



Enclustra

The perfect FPGA based System-on-Module for every application

June 2021

Diana Ungureanu

Manager, BU Standard Platforms



Who is Enclustra?

- Founded in 2004, based in Zurich, >50 employees
- Pioneer in SOM development -> first product in 2008
- Largest offering of FPGA and SOC modules on the market
 - 30+ standard products
 - 90+ product models
 - Three product families
- Over 1000 customers in 69 countries
- Sold over 70 000 modules
- Offering Modules and Design Services
 - Customized to your needs



Who is Enclustra?



Enclustra Global Offices & Sales Agents



Distribution: Avnet, Digi-Key and Enclustra Partners

- We help customers to:
 - Reduce time to market (TTM)
 - Reduce total cost of ownership (TCO)
 - Differentiate their products
- We deliver:
 - Highest proven quality
 - Standard and custom-made products
 - One of the largest SOM product offerings on the market



Our Offering



System on Modules



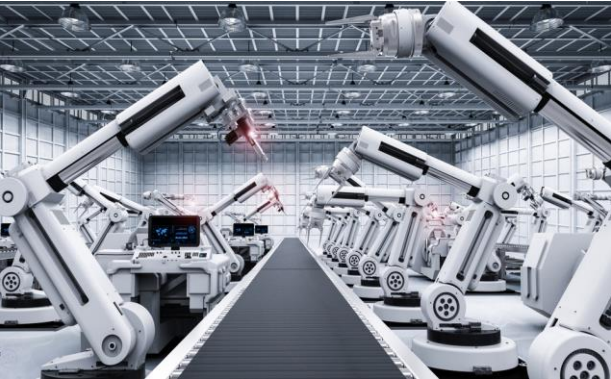
Design Services



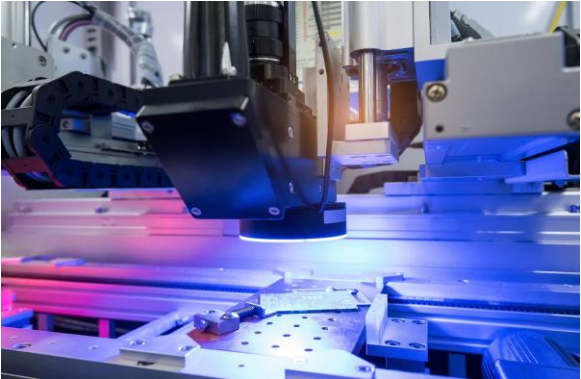
IP Cores

SOM Key Target Markets

ROBOTICS



VISION



SECURITY



MEASUREMENT



INDUSTRIAL





MEDICAL




- Team of 30 highly skilled engineers
- 200+ years of extensive experience in FPGA technology
- Supporting all areas of FPGA system design
 - High-speed FPGA hardware
 - HDL firmware
 - Embedded software
- Wide range of application areas
 - Wired and Wireless Comms
 - Smart camera
 - Drive/Motion control
 - Embedded interfaces




Communications/Networking

| IP Core | | Description | Status |
|--------------------------|---|--|-----------------------|
| FPGA Manager IP Solution |  | Enclustra's FPGA Manager solution allows for easy and efficient data transfer between a host and a FPGA over different interface standards like USB 2.0/3.0, Gigabit Ethernet and PCI Express. | Available now! |
| UDP/IP Ethernet IP Core |  | Enclustra's UDP/IP Ethernet IP core easily enables FPGA-based subsystems to communicate with other subsystems via Ethernet, using the UDP protocol. | Available now! |


Digital Signal Processing

| IP Core | | Description | Status |
|-----------------------|---|--|-----------------|
| Universal DSP Library |  | Enclustra's Universal DSP Library provides efficient FPGA implementations of the most common digital signal processing components. It allows building signal processing chains rapidly using Vivado's Block Design GUI, or by direct VHDL. | Available soon! |


Motion Control

| IP Core | | Description | Status |
|----------------------------|---|---|-----------------------|
| Universal Drive Controller |  | This IP core enables the independent position and/or velocity control of up to 8 DC, BLDC or stepper motors directly from the FPGA. | Available now! |

SoPC/Embedded Computing

| IP Core | | Description | Status |
|--------------------|---|---|-----------------------|
| Display Controller |  | This IP core enables the easy integration of displays into FPGA-based systems. With Linux, even multi-touch displays are supported. | Available now! |

Miscellaneous

| IP Core | | Description | Status |
|--------------------------|---|---|-----------------------|
| Stream Buffer Controller |  | A versatile stream to memory-mapped DMA bridge. | Available now! |



Everything FPGA

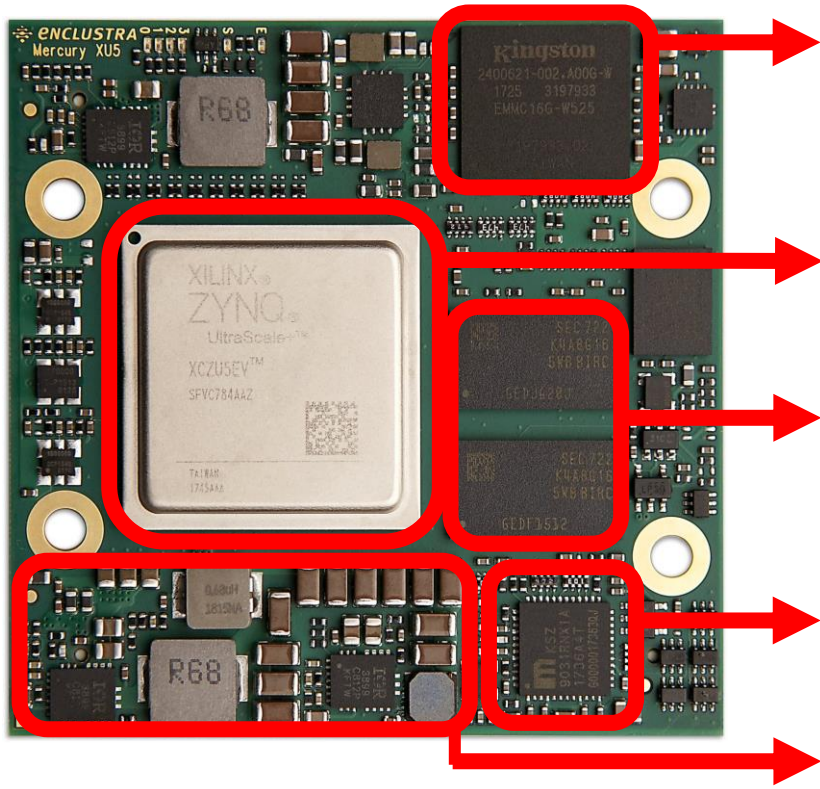
Product Overview

June 2021



Enclustra SOM

FRONT SIDE



Flash

FPGA/SoC
Pre-built Linux and
build environment

DDR

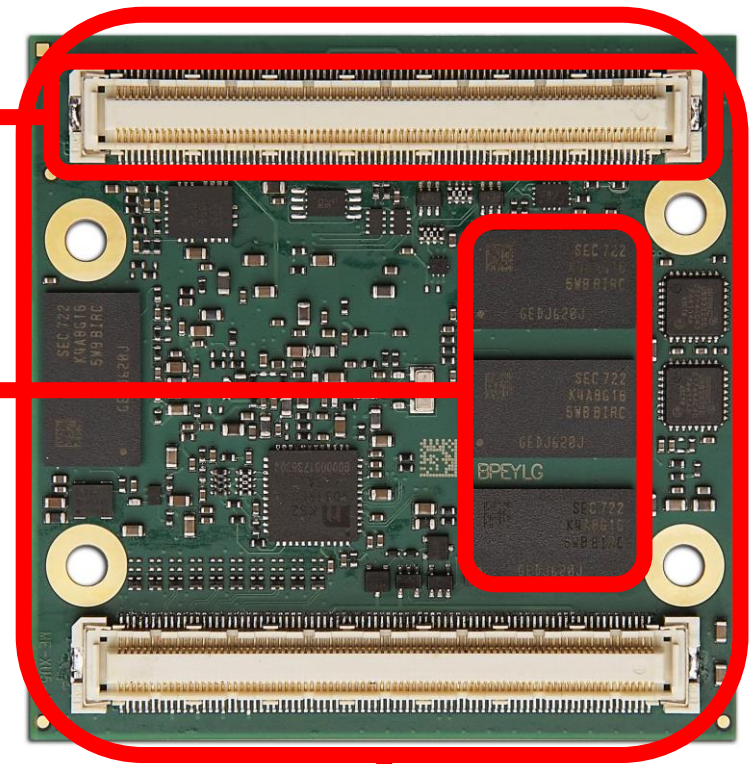
Standard interfaces
(e.g. Ethernet PHY,
USB PHY, etc.)

Power supply
Clock

REAR SIDE

High density
connectors

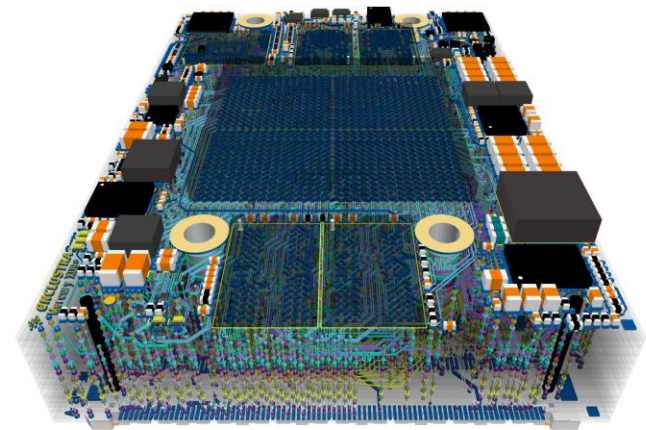
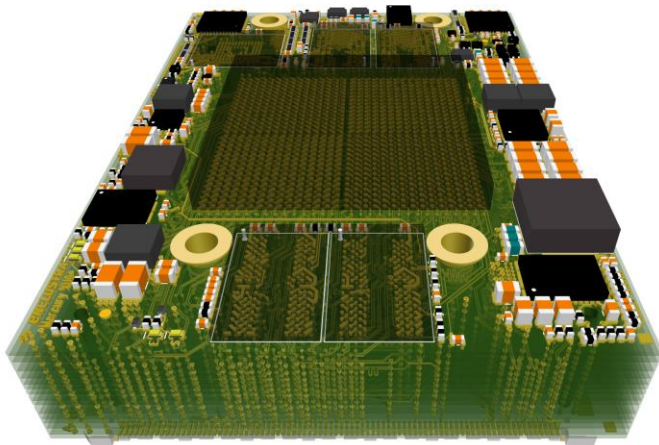
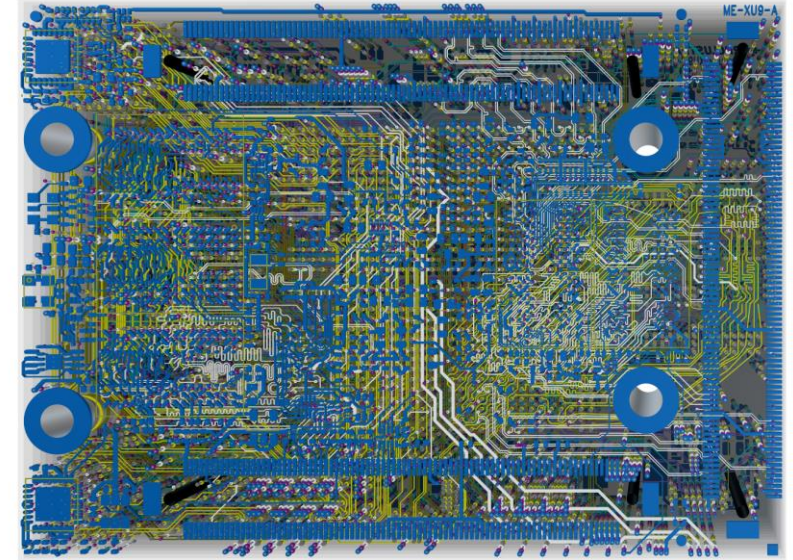
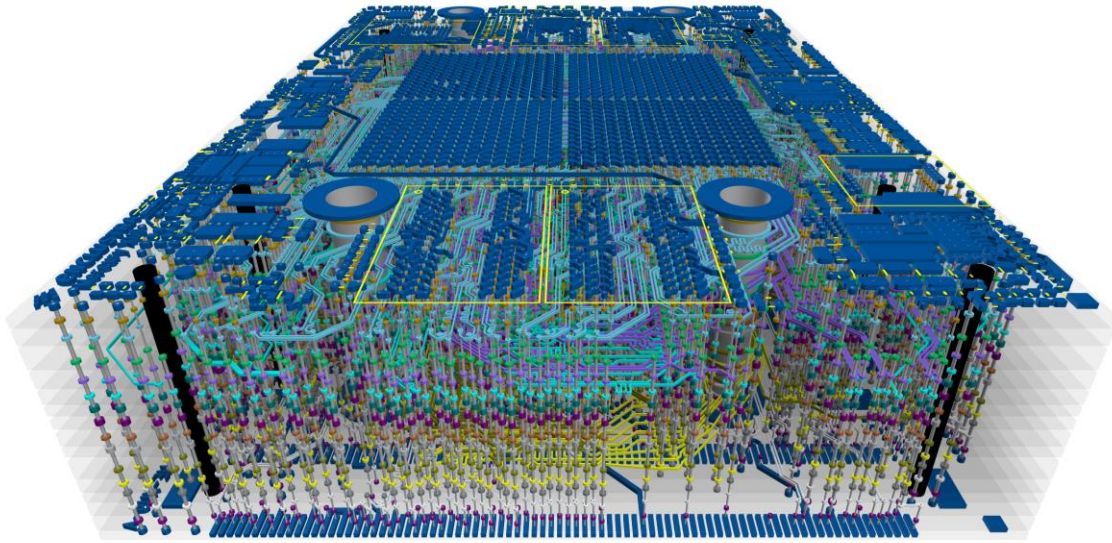
DDR



