PHYTEC Messtechnik GmbH

Application Note

How to use phyCAM camera modules with phyCORE-i.MX 8M Mini SBC on phyBOARD Polis

Revision History

Version	Changes	Author	Date
A0	Initial Release	H. Fendrich	02.03.2021
A1	Insert chapter 1 and 2	H. Fendrich	31.03.2021

Content

Support Hotline Europe:: +49 (6131) 9221-31 ♦http://www.phytec.de
Support Hotline North America: 1-800-278-9913 ♦http://www.phytec.com

1 Introduction

The SBC kits (Single-Board-Computer) from PHYTEC represent an ideal basis for testing and the first design steps with phyCAM camera modules. The kit forms the starting point for your development project. The kit can be used to develop and test application software before your own adapted hardware is available. It is also an optimal platform for testing special hardware extensions before you create a complex application board.

This optional step secures your design and at the same time provides software development with an opportunity to adapt newly added hardware components. Extensions can be easily connected via the extension interfaces of the baseboards included in the kit. In the simplest case, a freely wired hole grid structure is sufficient.

For certain combinations of controller modules, camera modules, and lenses, we offer ready-made kits. If the combination you require is not available, we recommend that you first put a similarly-equipped kit into operation and then connect the camera you require to the kit.

We will be happy to advise you on the best way to carry out your project.

2 SBC Kit Start-Up

Before you try out the camera and image processing functions, we recommend that you first familiarize yourself with the basic functions of the kit and the associated development environment. Each kit contains a Quick Start Guide that guides you through installing the software on your development PC and configuring the hardware.

2.1 Connecting and Starting the Camera

Note: Depending on the kernel version, the software installation may differ in some details from the described procedure.
When you purchase an SBC kit, you will receive the PHYTEC guarantee. If you encounter any difficulties during installation, PHYTEC support will help you!

Before You Start

The following components are required for the camera setup:

- PHYTEC SBC-Kit with power supply and serial connection (RS232 or USB)
- Camera module, with camera cable, lens holder, and lens

If you have ordered the corresponding kit as an Embedded Imaging Kit, all hardware components required for installation are already included.

As a rule, both the extension of an existing kit by one camera and the extension of an existing video kit by additional cameras is possible without problems with an existing phyCAM interface.

Before connecting and starting up the camera, carry out the first steps of the "Quick Start Instructions" of the respective kit. To start the scripts on the module, follow the steps to establish a serial connection with the module. You can use the terminal program suggested in the manual or your own terminal program.

To transfer saved image files to a PC (host) via Ethernet, follow the steps to establish an FTP connection. You can use the FTP server suggested in the manual or your own FTP server.

3 Overview

The i.MX 8M Mini Microcontroller supported one MIPI CSI-2 camera interfaces (see figure 1).

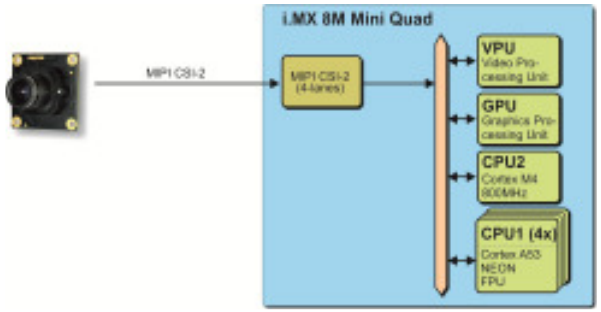

figure 1: Block Diagram Camera Interfaces of i.MX 8M Mini Controller (Quad)

On the phyCORE-i.MX 8M Mini the CSI1/CAM1 camera path go out as CSI-2 MIPI signal.

Support Hotline Europe:: +49 (6131) 9221-31 ♦http://www.phytec.de
Support Hotline North America: 1-800-278-9913 ♦http://www.phytec.com

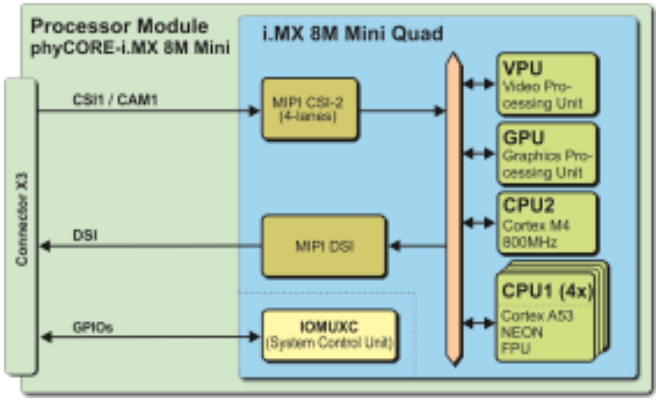

figure 2: CAM_1 (CSI1) present on connector X3

On the PHYTEC or customer carrier boards can the interfaces are led out as phyCAM-M. For more information to phyCAM-M see the phyCAM-Manuals on PHYTEC Homepage.
On the phyBOARD-Polis baseboard is the camera interfaces led out as phyCAM-M (MIPI CSI-2) interfaces (see figure 3). Here you can connect different phyCAM-M camera modules.

