uDMA CAMERA

A camera interface is a hardware block that interfaces with different image sensor interfaces and generates output that can be used for image processing.

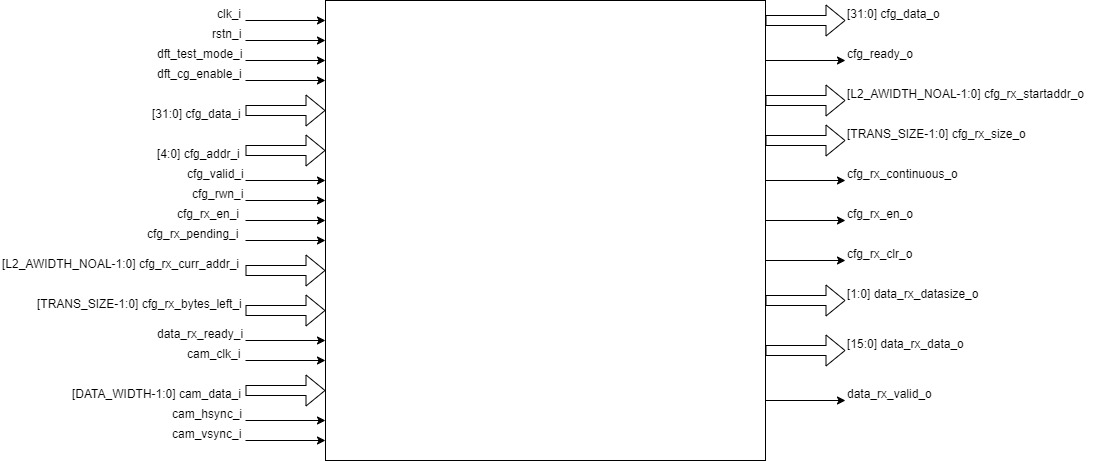
FEATURES:

* Supports RGB565, RGB555 ,RGB444, BYPASS\_LITEND and BYPASS\_BIGEND image formats.
* Allows windowing. It allows users to select a range of interest in the picture. It can be disabled by the user.
* Parallel data input line for carrying pixel data.
* There is a horizontal sync(HSYNC) input which indicates one line of the frame is transmitted.
* There is a vertical sync(VSYNC) input which indicates that one entire frame is transmitted. It can be configured for polarity.

THEORY OF OPERATION:

cam\_clk\_i is a pixel clock which changes on every pixel. Pixel data is taken as input through cam\_data\_i, and cam\_hsync\_i and cam\_vsync\_i indicate the horizontal and vertical sync value. It supports active low reset. It contains a udma dc fifo to store the pixel value before sending it to output.

BLOCK DIAGRAM:



* Read write input pin, cfg\_rwn\_i indicates if we want to write to the register or read from the register. If the input is high then the register is selected for reading and else for writing. Address of the register is provided through cfg\_addr\_i.
* Value read through the register is provided as output through cfg\_data\_o.
* cfg\_data\_i writes values to register.
* Data in register REG\_RX\_SADDR is passed through cfg\_rx\_startaddr\_o.
* Data in register REG\_RX\_SIZE is passed through cfg\_rx\_size\_o.
* Data in the REG\_RX\_CFG is passed through cfg\_rx\_continuous\_o, cfg\_rx\_en\_o, cfg\_rx\_clr\_o and data\_rx\_datasize\_o.
* Frame counter:
  + Frame counter is incremented at start of frame if frame drop is enabled.
  + Counter is reset if the counter value reaches frame drop value.or frame drop is disabled.
  + Frame drop enable status and frame drop value can be read from REG\_CAM\_CFG\_GLOB.
  + Non zero frame counter value indicates a valid frame.
* WINDOWING:
  + Window of interest can be selected by using the windowing feature.
  + Its enable or disable status can be read from REG\_CAM\_CFG\_GLOB.
  + Coordinates of the window can be written to and read from REG\_CAM\_CFG\_LL and REG\_CAM\_CFG\_UR.
  + A pixel is valid only if it is inside the window of interest if windowing is enabled.
  + If windowing is disabled, pixels will be valid for every valid frame.
* Row counter and column counter:
  + Counts the row and column at every camera clock.
  + Counter is reset at the start of the frame.
  + These counter values are used when windowing is enabled to check the validity of pixels.
  + Column counter is incremented at posedge of cam\_clk\_i. Column ends when the column counter reaches ROWLEN, which resets the counter value. ROWLEN value can be read from REG\_CAM\_CFG\_SIZE.
  + Row counter is incremented at the end of each column.
* IMAGE FORMAT: RGB565, RGB555, RGB444
  + RGB565: Five bits of data is allocated for the red and blue color component and 6 bits data for the green color component.
  + RGB555: Five bits of data is allocated for all the color components.
  + RGB444: Four bits of data is allocated for all the color components.
  + R, G, B pixel values can be read from cam\_data\_i.
  + Filter values for R, G, B can be obtained by multiplying their respective pixel values by their coefficients. Coefficient can be read from REG\_CAM\_CFG\_FILTER.
  + Filter values for all the pixels are added and then shifted right to get the final pixel value which is then passed to fifo. Number of bits needed to be shifted can be read from REG\_CAM\_CFG\_GLOB.
* IMAGE FORMAT: BYPASS\_LITEND, BYPASS\_BIGEND
  + These image formats are used for YUV images. In the YUV image a color is described as a Y component(luma) and two chroma components U and V. Luma represents the brightness of the image and chroma conveys the color information of the picture.
  + YUV pixel value can be read from cam\_data\_i.
  + Filter is not valid.
* Vertical sync:
  + Polarity can be read from REG\_CAM\_VSYNC\_POLARITY..
  + A start of frame is marked by high current vsync value and low previous vsync.
* udma\_dc\_fifo:
  + RGB or YUV pixel values are sent as input udma\_dc\_fifo.
  + Valid output is passed through data\_rx\_valid\_o if there is data in fifo to be read.
  + Data can be read from the fifo through data\_rx\_data\_o.