Compact
form factor

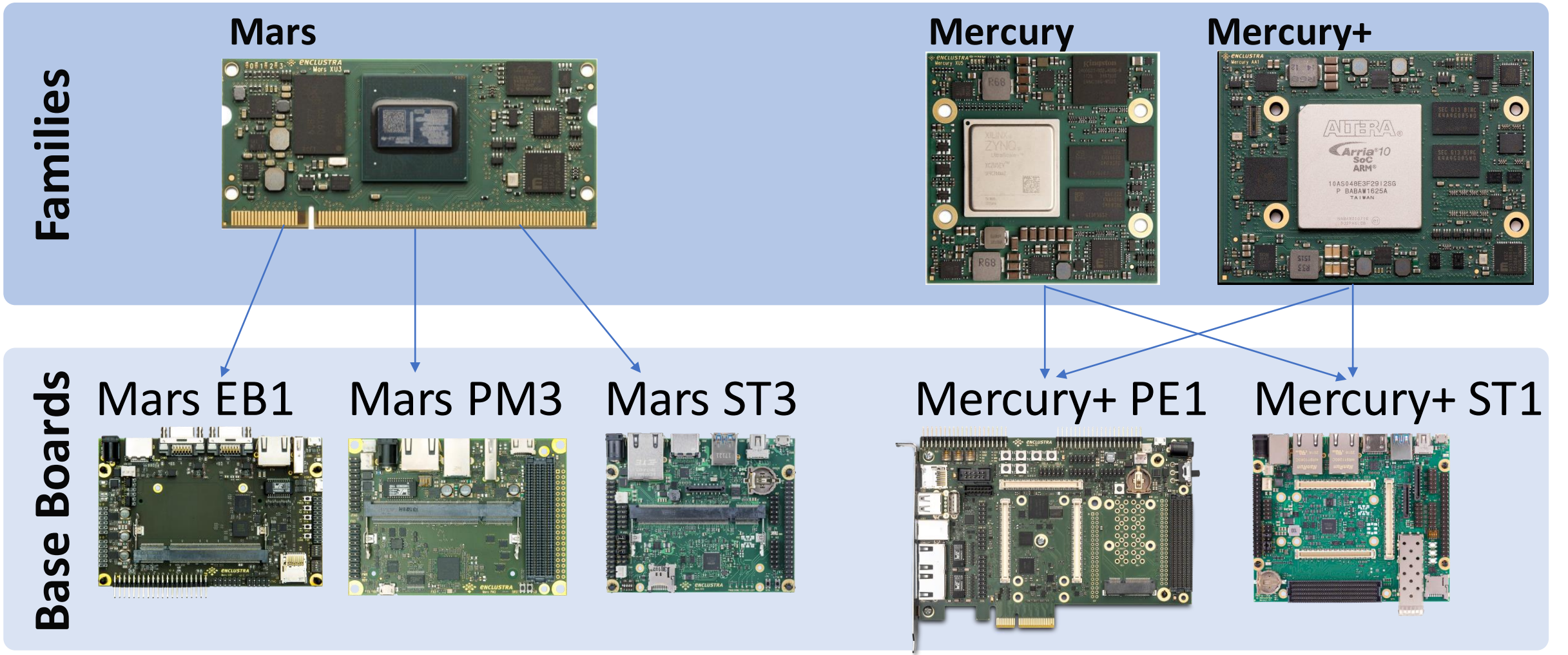


We Master Complexity in SOM



SOM cross-compatibility and base board support

Increase market share through supporting different FPGA sizes on a single base board



Enclustra SOM Families

Mars



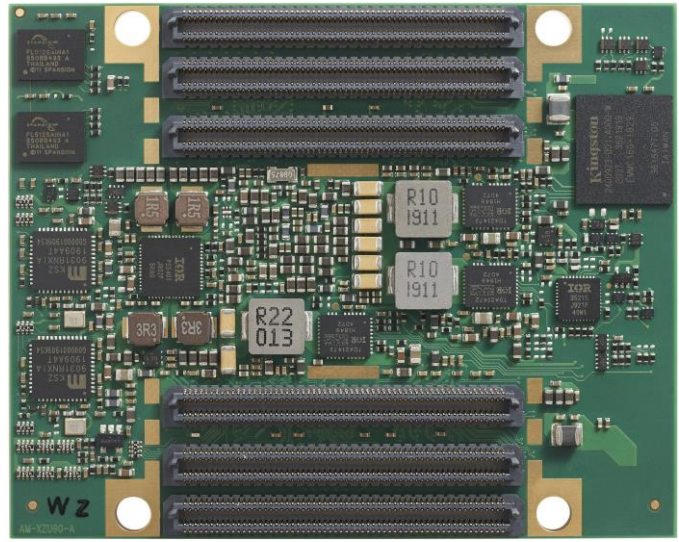
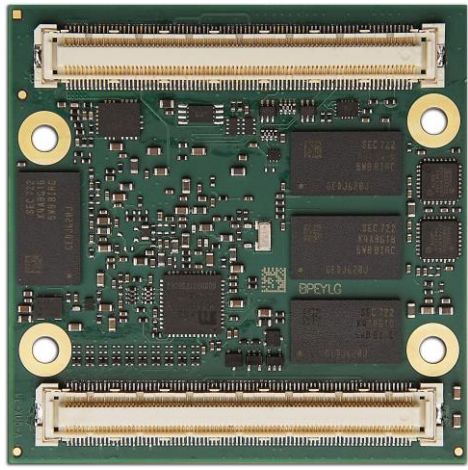
Mercury



Mercury+



Andromeda



Enclustra SOM Families

| | Dimensions [mm] | I/O Pins [max] | Transceivers [max] | Transceiver Speed [max] | SOMs [#] | Models [#] | Baseboard models [#] |
|------------------|-------------------------------|----------------|--------------------|-------------------------------|----------|------------|----------------------|
| Mars | 67.6 x 30 | 108 | 4 | 5 Gbps | 6 | 12 | 3 |
| Mercury | 56/64/72 x 54 | 178 | 8 | 15 Gbps | 6 | 21 | 5 |
| Mercury+ | 74 x 54 | 294 | 16 | 15 Gbps | 8 | 33 | 5 |
| Andromeda | 56 x 40 68 x 52 80 x 64 | 780 | 84 | 25 Gbps NRZ (56 Gbps PAM4) | 1 | 2 | Available soon |

Xilinx



XU3
Zynq Ultrascale+ ZU2-ZU3



ZX3
Zynq 7000 7020



ZX2
Zynq 7000 7010-7020



AX3
Artix-7 35-50-100



MX1
Spartan-6 45LX

SoC

FPGA

Intel SoC



MA3
Cyclone V SoC 5CSXFC6

Mercury/Mercury+ Xilinx SOM

Mercury



ZX5

Zynq 7000
7015-7030



ZX1

Zynq 7000
7030-7035-7045

Mercury+



XU5

Zynq Ultrascale+
ZU2-ZU3-ZU4-ZU5



XU6

Zynq Ultrascale+
ZU2-ZU3-ZU4-ZU5



XU8

Zynq Ultrascale+
ZU4-ZU5-ZU7



XU9

Zynq Ultrascale+
ZU4-ZU5-ZU7



XU1

Zynq Ultrascale+
ZU6-ZU9-ZU15



XU7

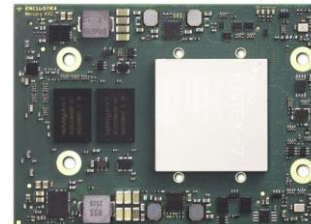
Zynq Ultrascale+
ZU6-ZU9-ZU15

SoC



KX1

Kintex 7
160-325



KX2

Kintex 7
160-410

FPGA

Enclustra Mercury SOM Comparison

| Module | <u>XU1</u> | <u>XU5</u> | <u>XU6</u> | <u>XU7</u> | <u>XU8</u> | <u>XU9</u> |
|--|-------------------------|-------------------------|----------------------|-------------------|-------------------|-------------------|
| ZU+ Devices | 6/9/15 | 2/3/4/5 | 2/3/4/5 | 6/9/15 | 4/5/7 | 4/5/7 |
| Sys. logic cells (k) | 469-747 | 103-256 | 103-256 | 469-747 | 192-504 | 192-504 |
| Main DDR4 PS/PL Bus Width (Bits) w/ECC(ECC) | 2-8 GB 64(ECC) | 1-8 GB 32/64(ECC) | 1-8 GB 32/64(ECC) | 2-8 GB 64(ECC) | 2-8 GB 64(ECC) | 2-8 GB 64(ECC) |
| Secondary DDR4 PL Bus Width (bits), w/out ECC | - | 0.5-2 GB 16 | - | 1-4 GB 32 | 1-4 GB 32 | 2-8 GB 64 |
| PS MGT (GTR) | 4 | 0/4 ²⁾ | 4 | 4 | 4 | 4 |
| PL MGT (GTH) | 12/16 ¹⁾ | 0/4 ²⁾ | 0/4 | 16 | 16 | 16 |
| H.264/265 Codec (VCU) | - | -/yes | -/yes | - | yes | yes |
| PCIe Gen3 Hard-IP | - | -/yes | -/yes | - | yes | yes |
| PL I/O (+PS I/O) | 200 ¹⁾ (+14) | 144 ²⁾ (+14) | 240 (+14) | 122 (+14) | 122 (+14) | 78 (+14) |
| Availability | now | now | 2Q2021 | now | now | now |

1) XU1...-G1 Version has 16 MGT but 20 User I/O less (=180)

2) XU5...-G1 Version has 4 PS + 4 PL MGT but 20 User I/O less (=124)

Mercury/Mercury+ Intel and Microchip SOM

Mercury



SA1

*Cyclone V SoC
5CSXFC6*

Mercury+



SA2

Cyclone V SoC 5CSTFD6



AA1

*Arria 10 SoC
10AS027- 10AS048*

Intel SoC



CA1

*Cyclone IV
EP4CE75 - EP4CE115*



MP1

*PolarFire SoC
250T - 460T*

Intel FPGA

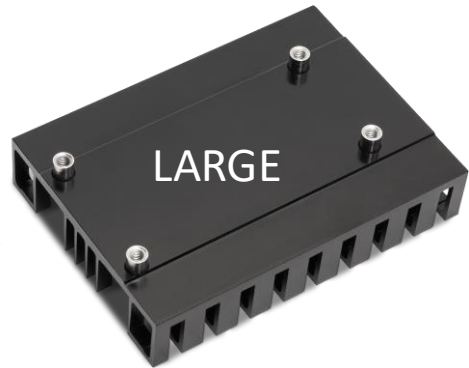
Microchip SoC (SOM in development)

Cooling solutions – Mercury/Mercury+ SOM



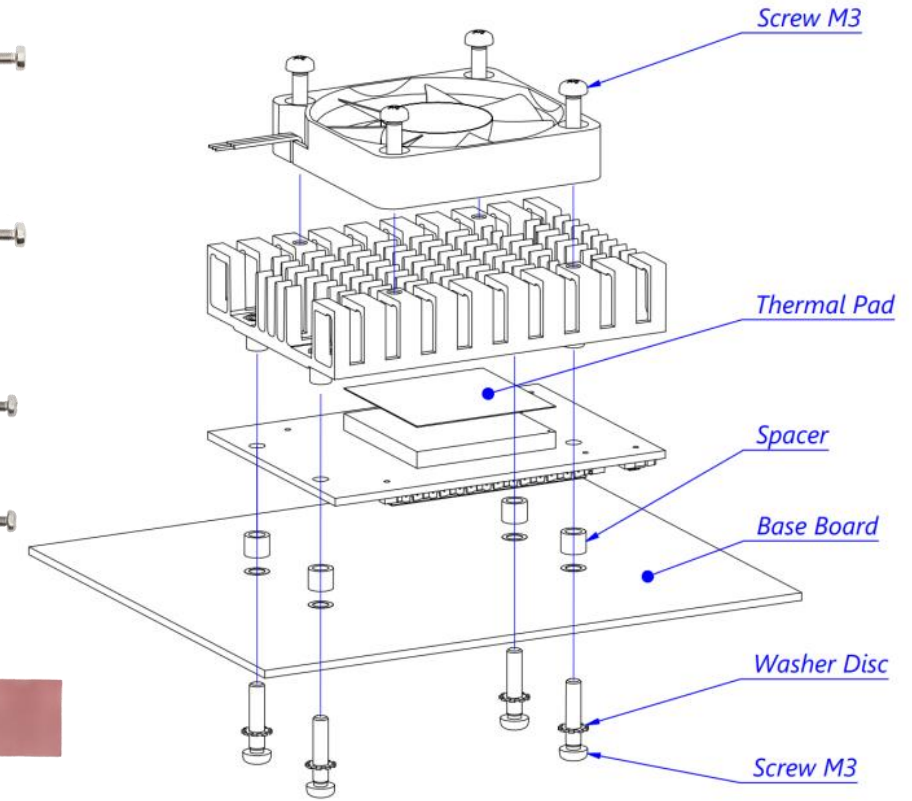
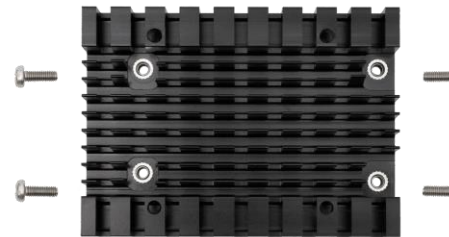
Supporting:

Mercury/ Mercury+
XU5 – XU6 – ZX1 – ZX5 (Xilinx)
SA1 – SA2 – CA1 (Intel)