Support Hotline Europe:: +49 (6131) 9221-31 ♦http://www.phytec.de
Support Hotline North America: 1-800-278-9913 ♦http://www.phytec.com

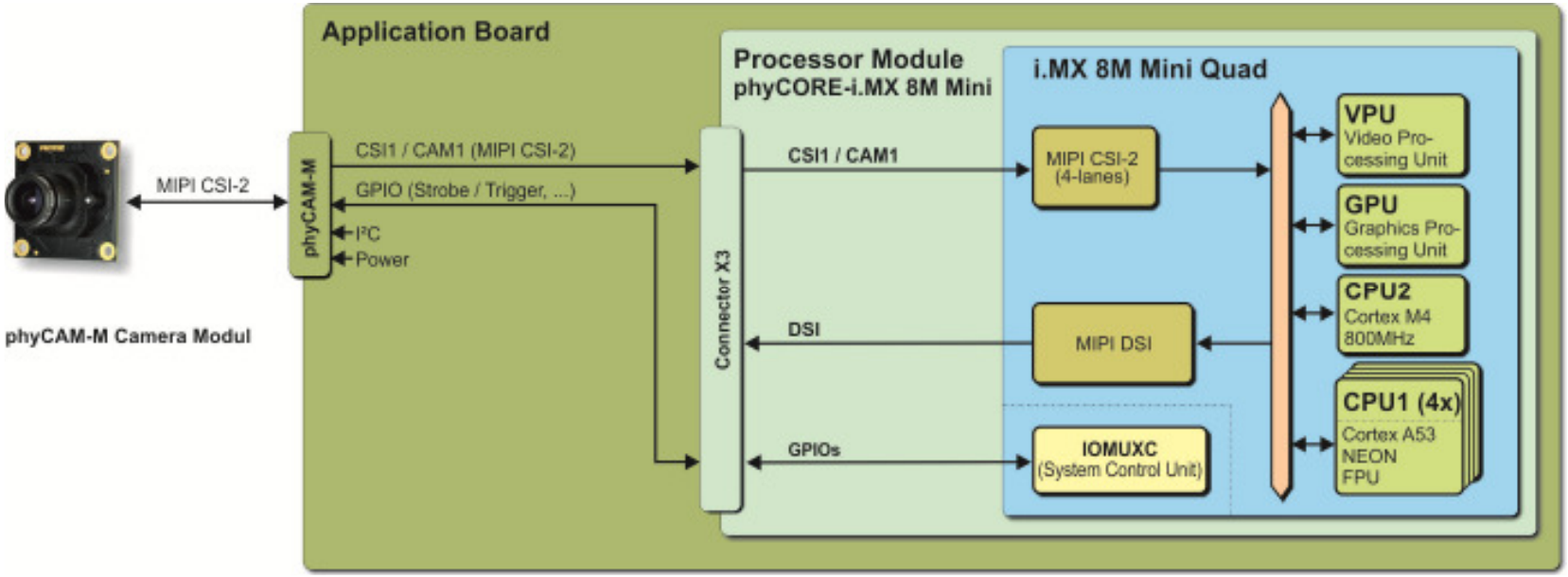

figure 3: Block Diagram of phyCAM-M Camera Interfaces of phyCORE-i.MX 8M Mini (Quad) and the go out on the phyBOARD-Polis-i.MX 8M Mini - SBC

The BSP shipped with the Kit includes already the software drivers for the supported phyCAM-M camera modules. The drivers are compatible with v4l2. Also GStreamer scripts are included for the evaluation of the camera modules. If you need the camera interface to connecting your own camera module, is an adapter to phyCAM-M necessary.

Support Hotline Europe:: +49 (6131) 9221-31 ♦http://www.phytec.de
Support Hotline North America: 1-800-278-9913 ♦http://www.phytec.com

4 Camera Connectors on the Polis - Carrier Boards

The development kits for the phyBOARD-Polis-i.MX 8M Mini contain:
- one carrier board (Polaris)
- one phyCORE-i.MX 8M modul SOM

The phyCORE-i.MX 8M Mini is direct soldering on carrier board.

