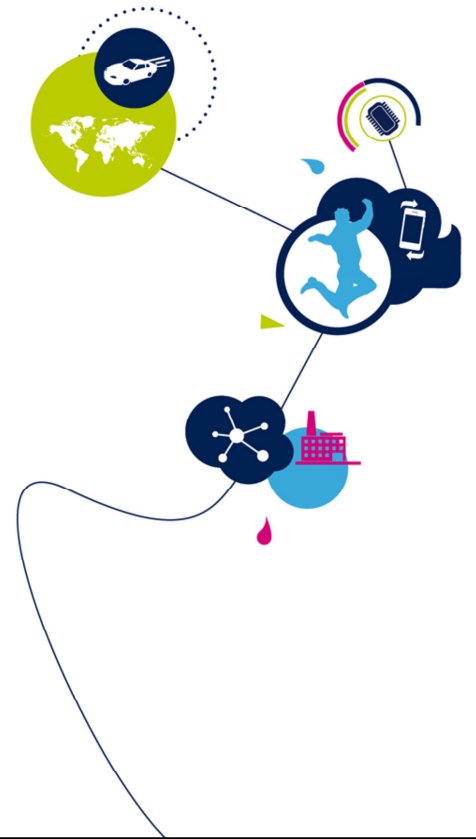


STM32MP1 – SYSCFG

System Configuration Controller
Revision 2.0



Hello, and welcome to this presentation of the STM32MP1 Series System Configuration Controller.

- The STM32MP1 Series microprocessors feature a set of configuration registers. The main purposes of the system configuration controller are the following:
 - Select the AXI masters data path used to access DDRCTRL
 - Select the Ethernet PHY interface
 - Manage the BOOT pins
 - Manage the I/O compensation cell
 - Enable timer break safety features
 - Enable /disable I2C Fast-mode Plus 20 mA high-drive
 - Enable /disable SPI/SDMMC/ETH/QUADSPI/TRACE 1.8V high-speed mode



The STM32MP1 microprocessors feature a set of configuration registers.

The system configuration controller gives access to the following features:

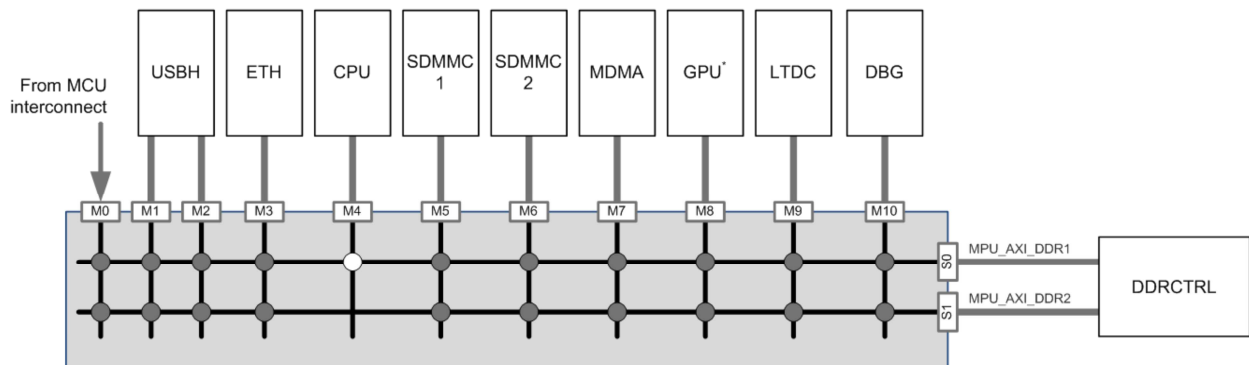
- Selection of the data path for AXI masters to the DDR controller,
- Selection of the Ethernet PHY interface,
- Management of the I/O compensation,
- Configuration of the 20 mA high-drive I/Os used for I²C Fast-mode Plus
- Configuration of the high-speed 1.8V I/Os used for SPI, SDMMC, Ethernet, Quad-SPI and trace.

