Keystone Bootloader

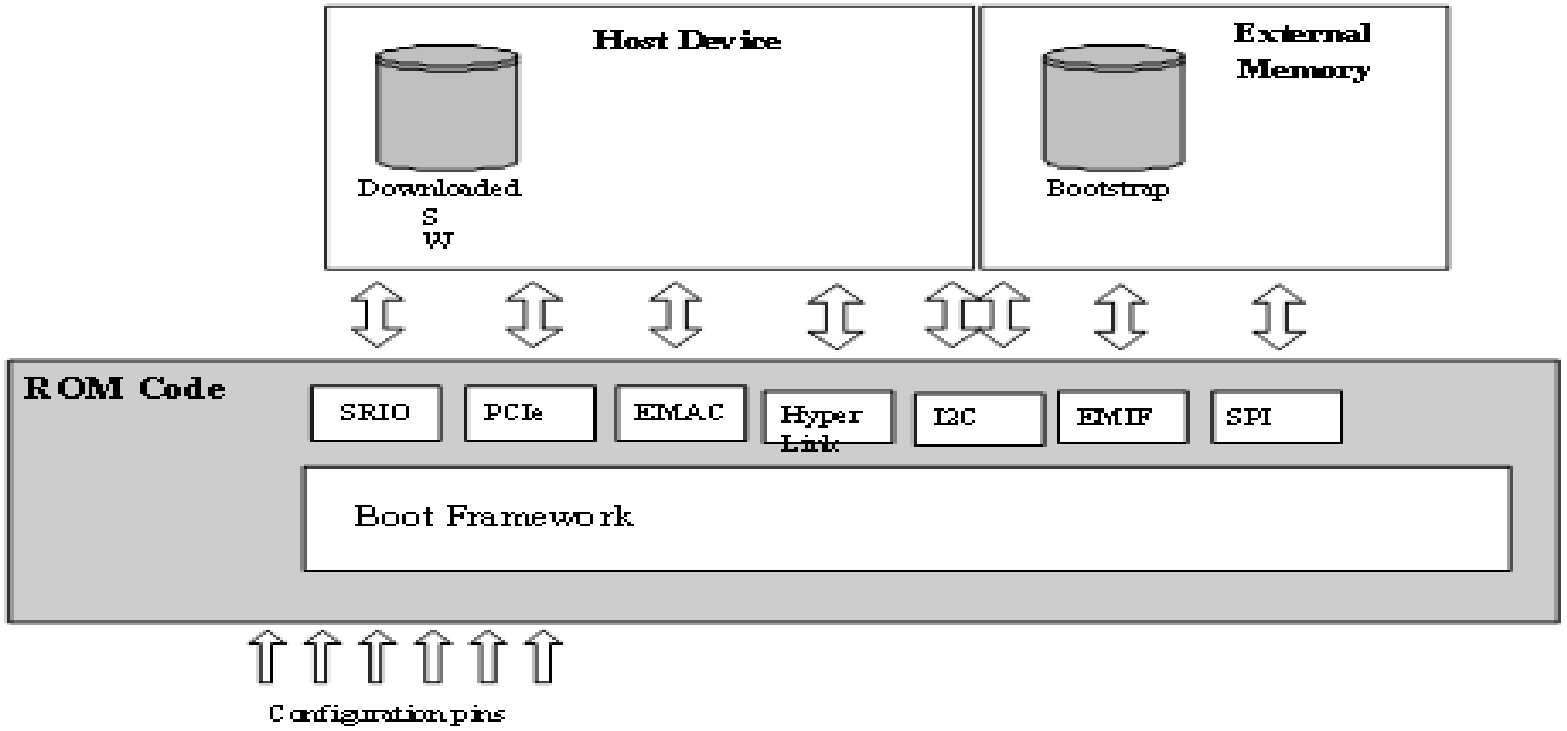
Keystone ROM Boot Loader (RBL)

- RBL is a code used for the device startup.
- RBL also transfers application code from memory or host to high speed internal memory or DDR3
- RBL code is burned in the DSP ROM (Non-modifiable)
- Base address for the RBL is 0x20B00000
- Seven different types of boot modes are supported
- These boot modes are broadly divided into two groups
  - Memory boot where the application code is stored in a slow external memory and DSP acts as a master and drives the boot process.
  - Host Boot where the DSP is configured as a slave and driven by a host device connected through fast transport.

