

Application Note

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

Revision History

Version	Changes	Author	Date
A0	Initial Release	H. Fendrich	05.02.2020
A1	Section 4, added reboot command	H. Fendrich	07.05.2020
A2	added phyCAM-M cable note	H. Fendrich	17.06.2020

Content

1	Overview.....	2
2	Camera Connectors on the Polaris - Carrier Boards	4
3	Change the cameras or camera parameters (YOCTO Linux PD19.1.0 and higher)	5
4	GStreamer function.....	6
5	GStreamer function for improved Tearing reduction	6
6	De-Bayering (demosacking) with NEON CoProcessor	7
7	OpenCV support.....	7

1 Overview

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

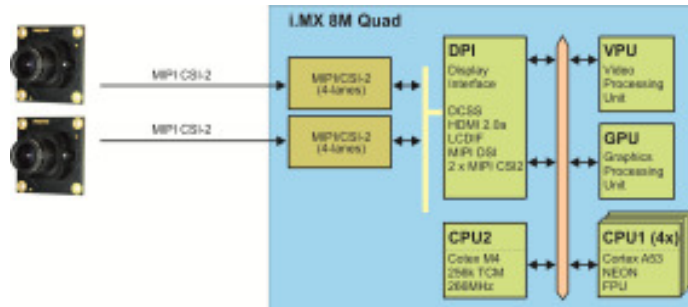


figure 1: Block Diagram Camera Interfaces of i.MX 8M Controller (Quad)

On the phyCORE-i.MX 8M the CSI1/CAM1 and CSI2/CAM2 camera path go out as CSI-2 MIPI signal. (see figure 2)

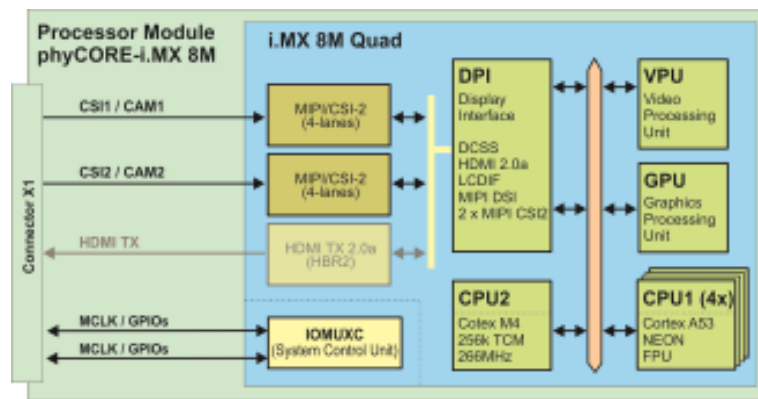


figure 2: CAM_1 (CSI1) and CAM_2 (CSI2) are present on connector X1

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-Polaris baseboard are both camera interfaces led out as phyCAM-M (MIPI CSI-2) interfaces (see figure 3). Here you can connect different phyCAM-M camera modules.