AXI master data path

3

Boosts performance

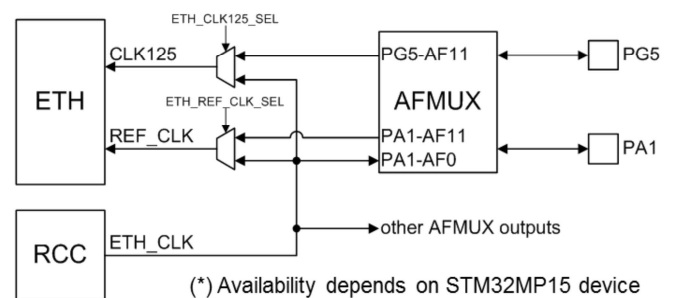
- Each master (except Cortex-A7 CPU) could use either DDRCTRL port1 or port2



(*) Availability depends on STM32MP15 device

The system configuration enables the control of the balance of the bandwidth between masters. Example are low latency masters (e.g. processor) versus high-bandwidth masters (e.g. Graphic processor or USB host) to get the best usage of the DDR memory bandwidth. Both DDR ports are using the same physical mapping address,. So there is no need to update the software and any change can be done during execution to dynamically adapt to different use cases.

- Control of PHY type
 - GMII*, RGMII*, MII and RMII are supported
 - Must be selected while ETH is in reset and before enabling any ETH clock
- Control of Ethernet Reference clock
 - Depending on external Ethernet PHY, ETH_CLK1 pin could have various usage.
 - GMII* and RGMII*: PG5* could be used to receive 125MHz from PHY or internal 125MHz clock
 - RMII: ETH_CLK at 50MHz could be output on PA1 and as well used by ETH REF_CLK
 - ALL modes: ETH_CLK could be used for 25MHz PHY reference clock (PA1, PB5 or PG8)
 - Must be selected while ETH is in reset and before enabling any ETH clock



The system configuration controller allows static configuration of the external Ethernet physical interface used. Supported Gigabit physical interface are GMII or RGMII with or without 125MHz from external physical interface. 10 or 100Mbit/s MII and RMII physical interface are also supported. ETH_CLK from RCC could be used to feed the external physical interface reference clock (usually 25 MHz) and thus saving a crystal.

- BOOT pins are used to control the source of the start code
 - BOOT[2:0] are sampled by BOOTROM code during initialization
 - Available in a SYSCFG register which could be also read by the application if required
 - Possible BOOT settings include
 - SD-Card interface on SDMMC1 with or without external level translator
 - eMMC 4 or 8 bits on SDMMC2
 - Serial NOR-Flash on QUADSPI
 - Serial NAND-Flash on QUADSPI
 - Parallel NAND-Flash 8 or 16 bits on FMC
 - Serial (USART2/3/6 and UART4/5/7/8)
 - USB High-Speed device (OTG) thru embedded PHY
 - Default interface pin (for SDMMC, QUADSPI) and interface instance (for SDMMC) can be overridden by OTP fuse settings
 - BOOT pin choices can be overridden by OTP fuse while keeping rescue mode thru Serial/USB with a single push-button or strap (e.g. after sales services)

