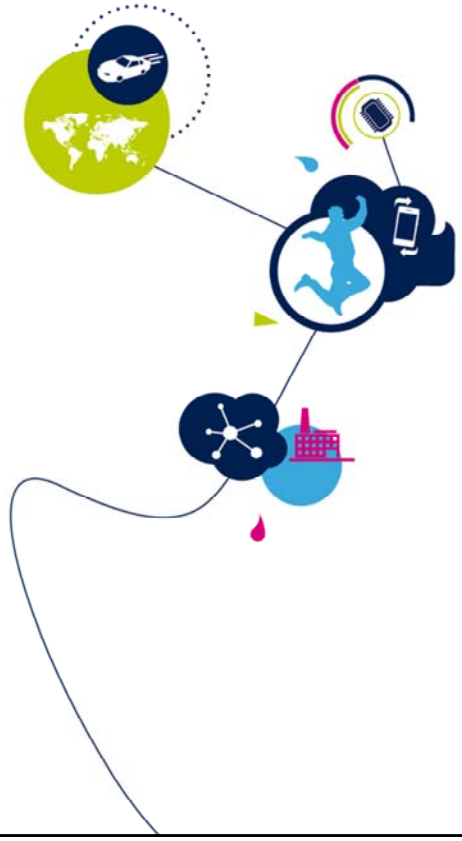
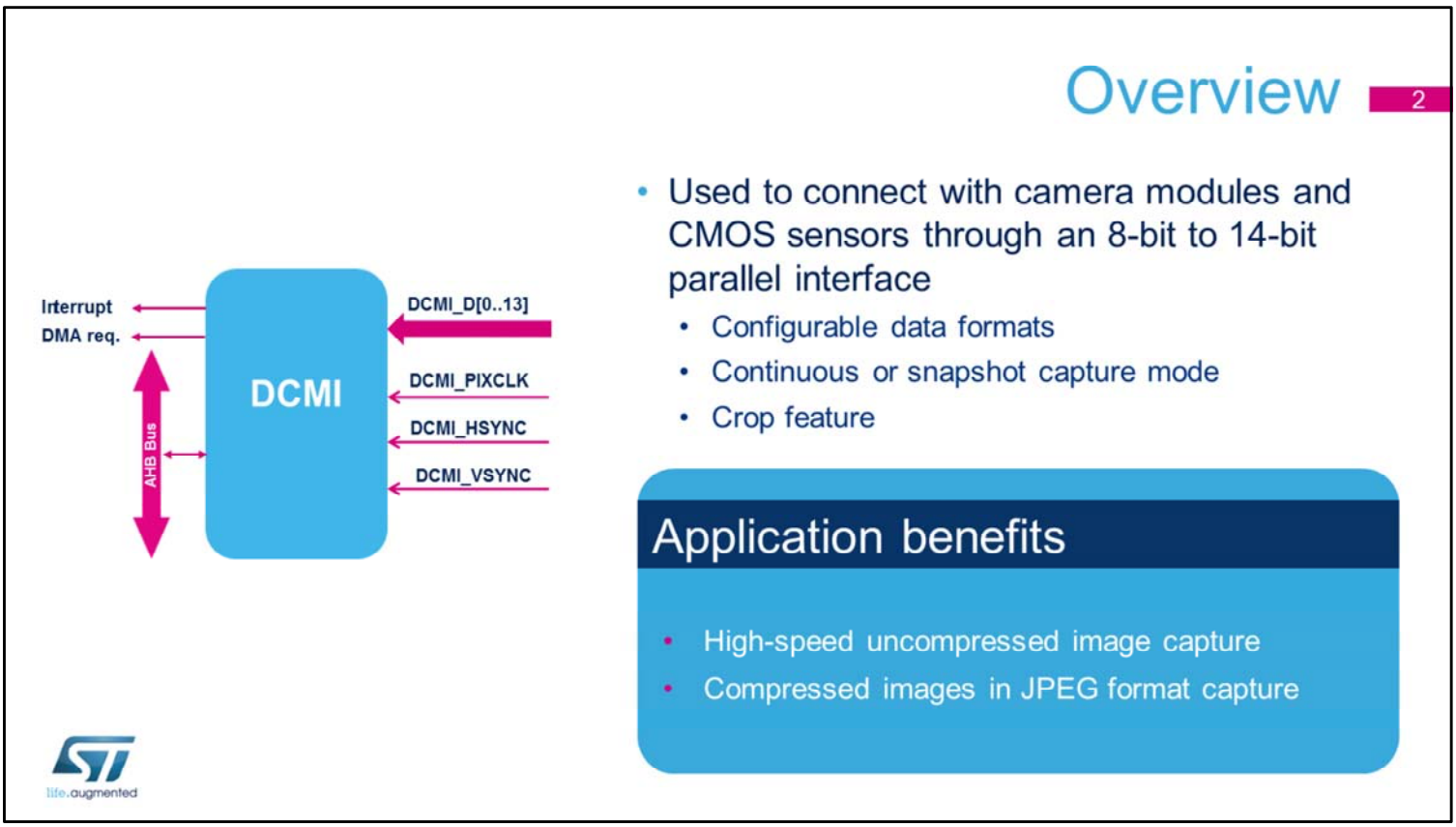
STM32MP1 - DCMI

