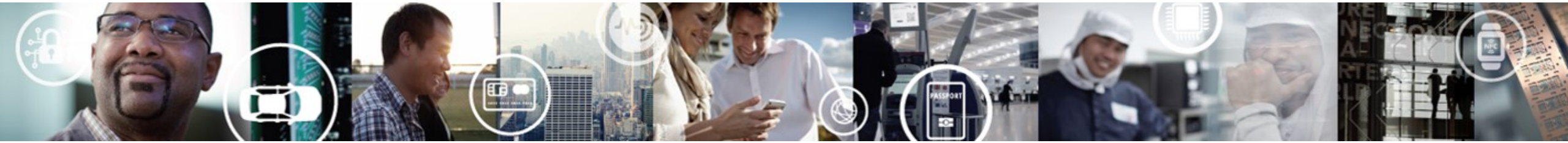
LPC55XX LOW POWER MODES

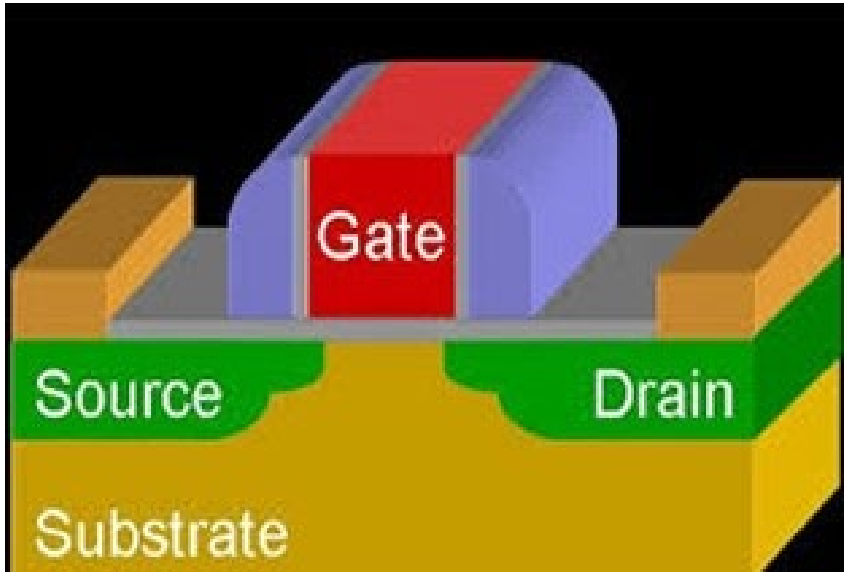
NOVEMBER, 2018

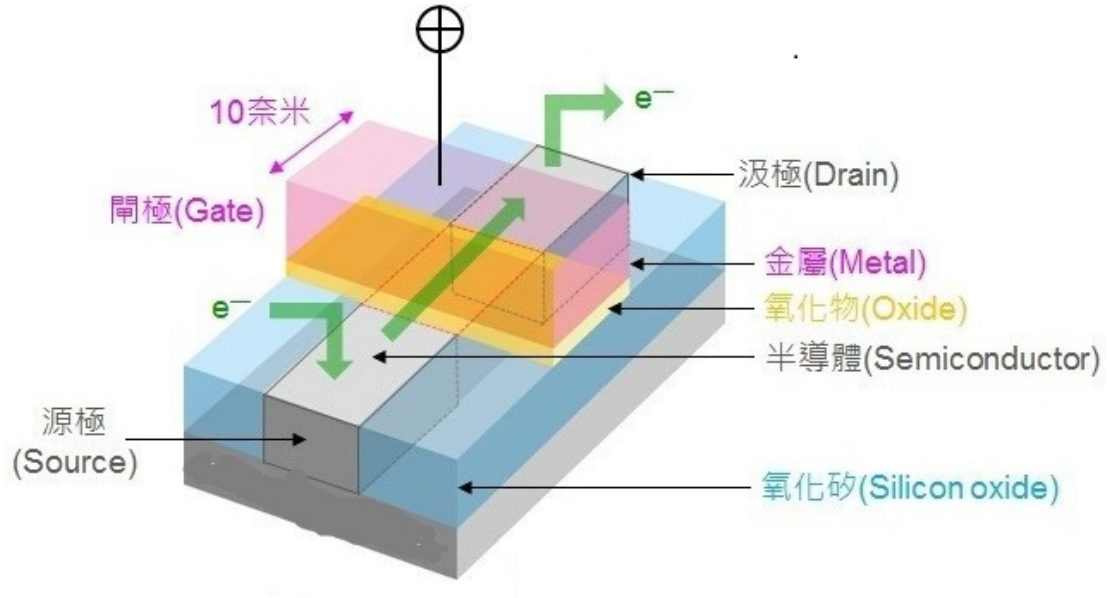


EXTERNAL USE

SECURE CONNECTIONS
FOR A SMARTER WORLD

LPC55XX : CMOS C040 Technology (40 nm channel length)





Compared to higher node technology like CMOS 090 or CMOS C140, CMOS 040 has such traits,

advantages	disadvantages
● **Higher intregration density** (this means that with a given area, we can put more features) ● **Better performances** in term of dynamic power consumption	**higher leakage power**

LPC55XX POWER SUPPLIES AND DOMAINS

LPC55xx Power Supplies

Power to the part is supplied via **6 on-chip regulators**

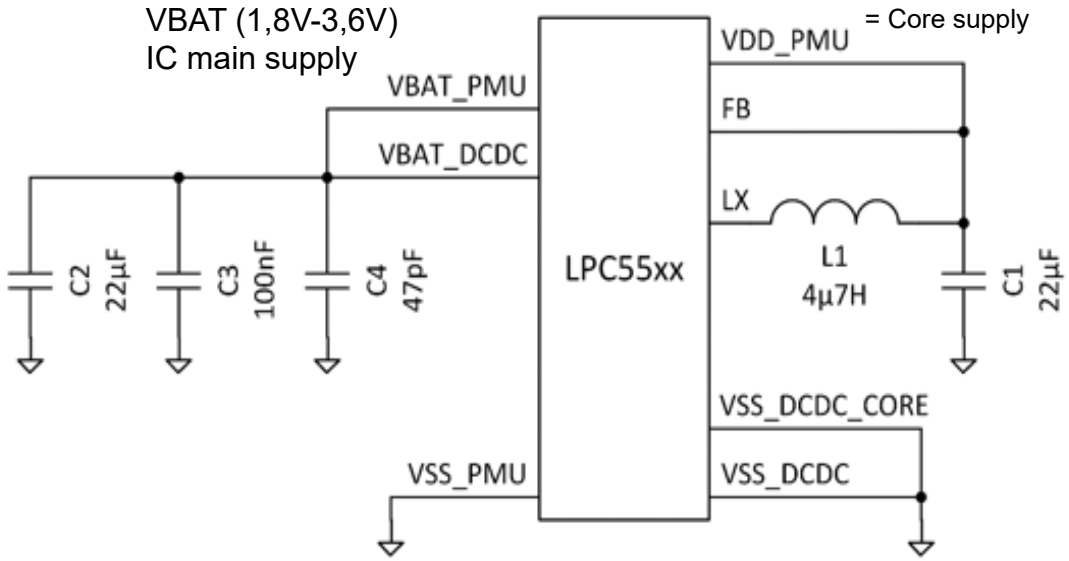
regulators	Characteristics
Bulk DCDC Converter	supplies main digital core logic , operating in the range 0.950 V – 1.200 V.
LDO_DEEP_SLEEP	used during “Deep-sleep” , operating in the range 0.9 V – 1.075 V.
LDO_AO	supplies the “Always on” digital logic , operating in the range 0.75 V – 1.220 V.
LDO_MEM	supplies SRAM during low power modes , operating in the range 0.75 V – 1.220 V.
LDO_USB_HS	supplies USB High Speed Interface , operating in the range 1.8V – 2.0 V.
LDO_FLASH_NV	supplies Flash Macro , operating in the range 1.8 V – 2.0 V.

All 6 previously mentioned internal regulators are supplied by the main external supply called **VBAT [1.8V – 3.6V]**

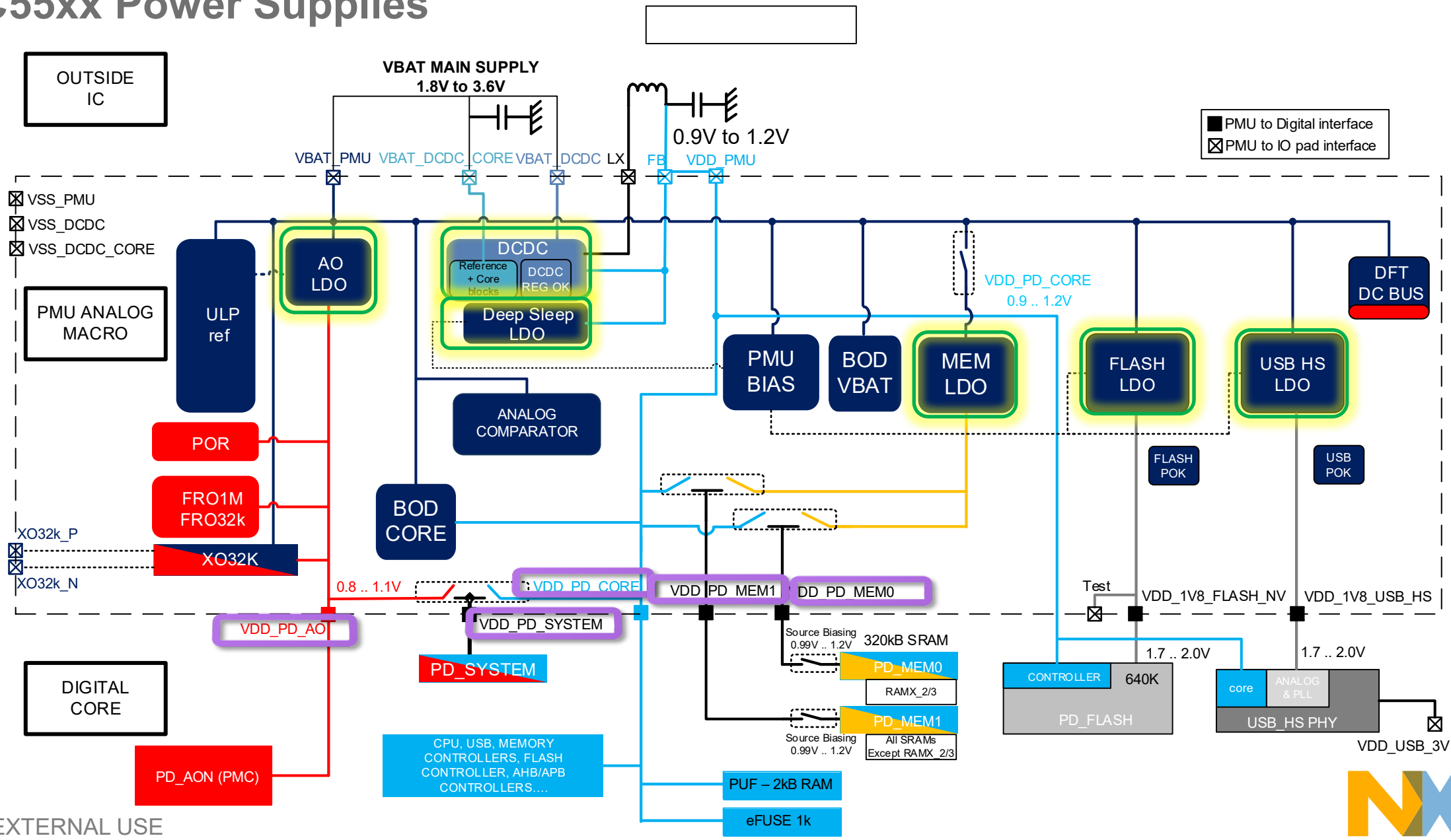
LPC55xx Power Management – DC-DC Converter

