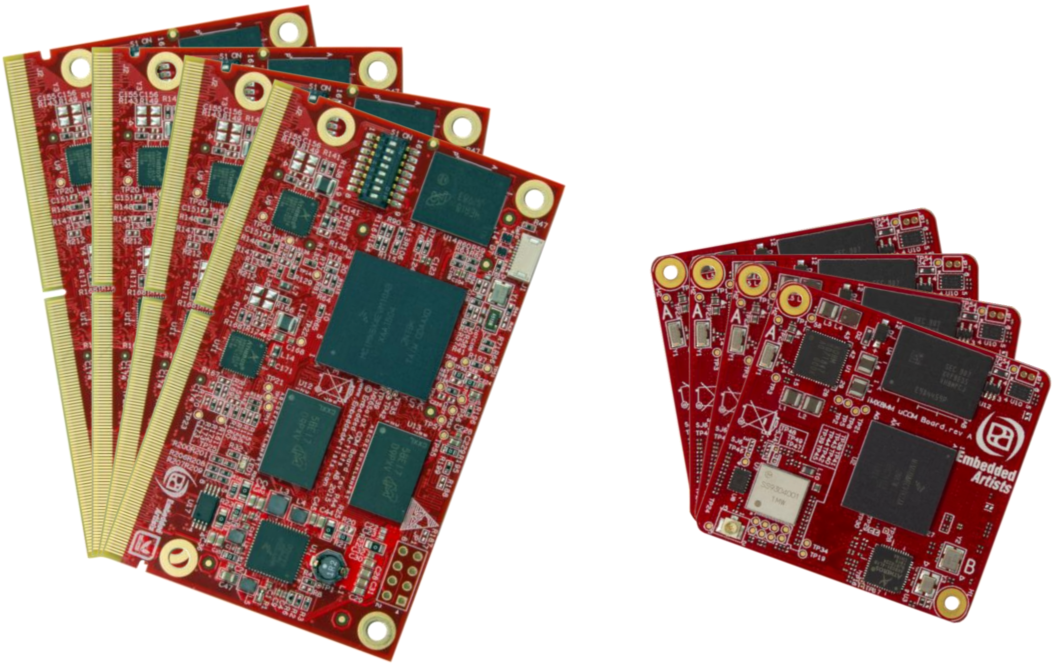
Copyright 2020 © Embedded Artists AB

EA(u)COM Selection Guide – Which i.MX to select?



Embedded Artists

Embedded Artists AB
Jörgen Ankersgatan 12
21145 Malmö
Sweden

https://www.EmbeddedArtists.com

Copyright 2020 © Embedded Artists AB. All rights reserved.

No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the prior written permission of Embedded Artists AB.

Disclaimer

Embedded Artists AB makes no representation or warranties with respect to the contents hereof and specifically disclaim any implied warranties or merchantability or fitness for any particular purpose. The information has been carefully checked and is believed to be accurate, however, no responsibility is assumed for inaccuracies.
Information in this publication is subject to change without notice and does not represent a commitment on the part of Embedded Artists AB.

Feedback

We appreciate any feedback you may have for improvements on this document.

Trademarks

All brand and product names mentioned herein are trademarks, services marks, registered trademarks, or registered service marks of their respective owners and should be treated as such.

Table of Contents

Copyright 2020 © Embedded Artists AB

1 Document Revision History

Revision	*Date*	*Description*
PA1	2016-02-02	First version.
PA2	2016-02-03	Added information about iMX7Solo/Dual.
PA3	2017-07-31	Added several tables.
A	2020-05-01	Updated with iMX7ULP, iMX8M, iMX8M Mini and iMX8M Nano. Also added reference to EA(u)COM.

Copyright 2020 © Embedded Artists AB

2 Introduction

This document give guidance in selecting a suitable application i.MX processor and EA(u)COM board to base a customer project around. Note that this document does not cover the i.MX RT crossover processor family.

This chapter gives a short overview of the main points to consider when selecting an EACOM or EAuCOM board, called EA(u)COM board for simplicity in the rest of the document. The following chapter will present each point to consider in more detail.

It is highly recommended to have worked out a detailed system or project requirement specification beforehand. From this document it should be relatively simple to extract the requirements (and answers) that will guide the EA(u)COM board selection. Below is a list of points that needs to be worked out (i.e., answered):

- **Determine what interfaces that are needed**
  List external wired communication interfaces like Ethernet, USB and CAN.
  List interface like SPI and I2C. Do not forget wireless interfaces that are typically implemented with RF-modules that have commonly used interfaces like SDIO and PCIe, and also SPI and USB.
  Do not forget if there is a need for external storage, like a USB/PCIe/SATA hard drive.
  Also list display interfaces (both internal and external).
- **Estimate the execution performance that is needed**
  This is probably the most difficult question to answer. There are however some guidelines around this further down in this document.
- **Determine what kind of display and graphical processing that is needed**
  Specify the resolution, frame update rate and pixel clock of the displays that are needed.
  Also specify if special graphical processing is needed, like video decoding, video encoding or 2D/3D processing.
- **Determine the amount of memory needed in the system**
  Specify amount of volatile memory (RAM) and non-volatile memory (FLASH).
  This can probably also be difficult to estimate with a detailed number. There are however some guidelines around this.
- **Determine if a real time coprocessor is needed - Heterogeneous Multicore Processing**
  Some i.MX family members have a Cortex-M core that can be used to handle real time critical tasks or, alternatively, to save power (since the Cortex-M core can do pre-processing before the more power hungry application processor is activated).
  The term **Heterogeneous Multicore Processing**, or HMP for short, is commonly used to describe different types of cores (Cortex-A and Cortex-M, in this case) operating together.
- **Determine what operating system that will be used**
  There are basically three operating systems to chose between for the application processor, Linux, Android or (some version of) embedded Windows. FreeRTOS support is available for the Cortex-M cores.
- **Determine if commercial or industrial temperature grade is needed, also consider longevity need**
  Specify the temperature range and longevity that is needed.
  Also specify what activity duty cycle the system will operate, for example 'always on' or 10%.
- **Determine need for low power consumption**
  Specify maximum power consumption the design can tolerate in active and inactive states.
- **Determine physical form factor**
  Physical requirements can dictate if an EACOM or EAuCOM board shall be considered.

Copyright 2020 © Embedded Artists AB

2.1 General guidelines

Besides the points above that needs answers, there are a few other general guidelines for EA(u)COM board selection.

- It is better to get started with something, than finding the optimal solution directly. The i.MX6/7/8 family members are very compatible and it is relatively easy to switch between members. Further, all different versions of EA(u)COM boards are not available in low volumes. Low volume users (<100 pcs/order) should select from the preferred, normally stocked EA(u)COM boards.
  Our general recommendation is to **start with a higher-end (more feature rich) EA(u)COM board - do the development and prove the design in the field. Then optimize which board to select in the end!** At that point the memory and execution needs are known with greater precision.
- Another general recommendation is to **focus on doing a proof-of-concept/prototype with one of the available iMX Developer's Kits**. By starting with a proven EA(u)COM based design, the project can move forward quickly. This is one of the main benefits with EA(u)COM boards - you can focus on your core business! ...and minimize your risk and time to market!
  - Do not go for a custom specific board or a Single Board Computer (SBC) solution before the design is tried out and proven in the field.
  - Go into production with an EA(u)COM board solution. Test the design. Make sure there are no (unforeseen) problems.
  - When production volumes start to increase for your system, a more cost optimized design can be worked out. A typical breakpoint is >10K/year, but it can vary from design to design.
- **Contact Embedded Artists**
  If you need guidance or have questions - just contact us!