Digital Camera Interface
Revision 1.0



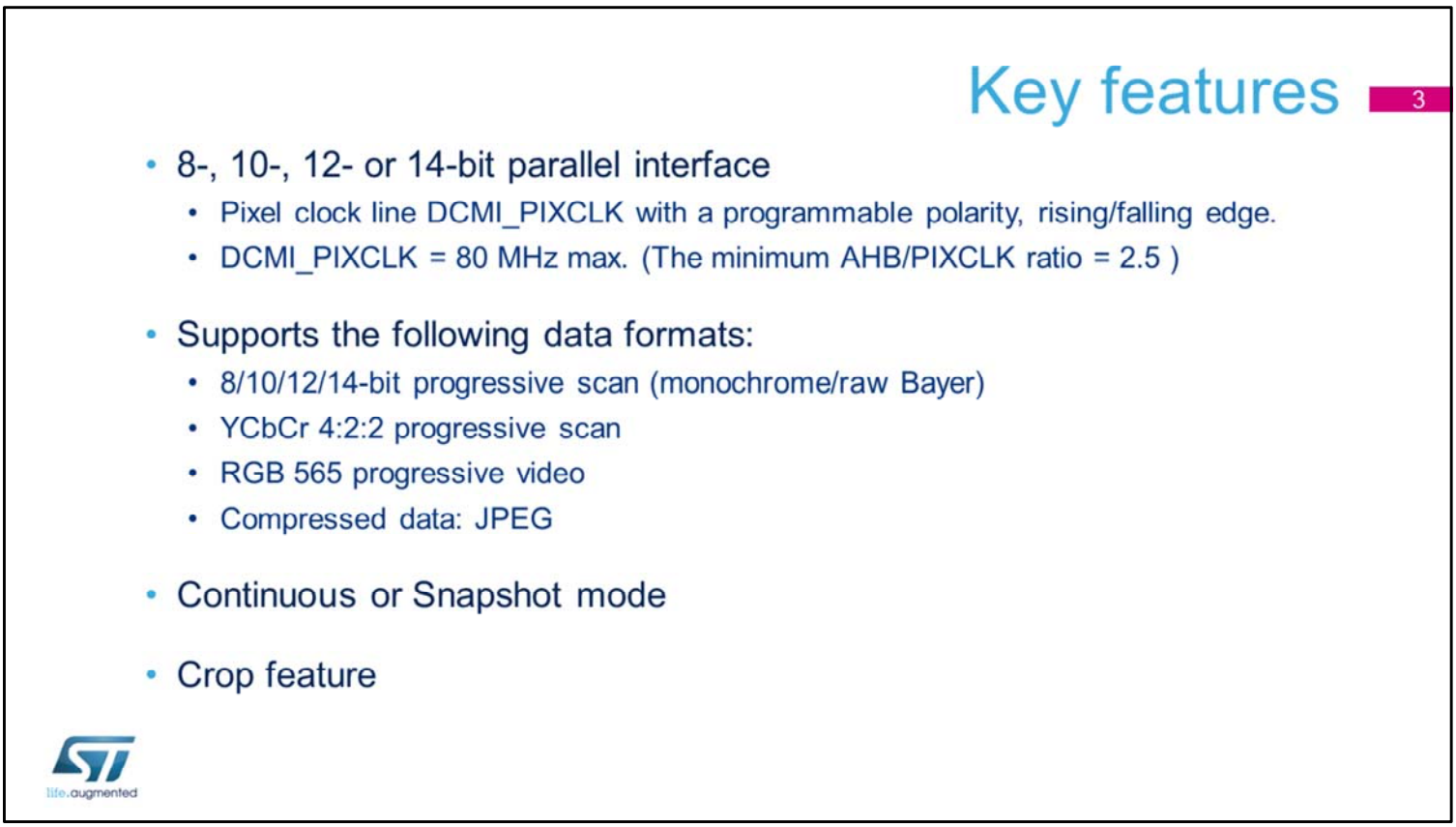
life.augmented

Hello, and welcome to this presentation of the STM32 digital camera interface controller. It covers all features of this interface.



DCMI stands for Digital Camera Interface. The DCMI interface is used to connect a parallel camera module to the STM32. The camera generates a parallel data flow together with a pixel clock signal (DCMI_PIXCLK) which allows the interface to capture the incoming data flow. Two optional signals (HSYNC and VSYNC) may be used to synchronize the image frame between the camera and the STM32. The DCMI supports also embedded line/frame synchronization code in the data flow.

The DCMI allows to perform continuous grabbing. This process starts on application request and continues until the CAPTURE bit is cleared. Alternatively, Snapshot mode allows to capture a single frame upon an application request.
With the crop feature, the camera interface can cut and store a rectangular portion of the received image.