Up to +80% Efficiency DC-DC that is directly controlled and started by the Power Management Controller (PMC)

Features
Input Voltage range:1.8V to 3.6V
Output voltage: 1.1V
Max Load current: 50mA
Unconditional stable, Constant ON-Time PFM (COT)
Low Ripple with proper user COT trimming and optimal valued external inductor and capacitor
DC-DC needs a 4µ7H inductor and a 22µF ceramic capacitor



LPC55xx Power Supplies



Power Domain

- LPC55xx is partitioned into 5 separate power domains:

Power domains	Characteristics
PD_CORE	Power Domain “Core”: most of all digital core logic (CPU0, CPU1, multilayer matrix ,Serial Peripherals, System DMA...).
PD_SYSTEM	Power Domain “System”: Some critical system components like clocks controller, reset controller and Syscon.
PD_AO	Power Domain “Always On”: PMC(Power management controller) and RTC. This domain always has power as long as sufficient voltage is available on VBAT ([1.8 V –3.6 V]).
PD_MEM_0	First Power Domain “Memories”: Two 4 KB SRAM instances.
PD_MEM_1	Second Power Domain “Memories”: All other SRAM instances

- How the different power domains are supplied :

Power domains	Powered by
PD_CORE	• By DCDC in active mode and sleep-mode. • By LDO_DEEP_SLEEP / DCDC in deep-sleep mode.
PD_SYSTEM	• By DCDC in active and sleep-modes • By LDO_AO in power-down modes.
PD_AO	• By LDO_AO only, whatever the power mode.
PD_MEM_0 PD_MEM_1	• By DCDC in active and sleep-power modes. • By LDO_MEM in deep-sleep, power-down and deep-power down modes.

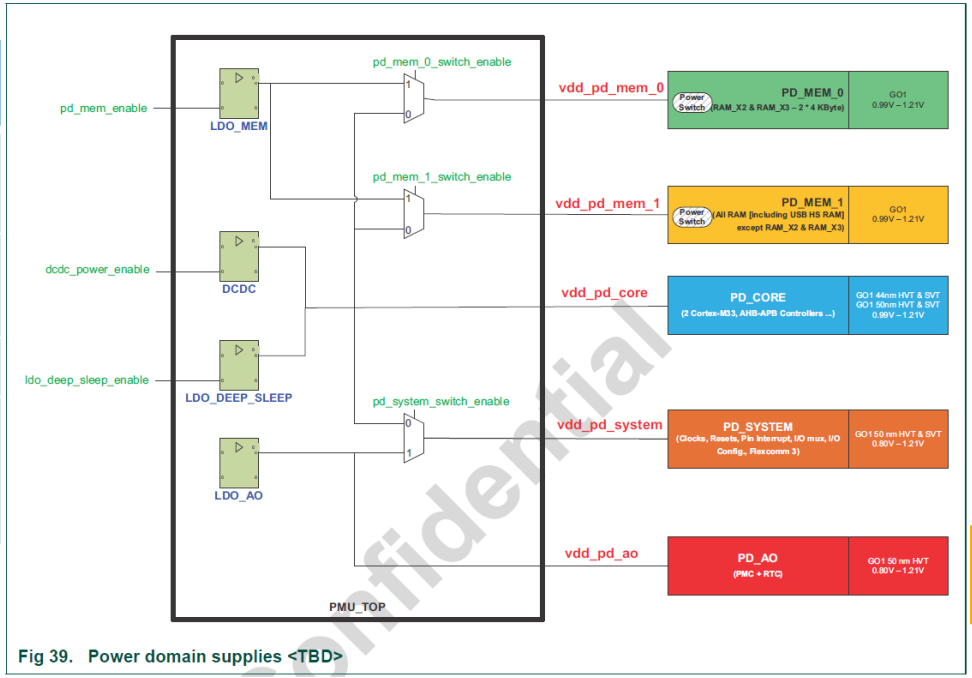
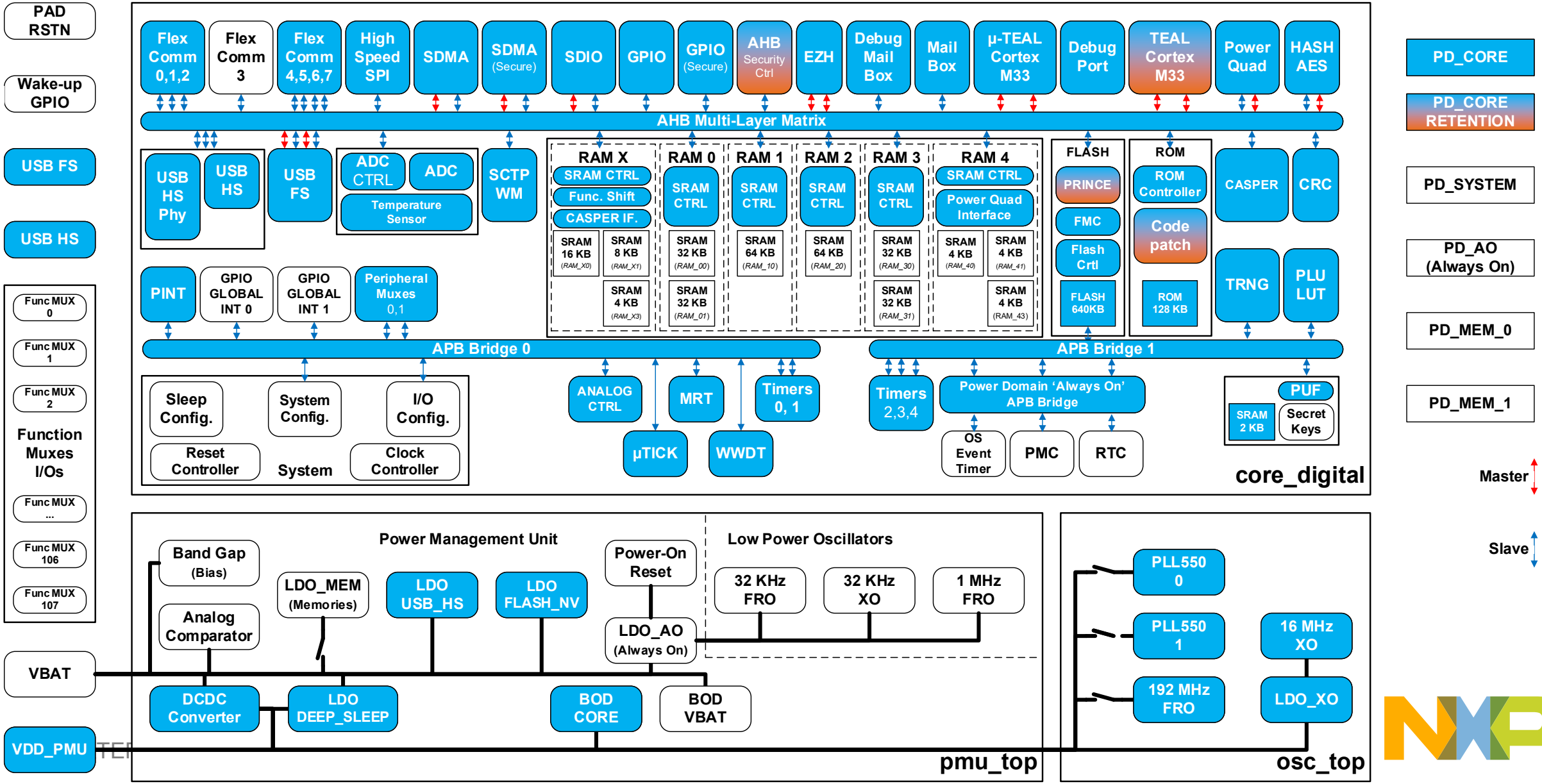
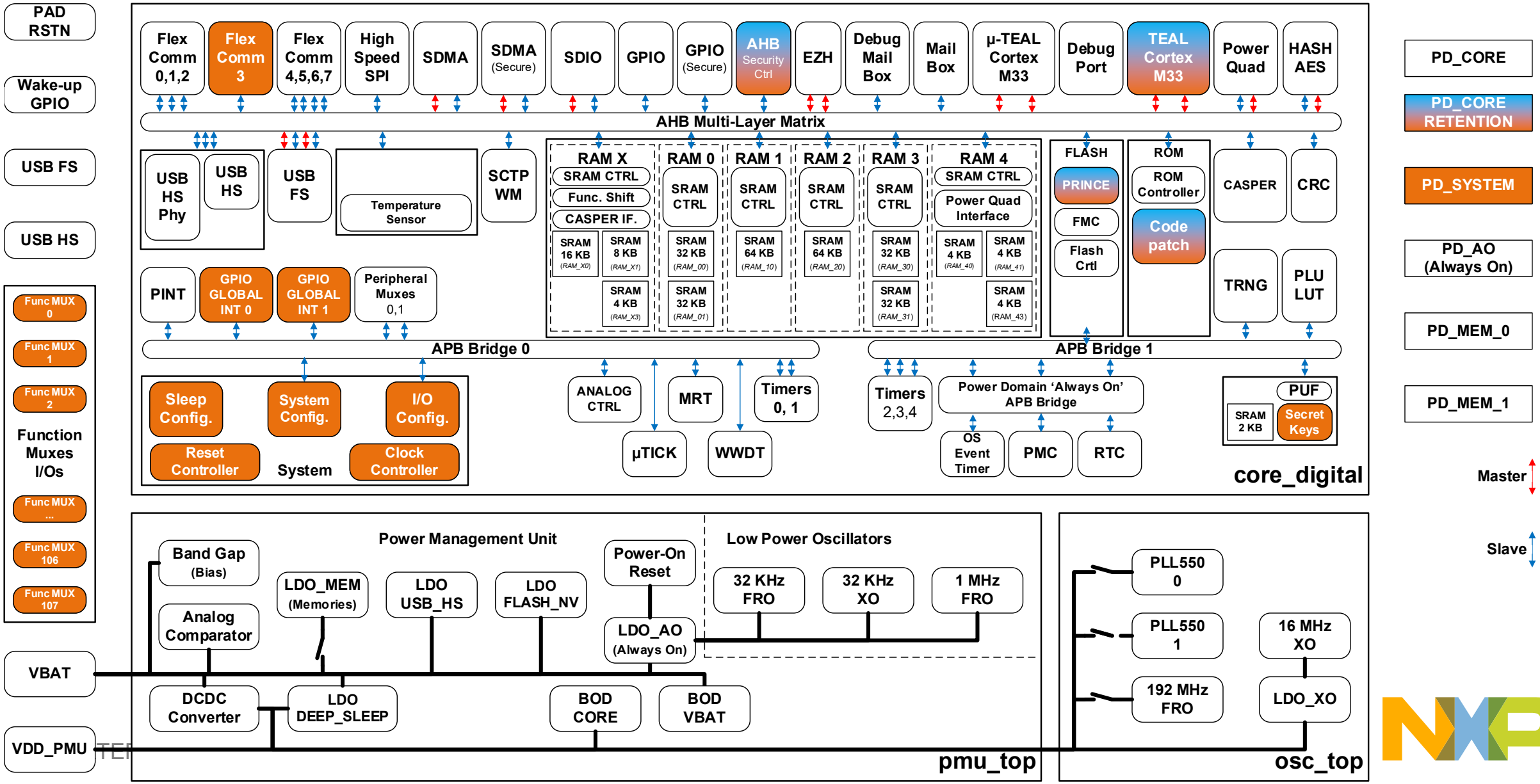