Copyright 2020 © Embedded Artists AB

3 System Specification

This chapter presents more details around each of the points presented in chapter 2 . Basically it is all about getting the information from the existing system requirements specification. Alternatively it is all about updating the existing specification with certain important information.

3.1 Interfaces - Connectivity

A good start on the system requirement specification is to determining which interfaces that are needed. It can be useful to divide the interfaces into external and internal interfaces.

- External interfaces are represented by the external connectors to the system and are often dictated by customer requirements and common standards. Examples of common external interfaces are:
    - Ethernet
    - CAN-bus
    - SD/MMC
    - USB
    - HDMI
    - LVDS to display. Depending on system architecture, this may be an internal interface.
- Internal interfaces are a little more flexible and subject to design decisions. It is typically internal busses on the carrier board that are needed to implement the different functions. Examples of common internal interfaces are:
    - I2C
    - UART
    - SPI
    - GPIO
    - PWM
    - I2S/SSI for audio
    - MIPI-DSI / MIPI-CSI

  Some internal interfaces are very standardized (with connectors), like PCIe and SATA. They still count as internal since they are typically not accessible from outside of the enclosure of the system.

Do not forget wireless interfaces that are typically implemented with RF-modules that have interfaces like SPI, SDIO, USB and PCIe. Also list display interfaces (both internal and external).

If a design requires a large number of certain interfaces, for example 24 GPIOs, 10 UART or 5 USB interfaces, the solution is to place expanders/multiplexors/bridges on the carrier board. For USB it is for example possible to add a USB hub on the carrier board. GPIOs are for example possible to create via I2C-GPIO expanders. Analog interface like ADC and DAC are often implemented on the carrier board with I2C or SPI interfaces to the ADC/DAC components. I2C is suitable for lower rate signals while SPI is used for higher data rate signals.

Assuming that the needed interfaces has been written down; **Step 1 is to determine if the needed interfaces are satisfied by the standard interfaces of EA(u)COM boards**. The general recommendation is to follow the EA(u)COM pinning standard. It gives flexibility to switch between

Copyright 2020 © Embedded Artists AB

boards (upgrade/downgrade) and future-proof the design - new boards will have EA(u)COM pinning. The table below lists the allocation of standard interfaces for EACOM as well as EAuCOM boards.

Interface	EACOM specification	EAuCOM specification
UART	3 ports	3 ports
SPI	2 ports	2 ports
I2C	3 ports	4 ports
SD/MMC	2 ports	1 port
Parallel LCD	24 databits, 4 ctrl	24 databits, 4 ctrl
LCD support	LCD power ctrl, Backlight power/contrast control, touch panel ctrl (RST and IRQ)	LCD power ctrl, Backlight power/contrast control, touch panel ctrl (RST and IRQ)
LVDS LCD	2 ports	-
MIPI-DSI	-	4 lanes
HDMI (TDMS)	1 port	-
Parallel Camera	8 databits, 4 ctrl	-
MIPI-CSI Camera	4 lanes	4 lanes
Gigabit Ethernet	2 ports	1 port
PCIe	1 port, 1 lane	1 port, 1 lane
SATA	1 port	-
USB	1 USB3.0 OTG 1 USB3.0 Host 1 USB2.0 Host	2 USB2.0 OTG
SPDIF	1 TX/RX port	-
CAN	2 ports	2 ports
I2S/SSI/AC97	1 port	1 port
Analog audio	Stereo output	-
GPIO	9 pins	12 pins
PWM	1 pin	-
ADC	8 inputs	4 inputs
JTAG	Available on separate connector	Available on expansion connector
Powering	Pins reserved for 3.3V On-board PMIC only powers the EACOM board.	Pins reserved for battery charging On-board PMIC can power circuits on the carrier board.

Note that all standard interfaces in the EA(u)COM specification are not implemented on all EA(u)COM boards. The reason can be that some interfaces are not implemented on all i.MX6/7/8 family members, like PCIe and SATA. The reason can also be limited number of pins, resulting in situations where one or the other interface can be used, but not both at the same time. Use the accompanying Excel sheet that is available to check which EA(u)COM boards that meets the requirements.

Copyright 2020 © Embedded Artists AB

If no EA(u)COM board fulfills all requirements there are three options:

1. **Rethink the interface requirements**. Very often small changes can be done on the carrier board design to meet the EA(u)COM pinning standard.
2. **Use EA(u)COM type-specific interfaces**. Besides the standard interfaces defined by EACOM there are also type-specific interfaces, which are the pins that are left after the standard interfaces have been allocated. Using these pins make full usage of the i.MX6/7/8 processor but limits the possibilities to switch between EA(u)COM boards. In many cases this is no problem at all.
   - Many pins on the EA(u)COM connectors are left unassigned. These pins are reserved for type-specific interfaces. Different boards can have different signals and interfaces assigned to these positions.
3. Ignore the EA(u)COM pinning standard and **make full use of all i.MX6/7/8 pins** that are available on the expansion connectors. This make it difficult to switch between EA(u)COM boards. In many cases this is no problem at all. The benefits of having complete flexibility in pin assignment outweigh the disadvantages.
   - There are accompanying Excel sheets for each EA(u)COM board with details about what peripheral functions each pin can carry.

Option 2) and 3) above contradicts the general recommendation to follow the EA(u)COM pinning standard, but sometimes this can be a reasonable trade-off between flexibility and efficiency.

The table below lists the major interfaces available. It can serve as a first start. Note that all of them might not be available at the same time due to pin multiplexing limitations.

EA(u)COM board	USB	Ethernet	UART	SPI	I2C	SD/ MMC	CA N	Serial Audio	Display interfaces	Other
iMX6 UltraLite, iMX6ULL COM	2xOTG	2x100/10Mbps PHY	8	3	4	1xSD	2	3xSAI/I2S, S/PDIF	RGB	Parallel CSI
iMX6 SoloX COM	2xOTG	2xGigabit PHY	6	4	4	3xSD	2	5xSSI/I2S, ESAI, SAI, S/PDIF	RGB, LVDS	Parallel CSI, PCIe
iMX6 DualLite COM	1xOTG, 1xHOST, 2xHSIC	1xGigabit PHY	5	4	4	3xSD	2	3xSSI/I2S, ESAI, S/PDIF	RGB, 2xLVDS, HDMI, MIPI-DSI	Parallel CSI, MIPI-CSI, PCIe
iMX6 Dual COM	1xOTG, 1xHOST, 2xHSIC	1xGigabit PHY	5	5	3	3xSD	2	3xSSI/I2S, ESAI, S/PDIF	RGB, 2xLVDS, HDMI, MIPI-DSI	Parallel CSI, MIPI-CSI, SATA, PCIe
iMX6 Quad COM	1xOTG, 1xHOST, 2xHSIC	1xGigabit PHY	5	5	3	3xSD	2	3xSSI/I2S, ESAI, S/PDIF	RGB, 2xLVDS, HDMI, MIPI-DSI	Parallel CSI, MIPI-CSI, SATA, PCIe
iMX7 Dual COM	2xOTG, 1xHSIC	1xGigabit PHY, 1xExternal Gigabit PHY possible	7	4	4	2xSD	2	3xSAI	RGB, MIPI-DSI	Parallel CSI, MIPI-CSI, PCIe
iMX7 Dual uCOM	2xOTG, 1xHSIC	2xExternal Gigabit PHY possible	7	4	4	2xSD	2	3xSAI	RGB, MIPI-DSI	Parallel CSI, MIPI-CSI, PCIe
iMX7ULP uCOM	1xOTG, 1xHSIC	Optional Wi-Fi/BT module	8	4	8	1xSD	-	3xI2S	MIPI-DSI	Parallel CSI
iMX8M COM	2x OTG 3.0	1xGigabit PHY	4	3	4	1xSD	-	5xSAI, S/PDIF	HDMI, MIPI-DSI	2xPCIe, 2xMIPI-CSI
iMX8M Mini uCOM	2xOTG	1xGigabit PHY Optional Wi-Fi/BT module	4	3	4	3xSD	-	5xSAI, 8xPDM, S/PDIF	MIPI-DSI	MIPI-CSI, PCIe
iMX8M Nano uCOM	1xOTG	1xGigabit PHY Optional Wi-Fi/BT module	4	3	4	2xSD	-	5xSAI, 8xPDM, S/PDIF	MIPI-DSI	MIPI-CSI