The camera interface has a configurable parallel data interface from 8 to 14 data lines, together with a pixel clock line DCMI_PIXCLK with a programmable polarity, rising/falling edge configuration and a maximum DCMI_PIXCLK of 80MHz. DCMI_PIXCLK and AHB clocks must respect the minimum ratio AHB/PIXCLK of 2.5
So this camera interface can achieve a data transfer rate of up to 140 Mbyte/s using a 80 MHz pixel clock and 14-bit data.

It supports color or monochrome cameras using different data formats:

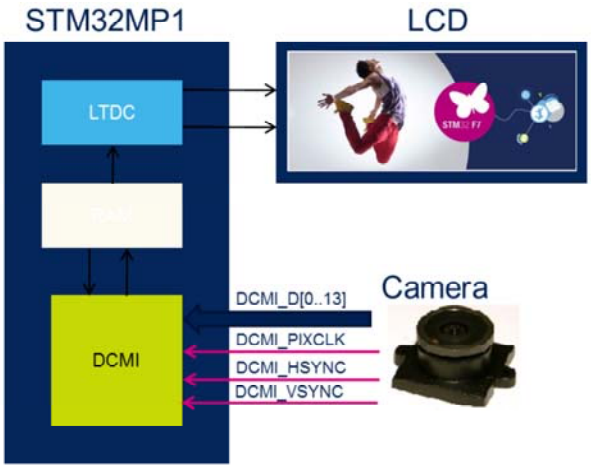
- Uncoded parallel data – also known as progressive scan, which can be either monochrome or color (raw Bayer)
- Luminance/color coded on 8-bit (4/2/2 progressive

scan)

- RGB 565 – red-green-blue information coded on 16-bit
- Some cameras also use this parallel interface to transmit compressed images in JPEG format

DCMI data transfer

- The data are packed into a 32-bit data register (DCMI_DR) connected to the AHB bus
- 8x32-bit FIFO with DMA handling



life.augmented

This is an example of a simple application used to transmit the camera image to the LCD display.

