
Mistral documentation

Release 1.0

Dec 27, 2020

CONTENTS

1	The Cyclone V FPGA	1
1.1	The FPGAs	1
1.2	Bitstream stucture	2
1.3	Logic blocks	2
1.4	Routing network	3
1.5	Programmable inverters	3
2	CycloneV internals description	5
2.1	Routing network	5
2.2	Inner logic blocks	6
2.3	Peripheral logic blocks	6

THE CYCLONE V FPGA

1.1 The FPGAs

The Cyclone V is a series of FPGAs produced initially by Altera, now Intel. It is based on a series of seven dies with varying levels of capability, which is then derived into more than 400 SKUs with variations in speed, temperature range, and enabled internal hardware.

As pretty much every FPGA out there, the dies are organized in grids.

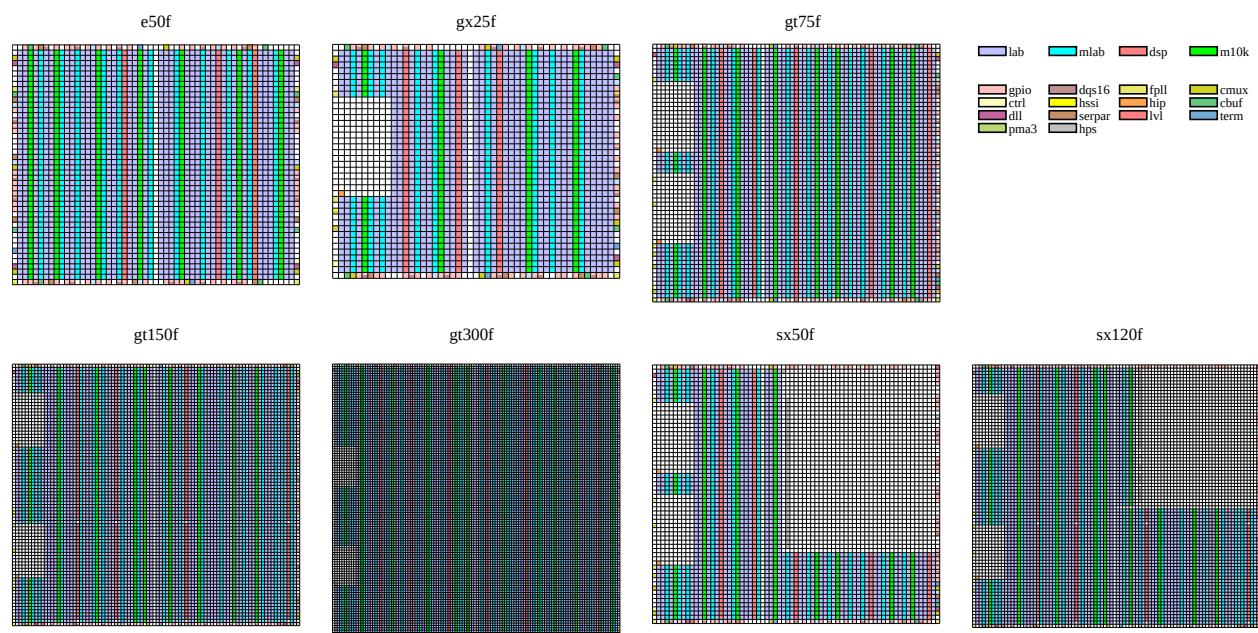


Fig. 1: Floor plan of the seven die types

The FPGA, structurally, is a set of logic blocks of different types communicating with each other either through direct links or through a large routing network that spans the whole grid.

Some of the logic blocks take visible floor space. Specifically, the notches on the left are the space taken by the high speed serial interfaces (hssi and pma3). Also, the top-right corner in the sx50f and sx120f variants is used to fit the hps, a dual-core arm.

1.2 Bitstream structure

The bitstream is build from three rams:

- Option ram
- Peripheral ram
- Configuration ram

The option ram is composed of 32 blocks of 40 bits, of which only 12 are actually used. It includes the global configurations for the chip, such as the jtag user id, the programming voltage, the internal oscillator configuration, etc.

The peripheral ram stores the configuration of all the blocks situated on the borders of the chip, e.g. everything outside of labs, mlabs, dsps and m10ks. It is build of 13 to 16 blocks of bits that are sent through shift registers to the tiles.

The configuration ram stores the configuration of the labs, mlabs, dsps and m10ks, plus all the routing configuration. It also includes the programmable inverters which allows inverting essentially all the inputs to the peripheral blocks. It is organised as a rectangle of bits.

Die	Tiles	Pram	Cram
e50f	55x46	51101	4958x3928
gx25f	49x40	54083	3856x3412
gt75f	69x62	90162	6006x5304
gt150f	90x82	113922	7605x7024
gt300f	122x116	130828	10038x9948
sx50f	69x62	80505	6006x5304
sx120f	90x82	99574	7605x7024

1.3 Logic blocks

The logic blocks are of two categories, the inner blocks and the peripheral blocks. To a first approximation all the inner blocks are configured through configuration ram, and the peripheral blocks through the peripheral ram. It only matters where it comes to partial reconfiguration, because only the configuration ram can be dynamically modified. We do not yet support it though.