Fig 39. Power domain supplies <TBD>

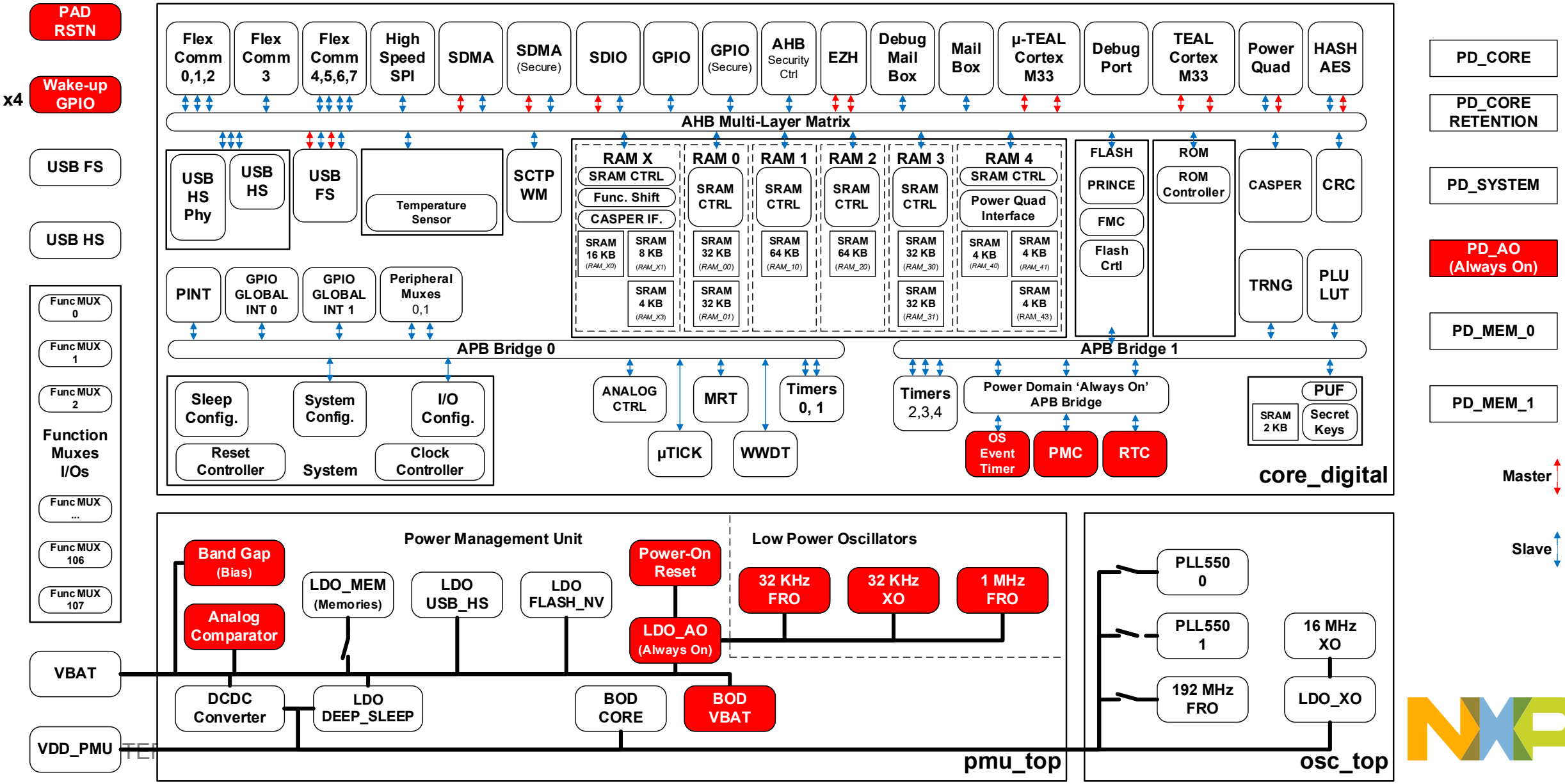
Power Domain : 1. CORE (PD_CORE)



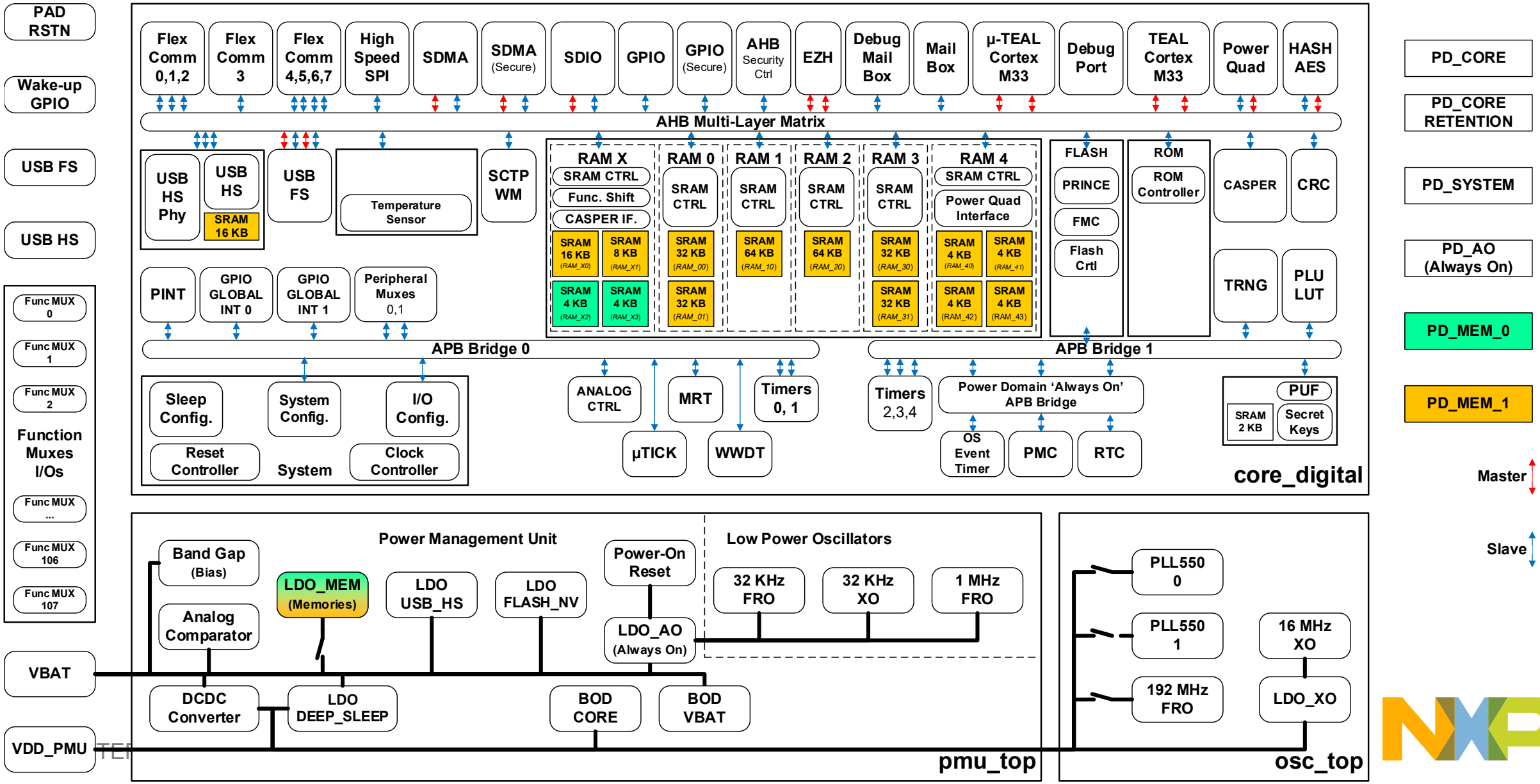
Power Domain : 2. SYSTEM (PD_SYSTEM)