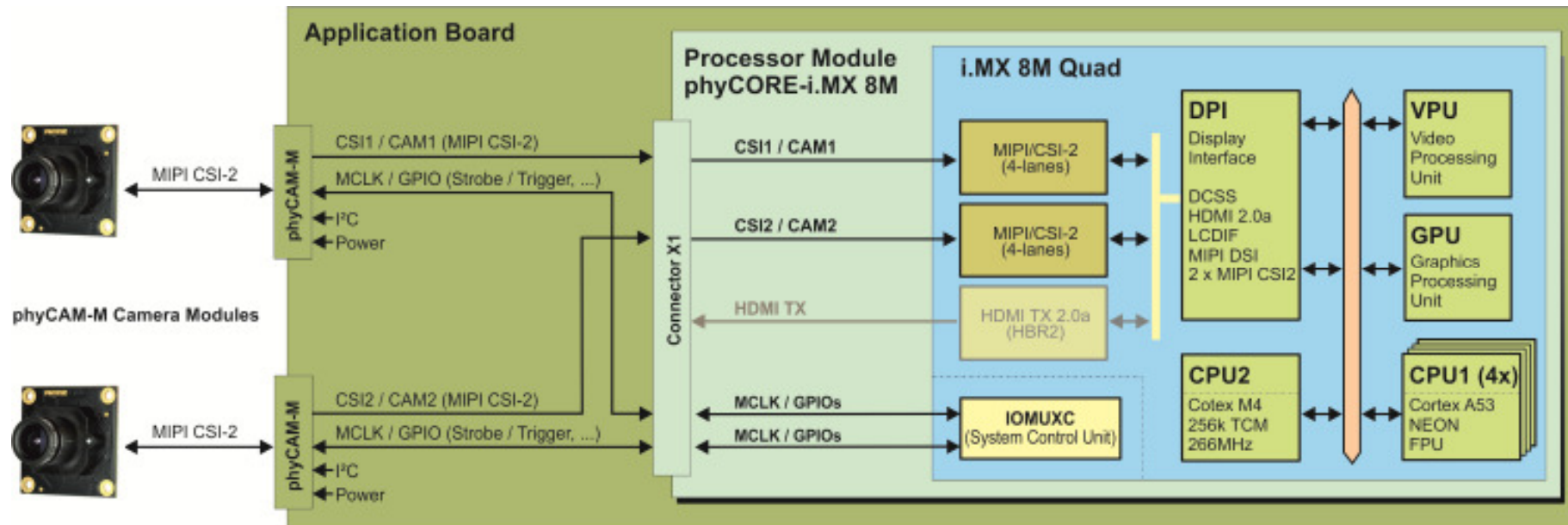


figure 3: Block Diagram of phyCAM-M Camera Interfaces of phyCORE-i.MX 8M (Quad) and the go out on the phyBOARD-Polaris-i.MX 8M - SBC

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.

Note that the both camera channels are independent. Therefore it is possible to work with 2 cameras parallel.

2 Camera Connectors on the Polaris - Carrier Boards

The development kits for the phyBOARD-Polaris-i.MX 8M contain:

- one carrier board (Polaris)
- one phyCORE-i.MX 8M modul SOM

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

On the base board Polaris (PBA-CD-12) we convert the both MIPI CSI-2 interfaces in phyCAM-M standard.

- phyCAM-M camera interface 1 (support on connector X10) / phyCAM-M camera interface 2 (support on connector X11)

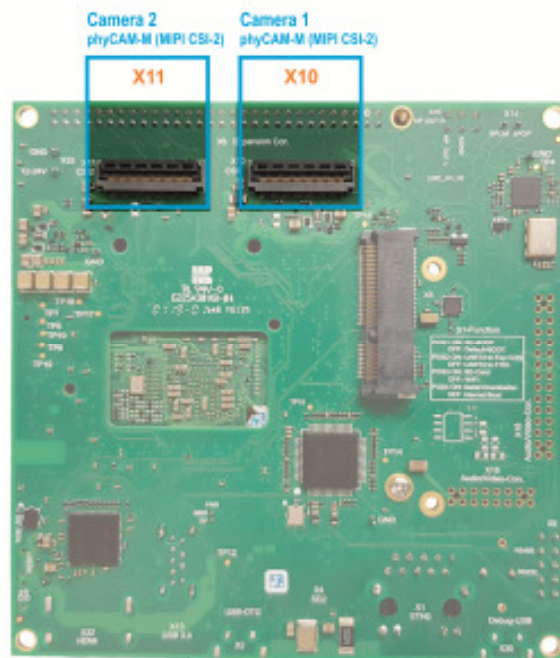


figure 4: Camera Interfaces on phyBOARD-Polaris up PCB Version PL1501.1 for the phyCORE-i.MX 8M SOM

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

3 Change the cameras or camera parameters (YOCTO Linux PD19.1.0 and higher)

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

For more information see on FTP:

ftp://ftp.phytec.de/pub/ImageProcessing/phyBOARD-Polaris-i.MX8M_linux_PD19.1.0/

In the PD19.1.0 are only the camera VM-016 (sensor AR0144) is supported. For use other cameras a cameras specific device tree is necessary.