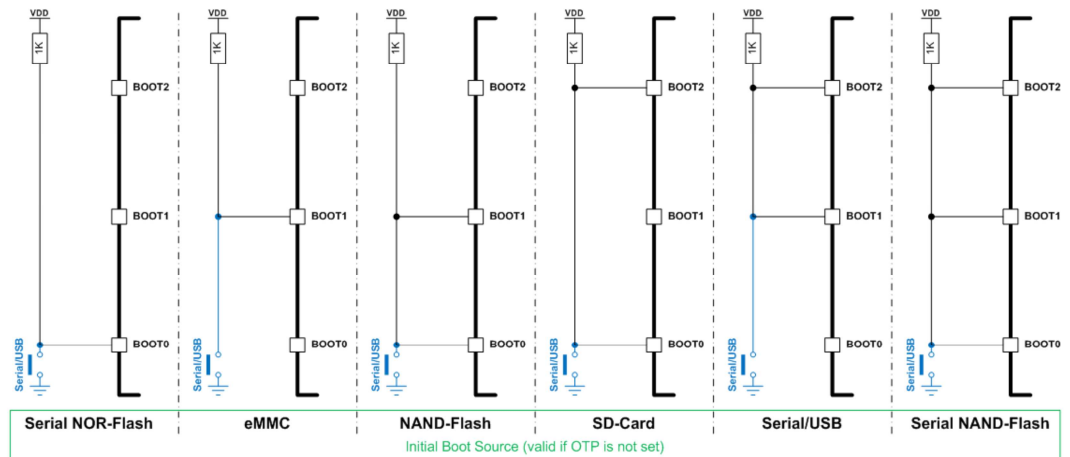


The values on the BOOT pins are latched during the boot phase. It is up to the user to set the BOOT pins to select the required boot mode before reset.

The BOOT pins are also resampled when the device exits Standby mode. Consequently, they must be kept in the required boot mode configuration when the device is in Standby mode.

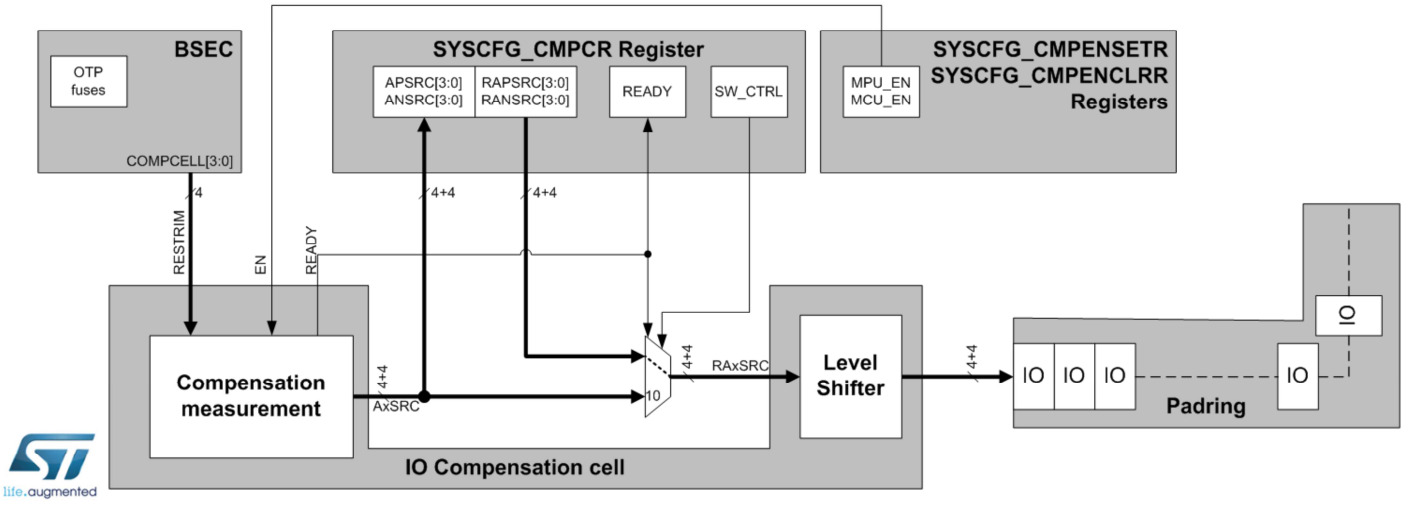
During STOP or RUN mode, the embedded pull-down on BOOT pins can be disabled to save power. The pull-down are automatically disabled during STANDBY mode and enabled again on STANDBY mode exit.

- Smart handling of the BOOT pins enables flexible boot scenario, e.g.:
 - Initial Boot from SD-Card which then program eMMC/NOR/NAND Flash memories
 - Rescue boot from Serial/USB port
 - Pre-Programmed Flash memory
 - Boot from USB to program Flash
 - Debug



Refer to the application note AN5031 (Getting started with STM32MP1 Series hardware development) and the Reference Manual for details on the BOOT pins.

