

AN13712

8MP ISP OS08A20 Sensor

Rev. 2 — 4 September 2023

Application note

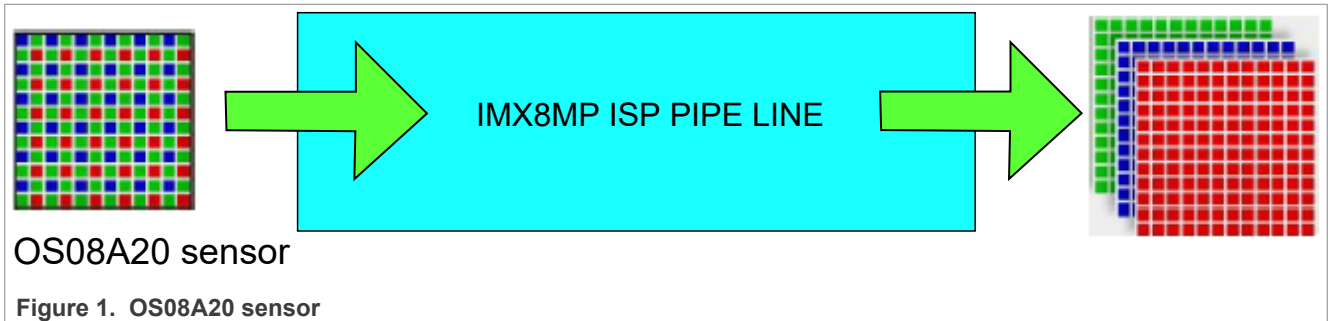
Document Information

Information	Content
Keywords	AN13712, 8MP ISP OS08A20 Sensor, i.MX Yocto SDK
Abstract	This document describes the OS08a20 sensor on the i.MX 8M Plus ISP. The OS08A20 sensor has image sizes of 4K, 2K, 1080p, and 720p.



1 Introduction

This document describes the OS08a20 sensor on the i.MX 8M Plus ISP. The OS08A20 sensor has image sizes of 4K, 2K, 1080p, and 720p. The output format is 12-bit / 10-bit RAW RGB. The sensor has 2-exposure staggered HDR and it supports the frame start input. The i.MX 8M Plus ISP has a demosaicing sensor for raw data and outputs the YUV format. The ISP also has denoise, sharpen, and gamma modules to improve the sensor image quality.



OS08A20 sensor

Figure 1. OS08A20 sensor

The following are the features of the OS08a20 sensor:

- Support for dual Os08a20 modules
- Support for 3 ISP output formats:
 - YUV422
 - NV16
 - NV12
- Support for 4 sensor modes:
 - 1920x1080 10-bit linear mode
 - 1920x1080 10-bit HDR mode
 - 3820x2160 12-bit linear mode
 - 3820x2160 10-bit HDR mode

2 Module and board

This section describes the module and the board.

2.1 i.MX 8M Plus EVK

[Figure 2](#) shows the i.MX 8M Plus EVK.

