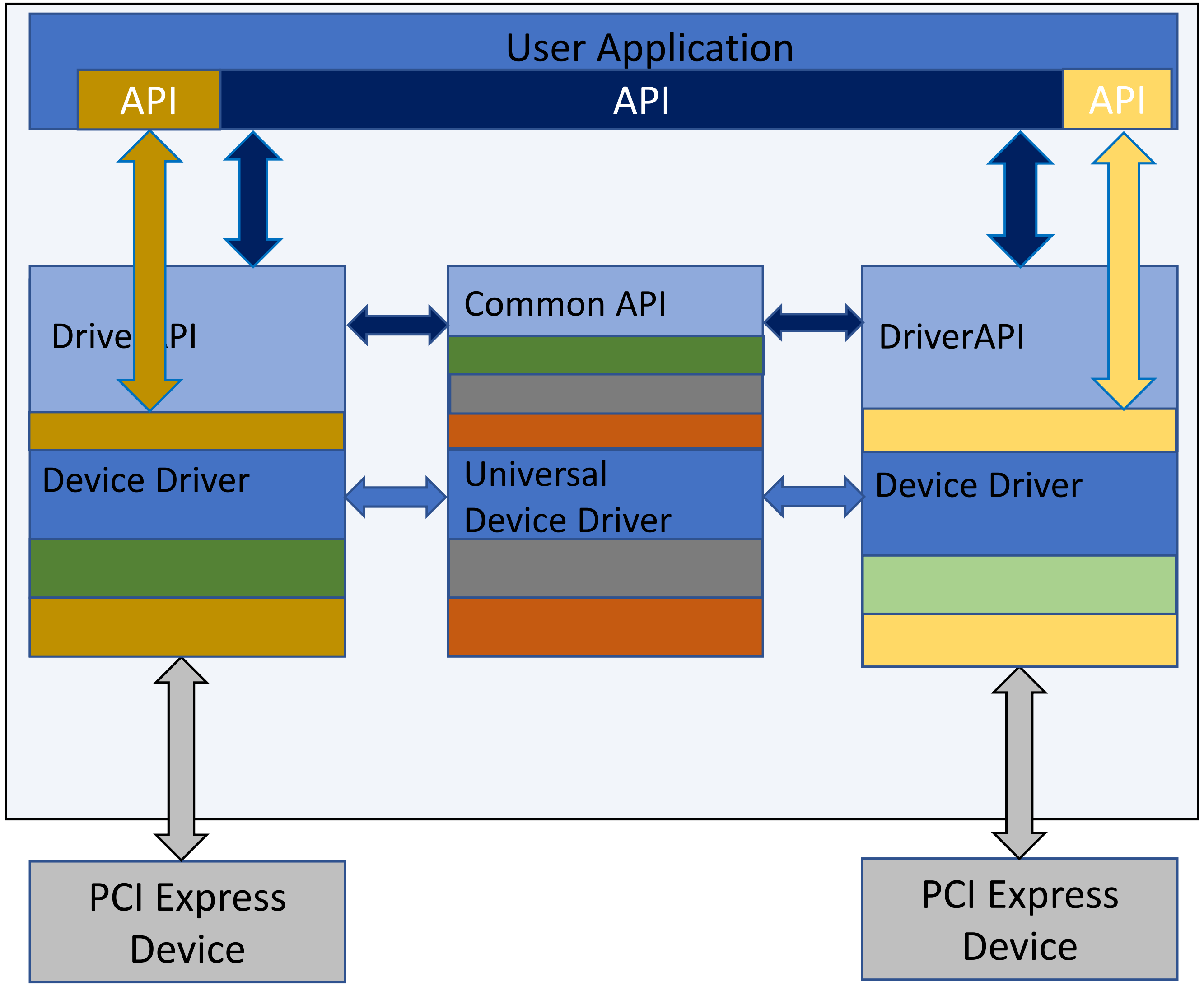
The Universal PCI Express Device Driver

- Basic PCI Express functionality
  - Mapping memories
  - **Read**, **write** and some common **ioctl**
  - Error handling
  - Hot Plug
- Standards or Guidlines functionalty
  - Standard Registers Set
  - SHAPI Registers Set (PICMG)
  - PICMG Standard Device Model
  - PCIe HotPlug functionality
- Device specific but has common API
  - DMA
- Device specific functionality