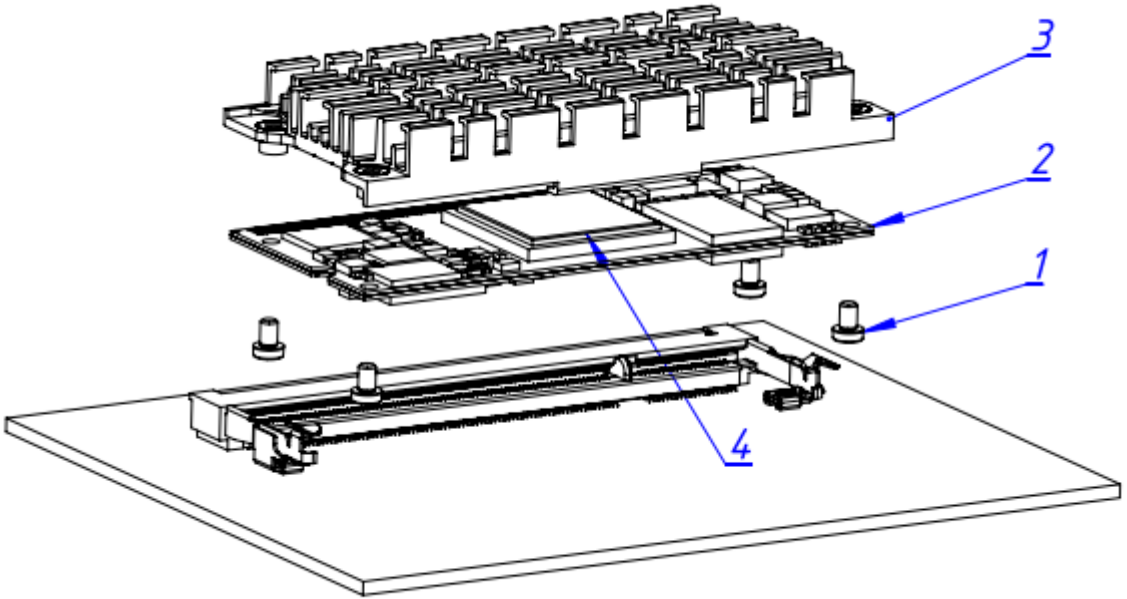
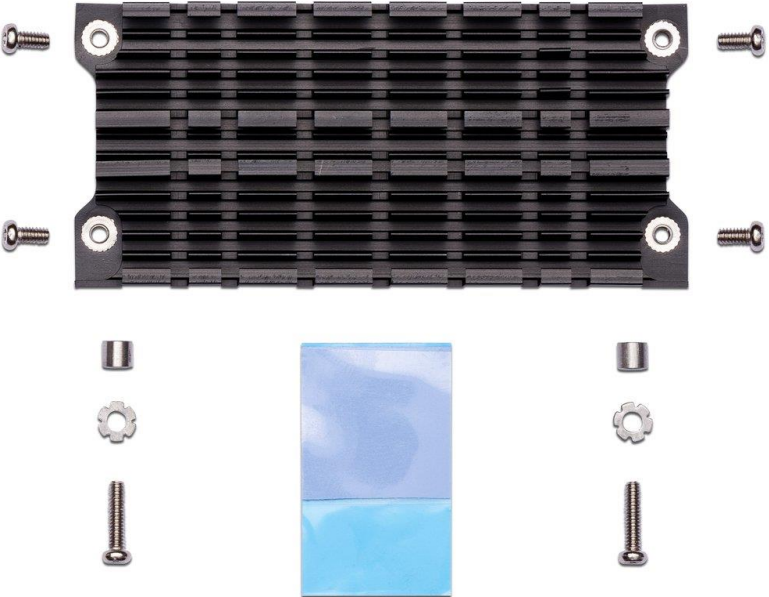


Supporting:

Mercury+
XU1 – XU7 – XU8 – XU9
– KX2 (Xilinx)
AA1 (Intel)
MP1 (Microchip)



Cooling solutions – Mars SOM



Supporting:

Mars
XU3 – ZX2 – ZX3 – AX3 (Xilinx)
MA3 (Intel)

We offer

- User Manuals
- Revision History
- User Schematics
- Pin Connection Guidelines
- 3D Models
- Module footprints for PADS, Altium, Orcad, Eagle
- Reference Designs
- Software (OS, Configuration, etc.)
- Application Notes
- Technical Support
- Design Services (paid service)





Mercury+ XU9 SoC Module

User Manual

Purpose

The purpose of this document is to present the characteristics of Mercury+ XU9 SoC module to the user and to provide the user with a comprehensive guide to understanding and using the Mercury+ XU9 SoC module.

Summary

This document first gives an overview of the Mercury+ XU9 SoC module followed by a detailed description of its features and configuration options. In addition, references to other useful documents are included.

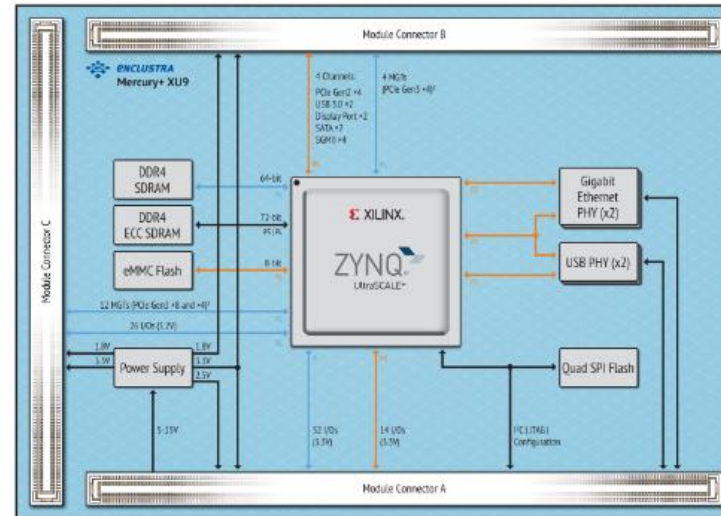
| Product Information | Code | Name |
|---------------------|--------|-------------------------|
| Product | ME-XU9 | Mercury+ XU9 SoC Module |

| Document Information | Reference | Version | Date |
|----------------------------|----------------|---------|------------|
| Reference / Version / Date | D-0000-463-001 | 03 | 16.02.2021 |

| Approval Information | Name | Position | Date |
|----------------------|------------|-----------------|------------|
| Written by | DIUN, MMOS | Design Engineer | 20.05.2019 |
| Verified by | MMOS, GKOE | Design Expert | 24.05.2019 |
| Approved by | DIUN | Manager, BU SP | 16.02.2021 |

2 Module Description

2.1 Block Diagram



1: PCIe Gen3 x4 is available at the system level by merging the MGTs from connectors B and C.

Figure 1: Hardware Block Diagram

The main component of the Mercury+ XU9 SoC module is the Xilinx Zynq Ultrascale+ MPSoC device. All available I/O pins (which are not routed to on-board peripherals) are connected to the Mercury+ module connector, making 92 regular user I/Os available to the user. Further, twenty MGT pairs are available on the module connector, making possible the implementation of several high-speed protocols such as PCIe Gen3 x16, PCIe Gen2 x4 and USB 3.0 (simultaneous usage of all the interfaces is limited to the available hardware resources i.e. number of transceivers and lane mapping).

The MPSoC device can boot from the on-board QSPI flash, from the eMMC flash or from an external SD card. For development purposes, a JTAG interface is connected to Mercury module connector.

The available standard configurations include a 16 GB eMMC flash, a 64 MB quad SPI flash, up to 4 GB DDR4 SDRAM with ECC connected to the Processing System (PS) and up to 2 GB DDR4 SDRAM connected to the Programmable Logic (PL).

Further, the module is equipped with two Gigabit Ethernet PHYs and two USB 2.0 PHYs, making it ideal for communication applications.

A real-time clock is available on the Xilinx Zynq Ultrascale+ MPSoC device.

5 Operating Conditions

5.1 Absolute Maximum Ratings

Table 47 indicates the absolute maximum ratings for Mercury+ XU9 SoC module. The values given are for reference only; for details please refer to the Zynq UltraScale+ MPSoC, DC and AC Switching Characteristics Datasheet [20].

| Symbol | Description | Rating | Unit |
|-------------|--|-------------------------------|------|
| VCC_MOD | Supply voltage relative to GND | -0.5 to 16 | V |
| VCC_BAT | Supply voltage for MPSoC battery-backed RAM and battery-backed RTC | 0 to 3.6 | V |
| VCC_IO_BN | Output drivers supply voltage relative to GND | -0.5 to 3.4 | V |
| VCC_IO_BO | Output drivers supply voltage relative to GND | -0.5 to 3.63 | V |
| V_CFG_MIO | Output drivers supply voltage relative to GND | -0.5 to 3.63 | V |
| V_IO | I/O input voltage relative to GND | -0.5 to V _{CC0} +0.5 | V |
| Temperature | Temperature range for extended temperature modules (E)* | 0 to +85 | °C |
| | Temperature range for industrial modules (I)* | -40 to +85 | °C |

Table 47: Absolute Maximum Ratings

5.2 Recommended Operating Conditions

Table 48 indicates the recommended operating conditions for Mercury+ XU9 SoC module. The values given are for reference only; for details please refer to the Zynq UltraScale+ MPSoC, DC and AC Switching Characteristics Datasheet [20].

| Symbol | Description | Rating | Unit |
|-------------|--|-------------------------------|------|
| VCC_MOD | Supply voltage relative to GND | 4.75 to 15.75 | V |
| VCC_BAT | Supply voltage for MPSoC battery-backed RAM and battery-backed RTC | 2.7 to 3.6 | V |
| VCC_IO_[x] | Output drivers supply voltage relative to GND | Refer to Section 2.9.5 | V |
| V_CFG_[x] | Output drivers supply voltage relative to GND | Refer to Section 2.9.5 | V |
| V_IO | I/O input voltage relative to GND | -0.2 to V _{CC0} +0.2 | V |
| Temperature | Temperature range for extended temperature modules (E)* | 0 to +85 | °C |
| | Temperature range for industrial modules (I)* | -40 to +85 | °C |

Table 48: Recommended Operating Conditions

Warning!

* The components used on the hardware are specified for the relevant temperature range. The user must provide adequate cooling in order to keep the temperature of the components within the specified range.

Product Deliverables – Revision History



Mercury+ XU8 SoC Module

Known Issues and Changes

Revision 2.1

Purpose

The purpose of this document is to describe the known issues and the revision changes related to the Mercury+ XU8 SoC module.

| Product Information | Code | Revision | Name |
|---------------------|--------|-------------------|-------------------------|
| Product | ME-XU8 | Mercury+ XU8 R2.1 | Mercury+ XU8 SoC Module |

| Document Information | Reference | Version | Date | Author |
|----------------------------|----------------|---------|------------|------------|
| Reference / Version / Date | D-0000-466-001 | 02 | 18.11.2019 | DIUN, MMOS |

Copyright Reminder

Copyright 2019 by Enclustra GmbH, Switzerland. All rights are reserved.

Unauthorized duplication of this document, in whole or in part, by any means is prohibited without the prior written permission of Enclustra GmbH, Switzerland.

Although Enclustra GmbH believes that the information included in this publication is correct as of the date of publication, Enclustra GmbH reserves the right to make changes at any time without notice.

All information in this document is strictly confidential and may only be published by Enclustra GmbH, Switzerland.

All referenced trademarks are the property of their respective owners.

Enclustra GmbH – Räflestrasse 28 – CH-8045 Zürich – Switzerland
Phone +41 43 343 39 43 – www.enclustra.com

1 Known Issues

In addition to the erratas from the components manufacturers, Enclustra is aware of the following issues with the Mercury+ XU8 R2.1 module.

| VCU power consumption limited to 4 A | |
|--------------------------------------|---|
| Description | The power consumption of the video codec unit (VCU) on the VCC_INT_VCU rail is limited to 4 A. This is due to current carrying limitations of a used component. |
| Workaround | There is no workaround available. |
| Status | This issue will be fixed in the next revision (R3). |