Copyright 2020 © Embedded Artists AB

3.2 Execution Performance

The next parameter that influence the EA(u)COM board selection heavily is the execution performance needed. It is a difficult question to answer but with some guidelines and rule-of-thumbs it is possible to estimate.

The execution performance is a combination of the following factors:

- If the i.MX6/7/8 processor has ARM Cortex-A53, Cortex-A9 or Cortex-A7 core(s)
- Number of cores in the i.MX6/7/8 processor
- Clock frequency for the core(s)
- Internal cache sizes (L1/L2)
- External memory bus transaction frequency
- External memory bus width
- If the i.MX6/7/8 processor has an ARM Cortex-M4 or Cortex-M7 core

Below is a table listing the different i.MX6/7/8 family members, which corresponds to EA(u)COM boards. The corresponding DMIPS (Dhrystone MIPS) number is also listed. Note that this number is approximate and there are definitively reasons to be skeptic to synthetic computing benchmarks. A DMIPS number does not capture all aspects and information about the processor/EA(u)COM board. High-resolution screens can, for example, consume considerable bandwidth of the memory system. In multi-core systems the application's ability to be parallelized is a very important factor. Nevertheless, it is possible to compare the relative speed between the EA(u)COM boards by comparing DMIPS numbers. The table also lists some typical comparative numbers for other ARM cores.

iMX family member and related EA(u)COM board	Core(s) / Max Frequency	Memory bus	DMIPS - Integer Performance (approximate)
iMX6UltraLite / iMX6ULL COM	A7 @ 528MHz A7 @ 900MHz	16-bit / 800MT/s	1000 DMIPS 1710 DMIPS
iMX6SoloX COM	A9 @ 1GHz + M4 @ 227MHz	32-bit / 800MT/s	2400 DMIPS + 208 DMIPS (M4)
iMX6Solo COM	A9 @ 1GHz	32-bit / 800MT/s	2400 DMIPS
iMX6DualLite COM	2x A9 @ 1GHz	64-bit / 800MT/s	4800 DMIPS
iMX6Dual COM	2x A9 @ 1GHz	64-bit / 1066MT/s	5700 DMIPS
iMX6Quad COM	4x A9 @ 1GHz	64-bit / 1066MT/s	11500 DMIPS
iMX7Solo COM/uCOM	A7 @ 800MHz + M4 @ 200MHz	32-bit / 1066MT/s	1520 DMIPS + 250 DMIPS (M4)
iMX7Dual COM/uCOM	2x A7 @ 1GHz + M4 @ 200MHz	32-bit / 1066MT/s	3800 DMIPS + 250 DMIPS (M4)
iMX7ULP uCOM	A7 @ 720MHz + M4 @ 200MHz	32-bit / 720MT/s	1400 DMIPS + 250 DMIPS (M4)
iMX8M COM	4x A53 @ 1.5GHz + M4 @ 266MHz	32-bit / 3200MT/s	13800 DMIPS
iMX8M Mini uCOM	4x A53 @ 1.8GHz + M4 @ 400MHz	32-bit / 3000MT/s	16500 DMIPS
iMX8M Nano uCOM	4x A53 @ 1.5GHz + M7 @ 750MHz	16-bit / 2400MT/s	13800 DMIPS

Copyright 2020 © Embedded Artists AB

Generic Cortex-M3/M4 for comparison	1.1-1.25 DMIPS/MHz, which translates to: 125 DMIPS @ 100MHz 225 DMIPS @ 180MHz
Generic Cortex-M7 for comparison	1.9-2.14 DMIPS/MHz, which translates to: 1500 DMIPS @ 750MHz
Generic ARM926 for comparison	1.25 DMIPS/MHz, which translates to: 220 DMIPS @ 200MHz 290 DMIPS @ 266MHz 550 DMIPS @ 454MHz

Note that it is not simple to compare DMIPS numbers from a microcontroller project running an RTOS with an application processor system running Linux as OS. The overhead in the Linux system is much bigger. Nevertheless, the numbers above indicates that the available raw processing power is much higher for i.MX members than Cortex-M3/M4/M7 based microcontrollers.

Few embedded systems require the sustained raw execution performance that four Cortex-A53 cores running at 1.8 GHz gives. However sometimes latency times are important. On average the needed performance is low, but during short periods of time the need is considerably higher. Examples of such systems are:

- Graphical user interface that must be responsive and free from irritating delays.
- Server application that must produce a response quickly, i.e., within a certain time.

Some system must be dimensioned for high peak load, although the peak load situation might be very uncommon. Example of such systems are:

- A system that has many simultaneous users during rush hour.
- An access system with many remote access points. It is unlikely that users access all remote access point to the system at the exact time, but it can happen and the system must be dimensioned for that.

The best way forward is to prototype and test the system early. Measure what response times the prototype system has and draw conclusions from that. Add possible margins for feature growth during the project. That is one strong reason for our recommendation to do a proof-of-concept as early as possible. It does not have to be the whole system - just the part that might be driving the execution performance needs. A great thing with EA(u)COM boards is that it is possible to downsize or upsize over a wide range of execution performance.

Copyright 2020 © Embedded Artists AB

3.3 Graphical Performance

Next thing to consider is what graphical performance the system needs. The list below outlines the parameters that needs to be evaluated.

1) Specify how many displays are needed, what their resolution and update rates are.
   - Each display consumes a certain bandwidth of the external memory bus. High-resolution displays can consume considerable bandwidth. For example, a 1024x768 pixels display with 50Hz frame rate consumes about 10% bandwidth of a 32-bit, 400MT/s memory bus.
   - A display not only consumes a constant bandwidth to refresh the display but also a load every time the content shall be updated.
2) Determine what kind of graphical acceleration that is needed.
   - Most graphical user interfaces to embedded systems have little need for 3D graphical acceleration. User interfaces typically do not look or behave like 3D games. There are always exceptions and parts of the user interface can require 3D acceleration. Evaluate the planned graphical user interface and estimate what kind of 2D/3D acceleration that is needed.
   - NXP has published a document called *i.MX Graphics User's Guide*, document number: **IMXGRAPHICUG**. The document provide good information on Linux graphic APIs and driver support.
   - A graphical user interface that requires a lot of updating (on the screen) in response to user events will require considerable graphical acceleration hardware to keep the user interface responsive.
3) Specify if video decoding/encoding is needed and the performance needed.
   - Some of the i.MX family members have on-chip video encoders/decoders. Typically, digital signage applications have need for this type of functionality (video decoding). Video encoding is closely related to application with cameras.