The Universal PCI Express Device Driver

- Split Device Driver into two parts follow the Linux Device Driver stacking Model
- Add all common functionality and API into universal part
  - Basic PCI Express functionality
  - Standards or Guidlines functionalty
- Add Common API for Device specific functionality into universal part but keep functionality in Device Driver side
  - Device specific but has common API

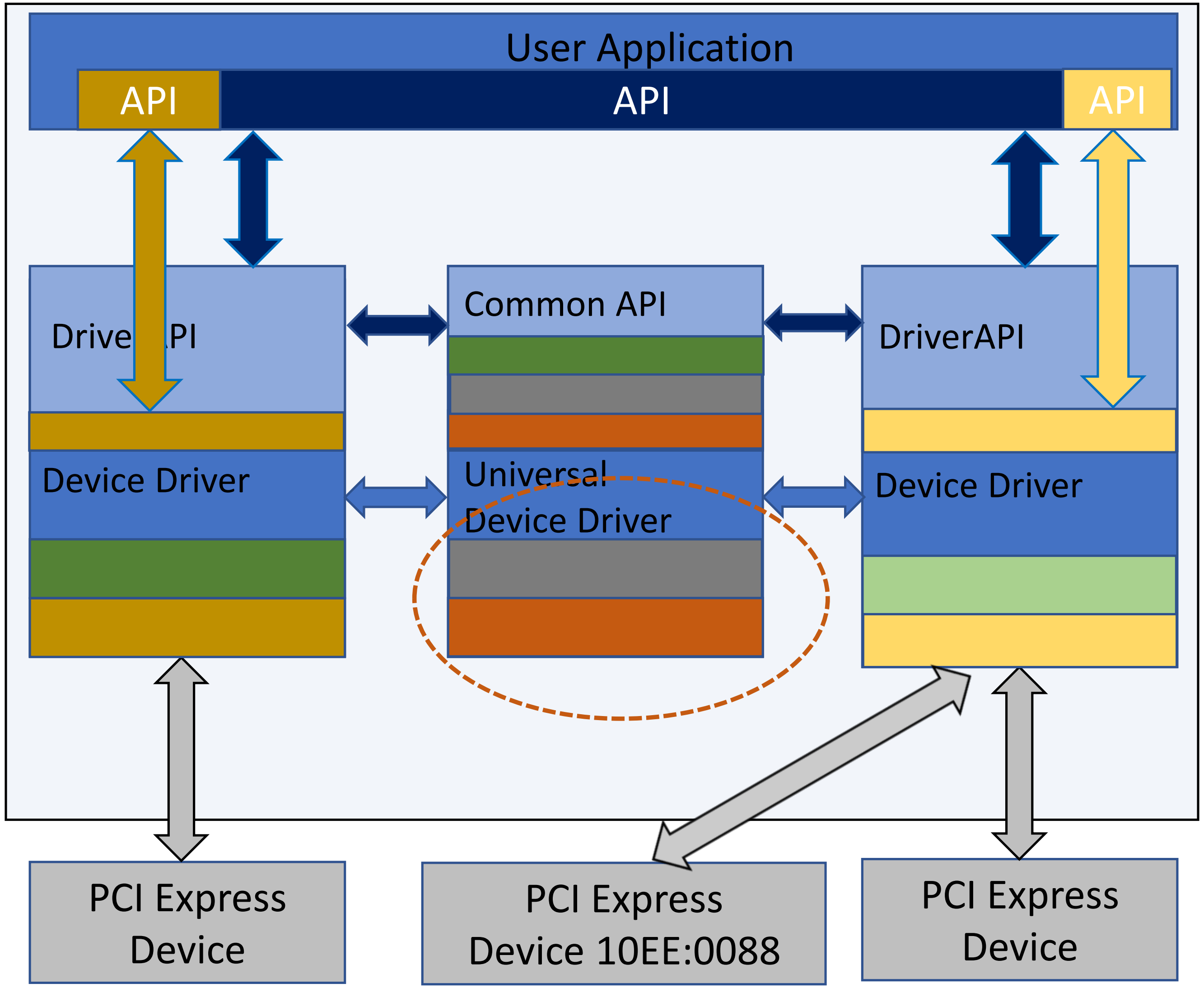


The Universal PCI Express Device Driver

- this approach facilitates creation of new drivers and user applications
- The Device Driver created on top of the ***universal*** driver has all necessary PCI Express functionalty
- It could be binded to any PCI Express Device

