CMOS Sensor Output Generator

Sahand Kashani

February 28, 2016

1 Core Overview

The cmos_sensor_output_generator core is an interface that generates the signals a CMOS sensor would normally output. The core is useful for testing CMOS sensor acquisition systems when the production sensor is not yet available.

The core is configurable at runtime through an Avalon Memory-Mapped (Avalon-MM) interface, and provides a Conduit interface identical to one a CMOS sensor would have.

Additionally, the core comes with a set of C library interfaces that can be used to configure it, as well as to start and stop its operation.

2 Generated Waveform

A CMOS sensor outputs 4 signals with which it is possible to sample its data:

- clock
- frame_valid (1-bit)
- line_valid (1-bit)
- data (n-bit)

Figure 1 shows the relationship between the different signals for 2 frames that contain 2 rows and 3 columns each.

The cmos_sensor_output_generator behaviour can be summarized as follows: The unit outputs sequentially increasing values as pixel data. The first pixel of each frame is assigned the value 0. This value is incremented for each subsequent pixel in the frame.

The maximum possible value is determined by the bit width of the unit's data output port. If the maximum value is smaller than the number of pixels in a frame, the output wraps around and restarts counting from 0.

This guarantees a deterministic output and allows one to verify CMOS sensor acquisition systems for correct behaviour.

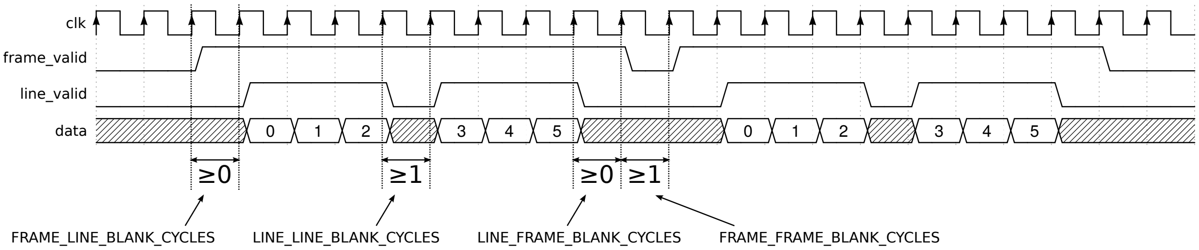


Figure 1: CMOS sensor output signals for two 2×3 frames with a pixel depth of 3 bits. Spacing requirements between the various signals are specified in clock cycles. The labels given to the 4 spacing intervals are the same ones used later in Section 4 to configure the core.

3 Block Diagram

Figure 2 shows a high-level view of the core. The `frame_valid`, `line_valid`, and `data` signals are generated with the same clock frequency as the core's input clock.

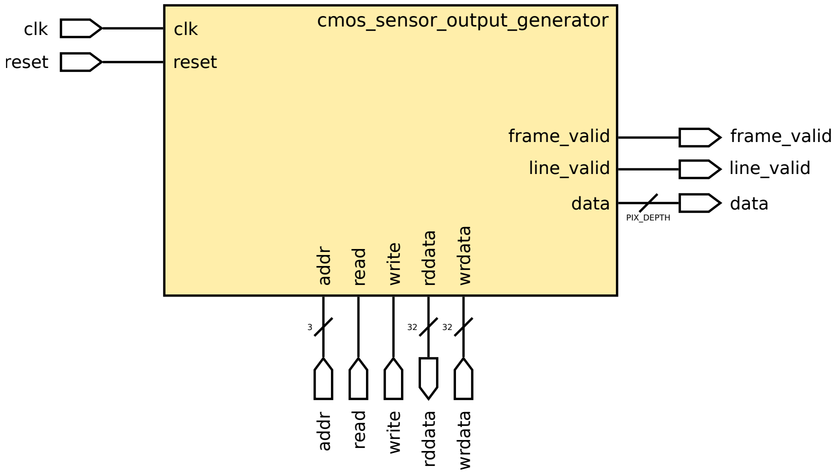


Figure 2: Block diagram.

Note that the core does not output the clock it receives as input as a normal CMOS sensor would do, because it is bad practice to route a clock from an FPGA's clock tree through standard logic in order to output it outside the clock tree.