The exact resolutions supported is not crystal clear when reading the i.MX datasheets and reference manuals. There is an upper limit resolution of the display controller but there is also an upper limit on the data rates caused by the display(s). There is also a question about how much memory bandwidth to reserve for the display(s) and processor cores. There must be enough execution performance left to be able to update the display content in a user friendly way, i.e., without noticeable delays.

The table below lists a few important numbers, but remember to always consult the datasheet and reference manual for exact details.

EA(u)COM board	Maximum recommended resolution @ 60Hz	Maximum pixel clock (combination of recommendation and datasheet limits)	Maximum number of simultaneous displays
iMX6 UltraLite / iMX6ULL COM	1366x768	Parallel RGB: 85 MHz	1
iMX6 SoloX COM	1366x768 or 2x 1280x720	Parallel RGB: 85 MHz (stretch to 150 MHz) LVDS: 85 MHz	2
iMX6 DualLite COM	1920x1200 or 2x 1366x768	Parallel RGB: 100 MHz (stretch to 200 MHz) HDMI: 264 MHz LVDS: 85 MHz each	2
iMX6 Dual COM	2x 2048x1536	Parallel RGB: 100 MHz (stretch to 225 MHz) HDMI: 240 MHz LVDS: 85 MHz each (165 MHz combined)	4

Copyright 2020 © Embedded Artists AB

iMX6 Quad COM	2x 2048x1536	Parallel RGB: 225 MHz HDMI: 240 MHz LVDS: 85 MHz each (165 MHz combined)	4
iMX7 Dual COM **iMX7 Dual uCOM**	1920x1080 via MIPI 1366x768 via Parallell RGB	Parallel RGB: 85 MHz (stretch to 150 MHz)	1
iMX7ULP uCOM	unspecified	MIPI-DSI: up to 792 Mbps bit rate	1
iMX8M COM	4096x2160 via HDMI 1920x1080 via MIPI	HDMI: 596 MHz MIPI-DSI: up to 1.5 Gbps bit rate	2
iMX8M Mini uCOM	1920x1080 via MIPI	MIPI-DSI: up to 1.5 Gbps bit rate	1
iMX8M Nano uCOM	1920x1080 via MIPI	MIPI-DSI: up to 1.5 Gbps bit rate	1

The table below lists the graphical acceleration hardware available in the different i.MX family members. The numbers can be difficult to relate to a typical need to *'have a responsive graphical user interface'*. Again, **the best way forward is to prototype and test the system early. Measure the graphical acceleration performance the selected system has. Determine if more or less acceleration is needed and change EA(u)COM board if needed**.

iMX family member and related EA(u)COM board	Hardware 2D/3D Graphics Acceleration APIs	3D GPU	2D GPU	Vector GPU	VPU - Video Processing Unit
iMX6UltraLite /iMX6ULL COM	-		Pixel processing pipeline (PXP)	-	Software only
iMX6SoloX COM	OpenGL ES 1.1/2.0 OpenVG 1.1 2DBLT 1 shader - 720 MHz	Vivante GC400T 1x shader, 27Mtri/s, 133Mpxl/s	Via 3D GPU 300Mpxl/s	-	Software only
iMX6Solo COM	OpenGL ES 1.1/2.0/3.0 OpenVG 1.1 2DBLT 1 shader - 528 Mhz	Vivante GC880 1x shader, 35Mtri/s 266Mpxl/s	Vivante GC320 600Mpxl/s	Via 3D GPU	HD1080p30 decode 1080p30 or dual 720p encode
iMX6DualLite COM	OpenGL ES 1.1/2.0/3.0 OpenVG 1.1 2DBLT 2 layer composition 1 shader - 528 MHz	Vivante GC880 1x shader, 35Mtri/s 266Mpxl/s	Vivante GC320 600Mpxl/s	Via 3D GPU	HD1080p30 decode 1080p30 or dual 720p encode
iMX6Dual COM	OpenGL ES 1.1/2.0/3.0 OpenCL 1.1 EP OpenVG 1.1 2DBLT 2 layer composition 4 shaders - 594 MHz	Vivante GC2000 4x shaders, 176Mtri/s 850Mpxl/s	Vivante GC320 600Mpxl/s	Vivante GC355 300Mpxl/s	Dual HD1080p30 or Single HD1080p60 decode 1080p30 or dual 720p encode
iMX6Quad COM	OpenGL ES 1.1/2.0/3.0 OpenCL™ 1.1 EP OpenVG™ 1.1 2DBLT 2 layer composition 4 shaders - 594 MHz	Vivante GC2000 4x shaders, 176Mtri/s 850Mpxl/s	Vivante GC320 600Mpxl/s	Vivante GC355 300Mpxl/s	Dual HD1080p30 or Single HD1080p60 video 1080p30 or dual 720p encode

Copyright 2020 © Embedded Artists AB

iMX7Solo & iMX7Dual COM	-		Pixel processing pipeline (PXP v3)	-	Software only
iMX7ULP uCOM	OpenGL ES 1.1/2.0 OpenVG™ 1.1	Vivante GCNanoUltra	Vivante GC320	Via 3D GPU	Software only
iMX8M COM	OpenGL ES 1.1/2.0/3.0/3.1 OpenCL 1.2 Vulcan	Vivante GC7000Lite 4x shaders 267Mtri/s 1600Mpxl/s 32GFLOPS	Via 3D GPU	Via 3D GPU	4Kp60 decode Software 1080p30 H.264 encode (use 3x A53 cores)
iMX8M Mini uCOM	OpenGL ES 1.1/2.0 OpenVG™ 1.1	Vivante GCNanoUltra 1x shader, 40Mtri/s 400Mpxl/s 6.4GFLOPS	Vivante GC320	Via 3D GPU	HD1080p60 decode 1080p60 encode
iMX8M Nano uCOM	OpenGL ES 1.1/2.0/3.0/3.1 OpenCL 1.2 Vulcan	Vivante GC7000UL 2x shader, 100Mtri/s 600Mpxl/s 9.6GFLOPS	Via 3D GPU	Via 3D GPU	Software only

Copyright 2020 © Embedded Artists AB

3.4 Memory

The next thing to determine is the memory need for the system. As with previous points, it is difficult to determine a fixed number without actually running the system. However, some guidelines exist.

The system memory (RAM) is in the region between 512 MByte and 2 GByte on standard EA(u)COM boards. This is in the upper range of what is typically needed. It is a design and marketing decision to have top-of-the-line boards that are competent and fulfill most customer's needs as opposed to marketing stripped boards where customers sooner or later encounter the limitations. The guidelines for system memory are as follows:

- 32 - 64 MByte is the absolute minimum for single core systems. The system must be carefully controlled to keep the size of the kernel to a minimum. Applications that run on the system must be careful with memory usage.
- 256 MByte will likely be more than enough for systems with a low-resolution display or no display at all.
- 512 MByte - 1 GByte will likely be sufficient for 1-4 core systems with high-resolution displays. As a reference, 512 MByte is the minimum recommended size for *Ubuntu Desktop*.
- 2 - 4 GByte might be needed for some systems, for example server applications (databases, webs servers, etc.) and communication systems/gateways.
- Android OS have detailed recommendations that depends heavily on screen resolution and dots-per-inch, but is in the region of 32 - 512 MByte.