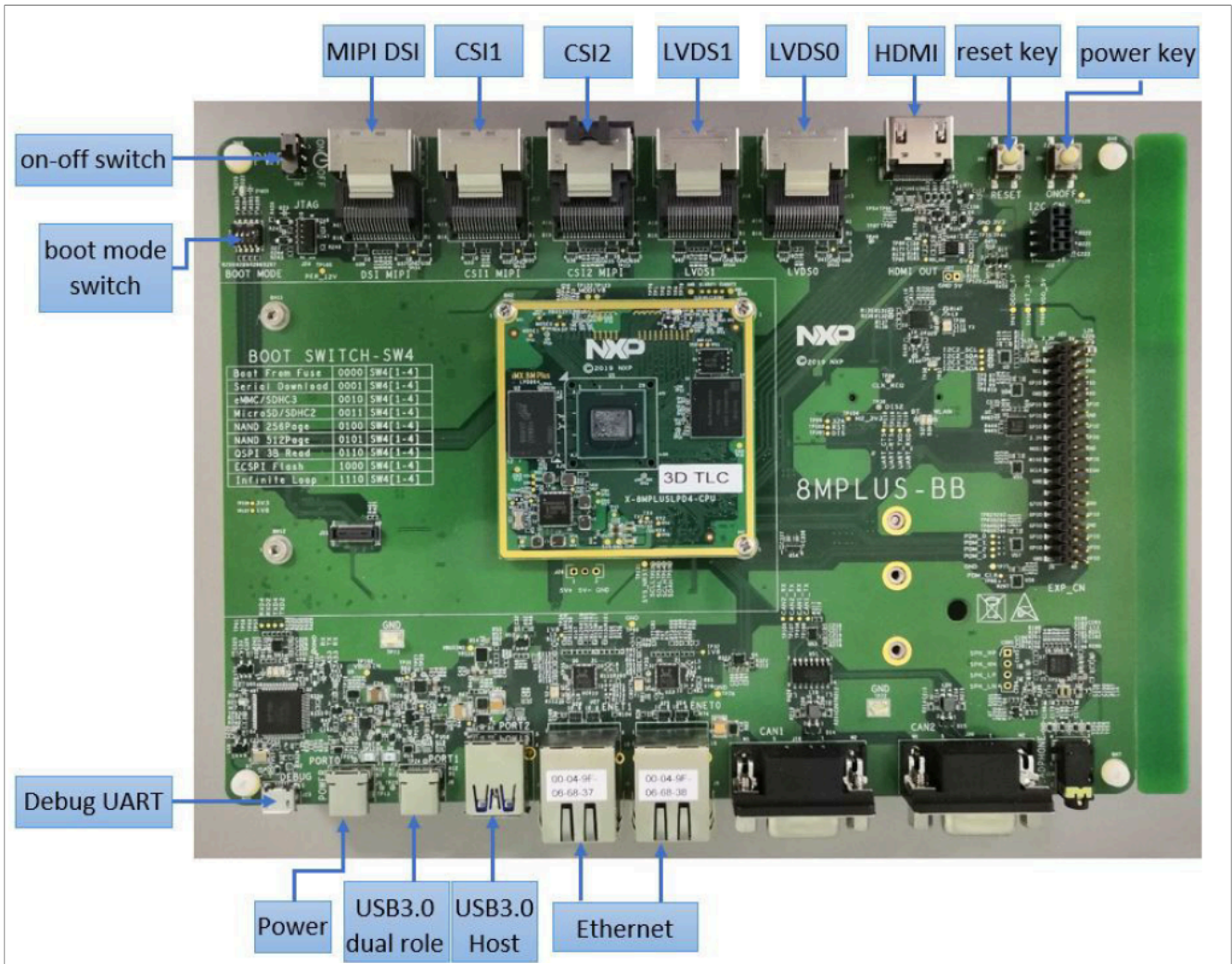


Figure 2. i.MX 8M Plus EVK

2.2 OS08A20 sensor module

Figure 3 shows the OS08A20 sensor module.



Figure 3. OS08A20 sensor module

2.3 Hardware connection

The sensor is connected to the switch board and the switch board uses the MiniSas cable to connect CSI1 or CSI2.

3 Configuring software

This section describes how to configure the software.

3.1 OS08A20 SDK HAL source code

```

OS08a20
├── calib
│   ├── OS08a20_8M_10_1080p_linear.xml
│   ├── OS08a20_8M_10_1080p_hdr.xml
│   ├── OS08a20_8M_10_4k_linear.xml
│   └── OS08a20_8M_10_4k_hdr.xml
├── source
│   └── OS08a20.c
└── dewarp
    ├── sensor_dwe_os08a20_1080P_config.json
    └── sensor_dwe_os08a20_4K_config.json
    
```

3.2 OS08A20 kernel driver source code

The OS08A20 kernel driver source code is as follows:

```

os08a20
├── os08a20_mipi_v3.c
├── os08a20_regs_1080p.h
├── os08a20_regs_1080p_hdr.h
├── os08a20_regs_4k.h
└── os08a20_regs_4k_hdr.h
    
```

3.3 Sensor mode table

[Table 1](#) describes the sensor modes.

Table 1. Sensor mode table

Mode	Index	Data format
1080P_linear	0	RAW10
1080P_hdr0	1	RAW10
4K_linear	2	RAW12
4K_hdr	3	RAW10

The "/opt/imx8-isp/bin/start_isp.sh" file has a mode-select parameter.

4 Building and testing

This section describes building and testing.

4.1 Creating the i.MX Yocto SDK and installing the toolchain

This section describes how to create the i.MX Yocto SDK and install the toolchain.