It is the user's responsibility to use the same clock that is driving the core for any other component down the acquisition pipeline that requires the CMOS sensor's generated output clock.

4 Register Map

The core is configured through multiple CONFIG_ registers, shown in Table 1. If a configuration register is written with a value smaller than those specified in column Minimum, then the core has undefined behaviour.

Offset	Type	Name	Default Value	Minimum
0x00	RW	CONFIG_FRAME_WIDTH	1	1
0x04	RW	CONFIG_FRAME_HEIGHT	1	1
0x08	RW	CONFIG_FRAME_FRAME_BLANK	1	1
0x0C	RW	CONFIG_FRAME_LINE_BLANK	0	0
0x10	RW	CONFIG_LINE_LINE_BLANK	1	1
0x14	RW	CONFIG_LINE_FRAME_BLANK	0	0
0x18	WO	COMMAND	N/A	N/A
0x1C	RO	STATUS	N/A	N/A

Table 1: Register map. Note that the CONFIG_ registers can only be modified if the core is not running (you must issue a STOP command before modifying any of these registers).

You can start or stop the controller by writing a command to its COMMAND register, shown in Table 2. Upon reception of a STOP command, the controller halts immediately and does not wait for the current frame to end.

Name	Value	Description
STOP	0	Stop generation
START	1	Start generation

Table 2: COMMAND Register Definitions.

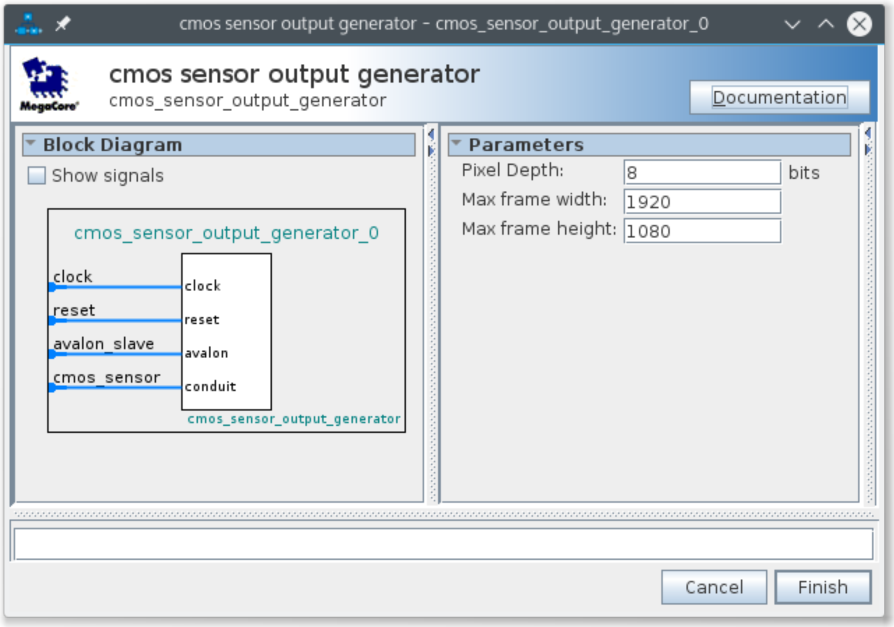
The STATUS register is described in Table 3.

Name	Value	Description
BUSY	0	Controller busy
IDLE	1	Controller idle

Table 3: STATUS Register Definitions.

5 Qsys Interface

Figure 3 shows the Qsys configuration interface for the core.


cmos sensor output generator - cmos_sensor_output_generator_0
cmos sensor output generator
cmos_sensor_output_generator
MegaCore
Documentation
Block Diagram
□ Show signals
cmos_sensor_output_generator_0
clock
reset
avalon_slave
cmos_sensor
clock
reset
avalon
conduit
cmos_sensor_output_generator
Parameters
Cancel
Finish

Pixel Depth: 8 bits
Max frame width: 1920
Max frame height: 1080

Figure 3: Qsys Configuration Interface.

Three paramters are needed to instantiate the core:

- `Pixel depth`
- `Max frame width`
- `Max frame height`

The `Max frame width` and `Max frame height` entries are used to determine the size of the internal registers. This helps keep the unit's footprint small if small resolutions are required.