For storage memory (FLASH), most EA(u)COM boards have on-board 4-8 GByte eMMC FLASH. This is typically more than needed for a common system and gives a lot of headroom for future expansions. The guidelines for storage memory are:

- Unless specific requirements for the system indicates other, a rule-of-thumb is to have a 1:4 relationship between system and storage memory.
- Android OS have a strong recommendation to have at least 3 GByte storage memory.
- Besides FLASH memory, the system can have special requirements to store large amounts of data. This can be done with a hard drive with SATA or USB interface. It can be a solid state disk (SSD) or with mechanical disks. A spare SD/MMC interface can also be used for FLASH expansion.

Our general recommendation is to start with a standard configuration; between 512-2 GByte system memory and 4-8 GByte storage memory. Prototype the system and measure the memory consumption in typical usage scenarios. There are Linux commands to checking system memory usage. The storage memory size is easy to measure. It is the size of the root file system plus any additional partitions.

The table below lists memory sizes, memory bus width and transaction speeds for the different boards.

EA(u)COM board	RAM	FLASH	Memory bus width and speed
iMX6 UltraLite/ iMX6ULL COM	512 MByte	4 Gbyte eMMC	16-bit / 800MT/s
iMX6 SoloX COM	1 GByte	4 Gbyte eMMC	32-bit / 800MT/s
iMX6 DualLite COM	1 GByte	4 Gbyte eMMC	64-bit / 800MT/s
iMX6 Dual COM	2 GByte	4 Gbyte eMMC	64-bit / 1066MT/s
iMX6 Quad COM	2 GByte	4 Gbyte eMMC	64-bit / 1066MT/s
iMX7 Dual COM	1 GByte	4 Gbyte eMMC	32-bit / 1066MT/s
iMX7 Dual uCOM	1 GByte	8 Gbyte eMMC	32-bit / 1066MT/s

Copyright 2020 © Embedded Artists AB

iMX7ULP uCOM	1 GByte	8 Gbyte eMMC	32-bit / 720MT/s
iMX8M COM	1 GByte	8 Gbyte eMMC	32-bit / 3200MT/s
iMX8M Mini uCOM	1 GByte	8 Gbyte eMMC	32-bit / 3000MT/s
iMX8M Nano uCOM	1 GByte	8 Gbyte eMMC	16-bit / 2400MT/s

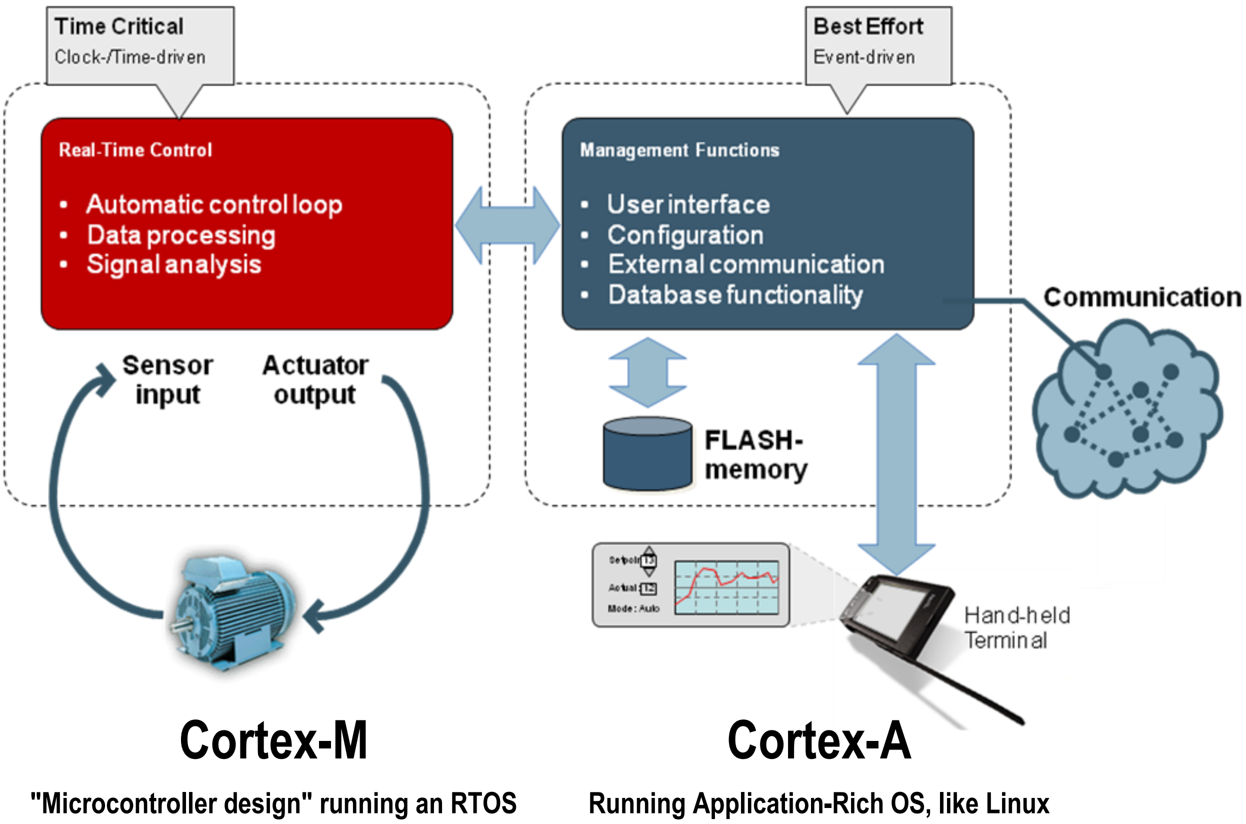
Copyright 2020 © Embedded Artists AB

3.5 Real-Time Coprocessor - HMP

Determine if an extra Cortex-M core is needed in the system. Some i.MX family members have this extra Cortex-M core that can be used to handle real time critical tasks or, alternatively, to save power. In the latter case, the Cortex-M core can do data collection and pre-processing before the more power hungry application processor is activated.

A processor with different types of processor cores, Cortex-A and Cortex-M in this case is said to have a **Heterogeneous Multicore Processing** system architecture, or HMP system architecture, for short.

Many embedded systems typically looks like the picture below. There is a core application with real time control and there are management functions that are not so focused on real time, but rather on complex functionality. The management functions have increased a lot in scope and complexity during the past couple of years. This trend show no signs of stopping. Customers require more and more functionality that not necessarily is related to the core functionality. The current microcontroller based design is pushed to its limits with little or no room to expand.



Until now the options have been limited to either continue and pack the microcontroller with more and more functionality or move to a Linux-based system. Each solution has its pros and cons.

Having an extra Cortex-M core can be a game changer. There are all the benefits of placing the management functions on the Cortex-A core(s) running an application-rich OS, like Linux - which is a modern, proven and stable operating system that has a wealth of free drivers and applications. A Cortex-A application processor has enough processing performance to drive advanced, high-resolution graphical user interfaces and to implement extensive communication solutions.