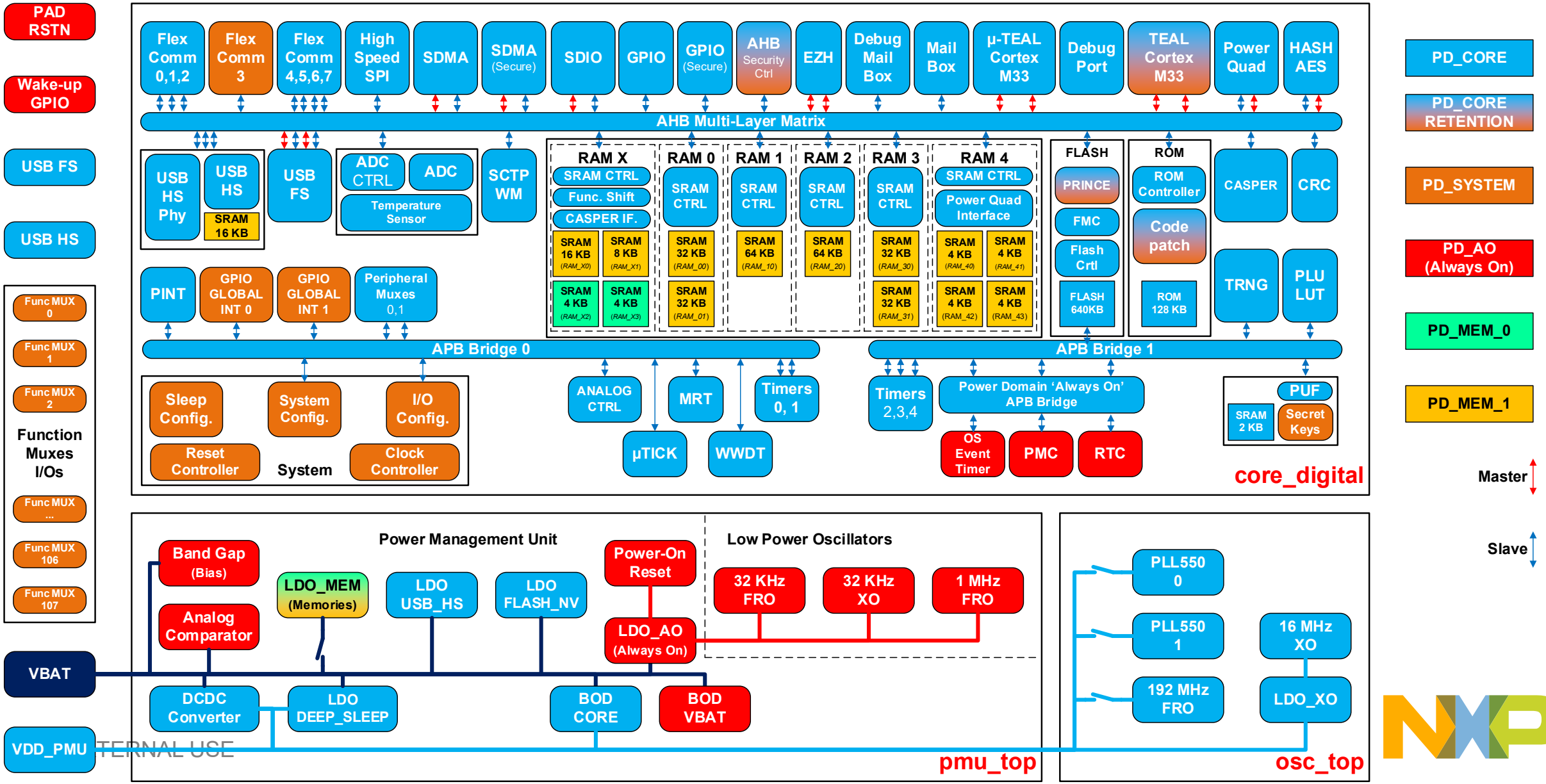
Power Domain : 3. ALWAYS-ON (PD_AO)



Power Domain : 4. MEMORIES (PD_MEM0 & PD_MEM1)

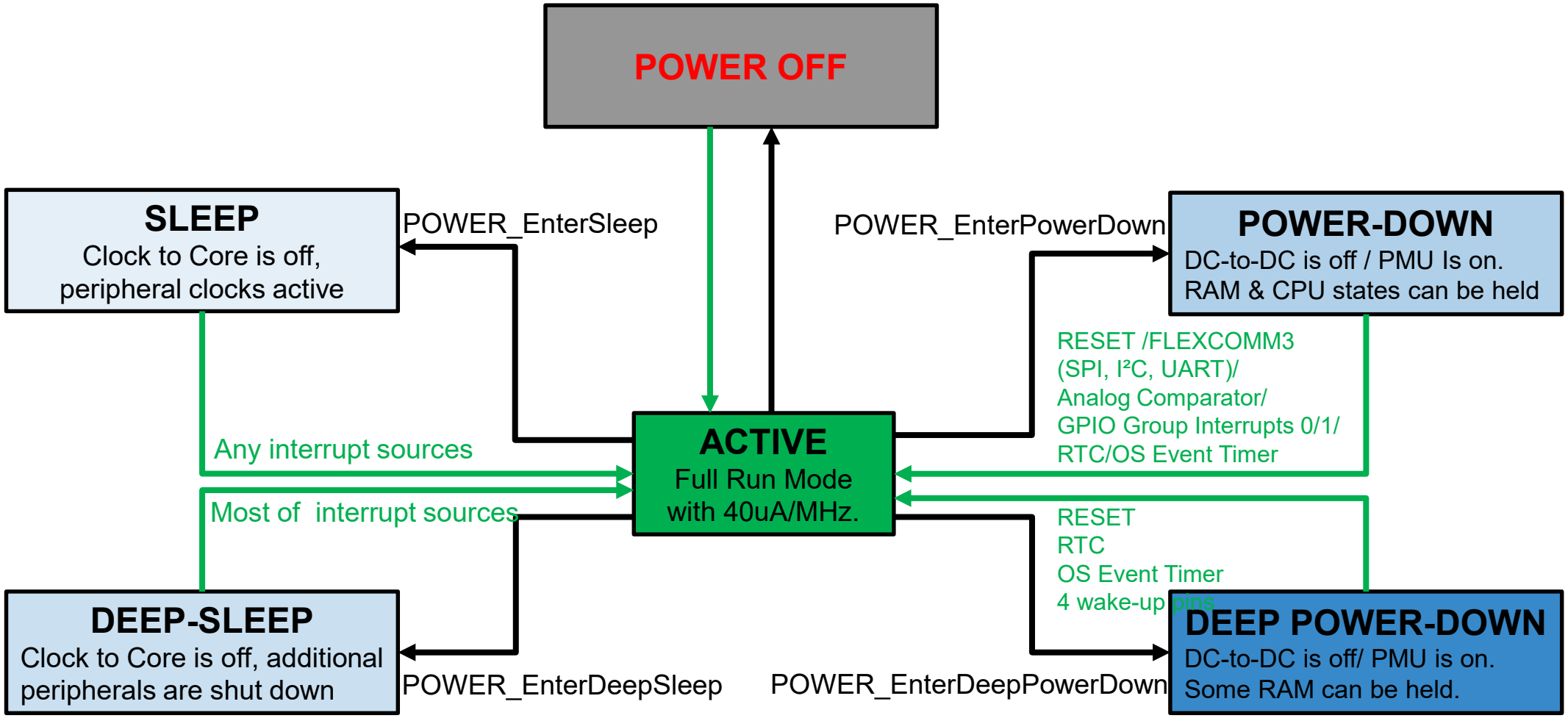


Power Domain : All together



LPC55XX LOW POWER MODES

POWER CYCLE

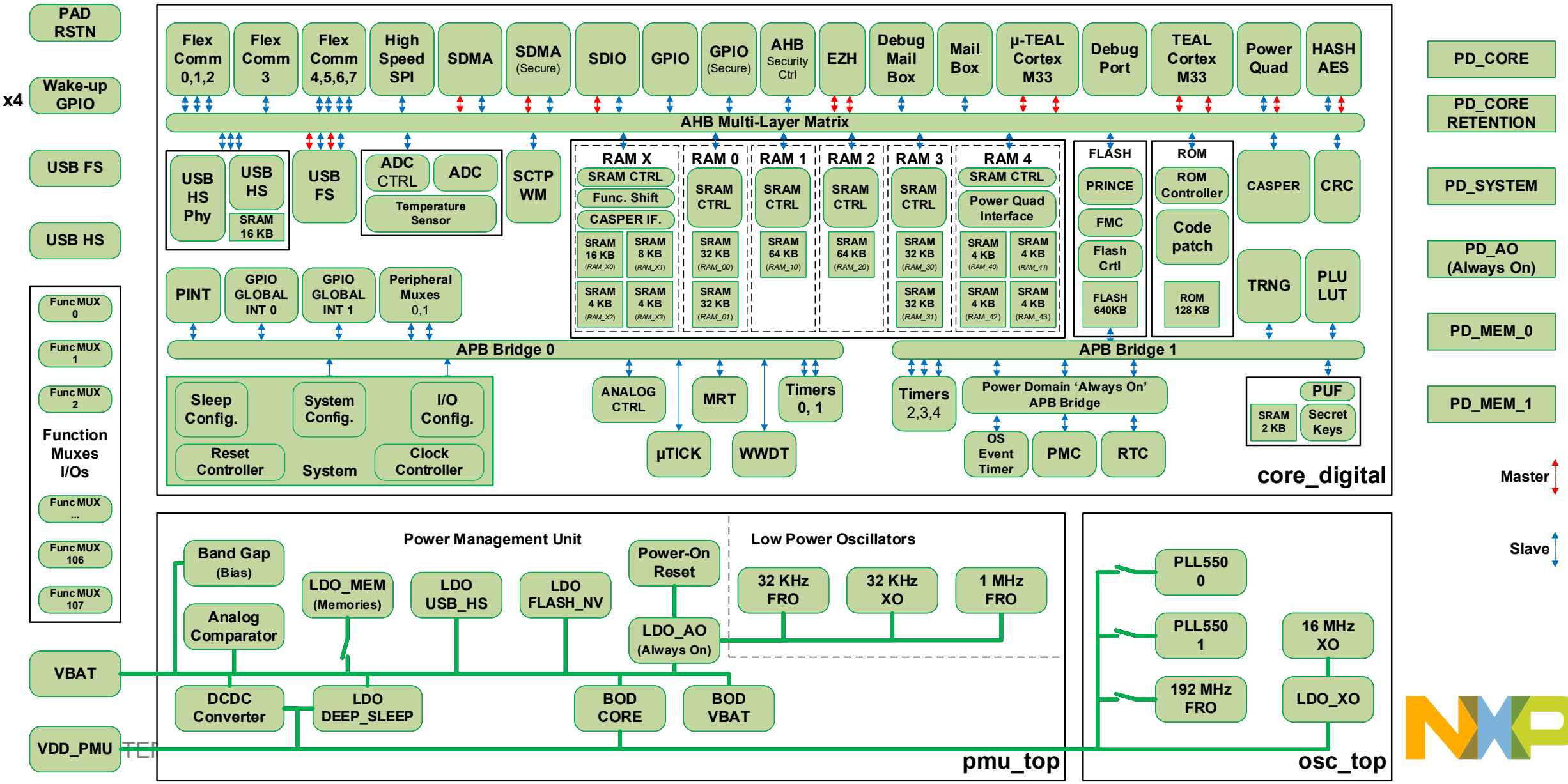

POWER OFF
SLEEP
Clock to Core is off,
peripheral clocks active
POWER_EnterSleep
POWER_EnterPowerDown
POWER-DOWN
DC-to-DC is off / PMU Is on.
RAM & CPU states can be held
RESET /FLEXCOMM3
(SPI, I²C, UART)/
Analog Comparator/
GPIO Group Interrupts 0/1/
RTC/OS Event Timer
Any interrupt sources
ACTIVE
Full Run Mode
with 40uA/MHz.
Most of interrupt sources
RESET
RTC
OS Event Timer
4 wake-up pins
DEEP-SLEEP
Clock to Core is off, additional
peripherals are shut down
DEEP POWER-DOWN
DC-to-DC is off/ PMU is on.
Some RAM can be held.
POWER_EnterDeepSleep
POWER_EnterDeepPowerDown

SLEEP

SLEEP : Overview

Sleep mode	Wake-up sources
● "**Sleep mode**" saves power by stopping CPU execution without affecting peripherals or requiring significant wake-up time. ● The clock to the CPU is shut off. ● Peripherals and memories are active and operational.	● Any interrupt enabled in the NVIC arrives at the processor ● Any system reset occurs (Power On Reset, Pin Reset, Watchdog Timer reset ...)