The standard way to use the camera interface is to store the received data in a frame buffer in RAM. The STM32 can then process this data or transmit it further through another interface (e.g. USB or Ethernet).
In order to limit the bus contention in the system and avoid missing data, despite of the high data rate of this interface, the received data are packed in a FIFO buffer, as shown on the next slide.

DCMI packing and extended data mode

- The camera interface can capture 14-, 12-, 10- or 8-bit data
- If less than 14 bits are used:
  - Unused input pins are available and can be assigned to other peripherals .
  - They must not be allocated to DCMI.

8-Bit

Byte address	31:24	23:16	15:8	7:0
0	Dn+3[7:0]	Dn+2[7:0]	Dn+1[7:0]	Dn[7:0]
4	Dn+7[7:0]	Dn+6[7:0]	Dn+5[7:0]	Dn+4[7:0]

10-Bit

Byte address	31:26	25:16	15:10	9:0
0	0	Dn+1[9:0]	0	Dn[9:0]
4	0	Dn+3[9:0]	0	Dn+2[9:0]

12-Bit

Byte address	31:28	27:16	15:12	11:0
0	0	Dn+1[11:0]	0	Dn[11:0]
4	0	Dn+3[11:0]	0	Dn+2[11:0]

14-Bit

Byte address	31:30	29:16	15:14	13:0
0	0	Dn+1[13:0]	0	Dn[13:0]
4	0	Dn+3[13:0]	0	Dn+2[13:0]

life.augmented

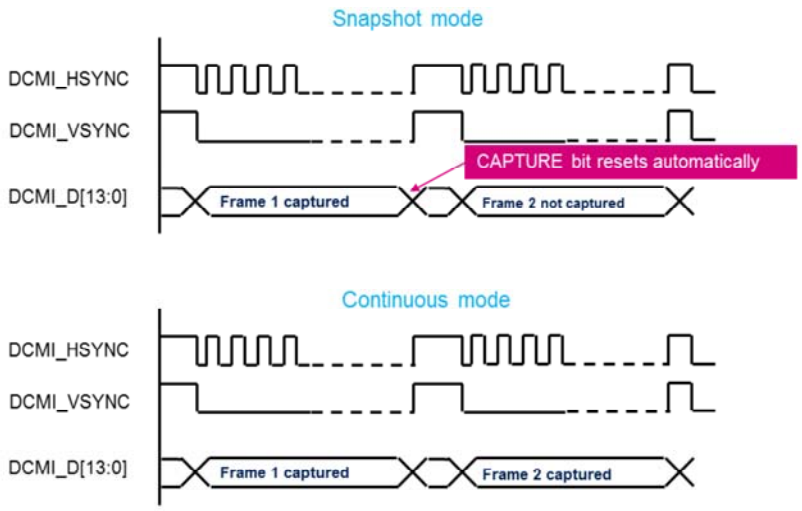
Depending on the interface size (8, 10, 12 or 14 bits), 2 or 4 data items are stored in a single 32-bit word. Once a complete 32-bit word is available, it is transferred by DMA to the memory. This allows to reduce the bus bandwidth used by the DCMI, even for high-speed cases.

The DCMI has also a hardware feature allowing to select 1 byte out of 2 or 1 byte out of 4. This feature is used to convert of a color image to B&W and/or the reduction of the image size. In this later case, in order to keep the form factor of the image, the DCMI may only store every other line, reducing the vertical resolution by a factor of two.

An 8-level FIFO is used in order to accommodate for any DMA response latency, without loosing data.