echo „10EE:0088“ > /sys/bus/devices/xxx/driver/new_id

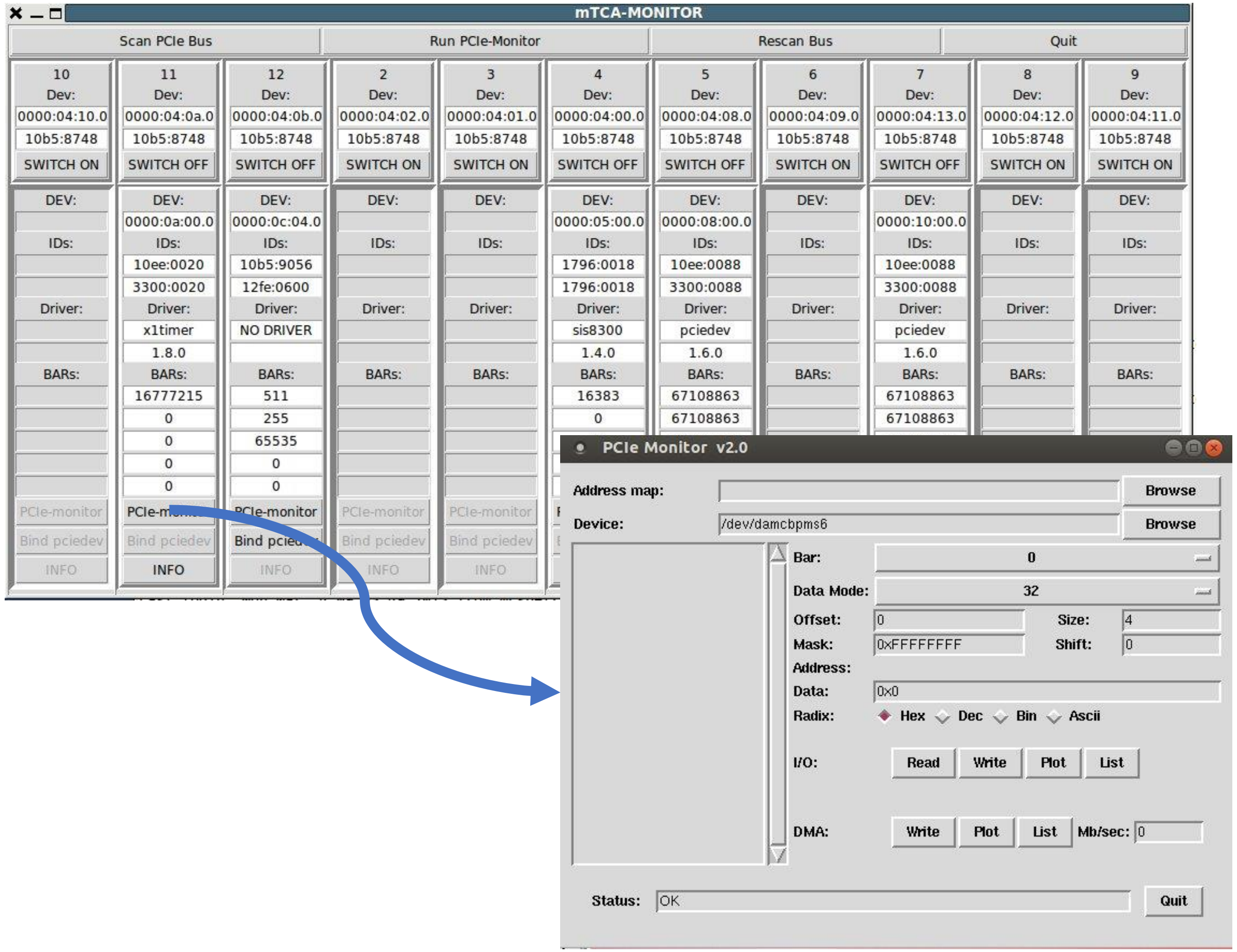
facilitates integration of new devices into the existing software