On the base board Polis (PB-02820) we convert the MIPI CSI-2 interfaces in phyCAM-M standard.
- phyCAM-M camera interface 1 (support on connector X10)

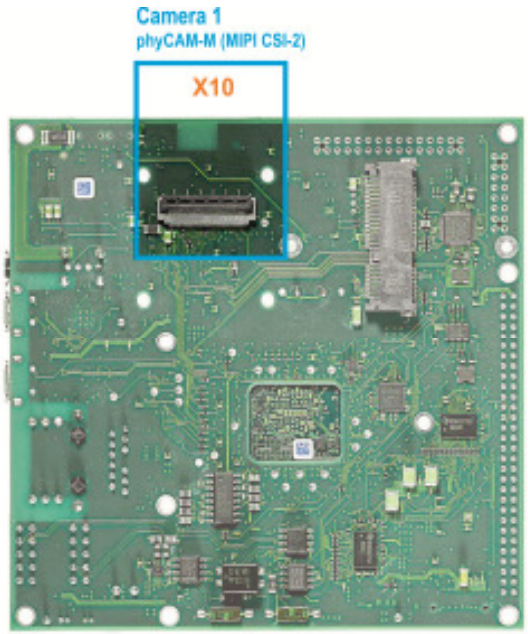

figure 4: Camera Interfaces on phyBOARD-Polis up PCB Version PL1532.1 for the phyCORE-i.MX 8M Mini SOM

Note: If you use the phyCAM-M interface, use an FFC cable that is especially suitable for FH41 connectors (e.g. Phytec WF271). Standard FFC cables can cause a short circuit.

4.1 Connecting the phyCAM-M Camera

phyCAM-M Flip Lock – FCC Socket Camera Connection

- Plug the 30-pin FFC cable into the FFC socket with the contact surfaces facing downwards until you feel the stop. The reinforcement of the FFC cable (usually highlighted in color) points to the bracket of the socket.
- Lock the FFC socket by carefully pressing down on the bracket.

A type-A FFC cable (stripped and reinforced on the same side) is supplied with the kit. This ensures the correct signal assignment between the camera and the baseboard in the kit.
The position of the camera socket on the baseboard can be found in the hardware manual of the kit. Parallel to the connector designation (e.g. X100), some basic boards also have "CAM" printed on the board. The phyCAM-M interface is a 30-pin FFC socket, which can be installed standing or lying on the board.
The FFC cable is plugged in exactly the same way as the cable is plugged into the camera FFC socket.

Note: In the vertical version of the FFC socket, the contacts are also on the side opposite the bracket.

Open the lock of the 30-pin FFC connector on the camera by lifting the lock upwards.

figure 5: phyCAM-M Flip Lock – FFC Socket camera connector

5 Supported camera boards (YOCTO Linux PD21.1.0 and higher)

The cameras and the camera interface supported as a sub-devices. To configure the v4l2 framework is the handler "media-ctl" used. See chapter 7

The device tree of the camera VM-016 (sensor AR0144) is preselected in PD21.1.0. To use other cameras (e.g. VM-017/117) the device tree must be changed. See chapter 0

5.1 Supported camera types in standard vision image (phytec-vision-image-phyboard-polis-imx8mm-3)

Per default are following Kameratyps are supported in the image:

- VM-016-BW-M series (VM-016-BW-M, VM-016-BW-M-M12, VM-016-BW-M-H, …) based on camera sensor “AR0144”
- VM-016-COL-M series (VM-016-COL-M, VM-016-COL-M-M12, VM-016-COL-M-H, …) based on camera sensor “AR0144”

After login, you can start working with the demo-scripts. (see chapter 6.1)

Hardware Configuration		Possible Parameters				default I²C-address (default jumper on camera and boards)
phyCAM camera model (part number)	connected to	csi_port	cam_bus_type	cam_type	cam_i2c_address	
VM-016-BW-M (-M12 / -H)	X10 on Polaris board	1	phyCAM-M	VM-016	0x10, 0x18	0x10
VM-016-COL-M (-M12 / -H)	X10 on Polaris board	1	phyCAM-M	VM-016	0x10, 0x18	0x10

Notes:

- I²C addresses of the camera are set by hardware configuration (jumper setting on the camera and / or on the baseboard. Please refer to the VM-016-M manual L-1018 and the hardware manual of the Polis board L-862.