SLEEP

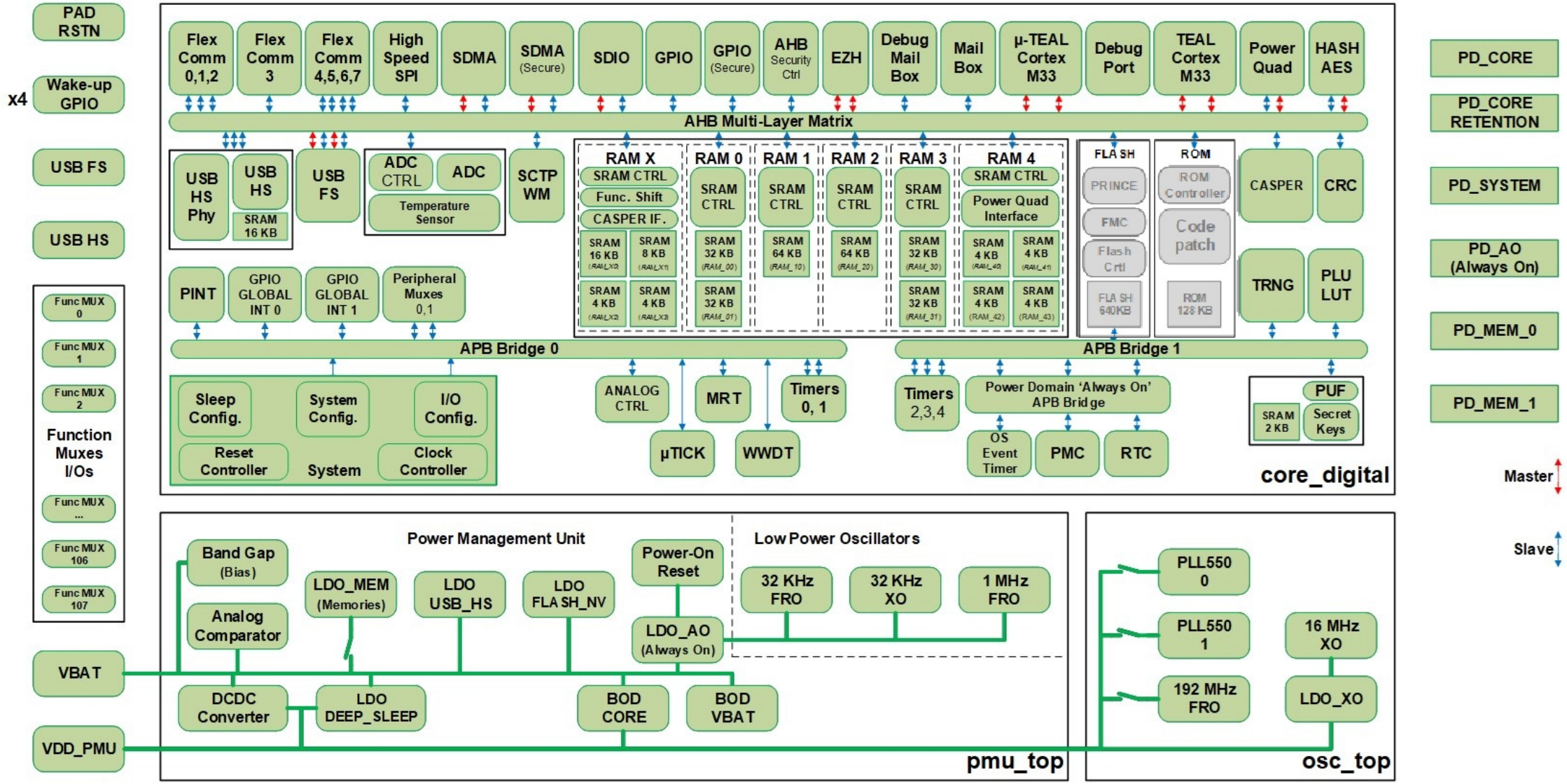

PAD RSTN
x4
Wake-up GPIO
USB FS
USB HS
Func MUX 0
Func MUX 1
Func MUX 2
Function Muxes I/Os
Func MUX ...
Func MUX 106
Func MUX 107
VBAT
VDD_PMU
Flex Comm 0,1,2
Flex Comm 3
Flex Comm 4,5,6,7
High Speed SPI
SDMA
SDMA (Secure)
SDIO
GPIO
GPIO (Secure)
AHB Security Ctrl
EZH
Debug Mail Box
Mail Box
µ-TEAL Cortex M33
Debug Port
TEAL Cortex M33
Power Quad
HASH AES
AHB Multi-Layer Matrix
USB HS Phy
USB HS
SRAM 16 KB
USB FS
ADC CTRL
ADC
Temperature Sensor
SCTP WM
RAM X
SRAM CTRL
Func. Shift
CASPER IF.
SRAM 16 KB (RAM_X0)
SRAM 8 KB (RAM_X1)
SRAM 4 KB (RAM_X2)
SRAM 4 KB (RAM_X3)
RAM 0
SRAM CTRL
SRAM 32 KB (RAM_00)
SRAM 32 KB (RAM_01)
RAM 1
SRAM CTRL
SRAM 64 KB (RAM_10)
RAM 2
SRAM CTRL
SRAM 64 KB (RAM_20)
RAM 3
SRAM CTRL
SRAM 32 KB (RAM_30)
SRAM 32 KB (RAM_31)
RAM 4
SRAM CTRL
Power Quad Interface
SRAM 4 KB (RAM_40)
SRAM 4 KB (RAM_41)
SRAM 4 KB (RAM_42)
SRAM 4 KB (RAM_43)
FLASH
PRINCE
FMC
Flash Crtl
FLASH 640KB
ROM
ROM Controller
Code patch
ROM 128 KB
CASPER
CRC
TRNG
PLU LUT
PINT
GPIO GLOBAL INT 0
GPIO GLOBAL INT 1
Peripheral Muxes 0,1
APB Bridge 0
APB Bridge 1
Sleep Config.
System Config.
I/O Config.
Reset Controller
System
Clock Controller
ANALOG CTRL
MRT
Timers 0, 1
µTICK
WWDT
Timers 2,3,4
Power Domain 'Always On' APB Bridge
OS Event Timer
PMC
RTC
PUF
SRAM 2 KB
Secret Keys
core_digital
PD_CORE
PD_CORE RETENTION
PD_SYSTEM
PD_AO (Always On)
PD_MEM_0
PD_MEM_1
Master
Slave
Power Management Unit
Band Gap (Bias)
Analog Comparator
LDO_MEM (Memories)
LDO USB_HS
LDO FLASH_NV
Power-On Reset
LDO_AO (Always On)
DCDC Converter
LDO DEEP_SLEEP
BOD CORE
BOD VBAT
Low Power Oscillators
32 KHz FRO
32 KHz XO
1 MHz FRO
pmu_top
PLL550 0
PLL550 1
192 MHz FRO
16 MHz XO
LDO_XO
osc_top

DEEP-SLEEP

DEEP SLEEP : Overview

Deep Sleep mode	Wake-up sources
● "**Deep-sleep**" reduces power by elimating almost all dynamic power and by lowering the core voltage supply down to 0.9V. ● The full IC remained powered but flash and ROM are shut down ● The clock to all CPUs is shut down ● The peripherals receive no internal clock (software configurable) ● All SRAM and registers maintain their internal states. All SRAM instances that are not configured to enter in 'retention state' will stay in active state (and therefore consume more power). ● Some peripherals can have DMA service during deep-sleep mode without waking up entire device. ● Through the power profile API, selected peripherals such as FLEXCOMM, WWDT, USB, Counters… can be left running to wake up the part.	● A reset from the RESET pin, or the Watchdog Timer (WWDT) ● The 8 pin interrupts from the Pin Interrupt Generator ● An interrupt from a block such as the watchdog interrupt or RTC interrupt ● Wake-up signal from any of the serial peripherals (FLEXCOMMs and High Speed SPI) that are operating in deep-sleep mode ● GPIO group interrupt signal ● RTC alarm signal or wake-up signal ● OS Event Timer