The Universal PCI Express Device Driver

Example: how to use

- Create PCIEDEV Device driver
  - Used for development
  - Used to bind to any new device
- PCIEDUMMY used as template to create new drivers
- Create test user application (mtcamonitor)
- Create Device Driver based on UPCIEDEV for each Device
- Mtcamonitor uses common API, could have access to any device (works with any driver based on UPCIEDEV)
- PCIEDEV could be binded to any new device, use Mtcamonitor to test any new device



mTCA-MONITOR

Scan PCIe Bus | Run PCIe-Monitor | Rescan Bus | Quit

10	11	12	2	3	4	5	6	7	8	9
Dev:	Dev:	Dev:	Dev:	Dev:	Dev:	Dev:	Dev:	Dev:	Dev:	Dev:
0000:04:10.0	0000:04:0a.0	0000:04:0b.0	0000:04:02.0	0000:04:01.0	0000:04:00.0	0000:04:08.0	0000:04:09.0	0000:04:13.0	0000:04:12.0	0000:04:11.0
10b5:8748	10b5:8748	10b5:8748	10b5:8748	10b5:8748	10b5:8748	10b5:8748	10b5:8748	10b5:8748	10b5:8748	10b5:8748
SWITCH ON	SWITCH OFF	SWITCH OFF	SWITCH ON	SWITCH ON	SWITCH OFF	SWITCH OFF	SWITCH ON	SWITCH OFF	SWITCH ON	SWITCH ON
DEV:	DEV:	DEV:	DEV:	DEV:	DEV:	DEV:	DEV:	DEV:	DEV:	DEV:
	0000:0a:00.0	0000:0c:04.0			0000:05:00.0	0000:08:00.0		0000:10:00.0		
IDs:	IDs:	IDs:	IDs:	IDs:	IDs:	IDs:	IDs:	IDs:	IDs:	IDs:
	10ee:0020	10b5:9056			1796:0018	10ee:0088		10ee:0088		
	3300:0020	12fe:0600			1796:0018	3300:0088		3300:0088		
Driver:	Driver:	Driver:	Driver:	Driver:	Driver:	Driver:	Driver:	Driver:	Driver:	Driver:
	x1timer	NO DRIVER			sis8300	pciedev		pciedev		
	1.8.0				1.4.0	1.6.0		1.6.0		
BARs:	BARs:	BARs:	BARs:	BARs:	BARs:	BARs:	BARs:	BARs:	BARs:	BARs:
	16777215	511			16383	67108863		67108863		
	0	255			0	67108863		67108863		
	0	65535								
	0	0								
	0	0								
PCIe-monitor	PCIe-monitor	PCIe-monitor	PCIe-monitor	PCIe-monitor						
Bind pciedev	Bind pciedev	Bind pciedev	Bind pciedev	Bind pciedev						
INFO	INFO	INFO	INFO	INFO						

PCIe Monitor v2.0

Address map: [Browse]

Device: /dev/damcbpms6 [Browse]

Bar: 0

Data Mode: 32

Offset: 0 Size: 4

Mask: 0xFFFFFFFF Shift: 0

Address:

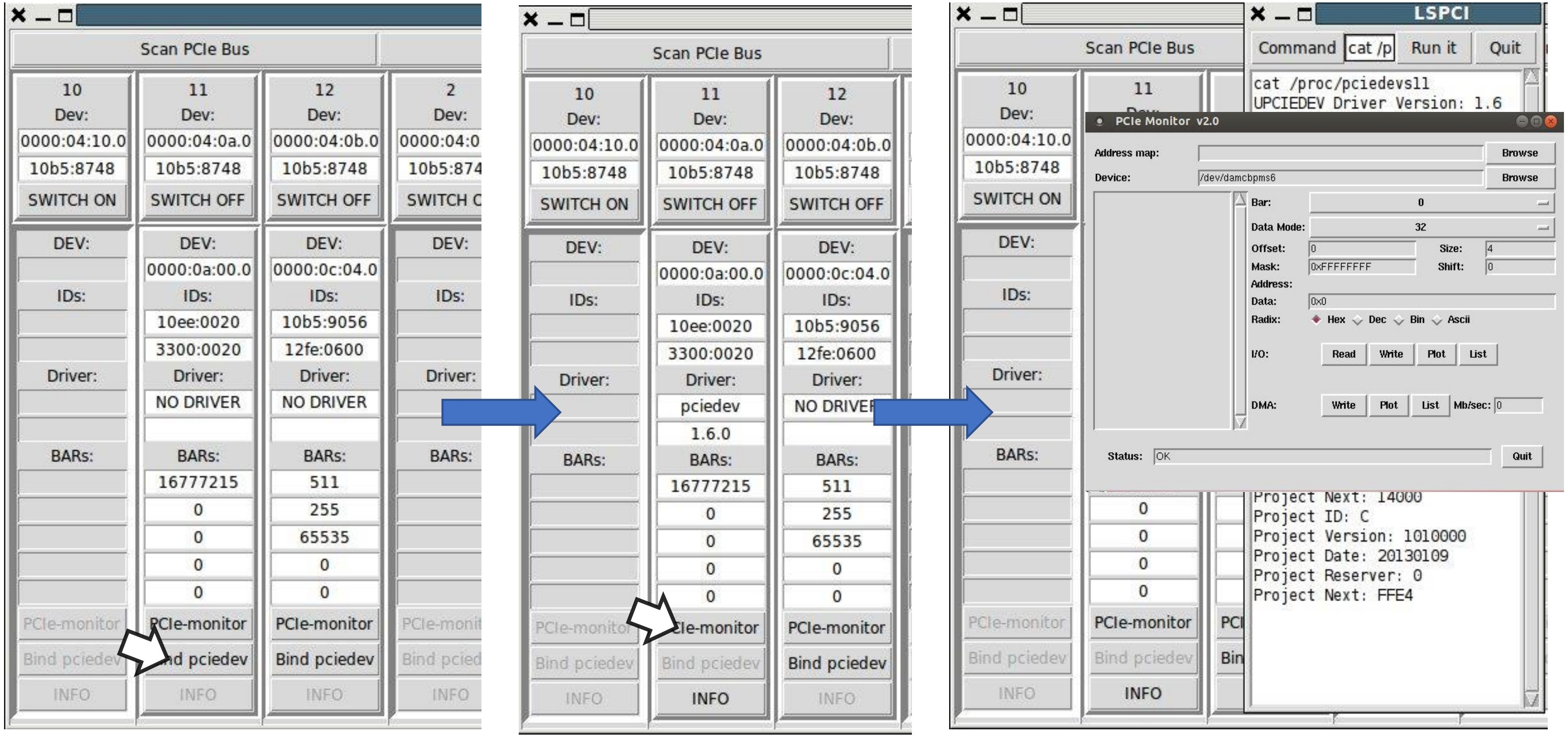
Data: 0x0

Radix: ● Hex ◇ Dec ◇ Bin ◇ Ascii

Mb/sec: 0

Status: OK

The Universal PCI Express Device Driver



Scan PCIe Bus

10	11	12	2
Dev:	Dev:	Dev:	Dev:
0000:04:10.0	0000:04:0a.0	0000:04:0b.0	0000:04:0
10b5:8748	10b5:8748	10b5:8748	10b5:874
SWITCH ON	SWITCH OFF	SWITCH OFF	SWITCH C
DEV:	DEV:	DEV:	DEV:
	0000:0a:00.0	0000:0c:04.0	
IDs:	IDs:	IDs:	IDs:
	10ee:0020	10b5:9056	
	3300:0020	12fe:0600	
Driver:	Driver:	Driver:	Driver:
	NO DRIVER	NO DRIVER	
BARs:	BARs:	BARs:	BARs:
	16777215	511	
	0	255	
	0	65535	
	0	0	
	0	0	
PCIe-monitor	PCIe-monitor	PCIe-monitor	PCIe-monit
Bind pciedev	Bind pciedev	Bind pciedev	Bind pcied
INFO	INFO	INFO	INFO

Scan PCIe Bus

10	11	12
Dev:	Dev:	Dev:
0000:04:10.0	0000:04:0a.0	0000:04:0b.0
10b5:8748	10b5:8748	10b5:8748
SWITCH ON	SWITCH OFF	SWITCH OFF
DEV:	DEV:	DEV:
	0000:0a:00.0	0000:0c:04.0
IDs:	IDs:	IDs:
	10ee:0020	10b5:9056
	3300:0020	12fe:0600
Driver:	Driver:	Driver:
	pciedev	NO DRIVER
	1.6.0	
BARs:	BARs:	BARs:
	16777215	511
	0	255
	0	65535
	0	0
	0	0
PCIe-monitor	PCIe-monitor	PCIe-monitor
Bind pciedev	Bind pciedev	Bind pciedev
INFO	INFO	INFO