At the same time, the familiar microcontroller world is preserved. All the "traditional" time critical microcontroller tasks are placed on the Cortex-M and an RTOS controls everything on it.

So, to sum up - **if you are coming from the microcontroller world, an extra Cortex-M core can give a smooth path for the next generation of your product. You get a powerful and flexible platform for your future needs!**

Copyright 2020 © Embedded Artists AB

The table below lists boards with a heterogeneous multiprocessing system architecture, i.e., with an extra Cortex-M core.

EA(u)COM board	Cortex-M4/M7 core?	Cortex-M core frequency
iMX6 UltraLite/iMX6ULL COM	No	-
iMX6 SoloX COM	Yes, M4	Up to 227 MHz
iMX6 DualLite COM	No	-
iMX6 Dual COM	No	-
iMX6 Quad COM	No	-
iMX7 Dual COM	Yes, M4	200 MHz
iMX7 Dual uCOM	Yes, M4	200 MHz
iMX7ULP uCOM	Yes, M4	200 MHz
iMX8M COM	Yes, M4	266 MHz
iMX8M Mini uCOM	Yes, M4	400 MHz
iMX8M Nano uCOM	Yes, M7	Up to 750 MHz

3.6 Operating System (OS)

In the previous point the subject of operating system was partly discussed. The OS will be discussed in more detailed here. There are a couple of OS options for Cortex-A cores:

- **Linux**
  This is the main option that is supported and gives an excellent starting point for development. Embedded Artists provides a Linux BSP that is based on NXP's release, with EA(u)COM board adaptations.
  There are many variants of Linux, for example with real-time patches. There are also a vast number of Linux distributions. There are benefits with using a well-known distribution but there will also be more work to apply patches from the official NXP port.

- **Android**
  NXP has Android BSP releases for most i.MX processors (except for i.MX 6UltraLite). Note that these are a bare ports that can require adaptation and certification for the final system. This is not supported by Embedded Artists but there are third parties that can create ports and provide professional support.

- **Windows**TM
  Some version of Windows Embedded, typically Windows Embedded Compact. This is not supported by Embedded Artists but there are third parties that can create ports and provide professional support.

- **RTOS**
  There are many third parties that provides generic real time operating systems also for Cortex-A cores. This can be an option if Linux cannot be used for some reason. One of the big benefits with Linux is all peripheral/device drivers that exist. When using a more generic RTOS make sure the needed drivers exist, for example for Ethernet, USB and PCIe. This is not supported by Embedded Artists but there are third parties that can create ports and provide professional support.

- **No RTOS, also known as Bare Metal**
  This is not a typical solution but NXP has released an SDK for i.MX 6ULL, mainly targeting the automotive industry. This is the only i.MX application processor that has this support.

Copyright 2020 © Embedded Artists AB

When it comes to the Cortex-M core that some i.MX family members have, the supported solution from NXP is FreeRTOS - period! Note that MQX (another RTOS) was supported by NXP before but that has now changed to FreeRTOS - and that is an excellent choice!

Other solutions might be offered by third-party companies. Check availability and support from your favorite RTOS supplier. If your development department has an RTOS solution that is proven and well-known there can definitively be reasons to keep this solution. There is nothing special with the Cortex-M core and any RTOS that supports Cortex-M cores can be ported. However do not forget that inter-processor communication with the Linux side (Cortex-A side) is also needed. This functionality also has to be ported.

3.7 Temperature Range, activity cycles and Longevity

It may seem strange to treat temperature range, activity cycle and longevity at the same time, but they are closely related for the i.MX6/7/8 family. The i.MX6/7/8 family members are classified into four main groups, but here grouped into three:

- **Consumer** and **Extended commercial** - this is the versions that offers highest operating frequency/performance.
  - The temperature range is 0 - 95 degrees Celsius for consumer and -20 to 105 degrees Celsius for extended commercial
  - The lifetime of the processor is estimated to be 5 years @ 50% activity cycle
  - NXP's longevity program for these processors is 5 years
- **Industrial** - this is the version that can operate "always on"
  - The temperature range is -40 to 105 degrees Celsius
  - The lifetime of the processor is estimated to be 10 years @ 100% activity cycle
  - NXP's longevity program for these processors is 10 years
- **Automotive** - this is a version that is not commonly used for embedded systems other than automotive applications.
  - The temperature range is -40 to 125 degrees Celsius
  - The lifetime of the processor is estimated to be 10 years @ 10% activity cycle
  - NXP's longevity program for these processors is 15 years

Note that the temperature ranges above are for the junction temperature of the processor. **The need for heat sinks or heat spreaders must always be considered.**

EA(u)COM boards are sold in three temperature versions:

- Commercial: 0 to 70 degrees Celsius. For these boards, consumer or extended commercial versions of the processors and memories are used.
- Extended: -20 or -25 to 85 degrees Celsius. For these boards, extended commercial or industrial versions of the processors and memories are used.
- Industrial: -40 to 85 degrees Celsius. For these boards, industrial versions of the processors and memories are used. Sometimes the temperature range can be somewhat limited by on-board Wi-Fi/BT modules.

If your system is an "always on" system or if you need at least 10 years longevity, select the industrial version of the EA(u)COM board.

If your system needs 15 years longevity, contact us for a special mounting version built with automotive versions of the i.MX family.

Copyright 2020 © Embedded Artists AB

There are application note documents from NXP where the lifetime estimations are explained. It is very much a function of operating temperature. The cooler the processor can run, the longer lifetime it will have.

3.8 Power Consumption Requirements

Next thing to consider is if there are special requirements regarding power consumption. There are three main reasons to keep power consumption as low as possible:

- For hand held or battery operated system the reason is obvious. Smaller batteries can be used (which translates to cheaper or lighter systems) or longer time between charging.
  - If a system is only active for short periods of time low idle current is very important. The i.MX7ULP has extremely low idle current and should be considered for this type of application.
- Thermal management becomes much simpler. Large heat sinks and complicated heat spreaders are no longer needed.
- Processors that run at lower temperature have longer lifetime.

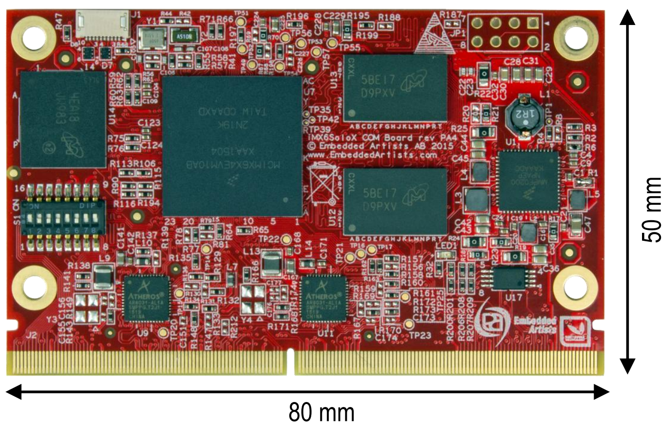
As a general guideline, do not oversize the system if power consumption is important. Also, lean towards the i.MX7 Dual/ULP or i.MX8M Mini/Nano families since these families targets low power applications.

