Introduction To EVMK2H

[4 x ARM + 8 x DSP]

•••

May 10, 2017

Muhammad Falak R Wani

* All the images are taken from Texas Instruments © website

What does the board look like ?

Components

Processing

- C66X DSP x 8
- ARM 15 x 4
- Co-Processors
- Queue MSS

Memory + I/O

- 2 GB ECC
- 2 GB DDR3
- 512 MB NAND
- 128 MB NOR
- 1 MB I^2C
- USB/ETH ...

Development

- CCS
- MCSDK
- SYS/BIOS
- Linux 3.10

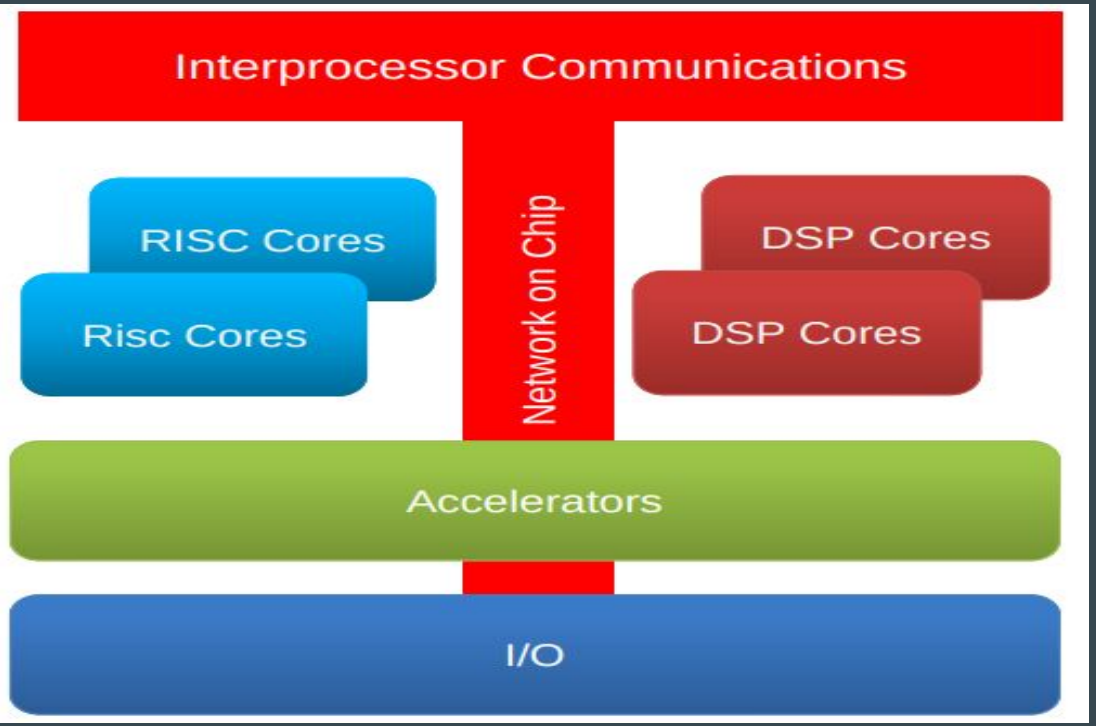
Zooming In:

The Components that make sense to us.