5.2 Change the camera types in standard vision image (phytec-vision-image-phyboard-polis-imx8mm-3)

The following camera types are interchangeable in the image:

- VM-017-BW-M series (VM-017-BW-M, VM-017-BW-M-M12, VM-017-BW-M-H, …) based on camera sensor “AR052x”
- VM-017-COL-M series (VM-017-COL-M, VM-017-COL-M-M12, VM-017-COL-M-H, …) based on camera sensor “AR052x”
- VM-117-BW-M series (VM-117-BW-M, VM-117-BW-M-M12, …) based on camera sensor “AR052x”
- VM-117-COL-M series (VM-117-COL-M, VM-117-COL-M-M12, …) based on camera sensor “AR052x”
- VM-017-BW-L series (VM-017-BW-L, VM-017-BW-L-M12, VM-017-BW-L-H, …) based on camera sensor “AR052x” (note: in progress)
- VM-017-COL-L series (VM-017-COL-L, VM-017-COL-L-M12, VM-017-COL-L-H, …) based on camera sensor “AR052x” (note: in progress)

Hardware Configuration		Possible Parameters				default I²C-address (default jumper on camera and boards)
phyCAM camera model (part number)	connected to	csi_port	cam_bus_type	cam_type	cam_i2c_address	
VM-017-BW-M (-M12 / -H) VM-117-BW-M (-M12)	X10 on Polaris board	1	phyCAM-M	VM-017	0x36, 0x37	0x36
VM-017-COL-M (-M12 / -H) VM-117-COL-M (-M12)	X10 on Polaris board	1	phyCAM-M	VM-017	0x36, 0x37	0x36
VM-017-BW-L (-M12 / -H)	X10 on Polaris board via VZ-018	1	phyCAM-M	VM-017	0x36, 0x37	0x36
VM-017-COL-L (-M12 / -H)	X10 on Polaris board via VZ-018	1	phyCAM-M	VM-017	0x36, 0x37	0x36

Notes:

- I²C addresses of the camera are set by hardware configuration (jumper setting on the camera and / or on the baseboard. Please refer to the VM-x17-_ manuals L-1020/1021/1022 and the hardware manual of the Polis board L-862.

How to change the device tree for VM-017/VM-117

Please following the instructions of “README.md” from PHYTEC FTP Server
ftp://ftp.phytec.de/pub/ImageProcessing/phyBOARD-Polis-i.MX8MM_linux_PD21.1.0/VM-017/

After reboot and login, load ar052x.ko camera driver: > insmod ar052x.ko

Support Hotline Europe:: +49 (6131) 9221-31 ♦http://www.phytec.de
Support Hotline North America: 1-800-278-9913 ♦http://www.phytec.com