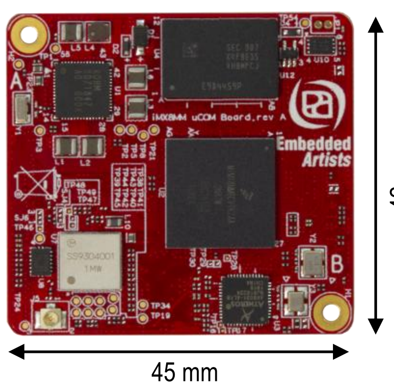
Expect power consumption to be in the range of 0.5-5W in highest power consumption mode. The mean power consumption can be considerably lower than this. It all depends on the application. Note that these numbers are for the complete EA(u)COM board, not just the processor.

Another general guideline is to measure the actual power consumption on the prototypes. Always add a healthy margin to what is measured when dimensioning the power supply and possible thermal management solutions (heat sink, heat spreader, etc).

3.9 Determine physical form factor

The final thing to consider is if there are special requirements on mechanical dimensions. EACOM boards are 80x50mm in size and have an MXM3 edge connector. EAuCOM boards are 42x45mm in size and have dual 100 pos and dual 40 pos DF40C connectors on the bottom side. The two pictures below illustrate the relative size of the two board families.





Copyright 2020 © Embedded Artists AB

4 i.MX Family and EA(u)COM Benefits

This document starts out assuming that EA(u)COM boards and the i.MX family are given for the project. If that is not the case, this chapter presents a lot of reasons to select the i.MX6/7/8 family and to use EA(u)COM boards.

4.1 Why the i.MX6/7/8 Family?

The i.MX6/7/8 application processors are **scalable**, multi-core platforms with single-, dual- and quad-core families. The key-word here is *scalable* that provides cost-effective scalability. Some processors also offer a heterogeneous processing architecture with an extra Cortex-M core, providing very flexible single-chip solution that can run sophisticated, feature rich operating systems (like Linux) and at the same time provide real time responsiveness.

The i.MX6/7/8 families reaches excellent levels of power versus performance. The processors are rich of peripherals, including support the continuously connected world; Ethernet, PCIe (wireless modules), SD/SDIO/MMC (wireless modules), USB, UARTs, I2S, CAN and SATA.

Display-centric devices requires an increasingly advanced and intuitive user interface to deliver the richest customer experience. Many members of the i.MX6/7/8 families integrates graphics accelerators to support both 2D and 3D graphics to create, and deliver, high-resolution, stunning graphical user interfaces.

The i.MX6/7/8 applications processors provides solutions across multiple market segments. The long product longevity is ideal for industrial applications. The table below list a few examples of applications suitable for the i.MX6/7/8 family.

Industrial	Smart (Mobile) Devices
• Smart meters • Smart energy • Building automation • Test and measurement equipment • Intelligent industrial control systems • Point-of-sale (POS) • (Mobile) Industrial HMI • IoT solutions	• Home automation / Smart home control • Residential gateway • Connected home audio • Digital signage • Smart monitors • Media hubs • Appliances • Wearables • Healthcare / Patient monitoring

All-in-all, **the i.MX6/7/8 family is an excellent choice to base your next project on. You will get a proven, feature-rich, powerful and scalable platform that has a lot of support!**

Copyright 2020 © Embedded Artists AB

4.2 Why EA(u)COM Boards?

The rationale behind Computer-on-Modules is to solve the development problem that *more features shall be implemented in less time, with less resources*. Embedded Systems development is also a multi-talented area. It is difficult to have a development department that have experts in all areas needed. Companies wants to focus on their core business and not on the infrastructure behind it. In general it is difficult to manage available engineering resources. On top of this, it is not always easy to get support when needed.

EA(u)COM stands for *Embedded Artists' (u)Computer-on-Modules*. These modules offers a solution to a number of common development challenges:

- You get a stable and proven platform to start development from - it lowers the development risk!
- You can focus on your core business and manage the available engineering resources!
- You get one off-the-shelf component with long term availability that is easy to integrate!
- You get shorter time to market!
- You get a scalable family of boards and a roadmap!
- You get lower total cost!
- You get a development partner in Embedded Artists and support directly from engineers!

The **EA(u)COM** board specification is available on Embedded Artists' website.

4.3 EA(u)COM Board Architecture

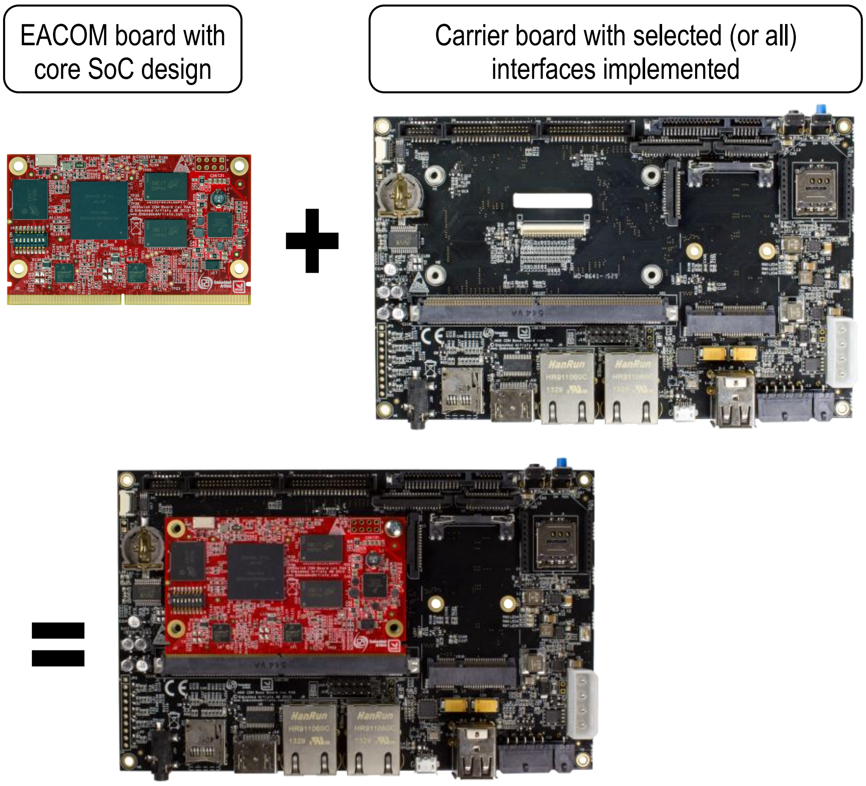
This section gives a general overview of the board architecture, interfaces and pinning.

An EA(u)COM based system solution has the following overall physical structure:

- **EA(u)COM board**, containing the core design that encapsulate a lot of the complexity of a modern, high-performance ARM SoC design.
- **Carrier board** that implements the needed interfaces in the specific solution. The carrier board also typically contains the powering solution and creates the mechanical entity that shall be mounted in box, or similar. The carrier board is typically a simpler design (i.e., less complex) than the EA(u)COM board. Either the carrier board is a fully custom specific design or more or less a copy of the *COM Carrier board* (used in the iMX Developer's Kits). In either case the *COM Carrier board* design can be used as a reference implementation for the different interfaces.
    - Pin assignment on the expansion connectors has been defined to simplify routing on the carrier board as much as possible. In many cases, a standard low-cost 4-layer PCB will be sufficient for the carrier board. The *COM Carrier board*, used in the iMX Developer's Kits is a 4 layer PCB design.