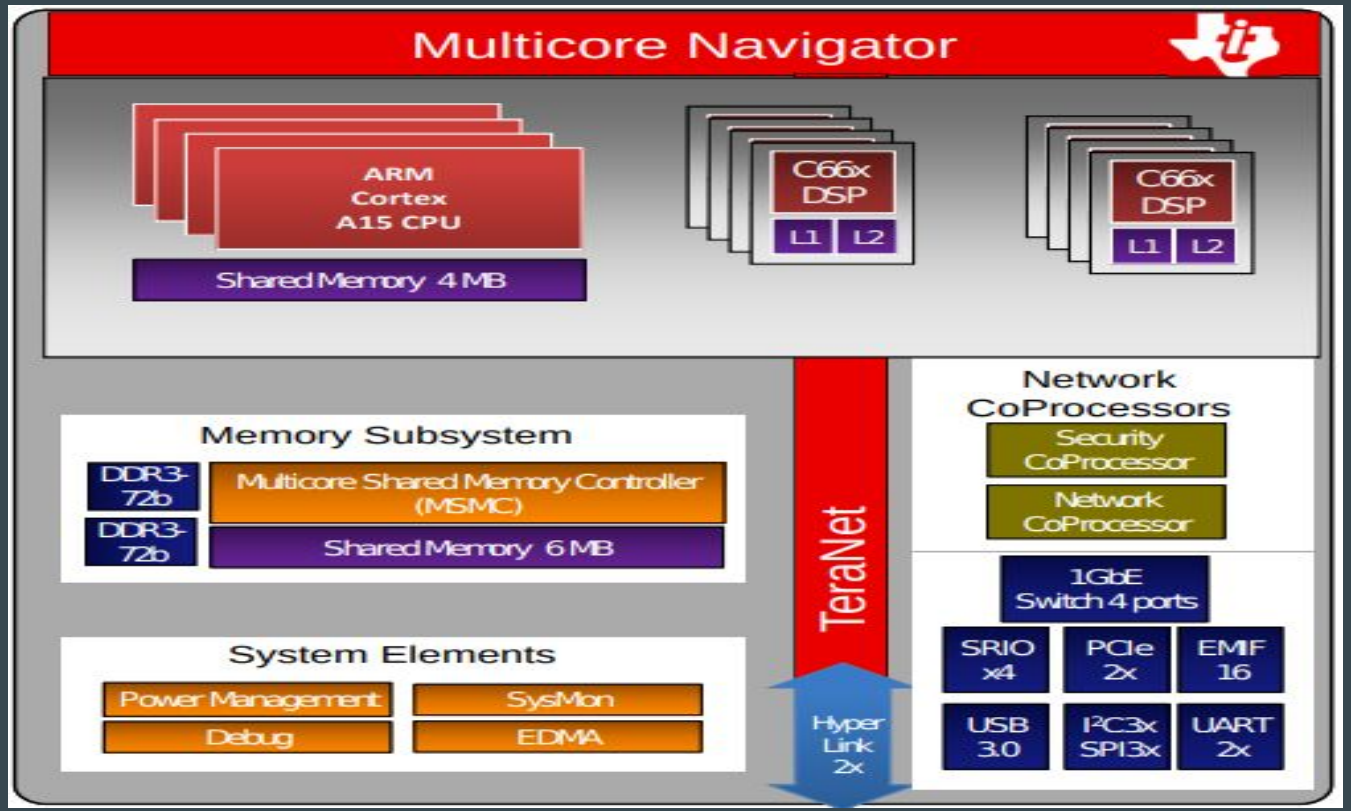
KeyStone II Architecture

High performance structure for integrating RISC and DSP

- MultiCore Navigator
- TeraNet
- Multicore SMC
- HyperLink



Functional Diagram ARM+DSP



Performance

The 66AK2H platform combines the quad ARM® Cortex™-A15 with eight TMS320C66x

Theoretical Performance:

- 5.6 GHz of ARM
- 9.6 GHz of DSP
- The C66X core :
  - 38.4 GMACS/core
  - 19.2 GFLOPS/core

Compute units

No Mem. Coherency between ARM & DSP

C66x DSP CorePac

Integrates the 8 DSP (@1.2 GHz each) cluster

Features:

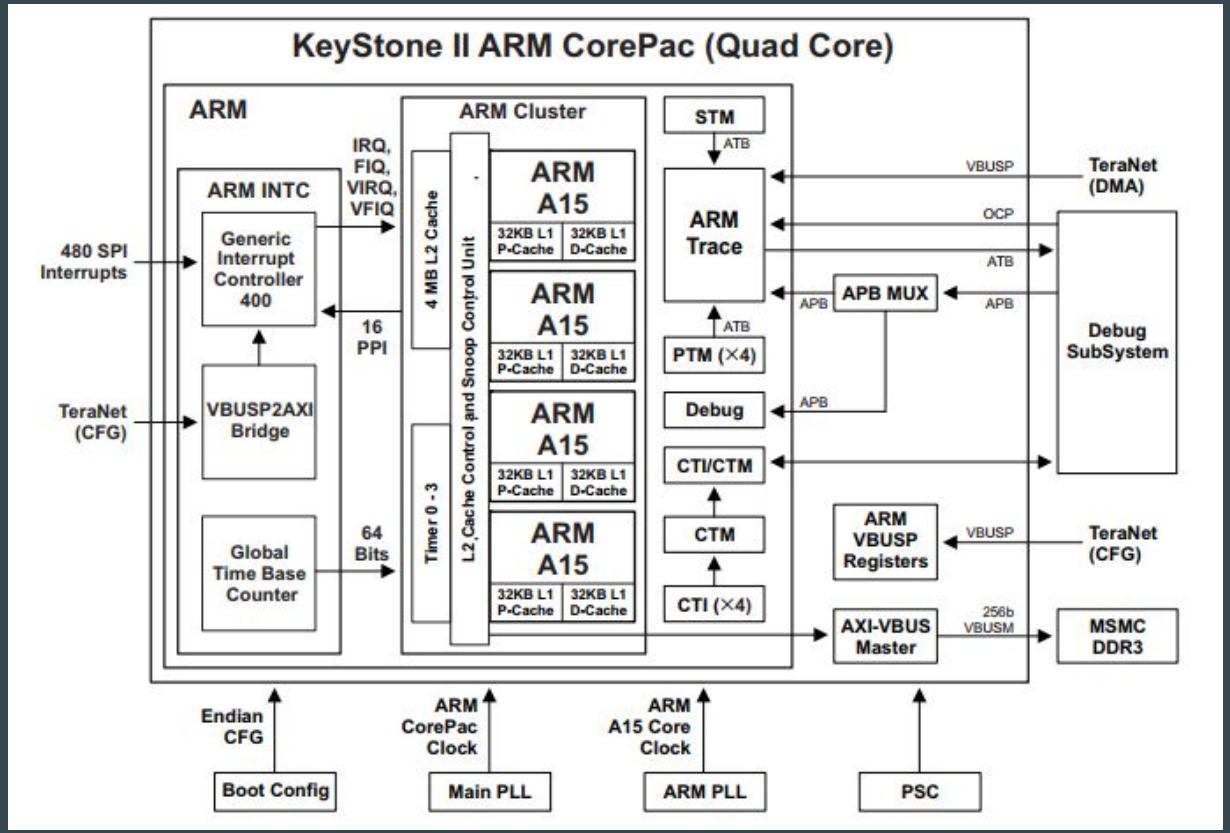
- 2-way SIMD operations for 16-bit data
- 4-way SIMD operations for 8-bit data
- 128-bit vector Instructions
- Improved vector processing capability
- 32KB of L1D/L1P
- L2 cache of 1MB
- 6Mb of scratch pad via MSMC

ARM CorePac

Integrates the 4 ARM-15 (@1.4 GHz) cluster

Features:

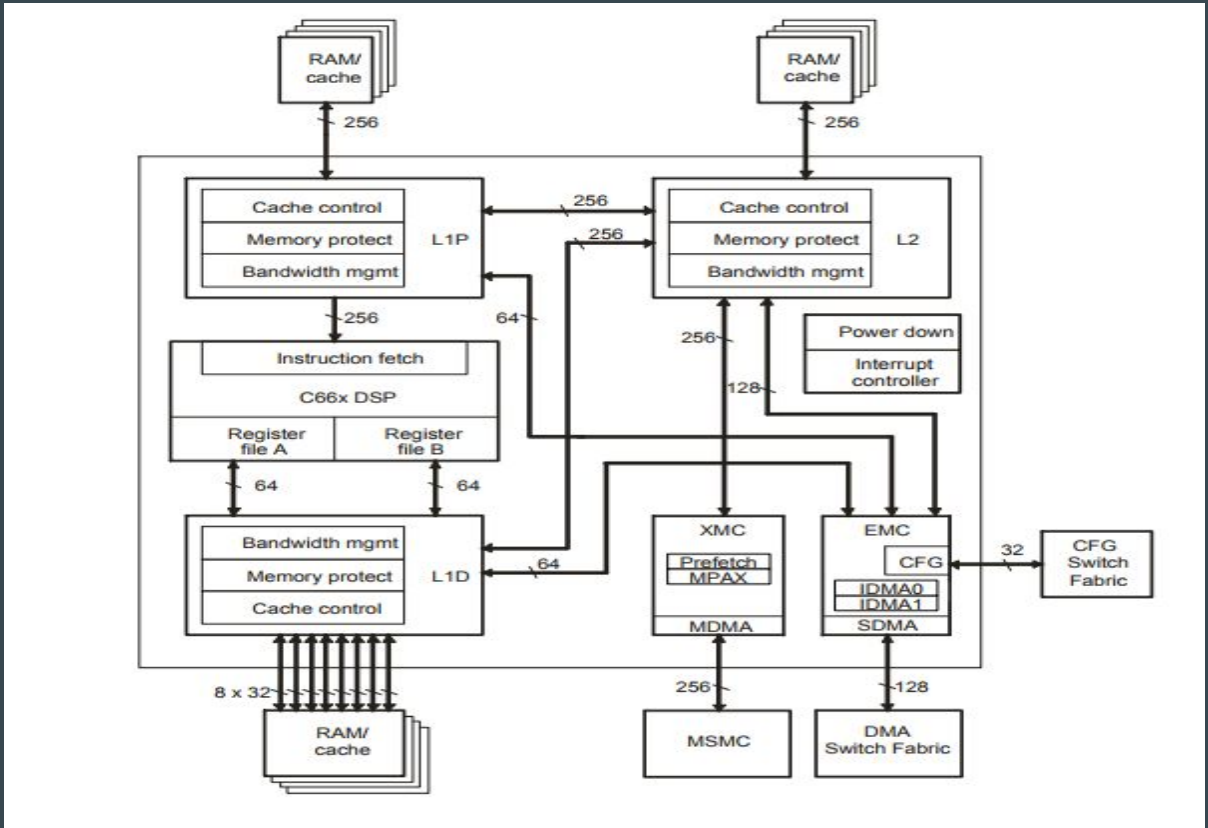
- ARMv7A-compatible, Multi-issue, out-of-order Processor
- Superscalar pipeline
- SIMDV2 (Neon technology)
- VFPv4 (Vector Floating Point)
- 32KB of L1D/L1P
- Includes a 4MB L2 (shared)
- L2 cache controller with snoop control
- ARM cores are fully cache coherent