The inner blocks are:

- lab: a logic blocks group with 20 LUTs with 5 inputs and 40 Flip-Flops.
- mlab: a lab that can be reconfigured as 64*20 bits of ram
- dsp: a flexible multiply-add block
- m10k: a block of 10240 bits of dual-ported memory

The peripheral blocks are:

- gpio: general-purpose i/o, a block that controls up to 4 package pins
- dqs16: a block that manage differential input/output for 4 gpio blocks, e.g. up to 16 pins
- fppll: a fractional PLL
- cmux: the clock muxes that drive the clock part of the routing network
- ctrl: the control block with things like jtag
- hssi: the high speed serial interfaces

- hip: the pcie interfaces
- cbuf: a clock buffer for the dqs16
- dll: a delay-locked loop for the dqs16
- serpar: TODO
- lvl: TODO
- term: termination control blocks
- pma3: manages the channels of the hssi
- hmc: hardware memory controller, a block managing sdr/ddr ram interfaces
- hps: a series of 37 blocks managing the interface with the integrated dual-core arm

All of these blocks are configured similarly, through the setup of block muxes. They can be of 4 types: * Boolean
 * Symbolic, where the choice is between alphanumeric states * Numeric, where the choice is between a fixed set of numeric value * Ram, where a series of bits can be set to any value

Configuring that part of the FPGA consists of configuring the muxes associated to each block.

1.4 Routing network

A massive routing network is present all over the FPGA. It has two almost-disjoint parts. The data network has a series of inputs, connected to the outputs of all the blocks, and a series of outputs that go to data inputs of the blocks. The clock network consists of 16 global clocks signals that cover the whole FPGA, up to 88 regional clocks that cover an half of the FPGA, and when an hssi is present a series of horizontal peripheral clocks that are driven by the serial communications. Global and regional clock signals are driven by dedicated cmux blocks (not the fppl in particular, but they do have dedicated connections to the cmuxes).

These two networks join on data/clock muxes, which allow peripheral blocks to select for their clock-like inputs which network the signal should come from.

1.5 Programmable inverters

Essentially every output of the routing network that enters a peripheral block can optionally be inverted by activating the associated configuration bit.

CYCLONEV INTERNALS DESCRIPTION

2.1 Routing network

The routing network follows a single-driver structure: a number of inputs are grouped together in one place, one is selected through the configuration, then it is amplified and used to drive a metal line. There is also usually one bit configuration to disable the driver, which can be all-off (probably leaving the line floating) or a specific combination to select vcc. The drivers correspond to a 2d pattern in the configuration ram. There are 70 different patterns, configured by 1 to 18 bits and mixing 1 to 44 inputs.

The network itself can be split in two parts: the data network and the clock network.

The data network is a grid of connections. Horizontal lines (H14, H6 and H3, numbered by the number of tiles they span) and vertical lines (V12, V4 and V2) helped by wire muxes (WM) connect to each other to ensure routing over the whole surface. Then at the tile level tile-data dispatch (TD) nodes allow to select between the available signals.

Generic output (GOUT) nodes then select between TD nodes to connect to logic blocks inputs. Logic block outputs go to Generic Input (GIN) nodes which feed in the connections. In addition a dedicated network, the Loopback dispatch (LD) connects some of the outputs from the labs/mlabs to their inputs for fast local data routing.

The clock network is more of a top-down structure. The top structures are Global clocks (GCLK), Regional clocks (RCLK) and Peripheral clocks (PCLK). They're all driven by specialized logic blocks we call Clock Muxes (cmux). There are two horizontal cmux in the middle of the top and bottom borders, each driving 4 GCLK and 20 RCLK, two vertical in the middle of the left and right borders each driving 4 GCLK and 12 RCLK, and 3 to 4 in the corners driving 6 RCLK each. The dies including an HPS (sx50f and sx120f) are missing the top-right cmux plus some of the middle-of-border-driven RCLK. That gives a total of 16 GCLK and 66 to 88 RCLK. In addition PCLK start from HSSI blocks to distribute serial clocks to the network.

The GCLK span the whole grid. A RCLK spans half the grid. A PCLK spans a number of tiles horizontally to its right.

The second level is Sector clocks, SCLK, which spans small rectangular zones of tiles and connect from GCLK, RCLK and PCLK. The on the third level, connecting from SCLK, is Horizontal clocks (HCLK) spanning 10-15 horizontal tiles and Border clocks (BCLK) rooted regularly on the top and bottom borders. Finally Tile clocks (TCLK) connect from HCLK and BCLK and distribute the clocks within a tile.