Copyright 2020 © Embedded Artists AB

The combination of an EA(u)COM board and accompanying Carrier board is very much like a Single Board Computer (SBC), but more flexible. The carrier board can be a much better fit for each specific application than a standard SBC. Normal design updates are more likely to be on the carrier board, which is simpler to update than a complete SBC would be. Upgrading a design for more execution performance or more memory is as easy as changing EA(u)COM board, as opposed to redesigning an SBC.



The block diagram below illustrates the typical components of an EACOM board (the same principles apply for EAuCOM boards):

- **SoC** - the main component, a member of the iMX6/7/8 family.
- **SDRAM** - a large memory array with 256 MByte - 4 GByte capacity. Typically DDR3L, DDR4, LPDDR4 memories to get low power consumption, yet high density.
- **Parallel Flash** - for storing Operating System and boot loader images. Typically an eMMC memory but can also be an unmanaged FLASH memory.
- **Serial Flash** - for storing code for (possible) Cortex-M core.
- **Power Management** - typically in the form of a PMIC that supports low-power operation including DVFS (Dynamic Voltage and Frequency Scaling).
- **Debug interface** - for JTAG debugging.
- **Boot control** - for controlling the boot source.
- **Parameter storage** - for retrieving important parameters during boot, like memory bus calibration parameters and MAC address(es).
- **Edge connector** - edge pads conforming to the MXM3-standard with 314 positions.

Copyright 2020 © Embedded Artists AB

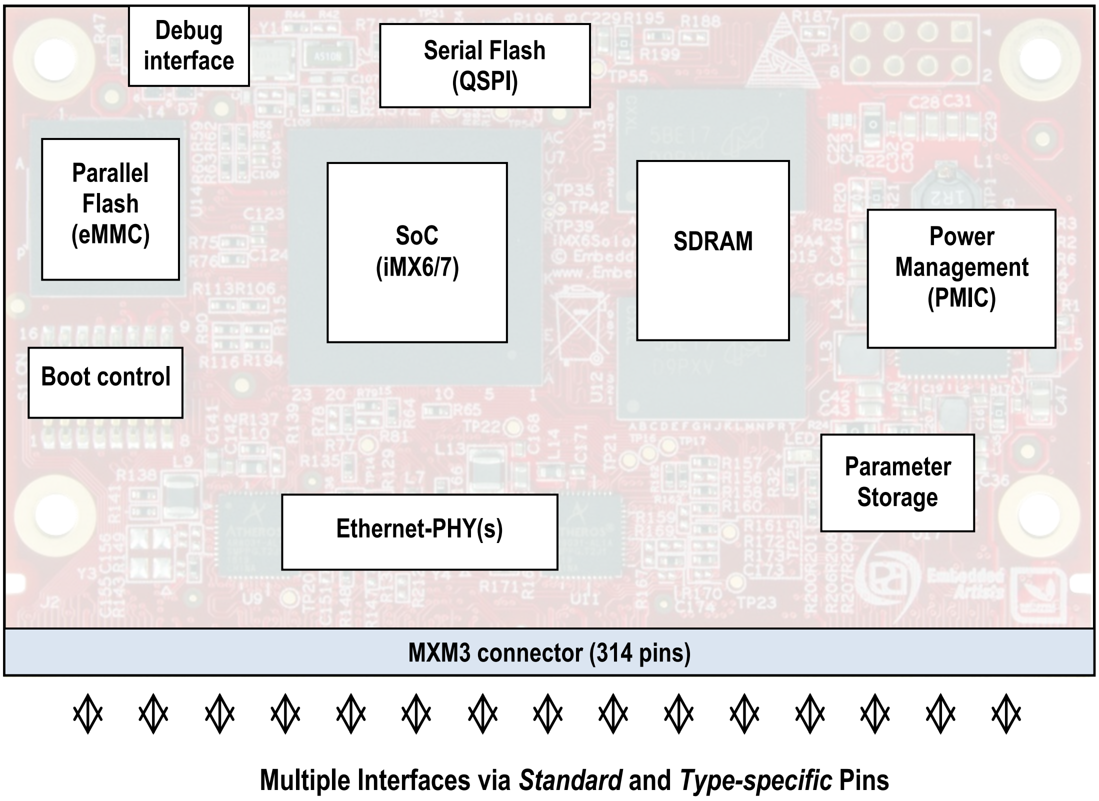


Figure 1 – EACOM Board Block Diagram

There are two types of interfaces to the EA(u)COM boards:

- ***Standard Interfaces***
  The EA(u)COM specification has defined a number of different interfaces and allocated positions for these interfaces on the expansion connector(s). These interfaces are reserved for their respective interface and will be the same on every EA(u)COM board. Note that every EA(u)COM board will assign signals to every interface whenever possible, but not necessarily all of them. Some interfaces may for example not be present on some SoC, like PCIe, SATA and a second Ethernet interface. Some SoC's may not have enough pins to assign all interfaces.
  It is important to note that **to guarantee electrical compatibility between (carrier board) designs, only make use of the standard interfaces.**
- ***Type-specific Interfaces***
  A number of positions on the expansion connector(s) have been left unassigned. Different EA(u)COM boards can have different signals and interfaces assigned to these positions. Note that using these pins on a carrier board design may result in lost compatibility between EA(u)COM boards, but not always. Details have to be checked in every specific case.

It can be limiting to only make use of the standard interfaces in the EA(u)COM specification. If compatibility between EA(u)COM boards is not a requirement then it is free to use every pin to whatever function the pin multiplexing allows.

Copyright 2020 © Embedded Artists AB

5 Disclaimers

Embedded Artists reserves the right to make changes to information published in this document, including, without limitation, specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Customer is responsible for the design and operation of their applications and products using Embedded Artists' products, and Embedded Artists accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the Embedded Artists' product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products. Customer is required to have expertise in electrical engineering and computer engineering for the installation and use of Embedded Artists' products.

Embedded Artists does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using Embedded Artists' products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). Embedded Artists does not accept any liability in this respect.

Embedded Artists does not accept any liability for errata on individual components. Customer is responsible to make sure all errata published by the manufacturer of each component are taken note of. The manufacturer's advice should be followed.

Embedded Artists does not accept any liability and no warranty is given for any unexpected software behavior due to deficient components.

Customer is required to take note of manufacturer's specification of used components, for example PMIC and eMMC. Such specifications, if applicable, contains additional information that must be taken note of for the safe and reliable operation. These documents are stored on Embedded Artists' product support page.

All Embedded Artists' products are sold pursuant to Embedded Artists' terms and conditions of sale: https://www.embeddedartists.com/terms-and-conditions/

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by Embedded Artists for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN EMBEDDED ARTISTS' TERMS AND CONDITIONS OF SALE EMBEDDED ARTISTS DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF EMBEDDED ARTISTS PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS EXPRESSLY APPROVED IN WRITING BY THE CEO OF EMBEDDED ARTISTS, PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, NUCLEAR, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE.

Resale of Embedded Artists' products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by Embedded Artists

Copyright 2020 © Embedded Artists AB

for the Embedded Artists' product or service described herein and shall not create or extend in any manner whatsoever, any liability of Embedded Artists.

This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from national authorities.

5.1 Definition of Document Status

Preliminary – The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. Embedded Artists does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information. The document is in this state until the product has passed Embedded Artists product qualification tests.

Approved – The information and data provided define the specification of the product as agreed between Embedded Artists and its customer, unless Embedded Artists and customer have explicitly agreed otherwise in writing.

Copyright 2020 © Embedded Artists AB