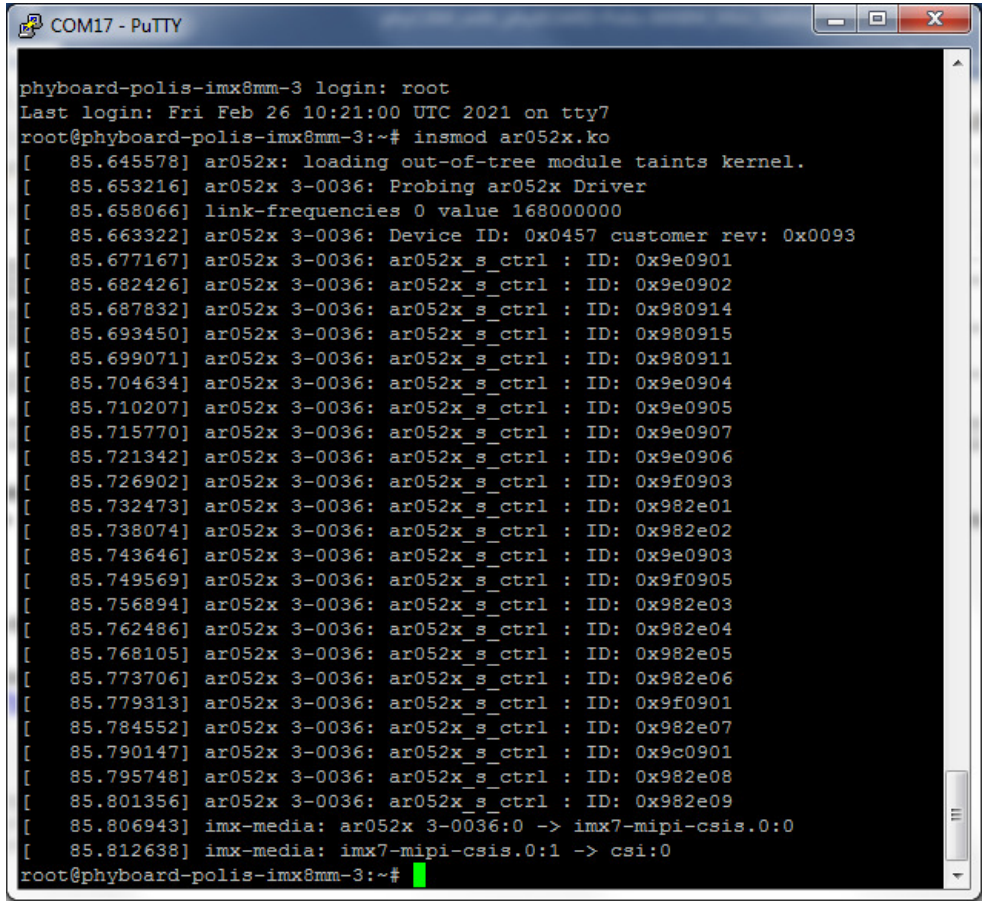

figure 6: output after load VM-017/117 driver AR052x

Now the VM-0x17 driver (AR052x) is ready for use.

Now you can start working with the demo-scripts. (see chapter 6.1)

Support Hotline Europe:: +49 (6131) 9221-31 http://www.phytec.de
Support Hotline North America: 1-800-278-9913 http://www.phytec.com

6 Demo scripts

If you want to see the live images, we recommend using a display on the Polis Board (for example 10” 1280x800 Display KPEB-AV-10-100.A0)

There are 3 sub directories with demo scripts for the cameras:
- gstreamer-examples
- v4l2_c-examples
- opencv-examples

Notes: Remove the qt-demo at the first start. With the \gstreamer-examples\remove_qt_demo.sh script, else the qt-demo is alway present on the display.

6.1 GStreamer scripts

After login, change into the directory: \gstreamer-examples\..
cd gstreamer_examples <ENTER>.

Now you can start working with the GStreamer demo-scripts.
Start the scripts with den word phrase "col" or "bw" depending on the connected camera color type.
- _cam-fbdev_1280x720.sh – scripts, show a livestream on display
- _cam-save_jpg_full_res.sh – scripts, save a JPG File in this directory
- _cam-save_raw_full_res.sh – scripts, save a RAW File in this directory
- func.sh – script, detect the camera typ and define the parameter for the scripts
- remove_qt_demo.sh – script, remove the qt-demo from autostart
- turn_off_on_wayland.sh – script, turn of/on if wayland necessary or not

Subdirectories:
- more_ar0144_scripts: contain more scripts for the VM-016 camera series (v4l-ctrl_ar0144.txt, list the v4l2-controls for this camera)
- more_ar052x_scripts: contain more scripts for the VM-017/117 camera series (v4l-ctrl_ar052x.txt, list the v4l2-controls for this camera)
- vpu_enc_dec_scripts: contain scripts how use the vpu-encoder (e.g networkstreaming or save H.264 streams)
- tools: contain scripts to get and set the camera register direct via i2c access
- phytec_usb_cam: scripts for use the Phytec USB-cameras

Support Hotline Europe:: +49 (6131) 9221-31 ♦http://www.phytec.de
Support Hotline North America: 1-800-278-9913 ♦http://www.phytec.com

