

STM32WB HSEM & IPCC Initialization

BLE_p2p Server

① main.c

HAL_Init();

HAL_MspInit();

RCC → AHB3ENR |= RCC_AHB3ENR_HSEMEN; // enables HW semaphore.

HAL_NVIC_SetPriority(HSEM_IRQn, 0, 0); // HSEM_IRQn = 46

HAL_NVIC_EnableIRQ(HSEM_IRQn);

Reset_Device();

Reset_IPCC();

RCC → AHB3ENR |= RCC_AHB3ENR_IPCCEN; // enables IPCC

IPCC → C1SCR |= (0x1 | 0x2 | 0x4 | 0x8 | 0x10 | 0x20); // Clear IPCC regs CPU1

IPCC → C2SCR |= (0x1 | 0x2 | 0x4 | 0x8 | 0x10 | 0x20); // Clear IPCC regs CPU2

IPCC → C1MR |= ((0x1 | ... | 0x20) << IPCC_C1MR_CHIFM_Pos); // Mask TX interrupts CPU1

IPCC → C2MR |= ((0x1 | ... | 0x20) << IPCC_C2MR_CHIFM_Pos); // Mask TX interrupts CPU2

IPCC → C1MR |= (0x1 | ... | 0x20); // Mask RX interrupts CPU1

IPCC → C2MR |= (0x1 | ... | 0x20); // Mask RX interrupts CPU2

Config_HSEC();

p_otp = (OTP_ID0_t *) OTP_Read(0);

RCC → HSECR |= HSE_CONTROL_UNLOCK_KEY; // = 0xCAFECAFE

MODIFY_REG(RCC → HSECR, RCC_HSECR_HSETUNE,
p_otp → hse_tuning << RCC_HSECR_HSETUNE_Pos);

SystemClock_Config();

PeriphClock_Config();

Init_Exti();

EXTI → IMR1 |= ~0; // 0-31

EXTI → IMR2 |= (// 32-63

(~0) & (EXTI_IMR2_IM36 | EXTI_IMR2_IM38) // except IPCC (36) and HSEM (38)

);

MX_GPIO_Init();

MX_DMA_Init();

② Research what is configured for DMA

MX_RTC_Init();

APPE_Init();

SystemPower_Config();

MODIFY_REG(PWR → C2CR1, 0x00, PWR_C2CR1_LPMS, PWR_C2CR1_LPMS-2); // Reset CPU2 value
// before standing C
// with C2BOOT 0x04

```
HW_TS_Init(); // Initialize the Time Server
APPD_Init(); // Debug initializations
// LED and Button inits
```

} Not directly related to BLE stack

```
LED_Init();
Button_Init();
app_TL_Init(); // Transport layers initialization
// Reference table initialization
```

```
TL_Init();
TL_RefTable.* = *; // sets all attributes of mapping table // TODO: NB-NEM1 addit
// objects
```

```
HW_IPCC_Init(); ①
```

```
RCC->AHB3ENR |= RCC_AHB3ENR_IPCCEN; // enables IPCC
```

```
IPCC->CICR |= IPCC_CICR_RXDIE; // enables RX occupied interrupt
```

```
IPCC->CICR |= IPCC_CICR_TXDIE; // enables TX free interrupt
```

```
NVIC_EnableIRQ(IPCC_CI_RX_IRQn); // = 44
```

```
NVIC_EnableIRQ(IPCC_CI_TX_IRQn); // = 45
```

// System channel initialization ② Research the whole system HCI middleware layer

```
TL_MM_Init();
// Configure TL_RefTable.p_mem_manager_table ③ Research all configuration defor
```

```
TL_Enable();
```

```
HW_TPCC_Enable(); ② // There is a good comment explaining the sequence
// in STM32-VDAN/Target/hw-ippc.c
```

```
EXTI->C2EMR2 |= EXTI_IMR2_IM41; // Enable EXTI Line Event request
// for Lines in range 32-63 for CP
// for Line 41
```

```
EXTI->RTSR2 |= EXTI_IMR2_IM41; // Enable EXTI Line Rising Edge Trigger
// for Lines in range 32-63
// for Line 41
```

```
--SEV(); // Set the interrupt event flag and send an event to the CPU2
```

```
--WFE(); // Clear the interrupt event flag
```

```
PWR->CR4 |= PWR_CR4_C2BOOT; // Boot CPU2 after reset or wakeup from
// stop or standby modes (by CPU1)
```