EF_QSPI_XIP_CTRL

A QSPI XiP Flash COntroller with a parameterized Direct-Mapped Cache.

The wrapped IP

The IP comes with an AHBL Wrapper

Wrapped IP System Integration

```
EF_QSPI_XIP_CTRL_AHBL INST (
        .HCLK(CLK),
        .HRESETn(RESETn),
        .HADDR(HADDR),
        .HWRITE(HWRITE),
        .HSEL(HSEL),
        .HTRANS(HTRANS),
        .HWDATA(HWDATA),
        .HRDATA(HRDATA),
        .HREADY(HREADY),
        .HREADYOUT(HREADYOUT),
        .sck(sck),
        .ce_n(ce_n),
        .din(din),
        .dout(dout),
        .douten(douten)
);
```

External IO interfaces

IO name	Direction	Width	Description
sck	output	1	SPI serial clock
ce_n	output	1	SPI chip select (Active Low).
dout	output	4	Flash controller SPI data out.
din	input	4	Flash controller SPI data in.
douten	output	4	Flash controller data out enable (Active Low)

Implementation example

The following table is the result for implementing the EF_QSPI_XIP_CTRL IP with different wrappers using Sky130 HD library and OpenLane2 flow.

Module	Number of cells	Max. freq
EF_QSPI_XIP_CTRL	1973	250
EF_QSPI_XIP_CTRL_AHBL	1973	250

Installation:

You can install the IP either by cloning this repository or by using IPM.

1. Using IPM:

- [Optional] If you do not have IPM installed, follow the installation guide here
- After installing IPM, execute the following command `ipm install EF_QSPI_XIP_CTRL` .

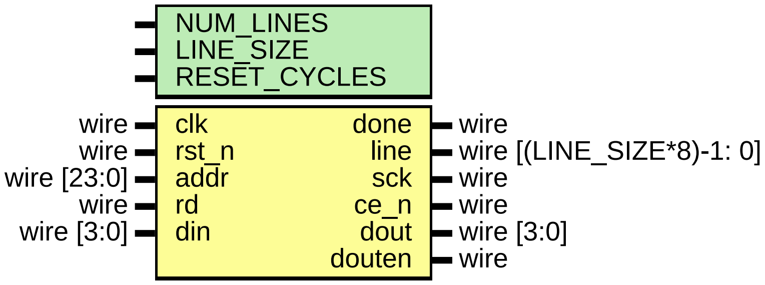
> **Note:** This method is recommended as it automatically installs EF_IP_UTIL as a dependency.

2. Cloning this repo:

- Clone EF_IP_UTIL repository, which includes the required modules from the common modules library, ef_util_lib.v. `git clone https://github.com/efabless/EF_IP_UTIL.git`
- Clone the IP repository `git clone github.com/efabless/EF_QSPI_XIP_CTRL`

The Wrapped IP Interface

> ***NOTE:*** This section is intended for advanced users who wish to gain more information about the interface of the wrapped IP, in case they want to create their own wrappers.



Module Parameters

Parameter	Description	Default Value
NUM_LINES	The cache number of lines.	16
LINE_SIZE	The cache line size in bytes.	32
RESET_CYCLES	The number of cycles needed for the s/w reset command; reset time = (RESET_CYCLES + 1) * 2 /(HCLK frequency).	999

Ports

Port	Direction	Width	Description
sck	output	1	SPI serial clock
ce_n	output	1	SPI chip select (Active Low).
dout	output	4	Flash controller SPI data out.
din	input	4	Flash controller SPI data in.
douten	output	4	Flash controller data out enable (Active Low)