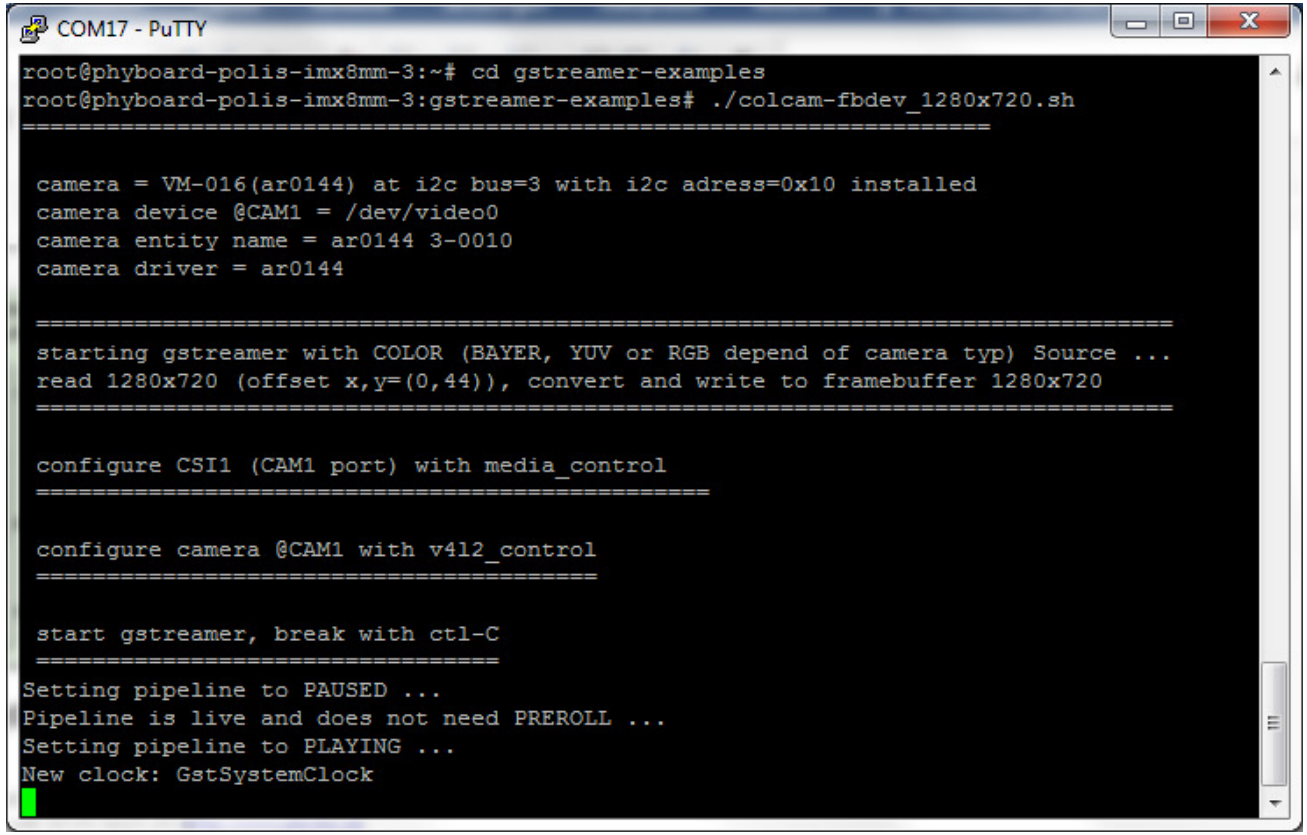

figure 7: call of "colcam-fbdev_1280x720.sh" for VM-016-COL (AR0144) series (A live image should show on the display)

All camera/video components get a separate "/dev/video[x]" or "dev/v4l-subdev[x]" device.
The v4l2 - capabilities are showed if you type: "v4l2-ctl -d [device] -L" e.g. "v4l2-ctl -d /dev/video0 -L".

For the first use, the camera and controller-camera interface must be configured with the tool v4l2-ctl. See chapter 7.

Support Hotline Europe:: +49 (6131) 9221-31 ♦http://www.phytec.de
Support Hotline North America: 1-800-278-9913 ♦http://www.phytec.com

6.2 Scripts to call C/C++ files based on v4l2 interface

After login, change into the directory: \v4l2_c-examples\..
cd v4l2_c-examples <ENTER>.

Now you can start working with the demo-scripts.
Start the scripts with den word phrase "col" or "bw" depending on the connected camera color type.

- ar0144_col/bw_full_save-raw – scripts, save a raw image from VM-016 in full resolution (8/10 and 12-Bit Formats are possible)
- ar052x_col/bw_full_save-raw – scripts, save a raw image from VM-017/117 in full resolution (8/10 and 12-Bit Formats are possible)

For saving the image we use the program Yavta. Yavta stands for "Yet Another V4L2 Test Application". This is a test application based on V4L2 Linux interface.

A other way to use in your C-program is the direct access call to the v4l2 interface. For example:

- *v4l2-ctl -d0 --stream-mmap --stream-count 1 --stream-to=raw_image1.raw*

For the first use, the camera and controller-camera interface must be configured with the tool v4l2-ctl. See chapter 7.

6.3 OpenCV scripts

PHYTEC i.MX 8M Mini BSP PD21.1.0 includes OpenCV4.4. For use OpenCV with image output we use window manager Wayland. PHYTEC tested openCV with PYTHON programming language. Examples of scripts can be found in the path “opencv-examples”:

- /opencv-examples/python/python3 phycam_video_v4l2.py: show the live image from the camera
- /opencv-examples/python/python3 face_detection.py: small application to detect faces

For the first use, the camera and controller-camera interface must be configured with the tool v4l2-ctl. See chapter 7.

7 Configure camera and controller-camera-interface with “media-ctl”

Modern System-on-Chip (SoC) devices support a wide range of functionality in the way of internal hardware blocks which has resulted in a variety of ways to interconnect functions within the SoC that alter the V4L device content.

The Media Controller kernel API has been designed to expose detailed information about media devices and capture devices to userspace in a way that allows them to be interconnected in dynamic and complex ways at runtime.

Media controller devices expose entities to the media controller framework. Each entity has one or more source pads and one or more sink pads. You use the media controller kernel API (ie via media-ctl) to query entities, link source pads to sink pads, and set formats of pads.