ROM Boot Modes

Supported Boot Modes

- I2C Boot
  - Master Boot (from I2C EEPROM)
  - Master-Broadcast Boot(Master Boot followed by broadcast to slave cores)
  - Passive Boot (external I2C host)
- SPI Boot (from SPI flash)
- SRIO Boot(from external host connected through SRIO)
- Ethernet Boot (boot from external host connected through Ethernet)
- PCIe Boot (boot from external host connected through PCIe)
- HyperLink Boot (boot from external host connected through HyperLink)
- EMIF16 NOR Boot(boot from NOR Flash)
  - Device Manual will detail supported types.
  - Some members have NAND boot as well



Boot Process Trigger

- The Boot Process is triggered when
  - The reset is asserted
- The Boot Process flow is determined by
  - The Boot Mode that is set by boot strap pins

Reset Types

- Power on Reset (POR) (Cold Reboot)
  - Resets everything
  - Latches the boot strap pins
  - RBL Process initiated
- RESETFULL (Warm Reboot)
  - Resets everything
  - Latches the boot strap pins
  - RBL Process initiated
- RESET (Can be configured as hard or soft)
  - Resets everything except EMU and reset isolated peripherals.
  - No latching of the boot strap pins.
  - For software reset PCIe, EMIF16, DDR3 and EMIF MMRs are also preserved.
  - RBL process is initiated.
- LRESET
  - Mostly used by watch dog timer
  - Just the corePac is reset all the memory are preserved.
  - No RBL process is initiated.

