

The Menta® eFPGA Core® is a high-density embedded programmable logic core IP. It is highly customizable for target applications and can be used in wide range of application like telecommunications, networking, digital signal processing, high-performance computing, video, image processing, industry, space and military.

The eFPGA Core is specially designed to provide a programmable solution to the industry for solving several key challenges. The small dedicated eFPGA Core IP inside a SoC can provides numerous benefits like:

- fast time to market
- product differentiation
- power consumption reduction
- IO flexibility
- future up gradation

It has been created as a highly customizable, very compact, fast and power efficient IP. The eFPGA Core technology is independent and allows the freedom to target any process node or fab immediately and remove any kind of barriers for the customers.

The eFPGA Core family consists of the eFPGA Core-S™ (synthesizable) and the eFPGA Core-H™ (hard macro) programmable logic architectures.

The eFPGA Core-S is the synthesizable version of the eFPGA Core and available through the eFPGA Creator® design environment development tool suite. It is ideal for low volume production and when only small level of programmability is required (200K gates).

The eFPGA Core-H, the hard macro cell version of the eFPGA Core-S, is for mass production. It is a full-custom implementation for a specific technology node of a specific foundry can be provided to have a very high silicon efficient IP.

Architecture Overview

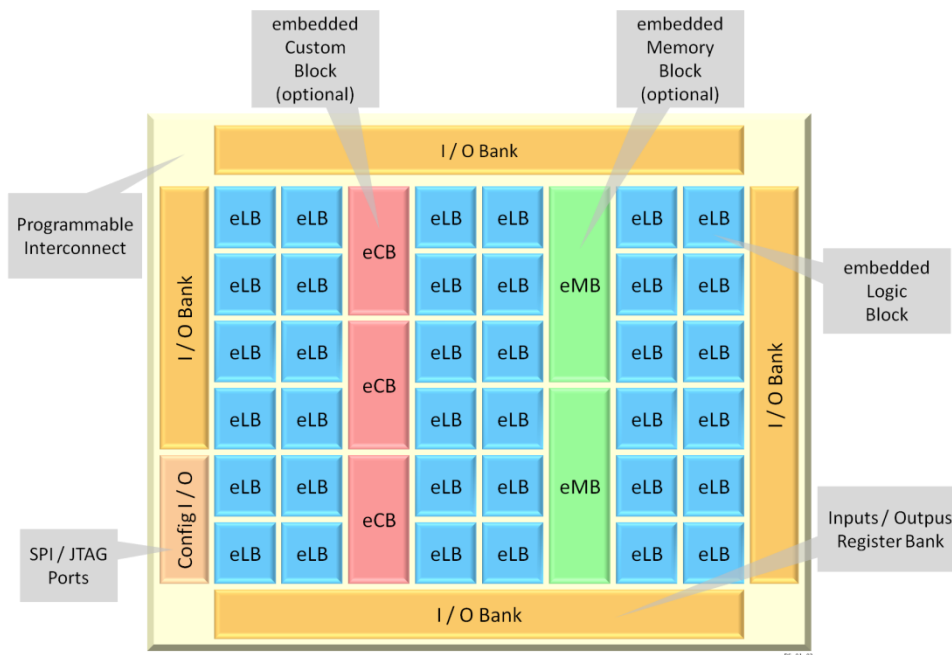


Figure 1: eFPGA Core architectural features

Figure 1 presents a simplified, conceptual overview of the eFPGA Core architecture. The architecture is highly scalable and regular. Since eFPGA Core is an IP this gives the advantage of greater degree of customization compared to classical FPGA device. The eFPGA Core architecture consists of programmable logic blocks, but unlike other FPGAs, the amount, type, size and position of blocks can be customized based on target. The eFPGA Core is composed of five basic elements which are connected to the programmable interconnect system through a Switch Box (SB).

- eLB (embedded Logic Block)
- eMB (embedded Memory Block)
- eAB (embedded Arithmetic Block)
- ePB (embedded Processor Block)
- eCB (embedded Custom Block)

eLB (embedded Logic Block)

The eLB is the basic configurable element of eFPGA Core. It provides combinatorial and synchronous logic capability. It consists of cluster of lookup-up tables (LUTs) and Flip Flops as shown in Figure 2, and provides the fine grained programmability for implementing any kind of logic functions. The size of LUT can be customized (LUT-3 to LUT-6) and there can also be a mix of different LUT sizes in an eLB.