DEEP-SLEEP

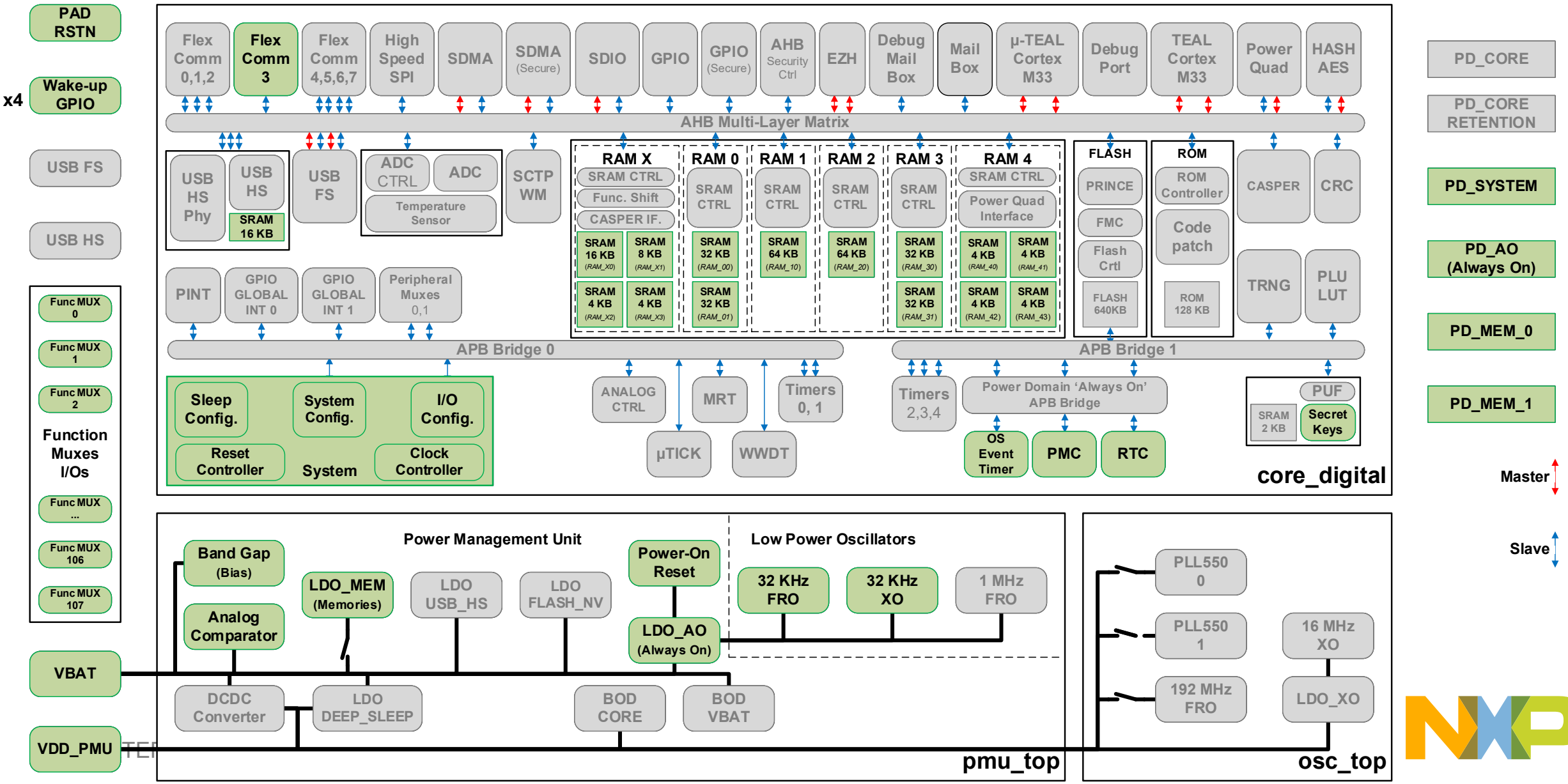

PAD RSTN
x4
Wake-up GPIO
USB FS
USB HS
Func MUX 0
Func MUX 1
Func MUX 2
Function Muxes I/Os
Func MUX ...
Func MUX 106
Func MUX 107
VBAT
VDD_PMU
Flex Comm 0,1,2
Flex Comm 3
Flex Comm 4,5,6,7
High Speed SPI
SDMA
SDMA (Secure)
SDIO
GPIO
GPIO (Secure)
AHB Security Ctrl
EZH
Debug Mail Box
Mail Box
µ-TEAL Cortex M33
Debug Port
TEAL Cortex M33
Power Quad
HASH AES
AHB Multi-Layer Matrix
USB HS Phy
USB HS
SRAM 16 KB
USB FS
ADC CTRL
ADC
Temperature Sensor
SCTP WM
RAM X
SRAM CTRL
Func. Shift
CASPER IF.
RAM 0
RAM 1
RAM 2
RAM 3
RAM 4
Power Quad Interface
FLASH
PRINCE
FMC
Flash Crtl
FLASH 640KB
ROM
ROM Controller
Code patch
ROM 128 KB
CASPER
CRC
TRNG
PLU LUT
PINT
GPIO GLOBAL INT 0
GPIO GLOBAL INT 1
Peripheral Muxes 0,1
APB Bridge 0
APB Bridge 1
Sleep Config.
System Config.
I/O Config.
Reset Controller
System
Clock Controller
ANALOG CTRL
MRT
Timers 0, 1
µTICK
WWDT
Timers 2,3,4
Power Domain 'Always On' APB Bridge
OS Event Timer
PMC
RTC
PUF
SRAM 2 KB
Secret Keys
core_digital
Power Management Unit
Band Gap (Bias)
Analog Comparator
LDO_MEM (Memories)
LDO USB_HS
LDO FLASH_NV
Power-On Reset
LDO_AO (Always On)
Low Power Oscillators
32 KHz FRO
32 KHz XO
1 MHz FRO
DCDC Converter
LDO DEEP_SLEEP
BOD CORE
BOD VBAT
pmu_top
PLL550 0
PLL550 1
192 MHz FRO
16 MHz XO
LDO_XO
osc_top
PD_CORE
PD_CORE RETENTION
PD_SYSTEM
PD_AO (Always On)
PD_MEM_0
PD_MEM_1
Master
Slave

POWER DOWN

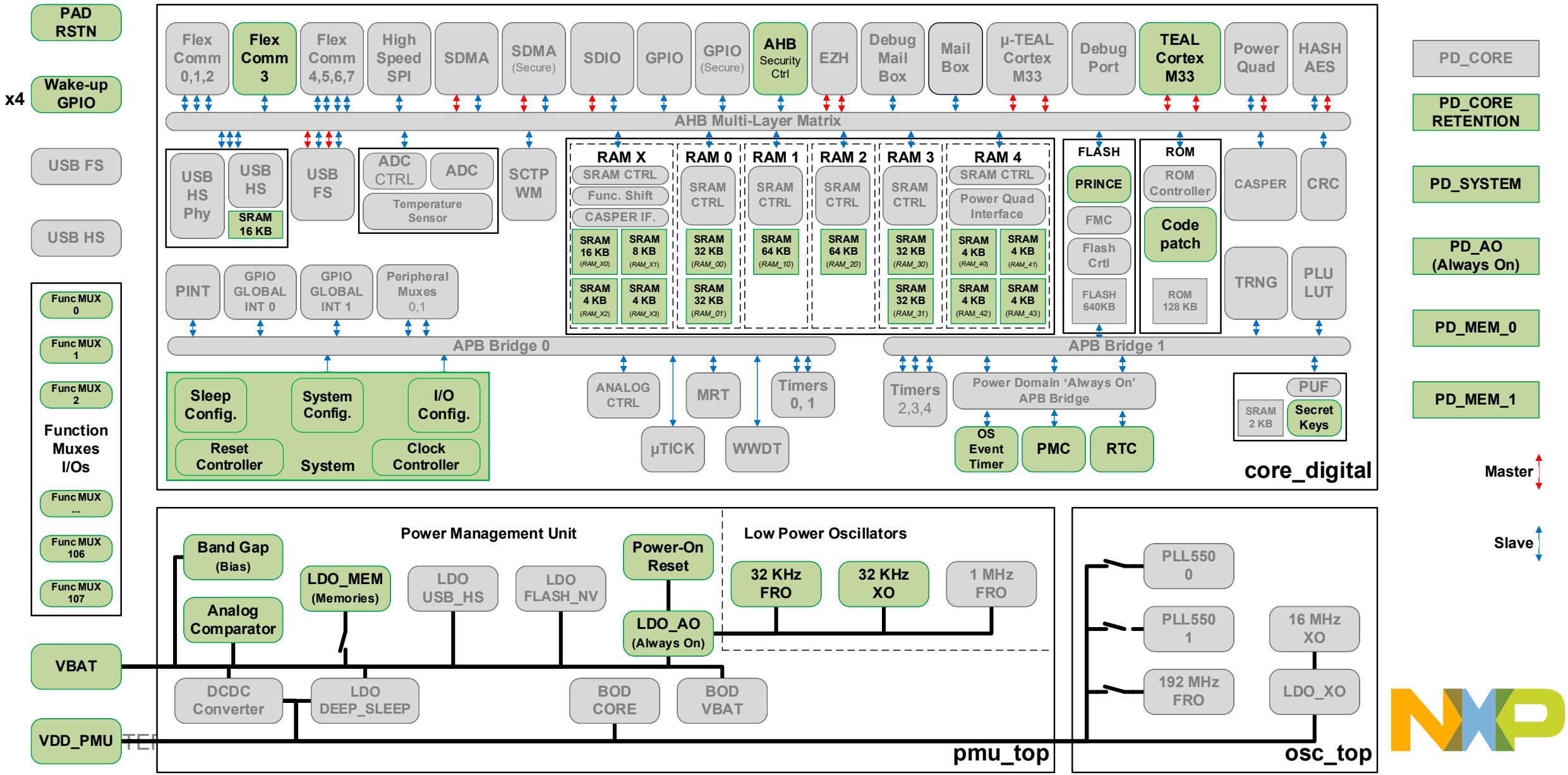
POWER DOWN: Overview

Power down mode	Wake-up sources
• " **Power-down**" mode turns off nearly all on-chip power consumption (by shutting down the DCDC) • The power-down mode affects the entire system, the clock to all CPUs and peripheral is shut down. • All registers lose their internal states except those located in the Power Domains PD_SYSTEM and PD_AO. • The contents of all SRAM instances can be retained, or will lose it. • It is possible to retain the internal state of the CPU0; in that case, the internal states of the AHB Security Controller and PRINCE will also be retained. • In case CPU0 state is retained, when wake-up event occurs, code execution will resume from where it has stopped,or will restarts from address 0x0. It is the responsibility of the application to re-configure all modules in power domain Core PD_CORE (all Flexcomm – except flexcomm3 -, SDMA, GPIO …)	• A reset from the RESET pin • Any Serial module inside FLEXCOMM3 (SPI, I²C, UART) • Analog Comparator • Any GPIO in Port0 & Port1 via GPIO Group Interrupts 0/1 • RTC alarm signal or wake-up signal • OS Event Timer

POWER-DOWN 2-1



POWER-DOWN 2-2 (with CPU State Retention)