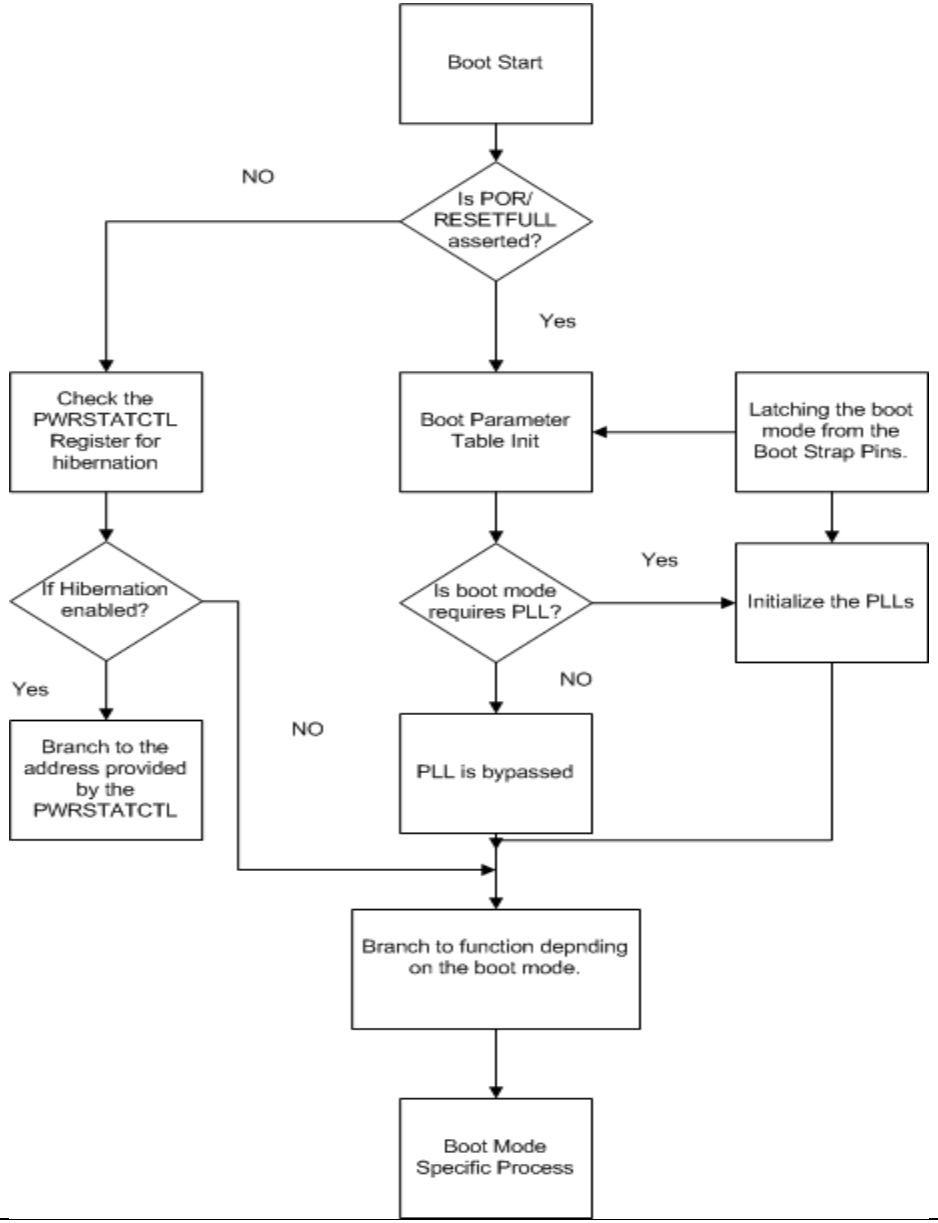
Boot Mode Configuration Pins

- Boot mode and configurations are chosen using bootstrap pins on the device.
  - Pins are latched and stored in13 bits of the DEVSTAT register during POR.
- The configuration format for these 13 bits are shown in the table:

Boot Mode Pins												
12	11	10	9	8	7	6	5	4	3	2	1	0
PLL Mult I2C/SPI Ext Dev Cfg			Device Configuration							Boot Device		

- Boot Device [2:0] is dedicated for selecting the boot mode
- Device Configuration [9:3] is used to specify the boot mode specific configurations.
- PLL Multi [12:10] are used for PLL selection. In case of I2C/SPI boot mode, it is used for extended device configuration. (PLL is bypassed for these two boot modes)

RBL Flow


Boot Start
Is POR/ RESETFULL asserted?
NO
Yes
Check the PWRSTATCTL Register for hibernation
Boot Parameter Table Init
Latching the boot mode from the Boot Strap Pins.
If Hibernation enabled?
Is boot mode requires PLL?
Yes
Initialize the PLLs
Yes
NO
NO
Branch to the address provided by the PWRSTATCTL
PLL is bypassed
Branch to function depnding on the boot mode.
Boot Mode Specific Process

Boot Device

- Boot Device Selection Values

Boot Mode Pins: Boot Device Values	
Value	Boot Device
0	Sleep(6670) / EMIF16[1]
1	Serial Rapid I/O
2	Ethernet (SGMII) (PA driven from core clk)
3	Ethernet (SGMII) (PA driver from PA clk)
4	PCIe
5	I2C
6	SPI
7	HyperLink

1. See the device-specific data manual for information.

- For interfaces supporting more than one mode of operation, the configuration bits are used to establish the necessary settings

PLL Configuration

The boot code sets the PLL multiplier based on the core frequency set in the EFUSE register.