2 Functional Changes

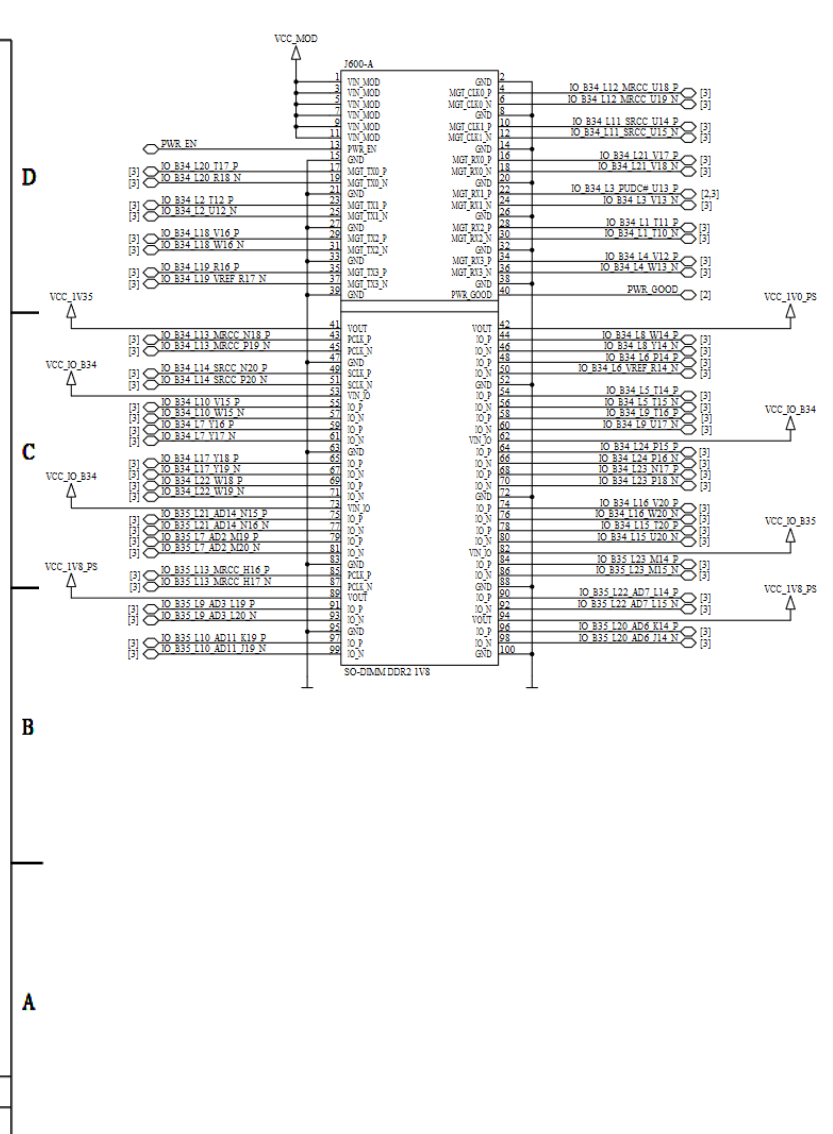
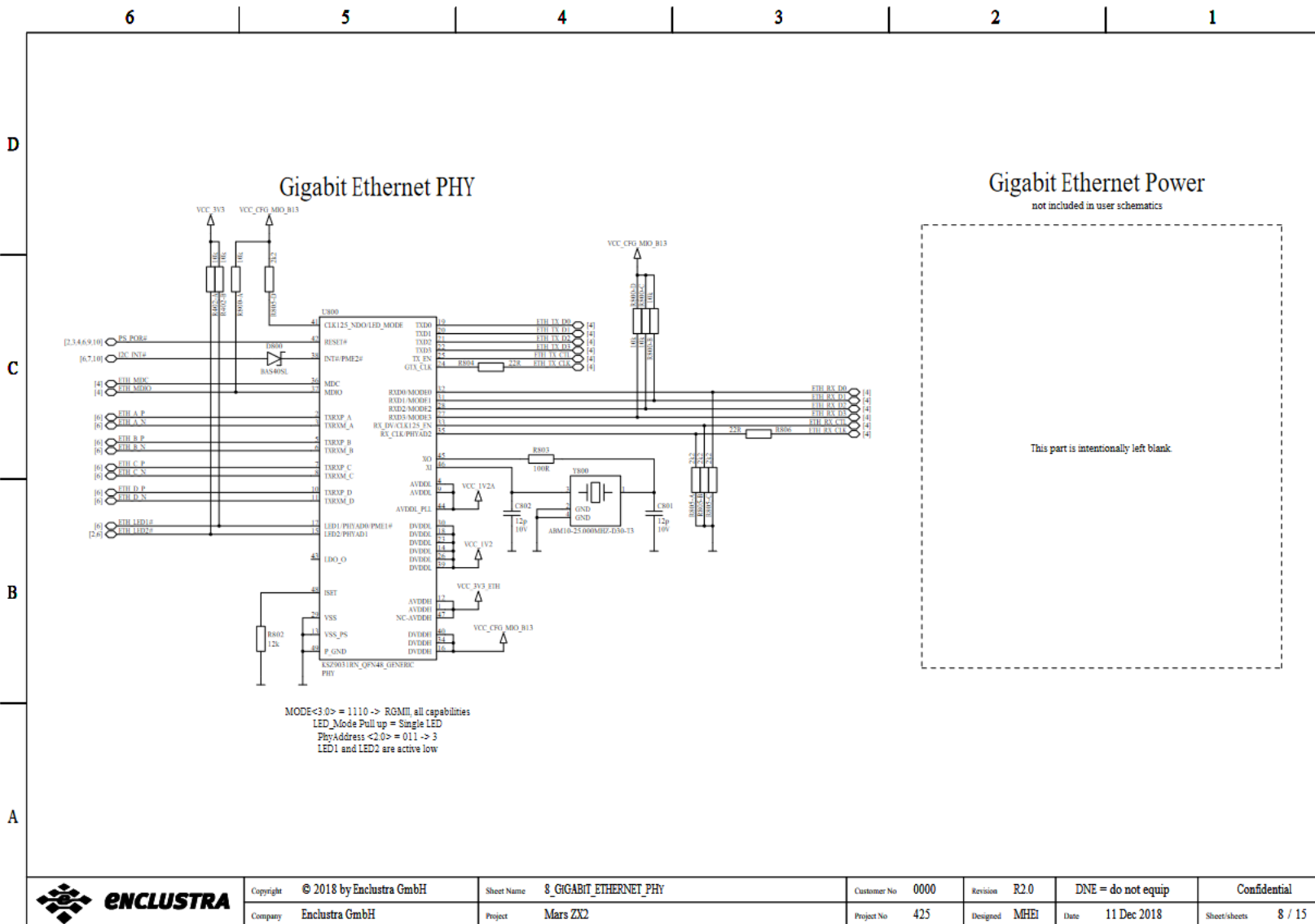
In addition to changes related to manufacturing process optimization and signal integrity improvements, the following functional changes were made from the Mercury+ XU8 SoC module revision R1.1 to R2.1.

| Added voltage monitoring output on module connector C | |
|---|---|
| Description | A new voltage monitoring output was added on the module connector pin C-8. This can be used to measure the on-board 1.2 V voltage, VCC_1V2. |

| 2.5 V supply is controllable by the PWR_EN signal | |
|---|---|
| Description | The 2.5 V supply (VCC_2V5) is now controllable by the PWR_EN signal. In this case, no additional power sequencing circuits are required on the base board. Please refer to the Mercury+ XU8 SoC module user manual for details. |

| Optional JTAG boot mode support | |
|---------------------------------|---|
| Description | Starting with revision 2, support for JTAG boot mode has been added to increase the usability of the module with Xilinx tools, for example for flash programming or bitstream loading. Please refer to the Mercury+ XU8 SoC module user manual for details on JTAG boot mode. |

Product Deliverables – User Schematics



Copyright © 2018 by Enclustra GmbH
Company Enclustra GmbH

Sheet Name 8_GIGABIT_ETHERNET_PHY
Project Mars ZX2

Customer No 0000
Project No 425

Revision R2.0
Designed MHEI

DNE = do not equip
Date 11 Dec 2018

Confidential
Sheet/sheets 8 / 15

Product Deliverables – Module Pin Connection Guidelines

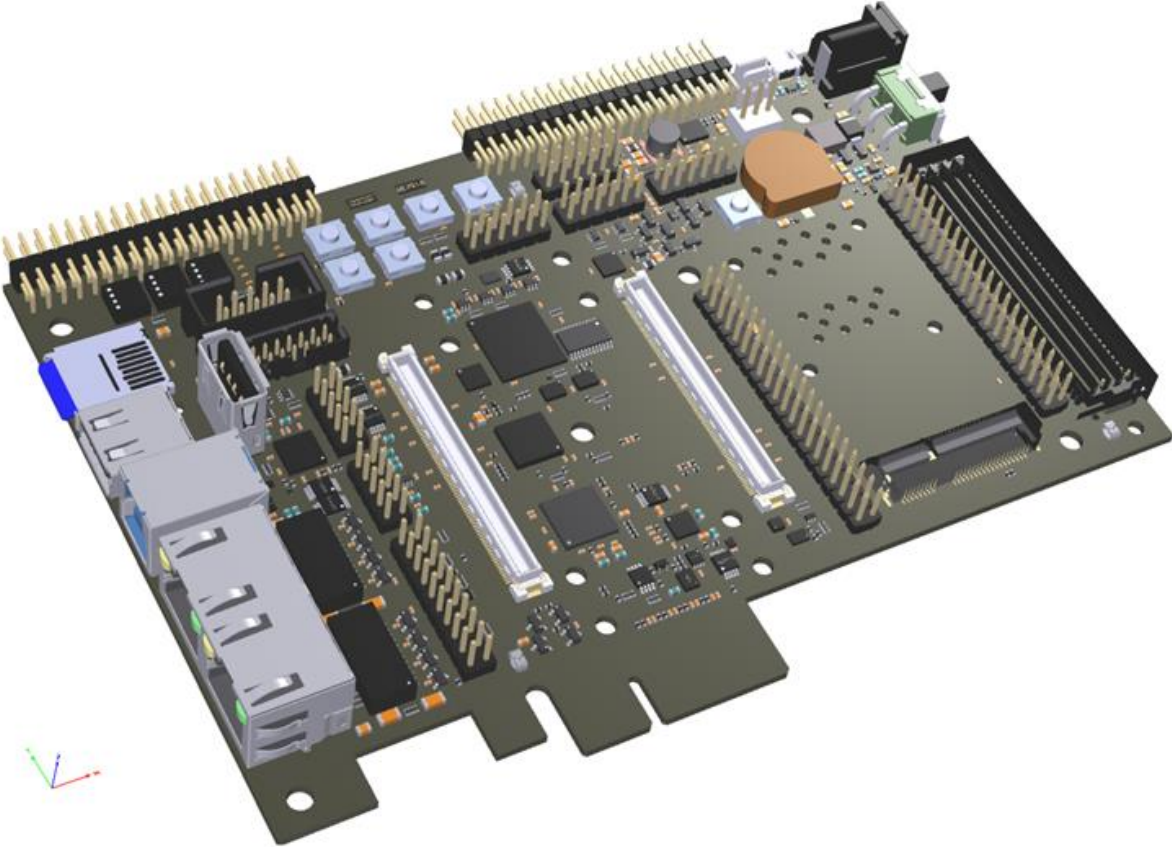
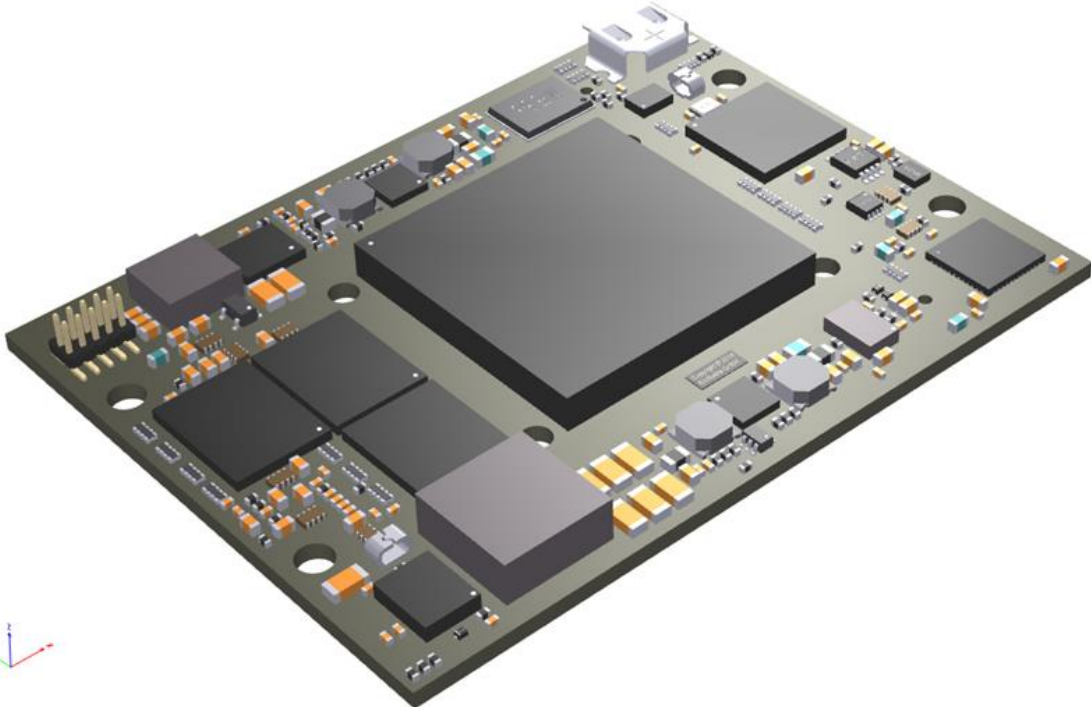
Mercury Mars Module Pin Connection Guidelines