We use media-ctl to set:
- the desired resolution and color format of the camera sensor
- the fitting resolution and color format in the CSI interface (controller-camera-interface)
- the way (and/or preprocessing) through the different hardware blocks in the controller
- the resulting camera stream is made available as “/dev/videox” device

The settings are necessary:
- once after restarting the system and using the camera for the first time
- set a new resolution, color format or bit-depth
- set a new path through the hardware blocks

Following entities are present at the i.MX 8M Mini Image (check with “media-ctl –p”):
- entity 1: csi (2 pads, 2 links)
  pad0: Sink
  pad1: Source
- entity 4: csi capture (1 pad, 1 link)
  pad0: Sink
- entity 10: imx7-mipi-csis.0 (2 pads, 2 links)
  pad0: Sink
  pad1: Source
- entity 13: ar0144 3-0010 (1 pad, 1 link) #(depend on selected camera)
  pad0: Source

For setting the media-ctl we use in our demo script following points:
- media-ctl –r reset all links to inactive
- media-ctl –l setup the links between the hardware blocks
- media-ctl –V formats the hardware blocks

For example we will set the VM-016-COL (AR0144) with full Resolution from CSI Port to a /dev/videox device:

First we set the links:

media-ctl -r
media-ctl -l "'ar0144 3-0010':0->'imx7-mipi-csis.0':0[1]"
media-ctl -l "'imx7-mipi-csis.0':1->'csi':0[1]"
media-ctl -l "'csi':1->'csi capture':0[1]"
 # Camera -> imx7_mipi_csis.0 -> CSI -> CSI capture (/dev/videoX)

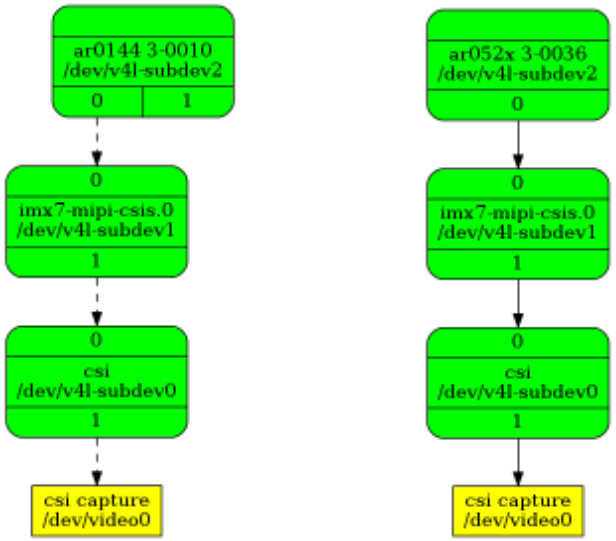

figure 8: video path VM-016 (AR0144) video path VM-017 (AR052x)

Then we formats the hardware blocks:

media-ctl -V "'ar0144 3-0010':0 [fmt:SGRBG8_1X8/1280x800 (0,4)/1280x800]"
media-ctl -V "'imx7-mipi-csis.0':1 [fmt:SGRBG8_1X8/1280x800]"
media-ctl -V "'csi':1 [fmt:SGRBG8_1X8/1280x800 field:none]"

Check the settings with “media-ctl –p”:

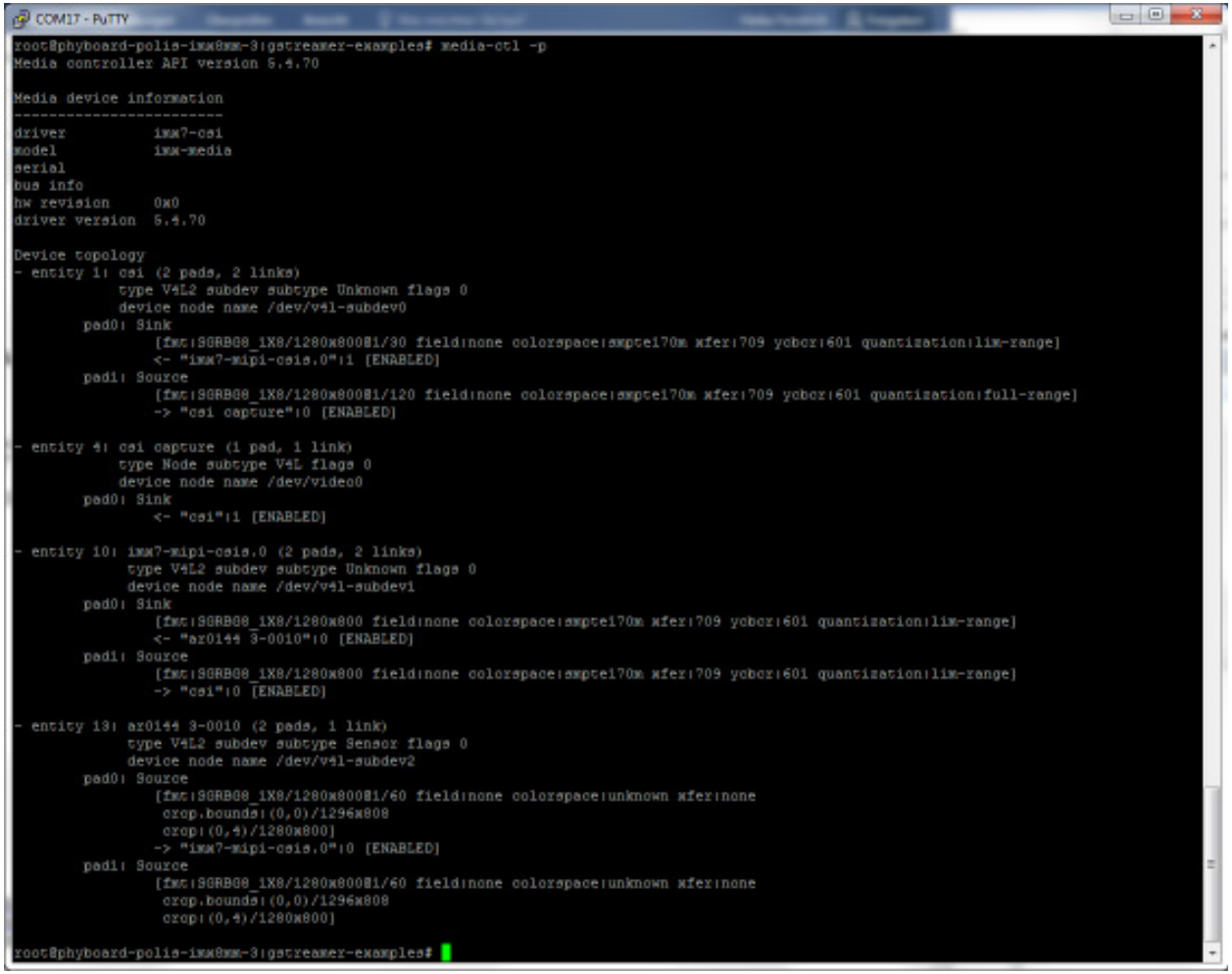

figure 9: Output “media-ctl –p”:

Now the Camera is present at “dev/video0” device.
The camera is ready for access.

8 Configuration of camera features

8.1 Configuration with v4l2-ctl

To set the various camera functions (e.g. exposure, gain, ...) use please the v4l2-ctl functions. You can get an overview of the existing functions by entering the commands:

- for example: v4l2-ctl -d /dev/video0 –all (list all)
- for example: v4l2-ctl -d /dev/video0 -L (list all detail)

With this control are many features usable. For example exposure. Set one exposure value (if the automatic disable):

- v4l2-ctl -d /dev/video0 --set-ctrl=auto_exposure=1 (set AR0144 AEC off)
- v4l2-ctl -d /dev/video0 --set-ctrl=exposure=800 (set the exposure time for a time, that the sensor need to generate 800 rows)

8.2 Configuration camera register direct

To set or get a camera register direct use the i2c functions in the path …/gstreamer-examples/tools/...

Note: Use this function only, if you know the register reaction. Read the register reference manual of the camera sensor manufacturer.

9 De-Bayering (demosaicking) with NEON CoProcessor

Most of CMOS color chips provide the image in the bayer mosaicing (bayer raw) format. For get a color image in RGB format is it necessary to convert the bayer raw image.
- https://en.wikipedia.org/wiki/Bayer_filter
- https://de.wikipedia.org/wiki/Bayer-Sensor

There are exist different algorithm for converting.

- https://en.wikipedia.org/wiki/Demosaicing

If the microprocessor does not include debayering hardware, have to do the converting via software. For this you need additional processing power and the framerate goes down. It is better to use the NEON coprocessor of the i.MX 8M Mini. For this support PHYTEC a special function.
It is present as GStreamer plugin "bayer2rgbneon" and in sources for use in an own C-program. We support a simple bilinear algorithm.

For use in GStreamer take "bayer2rgbneon" plugin. For more information to "bayer2rgbneon" parameters type:
- gst-inspect-1.0 bayer2rgbneon

The source are located at:
- https://git.phytec.de/bayer2rgb-neon/

Support Hotline Europe:: +49 (6131) 9221-31 ♦http://www.phytec.de
Support Hotline North America: 1-800-278-9913 ♦http://www.phytec.com