PLL Clock Configuration for KeyStone Devices									
Boot PLL Select [2:0]	Input Clock Freq (MHz)	core = 800 MHz		core = 1000 MHz		core = 1200 MHz		core = 1400 MHz	
		Clkr	Clkf	Clkr	Clkf	Clkr	Clkf	Clkr	Clkf
0	50.00	0	31	0	39	0	47	0	55
1	66.67	0	23	0	29	0	35	0	41
2	80.00	0	19	0	24	0	29	0	34
3	100.00	0	15	0	19	0	23	0	27
4	156.25	24	255	4	63	24	383	24	447
5	250.00	4	31	0	7	4	47	4	55
6	312.50	24	127	4	31	24	191	24	223
7	122.88	47	624	28	471	31	624	13	318

PLL Clock O/P = (Input Clock x (Clkf + 1))/(2 * (Clkr + 1))

Boot Configuration Format

- Boot Parameter Table
  - Map for the Boot Process
  - The boot process copies a default boot parameter table into a reserved L2 of Core0.
  - The first 10 byte offsets of the table are common across all the boot modes.
    - Length
    - Checksum
    - Boot Mode
    - Port Num
    - PLL configuration (most significant bits)
    - PLL configuration (least significant bits)
  - The rest of the table is boot mode dependent.

Boot Image Format

- Boot Table
  - Block of data that contains the code and data section.
  - The block is loaded from the host or an external memory to the internal memory or DDR by RBL.
  - The first 8 bytes of the boot table forms the header
    - 32 bit section bytes count
    - 32 bit section address where the block has to be moved.
  - The end of table is identified by writing 0's.

Register Configuration Format

- Boot Config Table
  - Provides read/modify/write capabilities to any memory on the DSP.
  - Each entry has three 32 bit wide elements.
  - First element is address to be modified
  - Second element is the set mask
  - Third element is the clear mask.
  - If all three elements are 0's the end of boot config table is reached.

Boot Configuration

I2C Master Mode

- In master mode the I2C Device Configuration uses 7 bits of device configuration instead of 5 bits used in passive mode.
- In this mode device will make the initial read of the I2C EEPROM while the PLL is in bypass.
- The initial boot parameter table will contain the desired clock multiplier which will be setup prior to any subsequent reads.

I2C Master Mode Device Configuration Bit Fields									
12	11	10	9	8	7	6	5	4	3
Rsvd	Speed	Address	Rsvd	Mode (0)	Parameter Index				

I2C Master Mode Device Configuration Field Descriptions		
Bit Field	Value	Description
Mode	0	Master Mode
	1	Passive Mode (bit field 9 is set to 1 and is used for this mode due to a bug in RBL)
Address	0	Boot From I2C EEPROM at I2C bus address 0x50
	1	Boot From I2C EEPROM at I2C bus address 0x51
Speed	0	I2C data rate set to approximately 20 kHz
	1	I2C fast mode. Data rate set to approximately 400 kHz (will not exceed)
Parameter Index	0-31	Identifies the index of the configuration table initially read from the I2C EEPROM

I2C Boot Parameter Table

Offset	Field
12	Option
14	Boot Addr in EEPROM
16	I2C Dev Addr of EEPROM
18	Broadcast Addr
20	Local Address
22	Device Frequency
24	Bus Frequency
26	Next Boot Addr in EEPROM after boot config
28	Next I2C Dev Addr of EEPROM after boot config
30	Address Delay

Utilities to go

- For getting the boot table
  - **Hex6x** (available in CGT)
  - Need a rmd file to provide details to hex6x
  - The output file is in the boot table format
  - If the EVM is set in little endian convert the boot table to big endian mode (that is used by the RBL) using **bconvert64x** utilities (available in MCSDK)
- Convert to a I2C format (to be loaded into the EEPROM) **b2i2c** (available in MCSDK)
- Append the boot parameter table to the boot table
  - **romparse** (Available in MCSDK)
  - romparse uses a map file to get the boot parameter tables.