Enclustra Confidential
Copyright © 2021 by Enclustra GmbH

Date Feb 12, 2021

| Pin Type | Direction (*1) | Voltage Level | Description | Comments |
|---------------------|----------------|---------------|-----------------------|---|
| General | | | | |
| All pins | | | | Applying a negative voltage or a voltage higher than the specified maximum, even for a short period of time, may damage the module permanently. The module does not have any protection against these hazards. The ESD protection level available on each module pin is module- and pin-specific. Enclustra recommends adding ESD protection circuitry to all signals connected to an externally-accessible connector. |
| User IO pins | | | | |
| IO_P IO_N | Bidirectional | Bank-specific | FPGA I/O pin | Please check the supported IO standard's compatibility with the FPGA/SoC tools. Some banks may only have a restricted voltage range, e.g. 1.8 V. Some FPGA/SoC families have restrictions when LVDS and single-ended signals are placed on the same bank. The FPGA/SoC vendor provides further information. The best way to check your pinout is to perform a full compilation of the FPGA/SoC design using the vendor tools. Some user I/O pins may have additional functionalities, such as analog input or PCIe reset, or PUDC, or reference voltage input. Some modules may have unconnected pins. Some FPGA/SoC architectures may have higher capacitance on some pins (e.g. VREF) and therefore offer lower performance on these pins. Some modules may not support differential driver or receiver functionality on all of the IO_P/N pin pairs. They may only be usable for single ended I/O standards. Some FPGA/SoC families may not support terminating differential signals on all pin pairs inside the FPGA/SoC. External termination on the base board could be required. This may reduce the performance of such pairs due to the trace stubs slightly. Certain modules contain differential termination resistors on the module on some of the IO_P/N pin pairs. These terminations cannot be switched off. The IO_P pin could be used as single-ended I/O with limited performance. The IO_N pin should be tri-stated in the FPGA/SoC design. Most modules have pull-ups on all IO_P/N pins before and during configuration. These pull-ups cannot be switched off. The voltage on these pins may never exceed the voltage on the corresponding bank supply voltage. Check the FPGA/SoC data sheet for details. |
| | | | PUDC pin | Some modules have the pull-up-during-configuration pin exposed to the user connector. Some modules apply a 1 kOhm pull-down resistor during configuration (FPGA_DONE is low). |
| | | | Analog I/O pin | Some modules may have analog I/O functionality on some of the IO_P/N pins. |
| | | | System I/O pin | Some modules may have system peripheral pins on some of the IO_P/N pins. These pins usually cannot be driven from inside the programmable logic. |
| PCLK_P PCLK_N | Bidirectional | Bank-specific | Primary clock input | When designing a new module, we try to put global clock input pins on PCLK_P (if single-ended) or PCLK_P/N (if differential) pins. The use of PCLK_N as a single-ended clock may not be supported by some FPGA/SoC families. Some FPGA/SoC families may not support terminating differential signals on all pin pairs inside the FPGA/SoC. External termination on the base board could be required. This may reduce the performance of such pairs, due to the trace stubs slightly. Some FPGA/SoC families may not support using clock input pins as outputs. |
| | | | Other pin | On some modules, these pins may not be clock capable. In this case, refer to the description of the IO_P/N pin type. |
| SCLK_P SCLK_N | Bidirectional | Bank-specific | Secondary clock input | When designing a new module, we try to put any remaining clock input pins on SCLK_P (if single-ended) or SCLK_P/N (if differential) pins. Some of these clock pins may only be usable as regional clocks, restricting the location of synchronous elements within the FPGA/SoC. The use of SCLK_N as a single-ended clock may not be supported by some FPGA/SoC families. Some FPGA/SoC families may not support terminating differential signals on all pin pairs inside the FPGA/SoC. External termination on the base board could be required. This may reduce the performance of such pairs due to the trace stubs slightly. |

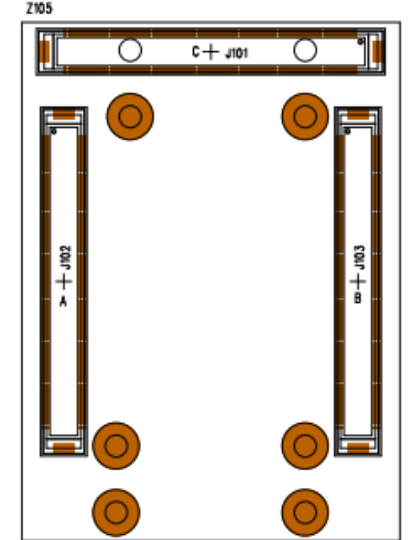
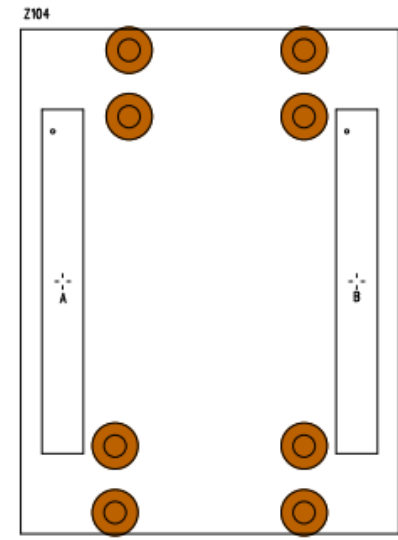
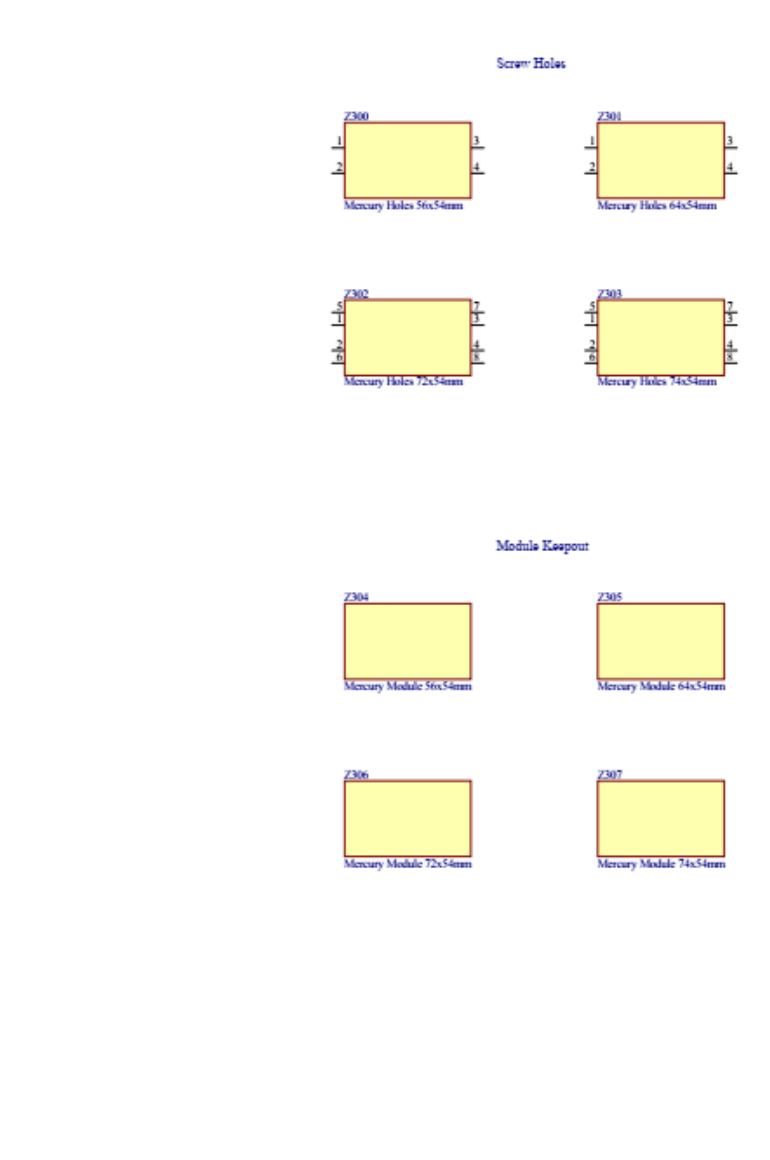
Product Deliverables – 3D Models