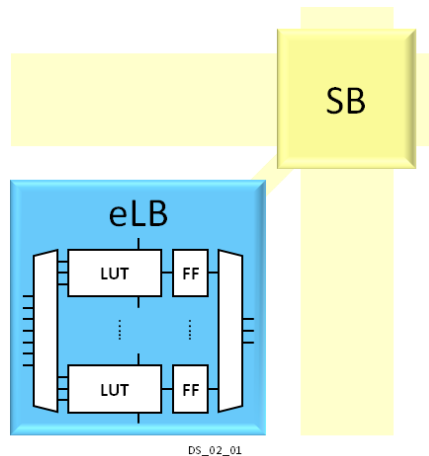


Figure 2: eLB overview

eMB (embedded Memory Block)

The eMB is the embedded SRAM memory block. The quantity (number of eMB blocks), position (in the middle, corners etc.) and properties (single port/dual port, data width, size etc.) are customizable based on the target application. The next section will highlight silicon properties of some of the possible eMBs.

eAB (embedded Arithmetic Block)

The eAB is the basic arithmetic element. It can be a simple ALU, multiplier, complex DSP element etc. Like eMB, the properties, quantity and position of eAB is customizable based on target application requirements.

ePB (embedded Processor Block)

The ePB is the embedded hard processor IP block. This element in addition to eAB provides unique capabilities to eFPGA Core by combining the powers of small RISC machines along with heterogeneous programmable elements of eFPGA Core (both fine and coarse grained). They can be built with small memory of their own or can use the eMB blocks.

The ePB block can be from 3rd party also (like Leon, MIPS, ARM ...) which will bring the advantages of rich software applications available for those processors. The 3rd party ePB will require special licensing and

royalty fees depending on decisions of 3rd party ePB providers. Menta will provide services and assistance for integrating these blocks in eFPGA.

The next section will provide some silicon details of a small RISC processor which our customers will get free with our eFPGA Core IP including its compiler, tools and several applications.

eCB (embedded Custom Block)

The eCB is the most distinguishing element of Menta eFPGA Core from other FPGAs. As eFPGA Core is an IP compared to a chip, this provides greater freedom of customization. Based on the target application domain of the SoC special custom blocks (including analog) can be integrated in the eFPGA Core in the form of eCB. They can be components like complete cryptographic cores, PLLs, SerDes etc.

Like ePB, eCB can also be from 3rd party or can be from the SoC vendor itself. Menta will provide services and assistance in integrating these special blocks in eFPGA Core.

eFPGA on Silicon

The Menta eFPGA Core IP is technology independent which provides the opportunity to migrate to any process node of same or any other fab. The compact interconnect architecture of eFPGA Core allows efficient mapping within 5-7 metal layers making it highly suitable for SoC integration without any need of additional expensive higher metal layers which are only suitable for FPGAs. This section highlights some of the features of eFPGA Core on process.

eLB

The table shows some key features like area, static power and speed of embedded Logic Block with four LUT6 per eLB for a eFPGA Core-S delivery on different technology nodes of STMicroelectronics.

Process	90 nm GP	65 nm LP	40 nm LP
Logic Density* (LUT-6/mm²)	150	380	740
Approximate Equivalent Gate / mm²	25K	60K	120K
eLB speed* (MHz)	1 800	1 000	1 000
Static Power* (Tile)	<35μW	< 1μW	<1μW
Configuration Bits (1 eLB+S)	620 bits		

* With nominal core voltage (1.0V for 90 and 65nm, 0.95V for 45nm), standard threshold voltage (SVT) and at 25°C.

Table 1: eFPGA Core-S results

These results are for the eFPGA Core-S implementation. The logic density of eFPGA Core-H targeted to a specific technology node with full-custom implementation can be up to 4 or 5 times more.

eMB

	Kbits	Area (mm ²)	Static Power* (μW)	Speed (MHz)
eMB_SP-16x512*	8	0.0109	0.9	900+
eMB_SP-32x512	16	0.0195	1.6	900+
eMB_DP-16x512*	8	0.0203	2.8	900+
eMB_DP-32x512	16	0.0337	4.2	900+

* SP (Single Port), DP (Dual Port), Static power in nominal conditions at 25°C

Table 2: eMB on 65 nm

The eMB can be of different sizes and types (single/dual port etc.). The table shows silicon properties of some of the eMB models (on ST 65nm LP). As an example eMB_DP-32x512 means that it's a dual port memory block with data width of 32 bits and depth of 512.

ePB

The table below shows the silicon properties of ePB-M1 processor (on ST 65nm LP) which is available free of charge to eFPGA Core customers. It's a small 32-bit MIPS like RISC processor with single instruction and data memory (Von Neumann). It has no cache, MMU or FPU. It has hardware multiplier and divider (multi cycle). It has a very small silicon foot print.