Scan PCIe Bus

10	11
Dev:	Dev:
0000:04:10.0	
10b5:8748	
SWITCH ON	
DEV:	
IDs:	
Driver:	
BARs:	
	0
	0
	0
	0
PCIe-monitor	PCIe-monitor
Bind pciedev	Bind pciedev
INFO	INFO

PCIe Monitor v2.0

Address map: Browse

Device: /dev/damcbpms6 Browse

Bar: 0

Data Mode: 32

Offset: 0 Size: 4

Mask: 0xFFFFFFFF Shift: 0

Address:

Data: 0x0

Radix: ● Hex ○ Dec ○ Bin ○ Ascii

I/O: Read Write Plot List

DMA: Write Plot List Mb/sec: 0

Status: OK Quit

LSPCI

Command cat /p Run it Quit

```
cat /proc/pciedevs11
UPCIEDEV Driver Version: 1.6
```

```
Project Next: 14000
Project ID: C
Project Version: 1010000
Project Date: 20130109
Project Reserver: 0
Project Next: FFE4
```

Bind pciedev to your AMC card

- *echo ‘VENDOR_ID DEVICE_ID’ > /sys/bus/pci/drivers/pciedev/new_id*
- Check */dev* directory, has to be */dev/pciedevsX,* X is slot number where You have your AMC

The Universal PCI Express Device Driver

Ubuntu and Debian 10 packages ***http://doocs.desy.de***

1. *#receive the DESY DOOCS key*
   - *Wget –O – http://doocs.desy.de/pub/doocs/DOOCS-key.gpg.asc | sudo apt-key add*
2. *#add the DOOCD repository:*
   - *Sudo sh –c ‘echo “deb http://doocs.desy.de/pub/doocs ‘lsb_release -sc main” > /etc/apt/sources.list.d/doocs.list*
3. *apt-get update*
4. *apt-get install upciedev-dkms*
5. *apt-get install pciedev-dkms*
6. *apt-get install pciedummy-dkms*
7. *apt-get install doocs-mtca-tools* (optional)
   - To run call *mtcamonitor*
8. The source directory of the drivers in /usr/src

There is an example c++ application in the driver source directory.
Use PCIEDEV as a template to crate your own device driver.

The Universal PCI Express Device Driver

Source codes https://github.com/MicroTCA

1. Download upciedev , pciedev amd pciedummy
2. Compile the drivers, just run *make*
3. In the source directory there is the file ./*sc*
4. Run ./*sc* it will copy all necessary files to */lib/modules/…*
5. *depmod –a*
6. *modeprobe upciedev*
7. *modprobe pciedev*
8. Download mtcamonitor (optional)
    1. Run make
    2. Run mtcamonitor

There is an example c++ application in the driver source directory.
Use PCIEDEV as a template to crate your own device driver.

The Universal PCI Express Device Driver

Crate new PCIe device driver based on upciedev, using pciedummy as template.

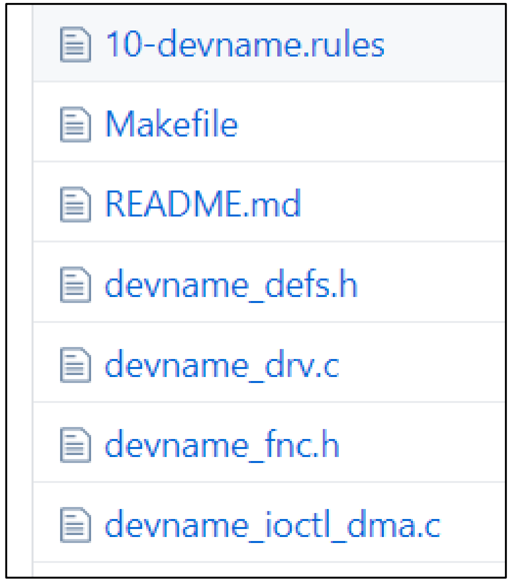
To crate the new driver (*as an example, we want to crate the diver for sis81600*):

- change all file names from devname* to your "device name“: *mv ./devname* ./sis8160**
- make changes in Makefile.