Product Deliverables – Module Symbols and Footprints

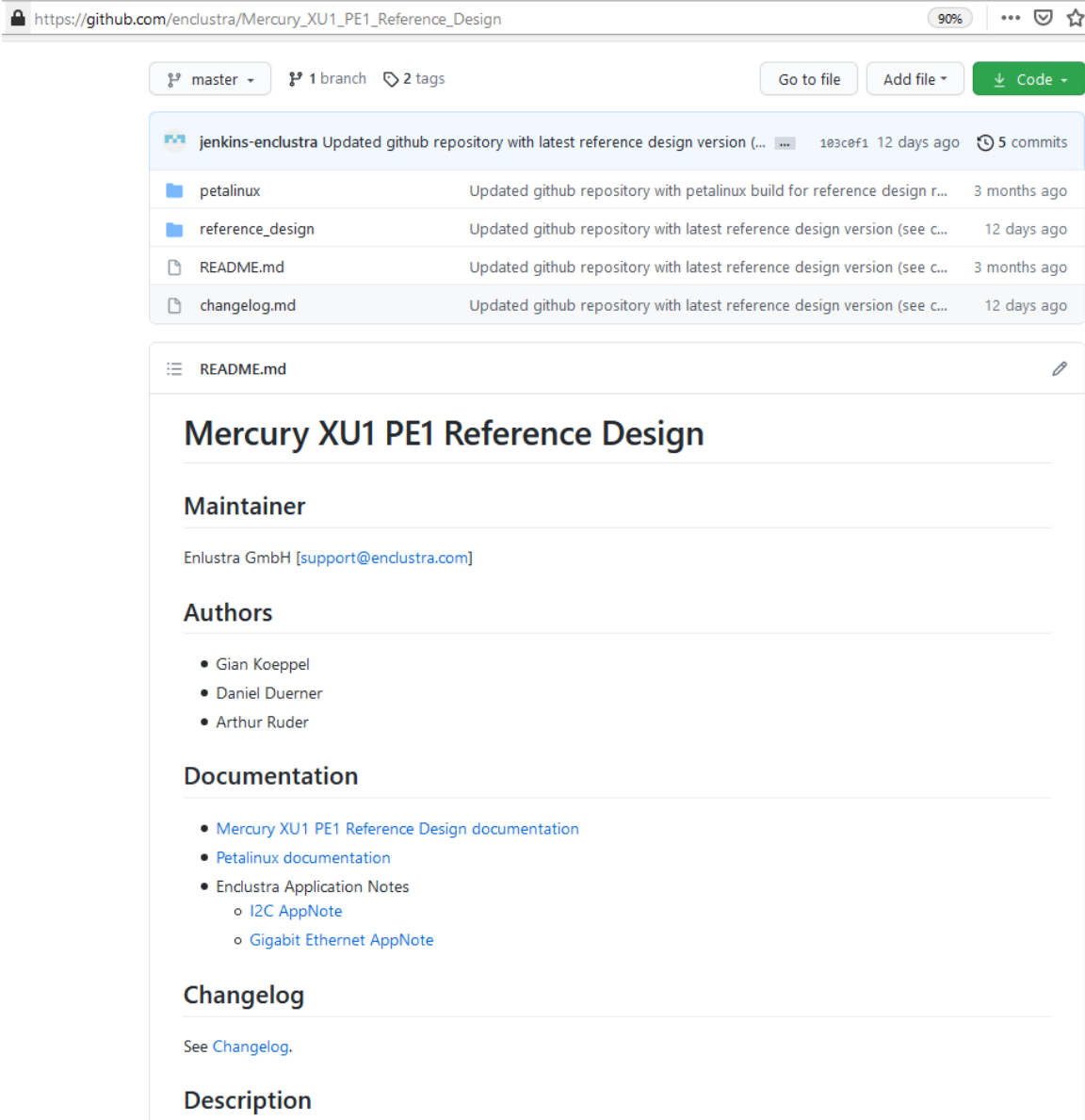
Connectors

| J300 | J301 | J302 |
|----------------|--------------|--------------|
| 1 VIN_MOD | 1 GND | 1 GND |
| 2 VIN_MOD | 2 SCLK_P | 2 REFCLK_P |
| 3 VIN_MOD | 3 SCLK_N | 3 MGT_CLK_N |
| 4 VIN_MOD | 4 SCLK_P | 4 MGT_CLK_N |
| 5 VIN_MOD | 5 SCLK_N | 5 VMON |
| 6 VIN_MOD | 6 PWR_EN | 6 IO_N |
| 7 VIN_MOD | 7 PWR_GOOD | 7 MGT_CLK_P |
| 8 GND | 8 GND | 8 MGT_CLK_N |
| 9 IO_P | 9 SCLK_P | 9 MGT_TX_P |
| 10 IO_P | 10 SCLK_N | 10 MGT_TX_N |
| 11 IO_N | 11 GND | 11 GND |
| 12 IO_N | 12 MGT_TX_P | 12 MGT_RX_P |
| 13 IO_P | 13 MGT_TX_N | 13 MGT_RX_N |
| 14 GND | 14 GND | 14 GND |
| 15 IO_N | 15 MGT_TX_P | 15 MGT_RX_P |
| 16 IO_P | 16 MGT_TX_N | 16 MGT_RX_N |
| 17 IO_N | 17 GND | 17 GND |
| 18 IO_P | 18 MGT_TX_P | 18 MGT_RX_P |
| 19 IO_N | 19 MGT_TX_N | 19 MGT_RX_N |
| 20 GND | 20 GND | 20 GND |
| 21 IO_P | 21 MGT_TX_P | 21 MGT_RX_P |
| 22 IO_N | 22 MGT_TX_N | 22 MGT_RX_N |
| 23 IO_P | 23 MGT_TX_P | 23 MGT_RX_P |
| 24 IO_N | 24 MGT_TX_N | 24 MGT_RX_N |
| 25 GND | 25 GND | 25 GND |
| 26 IO_P | 26 MGT_TX_P | 26 MGT_RX_P |
| 27 IO_N | 27 MGT_TX_N | 27 MGT_RX_N |
| 28 IO_P | 28 MGT_TX_P | 28 MGT_RX_P |
| 29 IO_N | 29 MGT_TX_N | 29 MGT_RX_N |
| 30 GND | 30 GND | 30 GND |
| 31 IO_P | 31 MGT_TX_P | 31 MGT_RX_P |
| 32 IO_N | 32 MGT_TX_N | 32 MGT_RX_N |
| 33 IO_P | 33 MGT_TX_P | 33 MGT_RX_P |
| 34 IO_N | 34 MGT_TX_N | 34 MGT_RX_N |
| 35 GND | 35 GND | 35 GND |
| 36 IO_P | 36 MGT_TX_P | 36 MGT_RX_P |
| 37 IO_N | 37 MGT_TX_N | 37 MGT_RX_N |
| 38 IO_P | 38 MGT_TX_P | 38 MGT_RX_P |
| 39 IO_N | 39 MGT_TX_N | 39 MGT_RX_N |
| 40 GND | 40 GND | 40 GND |
| 41 VIS_IO | 41 MGT_TX_N | 41 MGT_TX_N |
| 42 IO_N | 42 GND | 42 MGT_RX_N |
| 43 IO_P | 43 GND | 43 MGT_RX_N |
| 44 GND | 44 PCLK_P | 44 IO_P |
| 45 GND | 45 PCLK_N | 45 IO_N |
| 46 SCLK_P | 46 VOUT_3V3 | 46 IO_P |
| 47 SCLK_N | 47 IO_P | 47 IO_N |
| 48 GND | 48 IO_N | 48 IO_P |
| 49 GND | 49 IO_P | 49 IO_N |
| 50 VOUT | 50 IO_N | 50 IO_P |
| 51 IO_N | 51 GND | 51 GND |
| 52 IO_P | 52 MGT_TX_P | 52 MGT_TX_P |
| 53 IO_N | 53 MGT_TX_N | 53 MGT_TX_N |
| 54 GND | 54 GND | 54 GND |
| 55 IO_P | 55 MGT_TX_P | 55 MGT_TX_P |
| 56 IO_N | 56 MGT_TX_N | 56 MGT_TX_N |
| 57 GND | 57 GND | 57 GND |
| 58 IO_P | 58 MGT_TX_P | 58 MGT_TX_P |
| 59 IO_N | 59 MGT_TX_N | 59 MGT_TX_N |
| 60 GND | 60 GND | 60 GND |
| 61 IO_P | 61 MGT_TX_P | 61 MGT_TX_P |
| 62 IO_N | 62 MGT_TX_N | 62 MGT_TX_N |
| 63 GND | 63 GND | 63 GND |
| 64 IO_P | 64 MGT_TX_P | 64 MGT_TX_P |
| 65 IO_N | 65 MGT_TX_N | 65 MGT_TX_N |
| 66 GND | 66 GND | 66 GND |
| 67 IO_P | 67 MGT_TX_P | 67 MGT_TX_P |
| 68 IO_N | 68 MGT_TX_N | 68 MGT_TX_N |
| 69 GND | 69 GND | 69 GND |
| 70 IO_P | 70 MGT_TX_P | 70 MGT_TX_P |
| 71 IO_N | 71 MGT_TX_N | 71 MGT_TX_N |
| 72 GND | 72 GND | 72 GND |
| 73 IO_P | 73 MGT_TX_P | 73 MGT_TX_P |
| 74 IO_N | 74 MGT_TX_N | 74 MGT_TX_N |
| 75 GND | 75 GND | 75 GND |
| 76 IO_P | 76 MGT_TX_P | 76 MGT_TX_P |
| 77 IO_N | 77 MGT_TX_N | 77 MGT_TX_N |
| 78 GND | 78 GND | 78 GND |
| 79 VIS_CFG | 79 MGT_TX_N | 79 MGT_TX_N |
| 80 IO_P | 80 GND | 80 MGT_RX_N |
| 81 IO_N | 81 GND | 81 MGT_RX_N |
| 82 GND | 82 SCLK_P | 82 PCLK_P |
| 83 IO_P | 83 SCLK_N | 83 PCLK_N |
| 84 GND | 84 SCLK_N | 84 PCLK_N |
| 85 PCLK_P | 85 VOUT_3V3 | 85 IO_P |
| 86 PCLK_N | 86 IO_P | 86 IO_N |
| 87 VOUT | 87 IO_N | 87 IO_P |
| 88 IO_N | 88 GND | 88 GND |
| 89 IO_P | 89 MGT_TX_P | 89 MGT_TX_P |
| 90 IO_N | 90 MGT_TX_N | 90 MGT_TX_N |
| 91 GND | 91 GND | 91 GND |
| 92 IO_P | 92 MGT_TX_P | 92 MGT_TX_P |
| 93 IO_N | 93 MGT_TX_N | 93 MGT_TX_N |
| 94 GND | 94 GND | 94 GND |
| 95 IO_P | 95 MGT_TX_P | 95 MGT_TX_P |
| 96 IO_N | 96 MGT_TX_N | 96 MGT_TX_N |
| 97 GND | 97 GND | 97 GND |
| 98 IO_P | 98 MGT_TX_P | 98 MGT_TX_P |
| 99 IO_N | 99 MGT_TX_N | 99 MGT_TX_N |
| 100 GND | 100 GND | 100 GND |
| 101 IO_P | 101 VMON | 101 MGT_TX_P |
| 102 IO_N | 102 SCLK_P | 102 MGT_TX_N |
| 103 IO_P | 103 SCLK_N | 103 MGT_TX_P |
| 104 GND | 104 SCLK_N | 104 MGT_TX_N |
| 105 IO_P | 105 SCLK_P | 105 MGT_TX_P |
| 106 IO_N | 106 SCLK_N | 106 MGT_TX_N |
| 107 GND | 107 GND | 107 GND |
| 108 IO_N | 108 VOUT | 108 MGT_TX_P |
| 109 IO_P | 109 IO_N | 109 MGT_TX_N |
| 110 GND | 110 IO_P | 110 MGT_TX_P |
| 111 IO_N | 111 IO_N | 111 MGT_TX_N |
| 112 IO_P | 112 IO_P | 112 MGT_TX_P |
| 113 IO_N | 113 IO_N | 113 MGT_TX_N |
| 114 GND | 114 GND | 114 GND |
| 115 IO_P | 115 SCLK_P | 115 VIN_TO |
| 116 IO_N | 116 SCLK_N | 116 SCLK_N |
| 117 GND | 117 GND | 117 GND |
| 118 IO_P | 118 MGT_TX_P | 118 MGT_TX_P |
| 119 IO_N | 119 MGT_TX_N | 119 MGT_TX_N |
| 120 GND | 120 GND | 120 GND |
| 121 IO_P | 121 MGT_TX_P | 121 MGT_TX_P |
| 122 IO_N | 122 MGT_TX_N | 122 MGT_TX_N |
| 123 GND | 123 GND | 123 GND |
| 124 IO_P | 124 MGT_TX_P | 124 MGT_TX_P |
| 125 IO_N | 125 MGT_TX_N | 125 MGT_TX_N |
| 126 GND | 126 GND | 126 GND |
| 127 IO_P | 127 MGT_TX_P | 127 MGT_TX_P |
| 128 IO_N | 128 MGT_TX_N | 128 MGT_TX_N |
| 129 GND | 129 GND | 129 GND |
| 130 IO_P | 130 MGT_TX_P | 130 MGT_TX_P |
| 131 IO_N | 131 MGT_TX_N | 131 MGT_TX_N |
| 132 GND | 132 GND | 132 GND |
| 133 VIS_HDP | 133 MGT_TX_N | 133 MGT_TX_N |
| 134 ID_HDM | 134 GND | 134 GND |
| 135 GND | 135 IO_P | 135 MGT_TX_P |
| 136 USB_SSTX_P | 136 IO_N | 136 MGT_TX_N |
| 137 USB_SSTX_N | 137 GND | 137 GND |
| 138 GND | 138 MGT_TX_P | 138 MGT_TX_P |
| 139 USB_SSTX_P | 139 MGT_TX_N | 139 MGT_TX_N |
| 140 GND | 140 GND | 140 GND |
| 141 USB_SSTX_N | 141 MGT_TX_P | 141 MGT_TX_P |
| 142 GND | 142 MGT_TX_N | 142 MGT_TX_N |
| 143 GND | 143 GND | 143 GND |
| 144 ETH_D_N | 144 MGT_TX_N | 144 MGT_TX_N |
| 145 ETH_D_P | 145 GND | 145 GND |
| 146 ETH_LED_P | 146 IO_P | 146 MGT_TX_P |
| 147 ETH_LED_N | 147 IO_N | 147 MGT_TX_N |
| 148 ETH_LED_P | 148 GND | 148 GND |
| 149 ETH_LED_N | 149 GND | 149 GND |
| 150 ETH_C_N | 150 PCLK_P | 150 SCLK_P |
| 151 ETH_C_P | 151 PCLK_N | 151 SCLK_N |
| 152 ETH_C_P | 152 VOUT_3V3 | 152 SCLK_P |
| 153 ETH_C_P | 153 PCLK_N | 153 SCLK_N |
| 154 ETH_C_P | 154 MGT_TX_P | 154 MGT_TX_P |
| 155 ETH_C_P | 155 MGT_TX_N | 155 MGT_TX_N |
| 156 ETH_C_P | 156 MGT_TX_P | 156 MGT_TX_P |
| 157 ETH_C_P | 157 MGT_TX_N | 157 MGT_TX_N |
| 158 ETH_C_P | 158 MGT_TX_P | 158 MGT_TX_P |
| 159 ETH_C_P | 159 MGT_TX_N | 159 MGT_TX_N |
| 160 ETH_C_P | 160 MGT_TX_P | 160 MGT_TX_P |
| 161 ETH_C_P | 161 MGT_TX_N | 161 MGT_TX_N |
| 162 ETH_C_P | 162 MGT_TX_P | 162 MGT_TX_P |
| 163 ETH_C_P | 163 MGT_TX_N | 163 MGT_TX_N |
| 164 ETH_C_P | 164 MGT_TX_P | 164 MGT_TX_P |
| 165 ETH_C_P | 165 MGT_TX_N | 165 MGT_TX_N |
| 166 ETH_C_P | 166 MGT_TX_P | 166 MGT_TX_P |
| 167 ETH_C_P | 167 MGT_TX_N | 167 MGT_TX_N |
| 168 GND | 168 VMON | 168 GND |



- Supporting:
- PADS
 - ALTIUM
 - ORCAD
 - EAGLE

Product Deliverables – Reference Designs



The screenshot shows a GitHub repository page for 'Mercury_XU1_PE1_Reference_Design'. The repository is on the 'master' branch and has 2 tags. Recent updates include a commit by 'jenkins-enclustra' 12 days ago and updates to 'petalinux', 'reference_design', 'README.md', and 'changelog.md'. The 'README.md' file is open, displaying the title 'Mercury XU1 PE1 Reference Design', the maintainer 'Enclustra GmbH', authors 'Gian Koeppel', 'Daniel Duerner', and 'Arthur Ruder', and a list of documentation links including 'Mercury XU1 PE1 Reference Design documentation', 'Petalinux documentation', and 'Endustra Application Notes'.



Mercury+ XU1 SoC Module

Reference Design for Mercury+ PE1 Base Board User Manual

Purpose

The purpose of this document is to present to the user the overall view of the Mercury+ XU1 SoC module reference design and to provide the user with a step-by-step guide to the complete Xilinx® MPSoC design flow used for the Mercury+ XU1 SoC module.

Summary

This document first gives an overview of the Mercury+ XU1 SoC module reference design and then guides through the complete Xilinx MPSoC design flow for the Mercury+ XU1 SoC module in the getting started section. In addition, the internals and the boot options of the Mercury+ XU1 SoC module reference design are described.

| Product Information | Code | Name |
|---------------------|--------|-------------------------|
| Product | ME-XU1 | Mercury+ XU1 SoC Module |

| Document Information | Reference | Version | Date |
|----------------------------|----------------|---------------|------------|
| Reference / Version / Date | D-0000-460-002 | 2020.2_v1.1.1 | 04.02.2021 |

| Approval Information | Name | Position | Date |
|----------------------|-----------|-----------------|------------|
| Written by | DDUE/ARUD | Design Engineer | 25.01.2021 |
| Verified by | GKOE | Design Expert | 26.01.2021 |
| Approved by | DIUN | Manager, BU SP | 04.02.2021 |

Enclustra GmbH – Räfelfstrasse 28 – CH-8045 Zürich – Switzerland
Phone +41 43 343 39 43 – www.enclustra.com

- Xilinx SOM reference designs and PetaLinux BSPs available on GitHub

Product Deliverables – Reference Designs



Mercury+ AA1 SoC Module

Reference Design for Mercury+ PE1 Base Board User Manual

Purpose

The purpose of this document is to present to the user the overall view of the Mercury+ AA1 SoC module reference design and to provide the user with a step-by-step guide to the complete Intel® SoC design flow used for the Mercury+ AA1 SoC module.

Summary

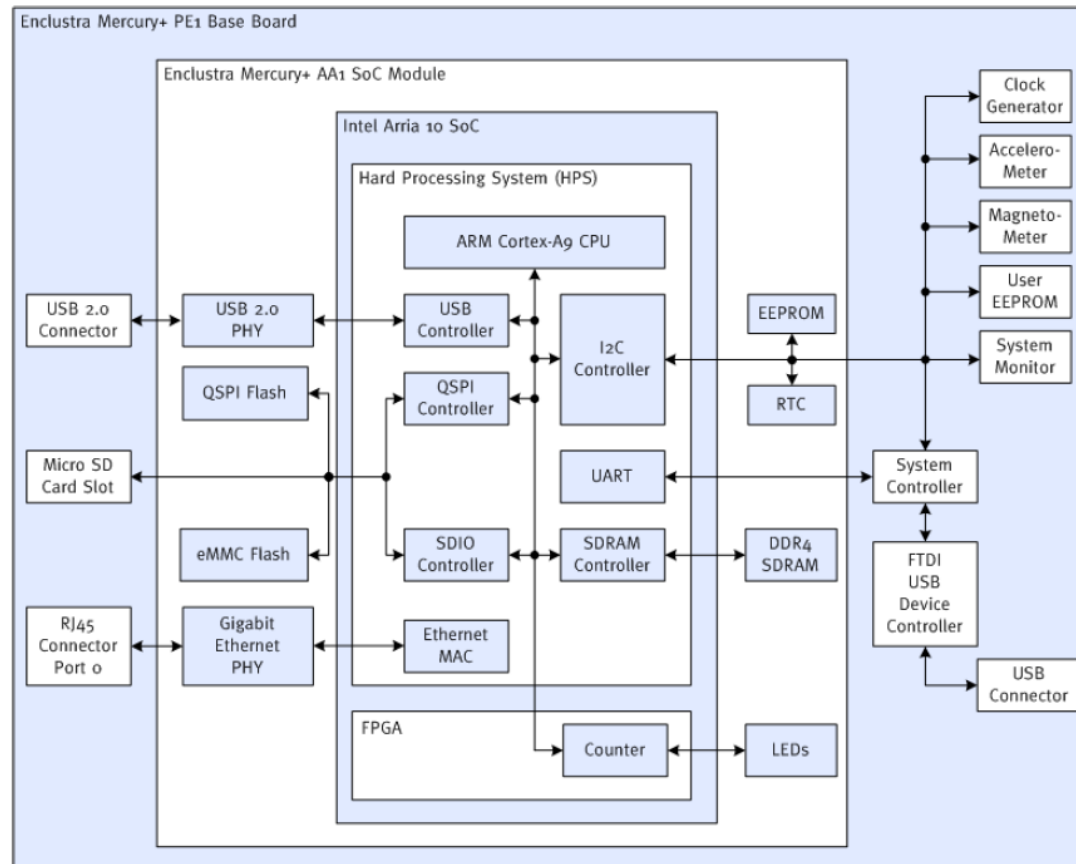
This document first gives an overview of the Mercury+ AA1 SoC module reference design and then guides through the complete Intel SoC design flow for the Mercury+ AA1 SoC module in the getting started section. In addition, the internals and the boot options of the Mercury+ AA1 SoC module reference design are described.