	Core Area (mm ²)	Static Power* (μW)	Dynamic Power* (μW/MHz)	Speed (MHz)
ePB-M1	0.0282	<10	<10	175+

* Under nominal conditions at 25°C

Table 3: ePB on 65 nm

System Integration

The eFPGA Core IP is integrated like classical ASIC flow for integrating IPs in the SoC. The scripts needed by the backend ASIC tools will be provided with the eFPGA IP.

The integration of eFPGA Core in the system is very simple. The proprietary very low gate count interface architecture of Menta makes it simple to integrate the eFPGA Core with buses (e.g. AMBA) and processors (as a coprocessor).

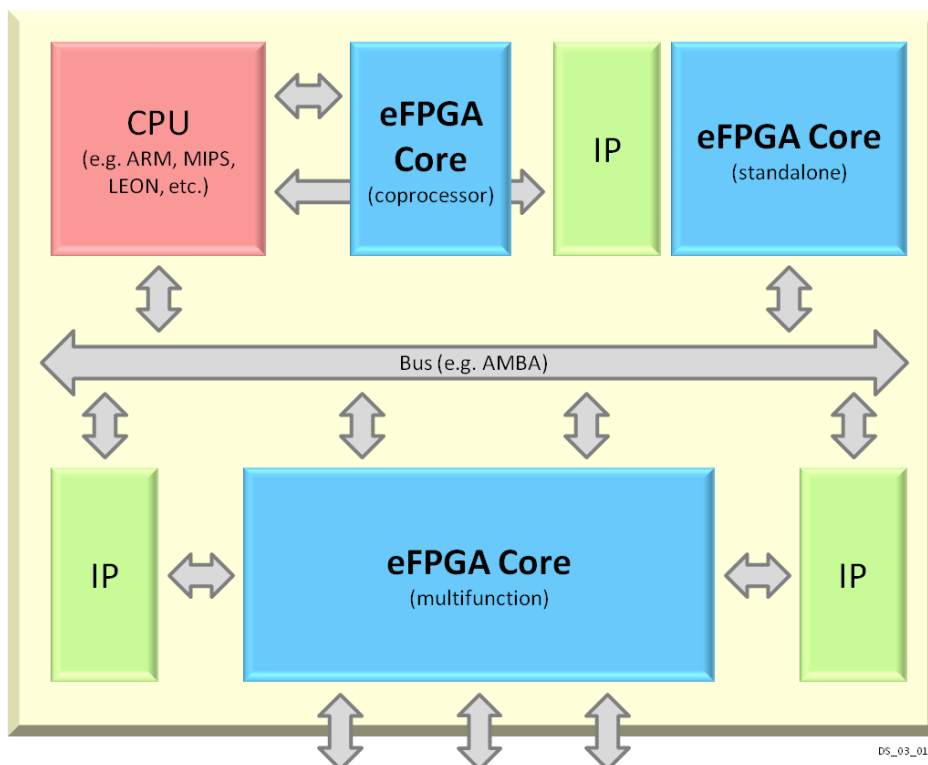


Figure 3: System integration example

It also facilitates to provide IOs count needed for the target domain (from low to very high). As an example a small eFPGA Core of an array of 12x12 eCBs (144 Tiles) can have an IO count from as low as 70

to 4000+. This highlights the possibilities which eFPGA Core can open to SoC designers. For I/O banks (in case eFPGA Core is used for interfacing with outside world) we are using the standard pads provided by the design kit. They provide several kinds of pads which can be selected depending on target needs.

Some revolutionary concepts are shown in figure with a large multifunction eFPGA Core IP. Due to parallel nature of the eFPGA Core it can at the same time provide IO flexibilities to outside world, multiple connections with the bus and even direct inter communications with communication intensive IPs.