4.1.1 Downloading repository (if needed)

```
$ mkdir ~/bin (this step may not be needed if the bin folder already exists)
$ curl https://storage.googleapis.com/git-repo-downloads/repo > ~/bin/repo
$ chmod a+x ~/bin/repo
$ export PATH=~/bin:$PATH
```

4.1.2 Setting up Git (if needed)

```
$ git config --global user.name "Your Name"
$ git config --global user.email "Your Email"
$ git config -list
```

4.1.3 Creating the Yocto build environment

```
$ mkdir imx-yocto-bsp
$ cd imx-yocto-bsp
$ repo init -u https://github.com/nxp-imx/imx-manifest -b imx-linux-mickledore -
m imx-6.1.22-2.0.0.xml
$ repo sync
$ DISTRO=fsl-imx-xwayland MACHINE=imx8mp-lpddr4-evk source imx-setup-release.sh
-b build
```

4.1.4 Installing the toolchain

To install the toolchain, perform the following steps:

1. From the "build" folder, run the `./tmp/deploy/sdk/fsl-imx-xwayland-glibc-x86_64-imx-image-full-armv8a-imx8mp-lpddr4-evk-toolchain-6.1-mickledore.sh` file.
2. The default directory of the toolchain is `/opt/fsl-imx-xwayland/6.1-mickledore`. If you install the toolchain in another location, replace the default path in the following sessions with your own path.

4.2 Building the NXP kernel

This section describes how to build the NXP kernel.

4.2.1 Downloading the latest version of NXP kernel

```
$ git clone https://github.com/nxp-imx/linux-imx.git -b lf-6.1.22-2.0.0
```

4.2.2 Building the kernel

```
$ source /opt/6.1-mickledore/environment-setup-armv8a-poky-linux
$ make mrproper
$ make ARCH=arm64 imx_v8_defconfig O=./build_v8
$ cd build_v8/
$ make ARCH=arm64 -j8
```

4.3 Building the isp-imx

This section describes how to build the isp-imx.

4.3.1 Downloading the latest version of isp-imx

```
$ wget https://www.nxp.com/lgfiles/NMG/MAD/YOCTO/isp-imx-4.2.2.22.0.bin
$ chmod +x isp-imx-4.2.2.22.0.bin
$ ./isp-imx-4.2.2.22.0.bin
```

In Yocto, "isp-imx" is in "tmp/work/aarch64-mx8mp-poky-linux/isp-imx".

4.3.2 Building the SDK

```
$ source /opt/6.1-mickledore/environment-setup-armv8a-poky-linux
$ ./build-all-isp.sh release partial
```

4.4 Building the isp-vvcam

This section describes how to build the isp-vvcam.

4.4.1 Downloading the latest version of isp-vvcam

```
$ git clone https://github.com/nxp-imx/isp-vvcam.git -b lf-6.1.22-2.0.0
```

In Yocto, "isp-vvcam" is in "build-wayland-8mp/tmp/work/imx8mpevk-poky-linux/kernel-module-isp-vvcam".

4.4.2 Building the vvcam

```
$ source /opt/6.1-mickledore/environment-setup-armv8a-poky-linux
$ export KERNEL_SOURCE_DIR = [the build path of the NXP kernel]
$ ./build-all-vvcam.sh
```

4.5 Storing useful files

This section describes how to store useful files.

4.5.1 Copying useful files to the output directory

Perform the following steps to copy useful files to the output directory:

1. Copy the kernel files to the build-out directory:

```
$ cp linux-imx/build_v8/arch/arm64/boot/dts/freescale/imx8mp-evk-*.dtb [your
build-out directory]/boot
$ cp linux-imx/build_v8/arch/arm64/boot/Image [the build-out directory]/boot
$ cp linux-imx/build_v8/drivers/staging/media/imx/imx8-media-dev.ko [the
build-out directory]/sdk
```

2. Copy the isp-imx files to the build-out directory:

```
$ cp -r ./isp-imx-4.2.2.22.0/build_output_release_partial/blob/* [the build-
out directory]/sdk
```

3. Copy the isp-vvcam files to the build-out directory:

```
$ cp ./isp-vvcam/modules/* [the build-out directory]/sdk
```

4.5.2 Sending a file to a board

```
$ scp -r [the build out directory]/sdk/* root@$EVK_IP_Address:/home/root/[your test directory in root]
$ scp [the build out directory]/boot/* root@$EVK_IP_Address:/run/media/boot-mmcblk1p1/
```

The output directory should contain the following files:

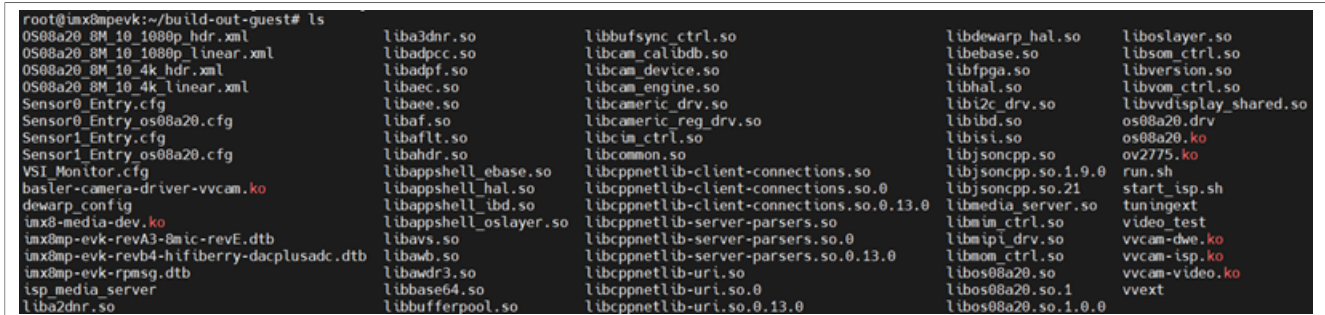


Figure 4. Output directory

4.6 Selecting a device tree

To select a device tree, perform the following steps:

1. "imx8mp-evk-os08a20.dtb" - # single os08a20, connect to CSI1
2. "imx8mp-evk-dual-os08a20.dtb" - # dual os08a20, connect to CSI1 and CSI2
3. "imx8mp-evk-os08a20-ov5640.dtb" - # ov5640 and os08a20 (os08a20 -> CSI1, ov5640 -> CSI2)

4.7 Editing the sensor-configuration file and selecting the correct mode

Sensor0_Entry.cfg (example):

```
name="os08a20" drv = "os08a20.drv"
mode= 2
[mode.0]
xml = "OS08a20_8M_10_1080p_linear.xml"
dwe = "dewarp_config/sensor_dwe_os08a20_1080P_config.json"
[mode.1]
xml = "OS08a20_8M_10_1080p_hdr.xml"
dwe = "dewarp_config/sensor_dwe_os08a20_1080P_config.json"
[mode.2]
xml = " OS08a20_8M_10_4k_linear.xml"
dwe = "dewarp_config/sensor_dwe_os08a20_4K_config.json"
[mode.3]
xml = " OS08a20_8M_10_4k_hdr.xml"
dwe = "dewarp_config/sensor_dwe_os08a20_4K_config.json"
```

4.8 Enabling the ISP and camera on the board

This section describes how to enable the ISP and camera on the board.

4.8.1 Adding to a path

```
$ export LD_LIBRARY_PATH=$pwd:$LD_LIBRARY_PATH
```

4.8.2 Stopping the default ISP

```
$ systemctl stop imx8-isp.service
```

4.8.3 Removing existing modules

```
$ rmmod vvcam-dwe  
$ rmmod vvcam-isp  
$ rmmod vvcam-video  
$ rmmod imx8-media-dev.ko  
$ rmmod os08a20.ko
```

4.8.4 Installing modules

```
$ insmod vvcam-dwe  
$ insmod vvcam-isp  
$ insmod vvcam-video  
$ insmod imx8-media-dev.ko  
$ insmod os08a20.ko
```

4.8.5 Starting the ISP media server

Single sensor:

```
$ ./isp_media_server CAMERA0&
```

Dual sensor:

```
$ ./isp_media_server DUAL_CAMERA&
```

4.9 OS08A20 test cases

Mode 0: 1080p linear:

- Change "Sensor0_Entry.cfg" to mode 0:

```
$ gst-launch-1.0 -v v4l2src device=/dev/video2 ! "video/x-raw,format=YUY2,width=1920,height=1080" ! queue ! waylandsink
```

Mode 1: 1080p HDR:

- Change "Sensor0_Entry.cfg" to mode 1:

```
$ gst-launch-1.0 -v v4l2src device=/dev/video2 ! "video/x-raw,format=YUY2,width=1920,height=1080" ! queue ! waylandsink
```

Mode 2: 4K linear:

- Change "Sensor0_Entry.cfg" to mode 2:

```
$ gst-launch-1.0 -v v4l2src device=/dev/video2 ! "video/x-raw,format=YUY2,width=3820,height=2160" ! queue ! waylandsink
```

Mode 3: 4K linear:

- Change "Sensor0_Entry.cfg" to mode 3:

```
$ gst-launch-1.0 -v v4l2src device=/dev/video2 ! "video/x-raw,format=YUY2,width=3820,height=2160" ! queue ! waylandsink
```

4.10 Disabling or bypassing dewarp

If you want to bypass the dewarp configuration, you may set the "dewarp bypass" parameter in the dewarp configuration file to "true".

```
{
  "dewarpConfigArray" : [
    {
      "source_image": {
        "width" : 1920,
        "height" : 1080
      },
      "?dewarpType": "LENS_CORRECTION, FISHEYE_EXPAND, SPLIT_SCREEN",
      "dewarpType": "FISHEYE_DEWARP",
      "scale": {
        "roix" : 0,
        "roiy" : 0,
        "factor" : 1.0
      },
      "split": {
        "horizon_line" : 540,
        "vertical_line_up" : 960,
        "vertical_line_down": 960
      },
      "bypass" : true,
      "hflip" : false,
      "vflip" : false,
      "camera_matrix" : [1.9584556270377586e+003,0.0, 9.6819933899253533e+002,0.0,0.0,0.0],
      "distortion_coeff": [-1.2839656060464022e-001, 1.4121087523973114e-001, 0.0, 0.0, 0.0, 0.0]
    }
  ]
}
```

Figure 5. Disabling or bypassing dewarp

If you want to disable the dewarp functionality, after starting "isp_media_server", you may run the following command before running the "gststream" command:

```
$ v4l2-ctl -d 2 -c viv_ext_ctrl='{<id>:<pipeline.s.dwe.onoff>;<enable>:false}'
```

5 Note about the source code in the document

Example code shown in this document has the following copyright and BSD-3-Clause license:

Copyright 2023 NXP Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials must be provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

6 Revision history

[Table 2](#) summarizes the changes done to this document.

Table 2. Revision history

Revision number	Release date	Description
2	4 September 2023	Update to Linux kernel version 6.1.22.
1	29 November 2022	Updated Section 1 .
0	24 August 2022	Initial release.

7 Legal information

7.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

7.2 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer.

In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out of the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages.

Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US\$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. - NXP B.V. is not an operating company and it does not distribute or sell products.

7.3 Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

i.MX — is a trademark of NXP B.V.

Contents

1	Introduction	2
2	Module and board	2
2.1	i.MX 8M Plus EVK	2
2.2	OS08A20 sensor module	3
2.3	Hardware connection	4
3	Configuring software	4
3.1	OS08A20 SDK HAL source code	4
3.2	OS08A20 kernel driver source code	4
3.3	Sensor mode table	4
4	Building and testing	4
4.1	Creating the i.MX Yocto SDK and installing the toolchain	5
4.1.1	Downloading repository (if needed)	5
4.1.2	Setting up Git (if needed)	5
4.1.3	Creating the Yocto build environment	5
4.1.4	Installing the toolchain	5
4.2	Building the NXP kernel	5
4.2.1	Downloading the latest version of NXP kernel	5
4.2.2	Building the kernel	5
4.3	Building the isp-imx	6
4.3.1	Downloading the latest version of isp-imx	6
4.3.2	Building the SDK	6
4.4	Building the isp-vvcam	6
4.4.1	Downloading the latest version of isp-vvcam	6
4.4.2	Building the vvcam	6
4.5	Storing useful files	6
4.5.1	Copying useful files to the output directory	6
4.5.2	Sending a file to a board	7
4.6	Selecting a device tree	7
4.7	Editing the sensor-configuration file and selecting the correct mode	7
4.8	Enabling the ISP and camera on the board	8
4.8.1	Adding to a path	8
4.8.2	Stopping the default ISP	8
4.8.3	Removing existing modules	8
4.8.4	Installing modules	8
4.8.5	Starting the ISP media server	8
4.9	OS08A20 test cases	8
4.10	Disabling or bypassing dewarp	9
5	Note about the source code in the document	10
6	Revision history	10
7	Legal information	11

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

© 2023 NXP B.V.

All rights reserved.

For more information, please visit: <http://www.nxp.com>

Date of release: 4 September 2023
Document identifier: AN13712