- Automatic compensation of I/O voltage and temperature variation
 - Not used by default. Should be used for VDD I/Os frequencies above 50MHz
 - IC process variation is compensated during the production by setting OTP fuses



In order to have best electrical characteristics whatever the conditions, the IO compensation automatically tunes the characteristics of the I/Os when the voltage or the temperature changes. Required only for VDD I/Os frequencies above 50 MHz.

Safety

- Safety features in Control Timer Break registers
 - PVD lock to connect PVD interrupt to TIM1/8/15/16/17 Break input and to lock PVD enable and threshold
 - CLL lock to connect Cortex M4 LOCKUP output to TIM1/8/15/16/17 Break input
 - Where a fault or supervisor call occurs at a priority of -1 or above the Cortex-M4 enters lockup state.
 - Put timers in application safe state in case of application problem detection.
 - Power voltage detector event
 - CORTEX-M4 LOCKUP state



The Control Timer Break Register contains the control bits related to the to safety and the robustness. Two control bits steer certain error detection events to the timers' break inputs. This allows timer outputs to be placed in a known state during an application crash. Once programmed, the connection is locked until the next system reset. These internal events are the power voltage detector event and the Cortex-M4 LOCKUP state.

- In Peripheral mode configuration register
 - I2C I/Os Fast-mode Plus 20 mA drive enable
 - High drive is only enabled on pads selected for I2C in AFMUX
 - Control of direct ADC inputs and analog performance
 - Improve ADC performance when VDDA is below 2.7V
- IO control register
 - When I/O voltage is below 2.5V, it enables full speed on specific interfaces
 - SPI, SDMMC, ETH, QUADSPI, TRACE
 - There is one bit for each interface type, only affects pads used for selected interface in AFMUX
 - As setting the voltage above 2.7V could damage the chip, this feature should be globally enabled with an OTP fuse (product_below_2V5)



The system configuration controller manages the I2C I/Os Fast-mode Plus 20 mA drive enable control bits. This mode is only enabled on pads selected for I2C in AFMUX.

There is also controls to improve ADC performance when analog supply is below 2.7V.

The system configuration controller manages as well specific high speed I/Os control when the voltage is below 2.5V. This control is available for SPI, SDMMC, ETH, QUADSPI interfaces and for TRACE pins. The mode is only effective on pads selected in AFMUX.

WARNING: Using this feature when the voltage is above 2.7V could damage the chip. To avoid unwanted programming, a global enable fuse is provided as OTP and must be set only for a platform expected to use an I/O voltage below 2.5V.

- The Bootloader supports following Serial/USB interfaces
 - USART2 (PA3/PA2)
 - USART3 (PB12/PB10)
 - USART6 (PC7/PC6)
 - UART4 (PB2/PG11)
 - UART5 (PB5/PB13)
 - UART7 (PF6/PF7)
 - UART8 (PE0/PE1)
 - OTG HS in Device mode (USB_DP2/USB_DM2)
 - Need HSE quartz or external clock (default 24 MHz)



The on-chip bootloader allows the user to program the Flash memory or set OTP fuses through a serial communication peripheral. The supported protocols are USART and USB. USB is recommended for a large amount of external memory.

Note

The USB Boot works with specific values of the external quartz oscillator on HSE with 24 MHz as default; 25 and 26 MHz are also possible with dedicated OTP fuse settings; OTP setting for HSE auto-detection allows 8, 10, 12, 14, 16, 20, 24, 28, 32, 36, 40, 48 MHz for USB Boot.

The USART/UART uses the internal HSI oscillator with automatic baud rate detection.

There is no restriction on the HSE frequency between 8 and 48MHz if the USB port is not used during the boot phase

(could be used by application with any HSE frequency).

- Refer to these training modules linked to this peripheral:
 - Reset and clock control (RCC)
 - Power controller (PWR)
 - Timers (TIM)
 - Ethernet (ETH)
 - Inter-Integrated Circuit (I²C)



In addition to this training, you can refer to the Reset and Clock Control, Power Controller, Timers, Ethernet and I²C trainings.