DCMI capture modes

- The DCMI interface supports two types of capture:
  - Snapshot (single frame)
  - Continuous grab

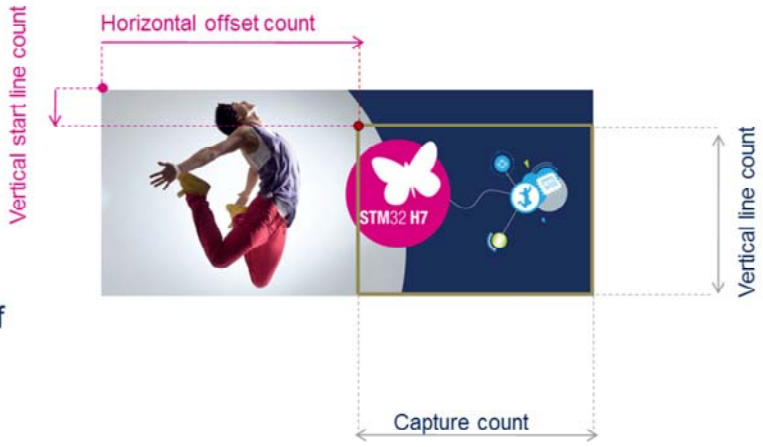


life.augmented

The camera interface allows to capture a single frame (synchronized on after a software request) or to continuously receive the video flow.
In capture mode, the capture is requested by setting the Capture bit in software and starts with the beginning of the next incoming frame and the DCMI clears the Capture bit when the single frame has been received.

DCMI “crop” feature

- The DCMI can select a rectangular window from the received image
  - The window size and coordinates are specified by two 32-bit registers DCMI_CWSTRT and DCMI_CWSIZE.
  - The size and position of the window is specified in number of pixel clocks (horizontal dimension) and in number of lines (vertical dimension)



Cropping is another way to reduce the image size, in addition to reducing the pixel resolution as mentioned previously.
This option is valid for both single frame capture and in continuous mode, but it is not supported for JPEG format.

Interrupts and DMA

Interrupt event	Description
IT_LINE	Indicates the end of line
IT_FRAME	Indicates the end of frame capture
IT_OVR	indicates the overrun of data reception
IT_VSYNC	Indicates the synchronization frame
IT_ERR	Indicates the detection of an error in the embedded synchronization frame detection

- DCMI interrupt is the logical OR of previous interrupts
- DMA interface: one DMA channel is needed for incoming data transfers
  - A DMA request is generated each time the camera interface receives a complete 32-bit data block in its FIFO

life.augmented

Five interrupts are generated. All interrupts are maskable by software. The global interrupt is the OR of all single interrupts.

The DMA interface is active when Capture mode is enabled. A DMA request is generated each time the camera interface receives a complete 32-bit data block in its FIFO. For DMA channels available to DCMI, please refer to the DMA section in the STM32H7 reference manual.

Low-power modes

Mode	Description
CRun	Active (enabled by software).
CSleep (MPU or MCU sub-system state)	Active. Peripheral interrupts cause the device to exit CSleep mode.
Stop + LP-Stop	Registers content is kept but the DCMI is not functional.
LPLV-Stop	Registers content is kept but the DCMI is not functional.
Standby	Powered-down. The peripheral must be reinitialized after exiting Standby mode.

life.augmented

Here is an overview of the status of the DCMI module in each of the low-power modes.
DCMI operations are not possible when the device is in Stop or Standby mode.

Related peripherals

10

- Refer to these trainings related to this peripheral
  - Direct memory access controller (DMA)
  - Interrupts (NVIC)

This is a list of peripherals related to the DCMI module. Refer to DMA and NVIC trainings for more information about the DCMI channel and interrupt configuration and to GPIO chapter for setting up the alternate function pins used by the DCMI.

Examples of configuration and use of camera interface are available in the STM32CubeMP1 package available at www.st.com.

References

- For more details, please refer to the following application note:
  - Application note AN5020: Digital camera interface (DCMI) for STM32 MCUs

life.augmented

For more details, please refer to the application note AN5020 about DCMI on STM32 devices.