In all files make Replace:

- DEVNAME to SIS8160
- devname to sis8160
- change in devname_fhc.h:
  - #define DEVNAME "devname“ to “sis8160”
  - #define DEVNAME_VENDOR_ID 0x10EE *// XILINX vendor ID* to 0x1796 *// STRUCK Vendor ID*
  - #define DEVNAME_DEVICE_ID 0x0088 to 0x0028 *//sis8160 Device ID*
- run "./sc" as root
- run "depmod -a" as root
- install 10-sis8160.rules in /etc/udev/rules.d/
- run “modprobe sis8160” as root
  - the driver will crate device file /dev/sis8160sX X-is a slot number where You have your device

compile, install and enjoy



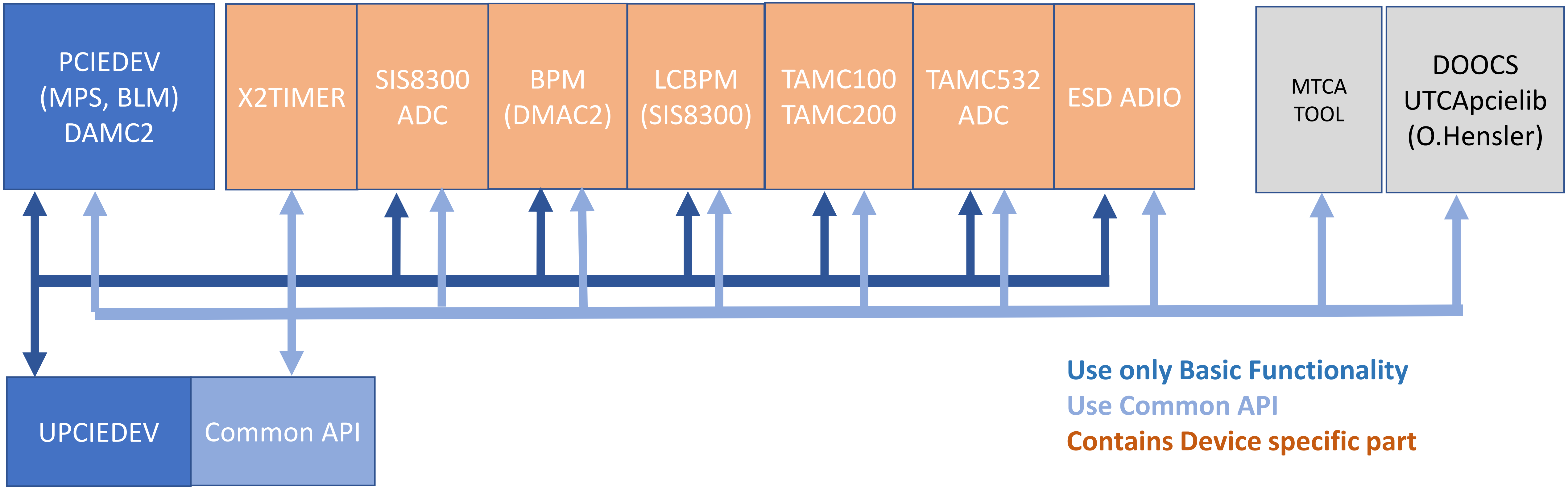
10-devname.rules
Makefile
README.md
devname_defs.h
devname_drv.c
devname_fnc.h
devname_ioctl_dma.c

sis8160 – DEVNAME
s – slot
X – Slot number

The Universal PCI Express Device Driver

Status

- The following drivers, tools and libraries are developed used
- PCIE HotPlug
- PCIE Non Transparent Bridging
- PCIE Point to Point transactions



Use only Basic Functionality

Use Common API

Contains Device specific part

The Universal PCI Express Device Driver

THANK YOU

- The source codes can be found on https://github.com/MicroTCA/
  - ***Any contributions and new developers are welcome!!!***
- The information and Linux packages can be found on a DOOCS web page ***http://doocs.desy.de***
  - *Device driver packages have no dependecies on DOOCS*
- Mail ***doocs@desy.de***