ARM CorePac

Components

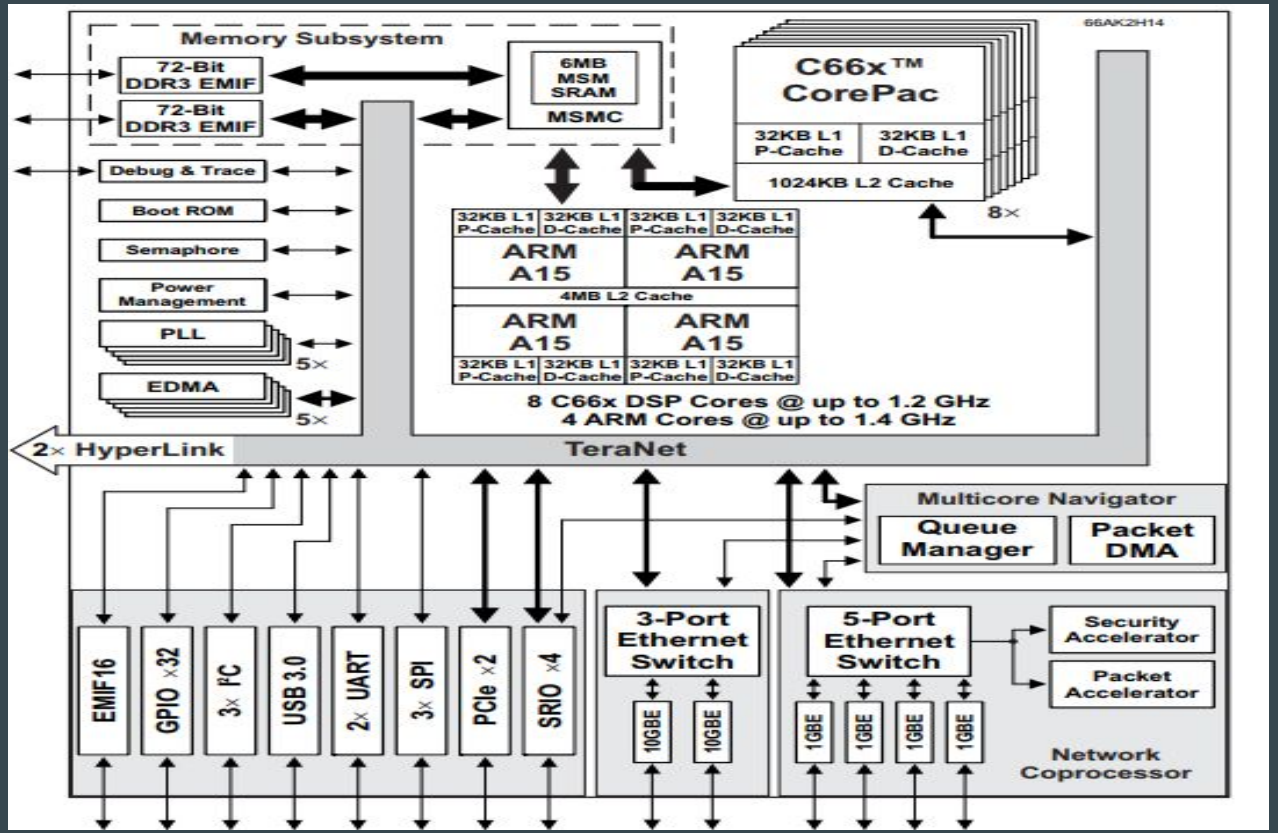
- ARM A15 Cluster
- AXI2VBUS Master
- ARM Interrupt Controller
- Local Power and Sleep Controller (LPSC)



C66x CorePac

Components

- C66x DSP
- Level 1 P/D (L1 P/D) Memory Controller
- Level 2 (L2) Memory Controller
- Internal DMA (IDMA)
- External Memory Controller (EMC)
- Extended Memory Controller (XMC)
- Power-Down Controller (PDC)