In addition the PMUX nodes at the entrance of plls select between SCLKs, and the GCLKFB and RCLKFB bring back feedback signals from the cmux to the pll.

Inner blocks directly connect to TCLK and have internal muxes to select between clock and data inputs for their control. Peripheral blocks tend to use a secondary structure composed from a TDMUX that selects one TD between multiple ones followed by a DCMUX that selects between the TDMUX and a TCLK so that their clock-like inputs can be driven from either a clock or a data signal.

Most GOUT and DCMUX connected to inputs to peripheral blocks are also provided with an optional inverter.

2.2 Inner logic blocks

2.2.1 LAB

The LABs are the main combinatorial and register blocks of the FPGA. A LAB tile includes 10 sub-blocks with 64 bits of LUT splitted in 6 parts, four Flip-Flops, two 1-bit adders and a lot of routing logic. In addition a common control subblock selects and dispatches clock, enable, clear, etc signals.

2.2.2 MLAB

A MLAB is a lab that can optionally be turned into a 640-bits RAM or ROM. The wiring is identical to the LAB, only some additional muxes are provided to select the RAM/ROM mode.

TODO: address/data wiring in RAM/ROM mode.

2.2.3 DSP

The DSP blocks provide a multiply-adder with either three 9x9, two 18x18 or one 27x27 multiply, and the 64-bits accumulator. Its large number of inputs and output makes it span two tiles vertically.

TODO: everything

2.2.4 M10K

The M10K blocks provide 10240 (256*40) bits of dual-ported rom or ram.

TODO: everything

2.3 Peripheral logic blocks

2.3.1 GPIO

The GPIO blocks connect the FPGA with the exterior through the package pins. Each block controls 4 pads, which are connected to up to 4 pins.

TODO: everything, GOUT/GIN/DCMUX mapping is done

2.3.2 DQS16

The DQS16 blocks handle differential signaling protocols. Each supervises 4 GPIO blocks for a total of 16 signals, hence their name.

TODO: everything

2.3.3 FPLL

The Fractional PLL blocks synthesize 9 frequencies from an input with integer or fractional ratios.

TODO: everything, GOUT/GIN/DCMUX mapping is done

2.3.4 CMUX

The Clock mux blocks drive the GCLK and the RCLK.

TODO: fpll feedback lines

2.3.5 CTRL

The Control block gives access to a number of ancillary functions of the FPGA.

TODO: everything, GOUT/GIN/DCMUX mapping is done

2.3.6 HSSI

The High speed serial interface blocks control the serializing/deserializing capabilities of the FPGA.

TODO: everything

2.3.7 HIP

The PCIe Hard-IP blocks control the PCIe interfaces of the FPGA.

TODO: everything

2.3.8 DLL

The Delay-Locked loop does phase control for the DQS16.

TODO: everything

2.3.9 SERPAR

Unclear yet.

TODO: everything

2.3.10 LVL

The Leveling Delay Chain does something linked to the DQS16.

TODO: everything

2.3.11 TERM

The TERM blocks control the On-Chip Termination circuitry

TODO: everything

2.3.12 PMA3

The PMA3 blocks control triplets of channels used with the HSSI.

TODO: everything

2.3.13 HMC

The Hardware memory controller controls sets of GPIOs to implement modern SDR and DDR memory interfaces. In the sx dies one of them is taken over by the HPS. They can be bypassed in favor of direct access to the GPIOs.

TODO: everything, and in particular the hmc-input -> GPIO input mapping when bypassed.

2.3.14 HPS

The interface between the FPGA and the Hard processor system is done through 37 specialized blocks or 28 different types.

TODO: everything. GOUT/GIN/DCMUX mapping is done except for HPS_CLOCKS.

HPS_BOOT

HPS_CLOCKS

HPS_CLOCKS_RESETS

HPS_CROSS_TRIGGER

HPS_DBG_APB

HPS_DMA

HPS_FPGA2HPS

HPS_FPGA2SDRAM

HPS_HPS2FPGA

HPS_HPS2FPGA_LIGHT_WEIGHT

HPS_INTERRUPTS

HPS_JTAG

HPS_LOAN_IO

HPS_MPU_EVENT_STANDBY

HPS_MPU_GENERAL_PURPOSE

HPS_PERIPHERAL_CAN

(2 blocks)

HPS_PERIPHERAL_EMAC

(2 blocks)

HPS_PERIPHERAL_I2C

(4 blocks)

HPS_PERIPHERAL_NAND

HPS_PERIPHERAL_QSPI

HPS_PERIPHERAL_SDMMC

HPS_PERIPHERAL_SPI_MASTER

(2 blocks)

HPS_PERIPHERAL_SPI_SLAVE

(2 blocks)

HPS_PERIPHERAL_UART

(2 blocks)

HPS_PERIPHERAL_USB

(2 blocks)

HPS_STM_EVENT

HPS_TEST

HPS_TPIU_TRACE