Due to customizable nature of eFPGA Core, different eFPGA Core IPs can be added in the SoC (shown in figure 3) providing features like product differentiation, reconfigurable acceleration, testing of new IPs in the SoC and much more. These features which eFPGA Core enables in the SoC are very hard to achieve with classical FPGAs which are available as a complete silicon solution for a relatively wide range of applications.

Programming flow

The eFPGA Core IP comes with the complete tools suite eFPGA Programmer® which is completely free for our customers. eFPGA Programmer uses commercial synthesis tools on front-end for synthesis. It is integration ready with the industry leading Design Compiler of Synopsys (not part of eFPGA Programmer, separate license is needed). In addition direct ESL programming from ANSI C/C++ through Catapult of Mentor Graphics for generating RTL for synthesis is also supported.

On the back-end eFPGA Programmer has complete flow for logic mapping, clustering, place & route and programming. In addition it also provides the possibility to generate simulation models at each step of the flow (with or without back-annotation SDF timing simulation) to facilitate verifications at each step.

The eFPGA Core architecture file provides all the necessary information about the architecture to eFPGA Programmer. The key information are like size of eFPGA Core, quantity and positions of Hard macro blocks (eMB, ePB etc.), Number and type of IOs, timing information of all the elements etc.

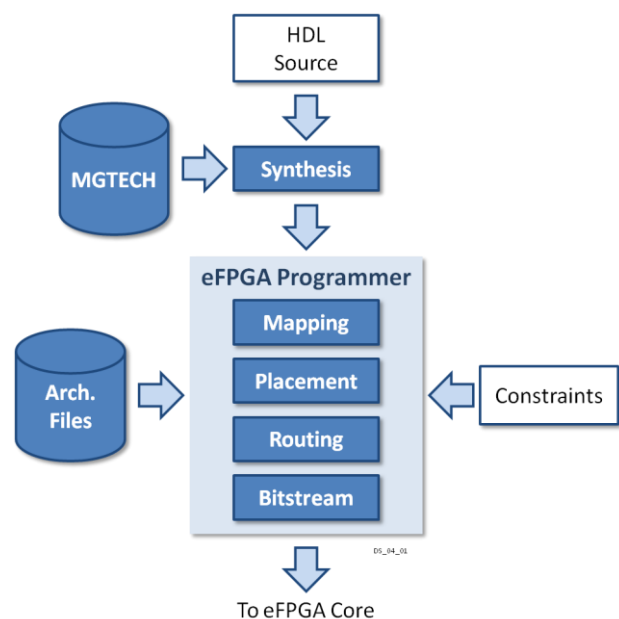


Figure 4: Programming flow

Configuration system

The embedded Configuration Unit (eCU) loads the configuration bitstream to the eFPGA Core. It can program eFPGA Core from variety of sources like DMA transfers through AMBA bus etc. It can be fully controlled by software through the main processor of SoC.

Deliverables

The eFPGA Core has three possible deliveries: Synthesizable, Firm or Hard.

Synthesizable delivery

A synthesizable delivery (eFPGA Core-S) includes all the HDL files. This kind of delivery provides greatest amount of freedom to the clients and will be subjected to a special contract and conditions as security requirements.

Firm delivery

A firm delivery includes pre-synthesized code and timing constraints files (sdc) of an eFPGA Core-S delivery. The customer will have flexibility for making the place and route of his SoC.

Hard delivery

A hard delivery (eFPGA Core-H) is the virtual component placed and routed in a specific technology (netlist + lef + timing socket). The timing socket (=liberty files) is a model which represents the different timing arc between IO of the IP:

- Timing arc between input and clock
- Timing arc between clock and output
- Timing arc between input and output

The eFPGA IP can be included in an automatic IC design flow:

- Synthesis at SoC level
- Perform STA at SoC level
- Place & Route at SoC level

Timing socket utilization:

- Main goal : eFPGA integration in SoC
 - o Logic glue = digital design using technologic library (one lib per corner Min and Max)
 - o eFPGA with 2 liberty files (1 for each corner : Min and Max)
- Synthesis with the following lib files:
 - o Specific technologic library corner max
 - o eFPGA_TimingSocket_Max (corner MAX)
- Place&Route with the following lib files :
 - o Run step : specific technologic library MAX/MIN + eFPGA_TimingSocket_Max/MIN
 - o Verification step : STA sign-off
 - o Optional correction steps : ECO

Others deliveries can be provided if needed in talks with the customer support.