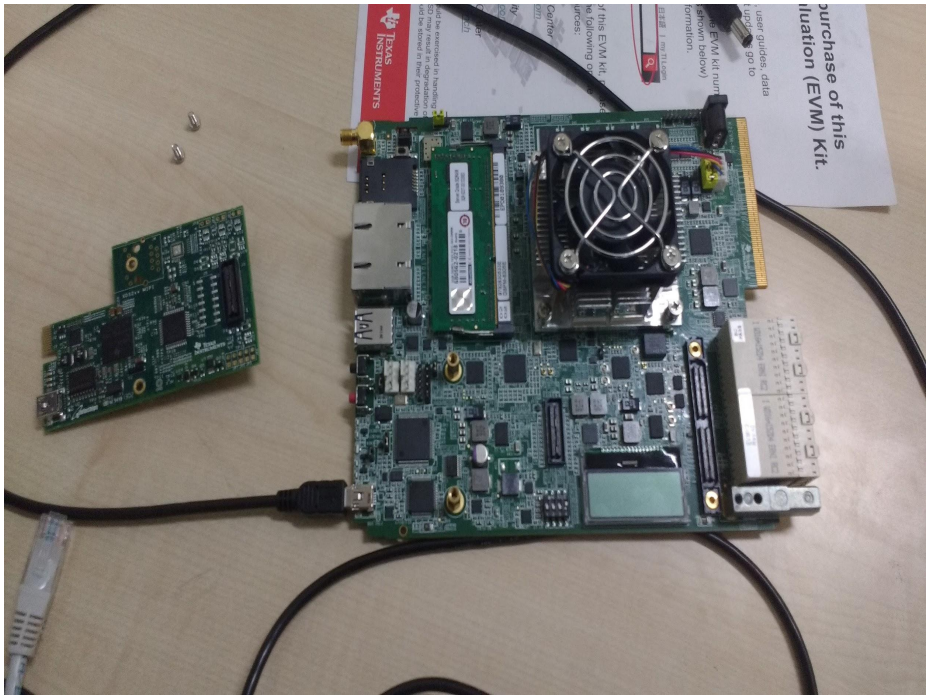
Complete Architecture (Scary slide)