Boot Configuration
I2C Passive Mode

- In passive mode the I2C Device Configuration uses 5 bits of device configuration instead of 7 used in master mode.
- In passive mode the device does not drive the clock, but simply acks data received on the specified address.
- The I2C address is calculated by adding 0x19 to the I2C address specified in the device configuration.
- Header format: (0x19 + I2C address) xx xx yy yy zz zz
  - xx xx = length, yy yy = checksum, zz zz = boot option

I2C Passive Mode Device Configuration Bit Fields						
9	8	7	6	5	4	3
Rsvd (Must be 1)	Mode (1)	Receive I2C Address			Reserved	

I2C Passive Mode Device Configuration Field Descriptions		
Bit Field	Value	Description
Mode	0	Master Mode
	1	Passive Mode
Address	0-7	The I2C Bus address the device will listen to for data

Boot Configuration – SPI Mode

Similar to I2C, the bootloader reads either a boot parameter table or boot table. The table loaded can contain a boot parameter table for any other boot mode.

SPI Device Configuration Bit Fields									
12	11	10	9	8	7	6	5	4	3
Mode (clk Pol/Phase)		4,5pin	Addr Width	Chip select		Parameter Table			

SPI Device Configuration Field Descriptions		
Bit Field	Value	Description
Mode	0	Data is output on the rising edge of SPICLK. Input data is latched on the falling edge.
	1	Data is output one half-cycle before the first rising edge of SPICLK and on subsequent falling edges. Input data is latched on the rising edge of SPICLK.
	2	Data is output on the falling edge of SPICLK. Input data is latched on the rising edge.
	3	Data is output one half-cycle before the first falling edge of SPICLK and on subsequent rising edges. Input data is latched on the falling edge of SPICLK.
4,5 pin	0	4 pin mode used
	1	5 pin mode used
Addr Width	0	16 bit address values are used
	1	24 bit address values are used
Chip Select	0-3	The chip select field value
Parameter Table Index	0-3	Specifies which parameter table is loaded

Boot Configuration – EMIF16 Mode

- EMIF16 mode is used to boot from the NOR flash.
- The boot loader configures the EMIF16 and then sets the boot complete bit corresponding to corePac0 in the boot complete register and then branches to EMIF16 CS2 data memory at 0x70000000.
- No Memory is reserved by the boot loader.

Sleep / EMIF16 Configuration Bit Fields						
9	8	7	6	5	4	3
Reserved		Wait Enable	Sub-Mode			

Sleep / EMIF16 Configuration Bit Field Description		
Bit Field	Value	Description
Sub-Mode	0b00	Sleep Boot
	0b01	EMIF16 boot
	0b10-0b11	Reserved
Wait Enable	0b0	Wait enable disabled (EMIF16 sub mode)
	0b1	Wait enable enabled (EMIF16 sub mode)

Boot Configuration – Ethernet

- Ethernet(SGMII) boot configuration sets SERDES clock and device ID.

Ethernet (SGMII) Device Configuration Bit fields						
9	8	7	6	5	4	3
SERDES Clock Mult		Ext connection		Dev ID		

Bit field	Value	Description
Ext connection	0	Mac to Mac connection, master with auto negotiation
	1	Mac to Mac connection, slave, and Mac to Phy
	2	Mac to Mac, forced link
	3	Mac to fiber connection
Device ID	0-7	This value is used in the device ID field of the Ethernet ready frame. Bits 1:0 are use for the SR ID.
SERDES Clock Mult The output frequency of the PLL must be 1.25 GBs.	0	x8 for input clock of 156.25 MHz
	1	x5 for input clock of 250 MHz
	2	x4 for input clock of 312.5 MHz
	3	Reserved

Boot Configuration – Serial RapidIO

- SRIO boot configuration sets the Clock, Lane configuration, and mode

Rapid I/O Device Configuration Bit Fields						
9	8	7	6	5	4	3
Lane Setup	Data Rate		Ref Clock			