| Product Information | Number | Name |
|---------------------|--------|-------------------------|
| Product | ME-AA1 | Mercury+ AA1 SoC Module |

| Document Information | Reference | Version | Date |
|----------------------------|----------------|---------|------------|
| Reference / Version / Date | D-0000-419-002 | 02 | 21.02.2018 |

| Approval Information | Name | Position | Date |
|----------------------|------|------------------|------------|
| Written by | DIUN | Design Engineer | 29.11.2017 |
| Verified by | GKOE | Technical Expert | 16.01.2018 |
| Approved by | RPAU | Quality Manager | 21.02.2018 |

Enclustra GmbH – Technoparkstr. 1 – CH-8005 Zürich – Switzerland
Phone +41 43 343 39 43 – www.enclustra.com



- Intel SOM reference designs available on Enclustra download site

Product Deliverables – I2C Application Note

<https://github.com/enclustra/I2CAppNote/blob/master/I2CApplicationNote.md>

94 lines (81 sloc) | 7.65 KB

The types of I2C chips used in Enclustra hardware are described in more detail on Enclustra hardware. The examples include sample code for all I2C devices

Table of contents

- Chapter 1 - Examples
 - 1.1 Mercury XU5 example
 - 1.1.1 EEPROM ATSHA204A-MAHDA-T
 - 1.1.2 System monitor Texas Instruments LM96080CIMT/NOPB
 - 1.1.3 Clock generator Silicon Labs Si5338
 - 1.2 Mars ZX2 example
 - 1.2.1 RTC PCF85063A
 - 1.2.2 System controller Lattice LCMXO2-4000HC-6MG132I
 - 1.3 Cosmos XZQ10 system board example
 - 1.3.1 RTC ISL12020M
 - 1.3.2 8-channel bus multiplexer NXP PCA9547
 - 1.3.3 EEPROM 24AA128T-I/MNY
- Chapter 2 - I2C device description
 - 2.1 EEPROM
 - 2.1.1 - Microchip ATSHA204A-MAHDA-T
 - 2.1.2 - Maxim DS28CN01U-A00+
 - 2.1.3 - Microchip 24AA128T-I/MNY
 - 2.2 RTC
 - 2.2.1 - ISL12020M
 - 2.2.2 - PCF85063A
 - 2.3 System Controller Lattice LCMXO2-4000HC-6MG132I
 - 2.4 System Monitor Texas Instruments LM96080CIMT/NOPB
 - 2.5 Clock generator Silicon Labs Si5338
 - 2.6 8-channel bus multiplexer NXP PCA9547

- Chapter 3 - Bare metal
 - 3.1 - EEPROM
 - 3.1.1 - ATSHA204A-MAHDA-T
 - 3.1.2 - DS28CN01U-A00+
 - 3.1.3 - 24AA128T-I/MNY
 - 3.2 - RTC
 - 3.2.1 - Helper functions
 - 3.2.2 - ISL12020M
 - 3.2.3 - PCF85063A
 - 3.3 - System Controller Lattice LCMXO2-4000HC-6MG132I
 - 3.4 - System Monitor LM96080CIMT/NOPB
 - 3.5 - Clock Generator Si5338
 - 3.6 - 8-channel bus multiplexer NXP PCA9547
- Chapter 4 - U-boot
- Chapter 5 - Linux
 - 5.1 - EEPROM
 - 5.2 - RTC
 - 5.3 - System Controller Lattice LCMXO2-4000HC-6MG132I
 - 5.4 - System Monitor LM96080CIMT/NOPB
 - 5.5 Clock generator Silicon Labs Si5338
 - 5.6 8-channel bus multiplexer NXP PCA9547
 - 5.7 Linux I2C tools

Chapter 3 - Bare metal

This chapter describes the usage of the aforementioned devices with the I2C bus and provides sample code snippets for a bare metal application. It is assumed that the hardware used is set up correctly using the corresponding user manual for the hardware in use. The source files for the provided code snippets are also available, e.g. [I2cInterface.h](#).

```
/**
 * \brief I2C subaddress modes.
 */
typedef enum
{
    EI2cSubAddressMode_None, ///< No subaddress
    EI2cSubAddressMode_OneByte, ///< One -byte subaddress
    EI2cSubAddressMode_TwoBytes ///< Two -byte subaddress
} EI2cSubAddressMode_t;

/**
 * \brief Perform any required initialisation for I2C operations.
 *
 * @return Result code
 */
EN_RESULT InitialiseI2cInterface ();

/**
 * \brief Perform a read from the I2C bus.
 *
 * \param[in] deviceAddress The device address
 * \param[in] subAddress Register subaddress
 * \param[in] subAddressMode Subaddress mode
 * \param[in] numberOfBytesToRead The number of bytes to read
 * \param[out] pReadBuffer Buffer to receive read data
 * \returns Result code
 */
EN_RESULT I2cRead(uint8_t deviceAddress,
                  uint16_t subAddress,
                  EI2cSubAddressMode_t subAddressMode,
                  uint32_t numberOfBytesToRead,
                  uint8_t* pReadBuffer);
```

Product Deliverables – Ethernet Application Note

Introduction

The following chapters and sections describe in detail the hardware setup for enabling Gigabit PHY used on all current Endustra modules is the [Microchip KSZ9031RNX](#).

Table of contents

- 1 - MAC address
 - 1.1 Endustra Module Configuration Tool (MCT)
 - 1.2 Bare metal
 - 1.3 U-boot
 - 1.4 Linux
- 2 - RGMII timing constraints
 - 2.1 PHY register configuration via MDIO
 - 2.2 Bare metal
 - 2.3 U-boot
 - 2.4 Linux

References

- [Microchip KSZ9031RNX data sheet](#)
- [Xilinx LWIP App Note XAPP1026](#)
- [Xilinx LWIP library](#)
- [Endustra MCT](#)
- [Endustra EBE kernel](#)

2.3 - U-boot

In U-boot the necessary driver needs to be selected which automatically configures the PHY delay registers with the default value stated in the [data sheet of the KSZ9031RNX](#). The driver for the KSZ9031 PHY chip is located in `Device Drivers -> Ethernet PHY (physical media interface) support -> Micrel Ethernet PHYs support -> Micrel KSZ90x1 family support`.

The MDIO registers can also be accessed and written to in U-boot. The commands `mdio` and `mii` provide read and write as well as other utility functions to configure the PHY chip. Please refer to the [data sheet of the KSZ9031RNX](#) for detailed instructions regarding writing and reading registers.

An example output for the command listing the present Ethernet PHY devices is shown below.

```
zynq-u-boot> mdio list
eth0:
3 - Micrel ksz9031 <--> ethernet@e000b000
zynq-u-boot>
```

The command `mii info` gives further information as shown below.


```
zynq-u-boot> mii info
PHY 0x03: OUI = 0x0885, Model = 0x22, Rev = 0x02, 1000baseT, FDX
zynq-u-boot>
```

With `mii dump` the PHY control registers can be accessed as well.












```
zynq-u-boot> mii dump 3 0
0. <1140>
<8000:0000> 0.15 = 0 -- PHY control register --
<4000:0000> 0.14 = 0 reset
<2040:0040> 0. 6,13 = b10 loopback
<1000:1000> 0.12 = 1 speed selection = 1000 Mbps
<0800:0000> 0.11 = 0 A/N enable
<0400:0000> 0.10 = 0 power-down
<0200:0000> 0. 9 = 0 isolate
<0100:0100> 0. 8 = 1 restart A/N
<0000:0000> 0. 7 = 0 duplex = full
<003f:0000> 0. 5- 0 = 0 collision test enable
<003f:0000> 0. 5- 0 = 0 <reserved>
```


Product Deliverables – PetaLinux BSP for Xilinx SOM

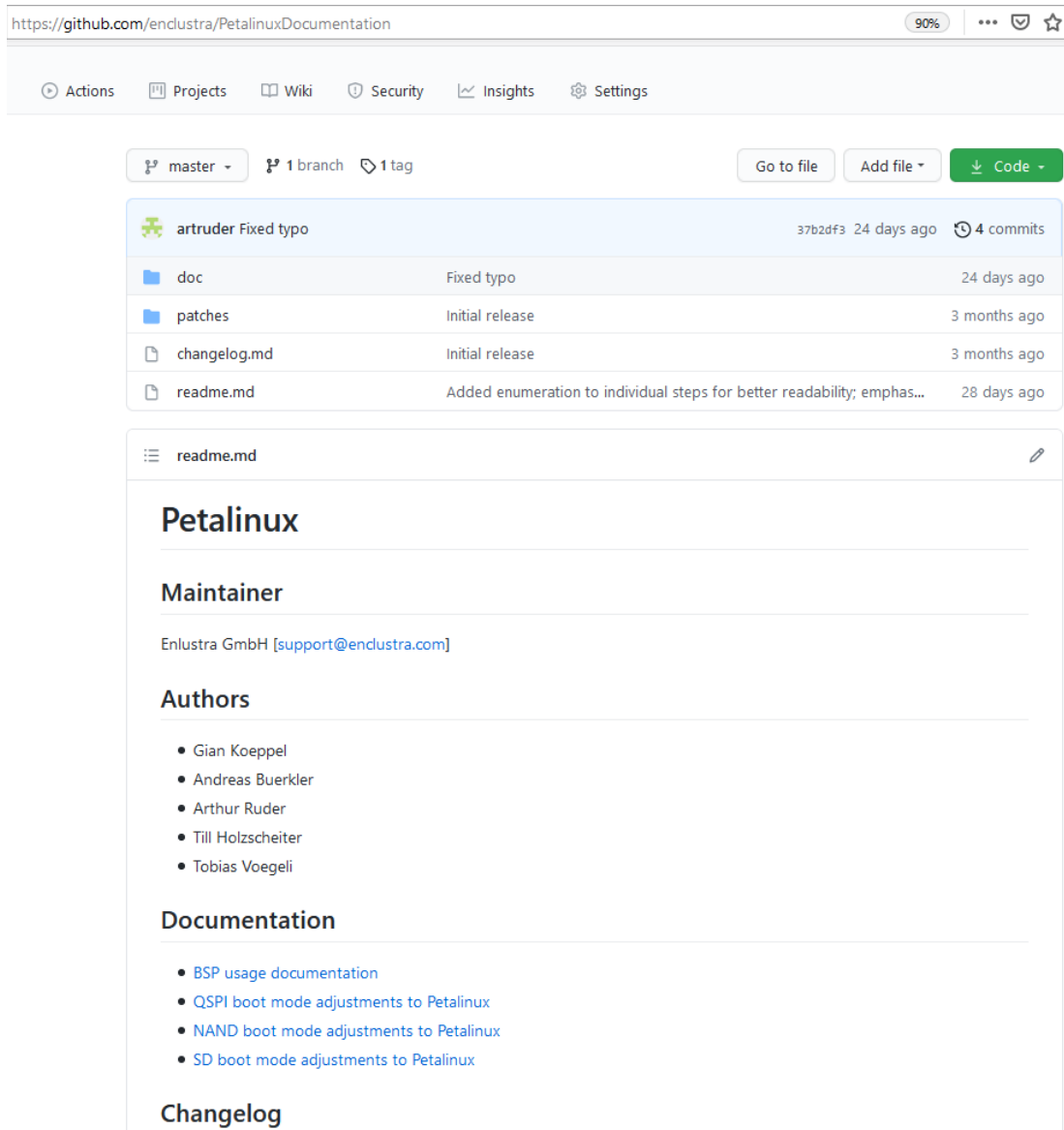
master [Mercury_XU1_PE1_Reference_Design / petalinux /](#) [Go to file](#) [Add file ▾](#) [⋮](#)

 jenkins-enclustra Updated github repository with petalinux build for reference design r... [10c5ee4](#) on Mar 8 [History](#)

..

| | | |
|--|--|--------------|
|  ME-XU1-15EG-1E-D12E-G1_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |
|  ME-XU1-15EG-2I-D12E-G1_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |
|  ME-XU1-15EG-2I-D12E_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |
|  ME-XU1-6CG-1E-D11E-G1_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |
|  ME-XU1-6CG-1E-D11E_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |
|  ME-XU1-6EG-1I-D11E-G1_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |
|  ME-XU1-6EG-1I-D11E_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |
|  ME-XU1-9EG-1E-D11E-G1_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |
|  ME-XU1-9EG-2I-D12E-G1_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |
|  ME-XU1-9EG-2I-D12E_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |
|  ME-XU1-9EG-3E-D12E_PE1 | Updated github repository with petalinux build for reference design r... | 3 months ago |

Product Deliverables – PetaLinux BSP Documentation



The screenshot shows the GitHub repository for PetaLinux Documentation. The repository is named "PetalinuxDocumentation" and is located at <https://github.com/enclustra/PetalinuxDocumentation>. The repository is currently on the "master" branch and has 1 tag. The repository was last updated 24 days ago by user "artruder" with 4 commits. The repository contains a "doc" directory, "patches", "changelog.md", and "readme.md". The "readme.md" file is currently selected and displays the following content:

Petalinux

Maintainer

Enclustra GmbH [support@enclustra.com]

Authors

- Gian Koeppel
- Andreas Buerkler
- Arthur Ruder
- Till Holzscheiter
- Tobias Voegeli

Documentation

- [BSP usage documentation](#)
- [QSPI boot mode adjustments to Petalinux](#)
- [NAND boot mode adjustments to Petalinux](#)
- [SD boot mode adjustments to Petalinux](#)

Changelog

BSP usage documentation

- [BSP usage documentation](#)
 - [Prerequisites](#)
 - [Petalinux project creation and build with BSP file](#)
 - [Updating the hardware file \(.xsa\) used by the Petalinux project](#)
 - [Accelerating Petalinux builds](#)
 - [Changes to Petalinux default configurations](#)
 - [Kernel changes](#)
 - [U-boot changes](#)
 - [Device tree changes](#)
 - [Root file system changes](#)
 - [CPU frequency](#)

Prerequisites

- Supported Linux OS for Petalinux 2020.1
- Petalinux 2020.1 installation and all required packages (for more details please refer to [Xilinx UG 1144](#))

Petalinux project creation and build with BSP file

The provided BSP file follows the naming convention of `Petalinux_${MODULE_NAME}_${BASEBOARD_NAME}_${BOOT_MODE}.bsp` (The BSPs themselves are located in the release page of each respective reference design, e.g. [Mercury XU5 PE1 release 2020.1_v1.1.0](#)). Using the Mercury ME-XU5-2EG-1I-D11E as an example in combination with the PE1 baseboard and SD card boot mode the file will have the name: `Petalinux_ME-XU5-2EG-1I-D11E_PE1_SD.bsp`

1. Open a console and source the Petalinux environment script: `source /<path-to-petalinux-installation-dir>/settings.sh`
2. To create a Petalinux project from the provided bsp file use: `petalinux-create --type project -s <path-to-bsp-file>.bsp`
3. The created project has the following structure:

Product Deliverables – Enclustra Build Environment (EBE)

```
Enclustra Build Environment (v1.5-18258de)
Running under Python version 2.7.13.

Copyright (c) 2015-2017 Enclustra GmbH, Switzerland.
All rights reserved.
```

< OK >

Which targets do you want to build?

- [*] **-Boot**
- [*] Buildroot
- [*] Linux
- [*] Load initial U-Boot configuration
- [*] Load initial Buildroot configuration
- [*] Load initial Linux configuration

< OK > < Back > < Help >

Choose the module type.

- ars XU3**
- Mercury XU1

< OK > < Back >

Setup paths for custom binaries.

| | |
|----------------|-------------------------|
| l31.elf | bl31.elf (default path) |
| fsbl.elf | fsbl.elf (default path) |
| fpga.bit | fpga.bit (default path) |
| pmu.elf | pmu.elf (default path) |

< OK > < Edit > < Back > <Default>

Product Deliverables – EBE Application Note

<https://github.com/enclustra/EBEAppNote/blob/master/EBEApplicationNote.md>

☰ 56 lines (48 sloc) | 4.62 KB

Table of contents

- 1 - Workflow
 - 1.1 Modules equipped with Xilinx SoC
 - 1.2 Modules equipped with Intel SoC
 - 1.3 Additional information
- 2 - U-Boot
 - 2.1 Bootloader
 - 2.2 Folder structure
 - 2.3 Configuration
 - 2.4 Device tree
 - 2.5 Command line interface
 - 2.6 Environment
 - 2.7 Boot scripts
 - 2.8 Additional information
- 3 - Linux kernel
 - 3.1 Folder structure
 - 3.2 Configuration
 - 3.3 Add an existing driver example
 - 3.4 Create and add a custom driver
 - 3.5 Device tree
 - 3.6 Additional information
- 4 - Buildroot
 - 4.1 Root file system
 - 4.2 Folder structure
 - 4.3 Configuration
 - 4.4 Toolchain
 - 4.5 File system modification
 - 4.6 Init system
 - 4.7 Create a custom application
 - 4.8 Debug a custom application
 - 4.9 Copy files to root file system
 - 4.10 Useful commands
 - 4.11 Additional information

🔗 4.8 - Debug a custom application

4.8.1 - Start a debug session in Xilinx SDK

Xilinx provides a System Debugger (TCF) which needs a software (tcf-agent) running on the target Linux. This software is built by default when a Linux build is generated in the Enclustra build environment with default configuration. The needed setting can be found in the Buildroot configuration menu in the `Target packages -> Debugging, profiling and benchmark` submenu.

Prepare Target Linux

- Create a Linux build as described in section 4.7.2
- Connect the Enclustra module via Ethernet to the host machine
- Boot Linux on the module
- Configure an IP address. For example:

```
ifconfig eth0 192.168.1.10
```

- Start Xilinx TCF agent:

```
tcf-agent &
```

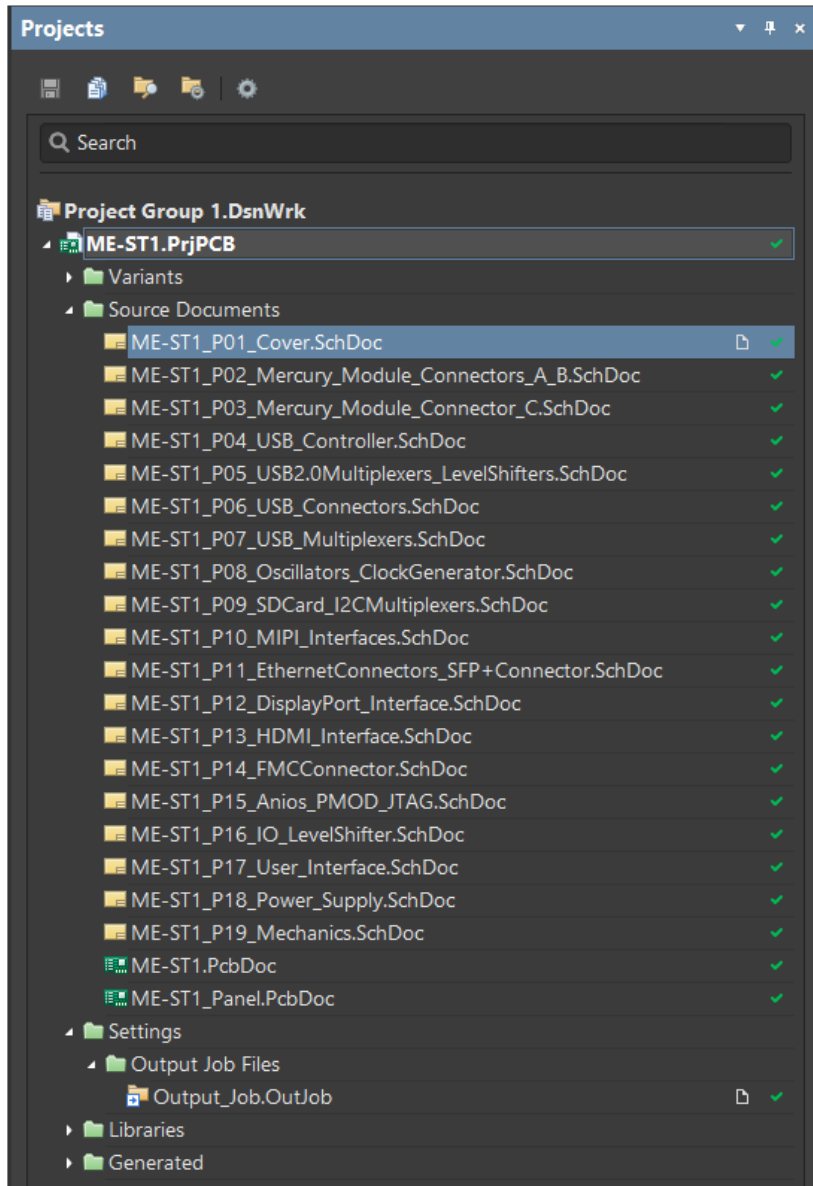
Create Application and Start Debugging

- Open Xilinx SDK
- Create a Linux application as described in section 4.7.2
- Open `Run -> Debug Configurations`
- Click `Xilinx C/C++ application (System Debugger)` to create a new configuration
- Select the `Target Setup` tab
- From the drop down list `Debug Type` select `Linux Application Debug`
- Click `New` to create a new connection
- Enter the IP address of the target module to the `Host` field and set a name for the target. Set the Port to `1534`.
- Click `Test connection` to check if SDK is able to connect to the tcf-agent application running on the target Linux
- Click `Ok` and select the `Application` tab
- Set the project to debug and optionally the working directory (the path to download the application to the target file system)
- Click `Apply`
- Start debugging by pressing `Debug`

4.8.2 - Start a debug session in Intel ARM DS-5

The following example provides instructions on how to start a debugging session in ARM DS-5 using GDB debugger.

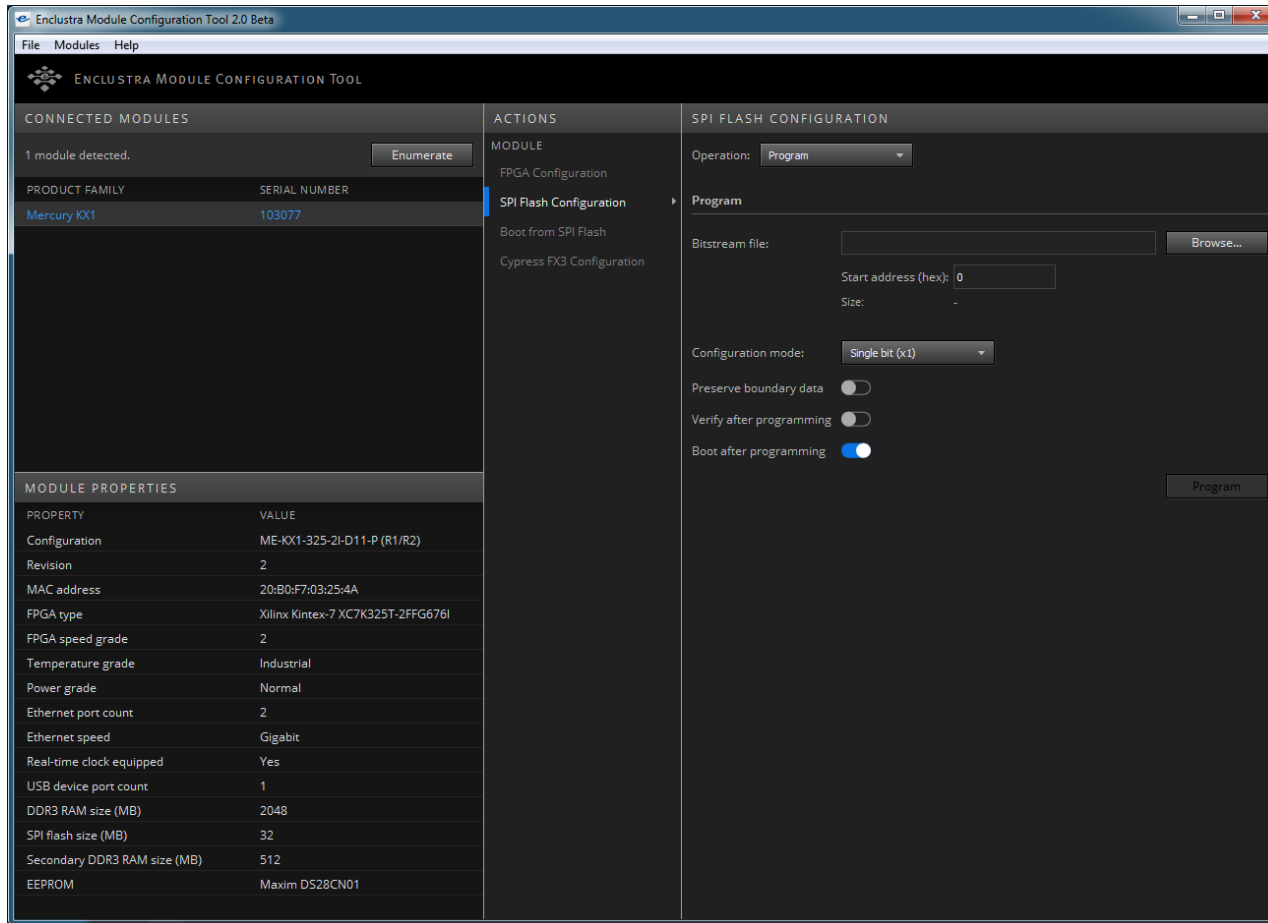
Product Deliverables – Design Files



- For Mercury+ ST1 and Mars ST3 base boards, the Altium design files are available on request by signing a license agreement
- Perfect starting point for a custom baseboard for Enclustra SOM

| Filename Extension | Description |
|--------------------|----------------------------|
| *.OutJob | Altium Output Job File |
| *.PcbDoc | Altium PCB Document |
| *.PrjPCB | Altium PCB Project |
| *.PvLib | Altium Pad Via Library |
| *.SchDoc | Altium Schematic Document |
| *.pdf | Design Files Release Notes |
| *.xlsx | BOM Template |

Product Deliverables – Module Configuration Tool (MCT)



- Configure Enclustra SOM and base boards via USB without additional hardware
 - Program FPGA
 - Program SPI flash
 - Read EEPROM configuration
 - Configure peripheral devices

- Custom base board schematics and PCB reviews
- Design-in support
- Custom modules
 - New product models (special assembly variant)
 - New products (SOM co-design)

If you are looking to...

- Reduce your Time-to-Market
- Increase your Market Share
- Reduce your Total Cost of Ownership
- Improve your quality
- Increase your flexibility
- Differentiate your offering



Enclustra can help you – let's talk



Everything FPGA