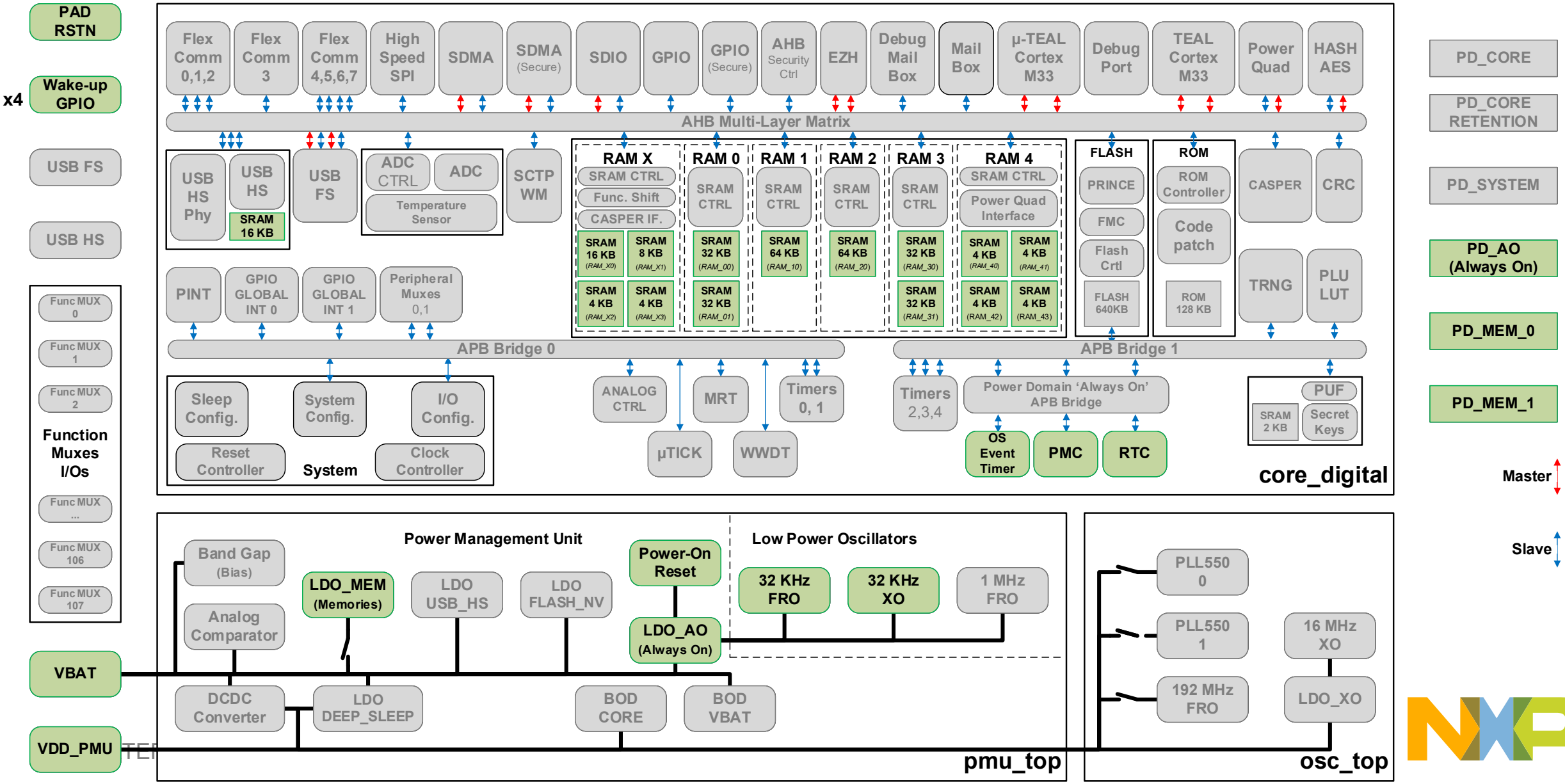


DEEP POWER-DOWN

DEEP POWER DOWN: Overview

Deep Power Down mode	Wake-up sources
● "**Deep power-down**" mode shuts down virtually all on-chip power consumption . ● Requires a significantly longer wake-up time . ● The entire system (CPUs and all peripherals) is shut down except for the PMU, the PMC, the RTC and the OS event timer .	● The RESET pin ● RTC alarm signal or wake-up signal ● OS Event Timer ● Any event (rising or falling edge) on one of the 4 wake-up pins

DEEP POWER-DOWN



LPC55xx System reduced power modes

	Active	Sleep	Deep-Sleep	Power-Down	Deep Power-Down
CPU	ON	Stopped	Stopped	CPU0	OFF
SRAM	Configurable	Configurable	Configurable	Configurable	Configurable
Flash	Configurable	Configurable	OFF	OFF	OFF
ROM	ON	ON	OFF	OFF	OFF
Peripherals	Configurable	Configurable	Configurable	ACMP Flexcomm3.Port0/1 Pint. RTC OS Event Timer	RTC OS Event Timer
Oscillators	Configurable	Configurable	Configurable	XTAL32K. FRO32K	XTAL32K. FRO32K

Peripheral configuration in reduced power modes

Peripharals		Reduced Power Modes			
Name	Description	SLEEP	DEEP-SLEEP	POWER-DOWN	DEEP POWER-DOWN
DCDC	*Bulk DCDC Converter*	ON	Software configured	OFF	OFF
BIAS	*Analog references*	ON	Software configured	Software configured	OFF
BoD Core	*Core Logic Brown Out Detector*	Software configured	Software configured	OFF	OFF
BoD VBAT	*VBAT Brown Out Detector*	Software configured	Software configured	OFF	OFF
FRO1M	*1 MHz Free Running Oscillator*	ON	Software configured	OFF	OFF
FRO192M	*192 MHz Free Running Oscillator*	ON	Software configured	OFF	OFF
FRO32K	*32 KHz Free Running Oscillator*	Software configured	Software configured	Software configured	Software configured
XTAL32K	*32 KHz Crystal Oscillator*	Software configured	Software configured	Software configured	Software configured
XTAL32M	*32 MHz Crystal Oscillator*	Software configured	Software configured	OFF	OFF
PLL0	*1st PLL550M*	Software configured	Software configured	OFF	OFF
PLL1	*2nd PLL550M*	Software configured	Software configured	OFF	OFF
USB_FS_PHY	*USB Full Speed Physical*	Software configured	Software configured	OFF	OFF
USB_HS_PHY	*USB High Speed Physical*	Software configured	Software configured	OFF	OFF
COMP	*Analog Compatator*	Software configured	Software configured	Software configured	OFF
TEMPSENS	*Temperature Sensor*	Software configured	Software configured	OFF	OFF
ADC	*General Purpose ADC*	Software configured	Software configured	OFF	OFF
LDO_MEM	*SRAM Regulator*	OFF	ON	Software configured	Software configured
LDO_DEEP_SLEEP	*SRAM Regulator*	Software configured	Software configured	OFF	OFF
LDO_USB_HS	*USB High Speed Regulator*	Software configured	Software configured	OFF	OFF
AUXBIAS	*ADC Analog references*	Software configured	Software configured	OFF	OFF
LDO_XO_32M	*32 MHz Crystal Oscillator Regulator*	Software configured	Software configured	OFF	OFF
LDO_FLASH_NV	*Flash Regulator*	ON	OFF	OFF	OFF
RNG	*True Random Number Generator*	Software configured	Software configured	OFF	OFF
PLL0_SSCG	*PLL0 Spread Spectrum Clock Generator*	Software configured	Software configured	OFF	OFF
ROM	*ROM*	ON	OFF	OFF	OFF

Note: *See parameter* **exclude_from_pd** *of Low Power API described here after.*

List of wake-up sources per Low Power Modes (2-1)

Bit	Wake-up Source	Description	DEEP-SLEEP	POWER-DOWN	DEEP POWER-DOWN
0	WAKEUP_SYS	Watchdog timer, BoDs	YES	NO	NO
1	WAKEUP_SDMA0	System DMA	YES	NO	NO
2	WAKEUP_GPIO_GLOBALINT0	GINT0	YES	YES	NO
3	WAKEUP_GPIO_GLOBALINT1	GINT1	YES	YES	NO
4	WAKEUP_GPIO_INT0_0	GPIO	YES	NO	NO
5	WAKEUP_GPIO_INT0_1	GPIO	YES	NO	NO
6	WAKEUP_GPIO_INT0_2	GPIO	YES	NO	NO
7	WAKEUP_GPIO_INT0_3	GPIO	YES	NO	NO
8	WAKEUP_UTICK	Micor-Tick timer	NO	NO	NO
9	WAKEUP_MRT	Multi Rate Timer	NO	NO	NO
10	WAKEUP_CTIMER0	Standard Counter/Timer 0	YES	NO	NO
11	WAKEUP_CTIMER1	Standard Counter/Timer 1	YES	NO	NO
12	WAKEUP_SCT	SCTimer/PWM	NO	NO	NO
13	WAKEUP_CTIMER3	Standard Counter/Timer 3	YES	NO	NO
14	WAKEUP_FLEXCOMM0	USART,SPI,I2C,I2S	YES	NO	NO
15	WAKEUP_FLEXCOMM1	USART,SPI,I2C,I2S	YES	NO	NO
16	WAKEUP_FLEXCOMM2	USART,SPI,I2C,I2S	YES	NO	NO
17	WAKEUP_FLEXCOMM3	USART,SPI,I2C,I2S	YES	YES	NO
18	WAKEUP_FLEXCOMM4	USART,SPI,I2C,I2S	YES	NO	NO
19	WAKEUP_FLEXCOMM5	USART,SPI,I2C,I2S	YES	NO	NO
20	WAKEUP_FLEXCOMM6	USART,SPI,I2C,I2S	YES	NO	NO
21	WAKEUP_FLEXCOMM7	USART,SPI,I2C,I2S	YES	NO	NO
22	WAKEUP_ADC	Genaral Purpose ADC	NO	NO	NO
23					
24	WAKEUP_ACMP_CAPT	Analog Comparator	YES	YES	NO
25					
26					
27	WAKEUP_USB0_NEEDCLK	USB Full Speed	YES	NO	NO
28	WAKEUP_USB0	USB Full Speed	YES	NO	NO
29	WAKEUP_RTC_LITE_ALARM_WAKEUP	RTC	YES	YES	YES
30	WAKEUP_EZH_ARCH_B	EZH Co-processor	NO	NO	NO
31	WAKEUP_WAKEUP_MAILBOX	Mailbox Interrupt	NO	NO	NO

Note: *See parameter* **wakeup_interrupts** *of Low Power API described here after.*

List of wake-up sources per Low Power Modes (2-2)

Bit	Wake-up Source	Description	DEEP-SLEEP	POWER-DOWN	DEEP POWER-DOWN
32	WAKEUP_GPIO_INT0_4		YES	NO	NO
33	WAKEUP_GPIO_INT0_5		YES	NO	NO
34	WAKEUP_GPIO_INT0_6		YES	NO	NO
35	WAKEUP_GPIO_INT0_7		YES	NO	NO
36	WAKEUP_CTIMER2	Standard Counter/Timer 2	YES	NO	NO
37	WAKEUP_CTIMER4	Standard Counter/Timer 4	YES	NO	NO
38	WAKEUP_OS_EVENT_TIMER	OS Event Timer	YES	YES	YES
39					
40					
41					
42	WAKEUP_SDIO	SDIO Controller Interrupt	NO	NO	NO
43					
44					
45					
46					
47	WAKEUP_USB1	USB High Speed	YES	NO	NO
48	WAKEUP_USB1_NEEDCLK	USB High Speed	YES	NO	NO
49	WAKEUP_SEC_HYPERVISOR_CALL	Hypervisor Security Violation	NO	NO	NO
50	WAKEUP_SEC_GPIO_INT0_0	Secure GPIO	YES	NO	NO
51	WAKEUP_SEC_GPIO_INT0_1	Secure GPIO	YES	NO	NO
52	WAKEUP_PLU	Programmable Logic	YES	NO	NO
53	WAKEUP_SEC_VIO	Security Violation	NO	NO	NO
54	WAKEUP_SHA	HASH-AES256	NO	NO	NO
55	WAKEUP_CASPER	CASPER	NO	NO	NO
56	WAKEUP_PUFF	Physically Unclonable Function	NO	NO	NO
57	WAKEUP_PQ	Power Quad	NO	NO	NO
58	WAKEUP_SDMA1	Secure System DMA	YES	NO	NO
59	WAKEUP_LSPI_HS	High Speed SPI	YES	NO	NO
63:60					

LOW POWER API

SLEEP Low Power API

Function prototype	API description
void POWER_EnterSleep(void);	• This API makes CPU to enter sleep mode.
void POWER_EnterDeepSleep (uint32_t exclude_from_pd, uint32_t sram_retention_ctrl, uint64_t wakeup_interrupts, uint32_t hardware_wake_ctrl);	• This API configures the chip then enters deep-sleep mode. • Configure the deep-sleep low power mode: allows controlling which peripherals are powered up and which SRAM instances are in retention state in deep-sleep.
Void POWER_EnterPowerDown (uint32_t exclude_from_pd, uint32_t sram_retention_ctrl, uint64_t wakeup_interrupts, uint32_t cpu_retention_ctrl);	• This API configures the chip then enters power-down mode. • Configure the power-down low power mode: allows controlling which peripherals are powered up , which SRAM instances and CPU are in retention state in power-down.
Void POWER_EnterDeepPowerDown (uint32_t exclude_from_pd, uint32_t sram_retention_ctrl, uint64_t wakeup_interrupts, uint32_t wakeup_io_ctrl);	• This API configures the chip then enters deep power-down mode. • Configure the deep power-down low power mode: allows controlling which peripherals are powered up and which SRAM instances are in retention state in deep power-down.