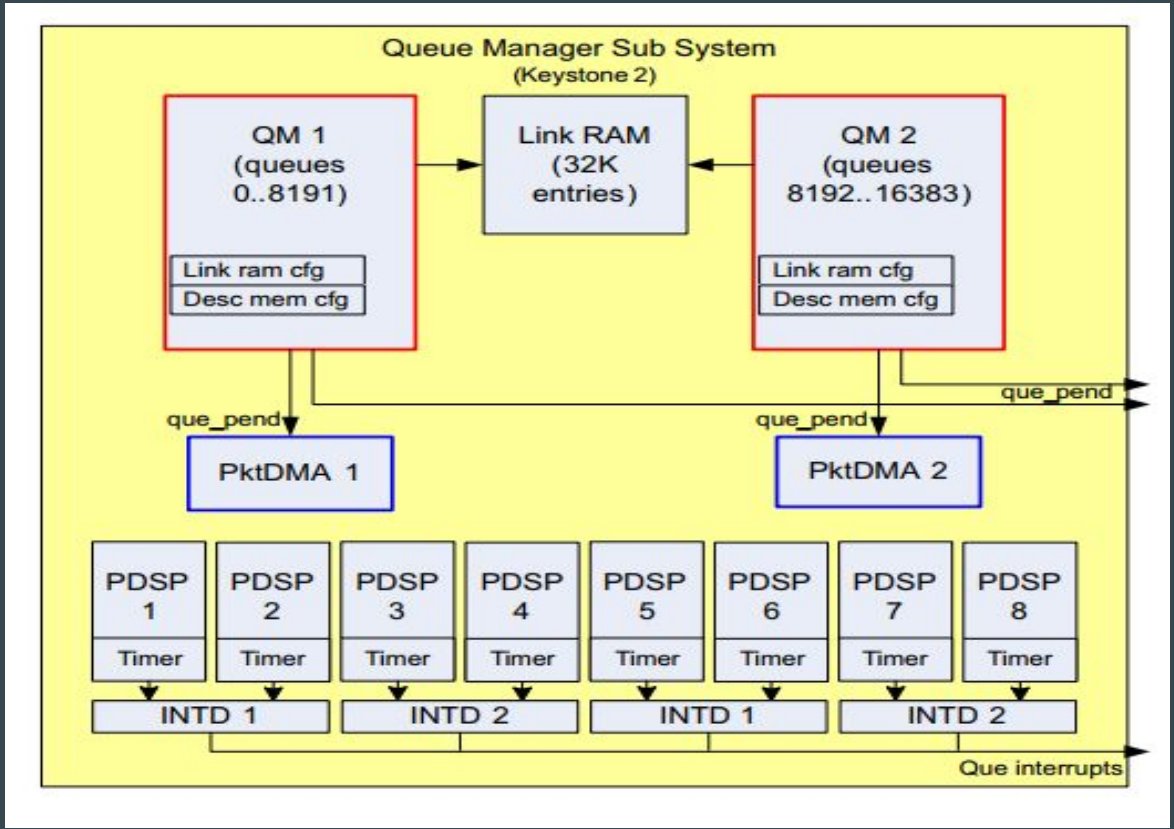
Important Stuff

- Multicore Navigator
  - Hardware queues
  - Packet DMA
- Semaphore x 64 (h/w)
- DMA Engine
- Network On Chip (2Tb/s bandwidth)

Demo Time

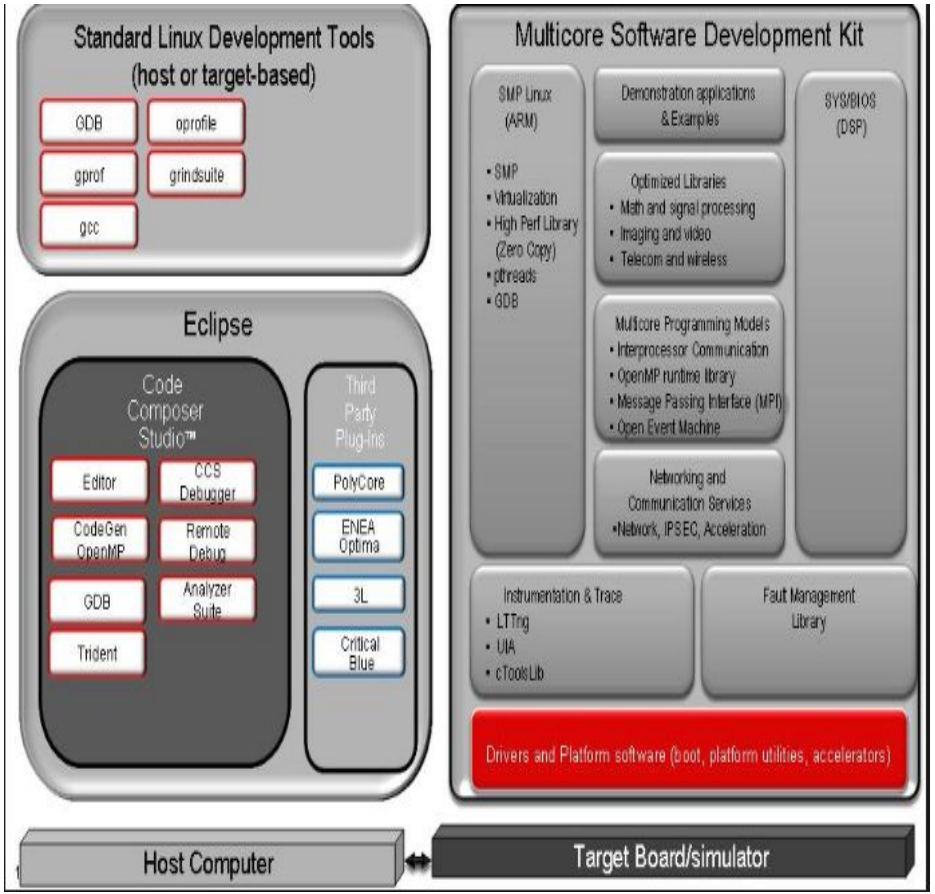
How did we bring-up the machine ?
How Much time did it take ?
Where is the HDD ?
Do we understand everything ?