SRIO Configuration Bit Field Descriptions		
Bit Field	Value	Description
Ref Clock	0	Reference Clock = 156.25 MHz
	1	Reference Clock = 250 MHz
	2	Reference Clock = 312.5 MHz
Data Rate	0	Data Rate = 1.25 GBs
	1	Data Rate = 2.5 GBs
	2	Data Rate = 3.125 GBs
	3	Data Rate = 5.0 GBs
Lane Setup	0	Port Configured as 4 ports each 1 lane wide (4 -1x ports)
	1	Port Configured as 2 ports 2 lanes wide (2 – 2x ports)

Boot Configuration – PCI Express

- In PCIe mode, most PCIE configuration registers should be setup by host remotely.
- And then the host loads all the sections directly to the memory.

PCI Device Configuration Bit Fields						
9	8	7	6	5	4	3
Rsvd	BAR Config				Reserved	

PCI Device Configuration Bit Fields		
Bit Field	Value	Description
SR ID	0-3	Smart Reflex ID
Bar Config	0-0xf	See Next Slide

Boot Configuration – PCI Express

BAR Config / PCIe Window Sizes								
		32 bit Address Translation					64 bit Address Translation	
BAR cfg	BAR0	BAR1	BAR2	BAR3	BAR4	BAR5	BAR1/2	BAR3/4
0b0000	PCIe MMRs	32	32	32	32	Clone of BAR4		
0b0001		16	16	32	64			
0b0010		16	32	32	64			
0b0011		32	32	32	64			
0b0100		16	16	64	64			
0b0101		16	32	64	64			
0b0110		32	32	64	64			
0b0111		32	32	64	128			
0b1000		64	64	128	256			
0b1001		4	128	128	128			
0b1010		4	128	128	256			
0b1011		4	128	256	256			
0b1100							256	256
0b1101							512	512
0b1110							1024	1024
0b1111							2048	2048

Boot Configuration
HyperLink Mode

- HyperLink boot mode boots the DSP through the ultra short range HyperLink.
- The host loads the boot image directly through the link and then generates the interrupt to wake the DSP.

MCM Boot Device Configuration						
9	8	7	6	5	4	3
Reserved	Data Rate		Ref Clock			

MCM Boot Device Configuration Field Descriptions		
Bit Field	Value	Description
SR Index	0-3	Smart Reflex Index
Ref Clock	0	156.25 MHz
	1	250 MHz
	2	312.5 MHz
Data Rate	0	1.25 GBs
	1	3.125 GBs
	2	6.25 GBs
	3	12.5 GBs

Booting Multiple Cores

- During the boot process, the boot loader code is loaded into the L2 of corePac0 from the ROM.
- The high 0xD23F (52K) bytes of L2 in all corePacs are reserved for the boot code. User should not overwrite this area.
- All the other Cores will execute an IDLE.
- User should load the image into the L2 of CorePacs they want to boot up.
- Before setting the boot complete register, the user should also set the start address of the code in the respective BOOT MAGIC ADDRESS of the CorePac L2.
- Finally, the user image should also write the IPC interrupt register to bring the required corePacs out of IDLE.

Second Stage Bootload Option

Second Stage Boot Load Process

Q: What if more boot parameters are needed than can be specified in the boot pins?

A:Other parameter values can be updated through I2C boot mode

• In this case, the I2C boot will start with a I2C boot parameter table which will in turn load a custom updated parameter table for a specific boot mode.

• Once the default parameter table is updated, the boot code executes using the updated boot parameter structure, using the same process as the primary boot mode.

Second Stage Boot Load Specifics

- The EEPROM image loaded will have two boot parameter tables
- The First one will be an I2C boot parameter table, setting the core clock and also the address of the next block.
- The next block will have the desired boot mode specific boot parameter table with the user desired values.
- After loading this image into the EEPROM, the boot mode in the boot strap is set for I2C master boot.
- After POR, the I2C boot code is executed as a first stage boot load, which will update the default boot parameter table and re-enter the boot code, executing the boot code utilizing the user specific parameters.