Supported camera types. Changing the setting of the config-expansions parameters is not necessary.

- 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, change into the directory: `\gstreamer_examples\.`
`cd gstreamer_examples <ENTER>`.

Now you can start working with the GStreamer demo-scripts. (see chapter 4)

Hardware Configuration		Possible Parameters				default I ² C-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
	X11 on Polaris board	2	phyCAM-M	VM-016	0x10, 0x18	0x18
VM-016-COL-M (-M12 / -H)	X10 on Polaris board	1	phyCAM-M	VM-016	0x10, 0x18	0x10
	X11 on Polaris board	2	phyCAM-M	VM-016	0x10, 0x18	0x18

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 phyCAM-manual L-748 and the hardware manual of the kit.
- If two cameras are used at the same time, csi[X]_cam address must be different for CAM1 and CAM2.

4 GStreamer function

After login, change into the directory: `\gstreamer_examples\.`
`cd gstreamer_examples <ENTER>`.

Note: At the first start disable the QT-Demo and the Wayland windows manager

- 1) Type "systemctl disable phytec-qt-demo" for disable the qt demo
- 2) Type "turn_off_on_wayland.sh" -> select "2) disable weston (remove from autostart)" for disable Wayland
- 3) Complete this with a restart <reboot>.

Now you can start working with the GStreamer demo-scripts.

Start the scripts with the word phrase "col" or "bw" depending on the connected camera color type. In the subdirectory more_[sensor-name]_scripts are located camera specific scripts. (e.g. "more_ar0144_scripts" path contains scripts for the VM-016 (with ar0144 camera sensor) camera board.

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

5 GStreamer function for improved Tearing reduction

Tearing is a visual artifact in video display where the display image shows information from two or more frames in a single screen draw. It appears as a horizontal border or line. The image part beyond the line seems to be shifted horizontally when the image content or the camera is moved. The tearing line(s) usually move(s) vertically across the image.

The artifact occurs when the camera framerate differs from the display frame or the camera readout cycle is not in sync with the display's refresh.

Definition and background information about the tearing effect can be found in http://en.wikipedia.org/wiki/Screen_tearing.

The Freescale i.MX-8M processor contains an anti-tearing mechanism in the IPU unit, which can reduce the tearing effect.

However, since - depending on camera settings and camera model - the frame rate might be very different from the display's refresh rate, tearing effects might still be visible even if the anti-tearing mechanism is active. For applications that are intended to display live camera images on the display, additional measures should be considered to obtain a perfect image quality. This measures can include frame rate control (trimming the camera frame rate to the display's refresh rate), multi-buffering of the camera image etc.

We recommend to activate the anti-tearing mechanism of the i.MX 8M when live camera images are shown on the display. For evaluation purposes with the development kits, Phytec added GStreamer examples, that use a different kmssink – function, that activates the anti-tearing mechanism.

For more information to "kmssink" parameters type:

- gst-inspect-1.0 kmssink

Note: The kmssink parameter "connector=___" define the output device. If you change the output device e.g. HDMI, set the right parameter.

6 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. 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/>

7 OpenCV support

PHYTEC BSP PD19.1.0 includes OpenCV3.3. For use OpenCV with image output we recommend windows manager Wayland. To turn/off/on wayland use the script in the path ".../gstreamer_examples/turn_off_on_wayland.sh"

PHYTEC tested openCV with pyCAM cameras and the windows manager Wayland. Examples of scripts can be found on our ftp:

- ftp://ftp.phytec.de/pub/ImageProcessing/phyBOARD-Polaris-i.MX8M_linux_PD19.1.0/Software/OpenCV/

If you use the PHYTEC phyCAM - camera modules the camera and the camera interface driver have to configure with the v4l2-ctl tool. Examples are in the scripts at upper link.