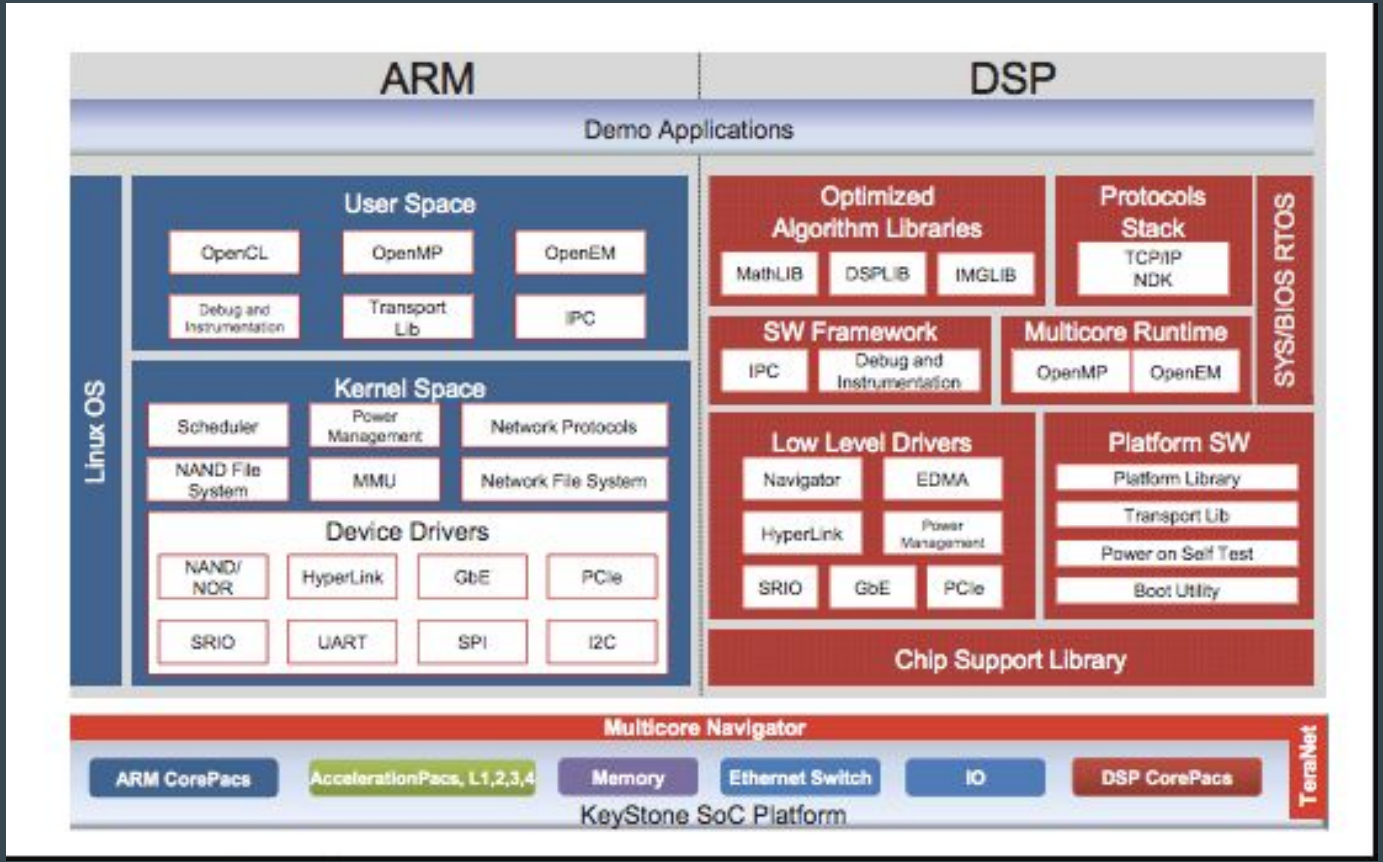




Queue Management Subsystem

How do we develop ?





ARM+DSP development support by TI

Components

- CCS IDE
- cl6x (opt. compiler)
- Assembler
- gcc/g++ (arm)
- Installed a go compiler
- ....

CCS and Compilers + Toolchain from TI

PS: I don't know much about CCS

Questions ?

Feel free to drop a mail:
falakreyaz@gmail.com