uDMA CAMERA CSR’s

REG\_RX\_SADDR:

Offset = 0x00

| Field | Bits | Type | Default | Description |
| --- | --- | --- | --- | --- |
| SADDR | 31:0 | RW |  | Address of receive memory buffer: |
|  |  |  |  | Read: value of pointer until transfer is over, then 0 |
|  |  |  |  | Write: set memory buffer start address |

REG\_RX\_SIZE:

Offset = 0x04

| Field | Bits | Type | Default | Description |
| --- | --- | --- | --- | --- |
| SIZE | 15:0 | RW | 0x00 | Buffer size in bytes (1MB max) |
|  |  |  |  | Read: bytes remaining until transfer complete |
|  |  |  |  | Write: set number of bytes to transfer |

REG\_RX\_CFG

Offset = 0x08

| Field | Bits | Type | Default | Description |
| --- | --- | --- | --- | --- |
| CLR | 6:6 | WO | 0x00 | Clear the receive channel |
| PENDING | 5:5 | RO | 0x00 | Receive transaction is pending |
| EN | 4:4 | RW | 0x00 | Enable the receive channel |
| DATASIZE | 2:1 | RW | 0X02 | Controls uDMA address increment |
|  |  |  |  | 0x00: increment address by 1 (data is 8 bits)  0x01: increment address by 2 (data is 16 bits) |
|  |  |  |  | 0x02: increment address by 4 (data is 32 bits) |
|  |  |  |  | 0x03: increment address by 0. |
| CONTINUOUS | 0:0 | RW | 0x00 | 0x0: stop after last transfer for channel |
|  |  |  |  | 0x1: after last transfer for channel |
|  |  |  |  | Reload buffer size and start address and restart channel. |

REG\_CAM\_CFG\_GLOB

Offset = 0x20

| Field | Bits | Type | Default | Description |
| --- | --- | --- | --- | --- |
| EN | 31:31 | RW | 0x00 | Enable data RX from camera interface |
|  |  |  |  | Enable/disable only happens at start of frame |
|  |  |  |  | 0x0: disable |
|  |  |  |  | 0x1: enable |
| SHIFT | 14:11 | RW | 0x00 | Number of bits to right shift final pixel value |
|  |  |  |  | Note: not used if FORMAT == BYPASS |
| FORMAT | 10:8 | RW | 0x00 | Input frame format: |
|  |  |  |  | 0x0: RGB565 |
|  |  |  |  | 0x1: RGB555 |
|  |  |  |  | 0x2: RGB444 |
|  |  |  |  | 0x4: BYPASS\_LITTLEEND |
|  |  |  |  | 0x5: BYPASS\_BIGEND |
| FRAMEWINDOW\_EN | 7:7 | RW | 0x00 | Windowing enable: |
|  |  |  |  | 0x0: disable |
|  |  |  |  | 0x1: enable |

REG\_CAM\_CFG\_LL:

Offset = 0x24

| Field | Bits | Type | Default | Description |
| --- | --- | --- | --- | --- |
| FRAMEWINDOW\_LLY | 31:16 | RW | 0x00 | Y coordinate of lower left corner of window. |
| FRAMEWINDOW\_LLX | 15:0 | RW | 0x00 | X coordinate of lower left corner of window. |

REG\_CAM\_CFG\_UR:

Offset=0x28

| Field | Bits | Type | Default | Description |
| --- | --- | --- | --- | --- |
| FRAMEWINDOW\_URY | 31:16 | RW | 0x00 | Y coordinate of upper right corner of window. |
| FRAMEWINDOW\_URX | 15:0 | RW | 0x00 | X coordinate of upper right corner of window. |
|  |  |  |  |  |

REG\_CAM\_CFG\_SIZE

Offset = 0x2C

| Field | Bits | Type | Default | Description |
| --- | --- | --- | --- | --- |
| ROWLEN | 31:16 | RW | 0x00 | N-1 where N is the number of horizontal pixels |
|  |  |  |  | (used in window mode) |

REG\_CAM\_CFG\_FILTER

Offset = 0x30

| Field | Bits | Type | Default | Description |
| --- | --- | --- | --- | --- |
| R\_COEFF | 23:16 | RW | 0x00 | Coefficent that multiplies R component |
|  |  |  |  | Note: not used if FORMAT == BYPASS |
| G\_COEFF | 15:8 | RW | 0x00 | Coefficent that multiplies G component |
|  |  |  |  | Note: not used if FORMAT == BYPASS |
| B\_COEFF | 7:0 | RW | 0x00 | Coefficent that multiplies B component |
|  |  |  |  | Note: not used if FORMAT == BYPASS |

REG\_CAM\_VSYNC\_POLARITY

Offset=0x34:

| Field | Bits | Type | Default | Description |
| --- | --- | --- | --- | --- |
| VSYNC\_POLARITY | 0:0 | R/W | 0x00 | Set vsync polarity: |
|  |  |  |  | 0x0: Active low |
|  |  |  |  | 0x1: Active high |