SLEEP

SLEEP Low Power API

```
void POWER_EnterSleep (void)
{
    /* Configure the Cortex-M33 in Light Sleep mode */
    SCB->SCR = SCB->SCR & ~SCB_SCR_SLEEPDEEP_Msk;
    /* Enter in low power mode */
    __WFI();
}
```

Note: The System control register (SCR) controls entry to and exit from a low power state , allowing to put the ARM core into sleep mode or the entire system in deep-sleep or power-down mode.

DEEP-SLEEP

DEEP-SLEEP Low Power API: « sram_retention_ctrl » param

- Defines which SRAM instances will be put in “retention” mode during deep-sleep.
- SRAM “retention mode” SRAM instances in “retention mode” do not lose their content but they cannot be involved in a DMA transfer during deep-sleep.

Bit	SRAM instance	Description	Value
0	RAM_X0 (16 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
1	RAM_X1 (8 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
2	RAM_X2 (4 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
3	RAM_X3 (4 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
4	RAM_00 (32 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
5	RAM_01 (32 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
6	RAM_10 (64 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
7	RAM_20 (64 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
8	RAM_30 (32 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
9	RAM_31 (32 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
10	RAM_40 (4 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
11	RAM_41 (4 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
12	RAM_42 (4 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
13	RAM_43 (4 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
14	RAM_USB (16 KBytes)		0 : SRAM keeps current state during Deep-Sleep 1 : SRAM in Retention mode during Deep-Sleep
31:15			

DEEP-SLEEP Low Power API: « hardware_wake_ctrl » param

- Provides the possibility to all FLEXCOMMs and to the High Speed SPI to have DMA service during Deep-sleep without waking up entire device. The IC stays in Deep-sleep mode until a DMA event occurs to wake-up the device from Deep-sleep mode.
- These wake-ups are based on FLEXCOMMs/High Speed SPI peripherals FIFO levels.

Bit	Symbol	Description	Value
0	HWWAKE_FORCED	Force peripheral clocking to stay on during deep-sleep mode.	*Shall always be set to '0'*
1	HWWAKE_PERIPHERALS	Wake for Flexcomms. Any Flexcomm FIFO reaching the level specified by its own TXLVL will cause peripheral clocking to wake up temporarily while the related status is asserted	0 : Disabled 1 : Enabled
2			
3	HWWAKE_SDMA0	Wake for DMA0. DMA0 being busy will cause peripheral clocking to remain running until DMA completes. Used in conjonction with HWWAKE_PERIPHERALS	0 : Disabled 1 : Enabled
4			
5	HWWAKE_SDMA1	Wake for DMA1. DMA0 being busy will cause peripheral clocking to remain running until DMA completes. Used in conjonction with HWWAKE_PERIPHERALS	0 : Disabled 1 : Enabled
6:30			
31	HWWAKE_ENABLE_FRO192M	Need to be set if FRO192M is disable - via PDCTRL0 - in Deep Sleep mode and any of HWWAKE_PERIPHERALS, HWWAKE_SDMA0 or HWWAKE_SDMA1 is set.	0 : Disabled 1 : Enabled

DEEP-SLEEP Low Power API usage : example

- **Enter Deep-sleep mode with wake up by System DMA0**

- 1 – **Configure any flexcomm in SPI receiver mode, and System DMA0 such that data received during Deep Sleep on SPI will be transferred to RAM_X2 by System DMA0; a wake-up event will be fired when the required number of data transfer by DMA0 is reached**
- 2 – **Set up low power API function parameters :**
  - exclude_from_pd = 0; /* All analog modules shut down */
  - sram_retention_ctrl = 0x7FFF & (~LOWPOWER_SRAMRETCTRL_RETEN_RAMX2); /* All RAM instances will be retained, except RAM_X2 RAM instance which will be kept in Active state because it is involved in DMA transfer during Deep-sleep */
  - wakeup_interrupts = WAKEUP_SDMA0; /* System DMA0 */
  - hardware_wake_ctrl = LOWPOWER_HWWAKE_ENABLE_FRO192M | LOWPOWER_HWWAKE_SDMA0 | LOWPOWER_HWWAKE_PERIPHERALS; /* Allow DMA transfer without leaving Deep Sleep mode */
- 3 – **Enter Deep-sleep :**
  - POWER_EnterDeepSleep (exclude_from_pd,sram_retention_ctrl, wakeup_interrupts, hardware_wake_ctrl);

POWER-DOWN

POWER-DOWN Low Power API usage : example

- **Enter Power-down mode with wake up by Flexcomm3 (SPI or I²C), CPU state retained**

- 1 – Configure the Flexcomm3 as SPI or I²C, in receiver mode
- 2 – **Set up low power API function parameters :**
  - exclude_from_pd = LOWPOWER_PDCTRL0_PDEN_LDOMEM; /* Memory retention required */
  - sram_retention_ctrl = LOWPOWER_SRAMRETCTRL_RETEN_RAMX2 | LOWPOWER_SRAMRETCTRL_RETEN_RAMX3; /* RAM instances RAM_X2 & RAM_X3 content will be retained, because they contain CPU stacks and variables for instance */
  - wakeup_interrupts = WAKEUP_FLEXCOMM3; /* Flexcomm 3 */
  - cpu_rentention_ctrl = 1; /* CPU state retention enabled */
- 3 – **Enter power-down :**
  - POWER_EnterPowerDown(exclude_from_pd,sram_retention_ctrl, wakeup_interrupts, cpu_rentention_ctrl);

- ***Note**: In case UART is used as wake-up source in Flexcomm3, a 32-KHz clock source need to be enabled inside the IC. The unique baudrate supported is 9600 Baud.*

DEEP POWER-DOWN

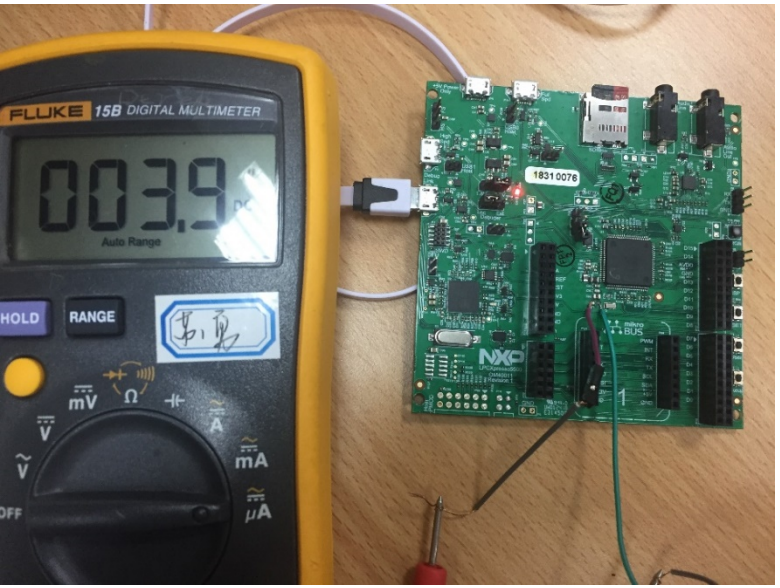
DEEP POWER-DOWN Low Power API usage : example

- **Enter Deep power-down mode with wake up by RTC, using FRO32KHz as clock source, content of RAM_X2 and RAM_X3 retained**

- 1 – **Configure RTC and FRO 32KHz**
- 2 – **Set up low power API function parameters :**
  - exclude_from_pd = LOWPOWER_PDCTRL0_PDEN_FRO32K | LOWPOWER_PDCTRL0_PDEN_LDOMEM; /* The RTC will use the FRO 32 KHz as clock source */
  - sram_retention_ctrl = LOWPOWER_SRAMRETCTRL_RETEN_RAMX2 | LOWPOWER_SRAMRETCTRL_RETEN_RAMX3; /* RAM instances RAM_X2 & RAM_X3 content will be retained, all other RAM instances will lose their content */
  - wakeup_interrupts = WAKEUP_RTC_LITE_ALARM_WAKEUP; /* RTC */
  - wakeup_io_ctrl = 0; /* All wake-up pin disabled */
- 3 – **Enter Deep power-down :**
  - POWER_EnterDeepPowerDown (exclude_from_pd,sram_retention_ctrl, wakeup_interrupts, wakeup_io_ctrl);

LPC55XX POWER MEASUREMENTS

LPC55xx Low Power Measurements



Power Mode	Current VBAT@3V	Wake Time	Description
Active mode (Core @1.1V, @96Mhz)	32 uA/MHz	-	Core running @1.1V, 96MHz FRO
Deepsleep (All 320kB SRAM retained)	92uA	100us	Logic is powered but @0.9V, Flash/ROM off, SRAM/Padring on, Clocks off (except wakeup sources)
Powerdown with Retention (with 8kB SRAM retained)	3.5uA	325us	DCDC off, RAM/Padring on, CPU0 and some periph. retained, PD_System powered , PD_core is off
Deep Powerdown (with 4kB retained)	0.7uA	15ms*	Only wakeup (4x) and reset pads, and PD_AO (PMC, RTC, OS timer, PMU) are powered,Rest of the logic is off

*Due to SW boot

LPC55XX POWER REDUCTION TIPS

LPC55xx Power Reduction Tips

keyword	Description
Clock	Disable the clock for unnecessary module during low power mode. That means, programmer can disable the clocks before entering the low power mode and re-enable them after exiting the low power mode when necessary.
Function	Disable the function for unnecessary part of a module when other part would keep working in low power mode. At the most time, more powerful function means more power consumption. For example, disable the digital function for the unnecessary pin mux, and so on.
Pin state	Set the proper pin state (direction and logic level) according to the actual application hardware. Otherwise, the pin current would be activated unexpectedly waste some energy.
Actual application	Other low power consideration based on the actual application hardware.
Configuration	In order to meet typedef power consumption of UM , Please configure MCU under the following conditions. ● Configure all pins as GPIO with pull-up resistor disabled in the IOCON block. ● Configure GPIO pins as outputs using the GPIO DIR register. ● Write 1 to the GPIO CLR register to drive the outputs LOW. ● All peripherals disabled.
Hardware	Hardware (board)

恩智浦MCU加油站

- **恩智浦工程师原创技术分享**
- 欢迎关注，欢迎投稿

SECURE CONNECTIONS
FOR A SMARTER WORLD