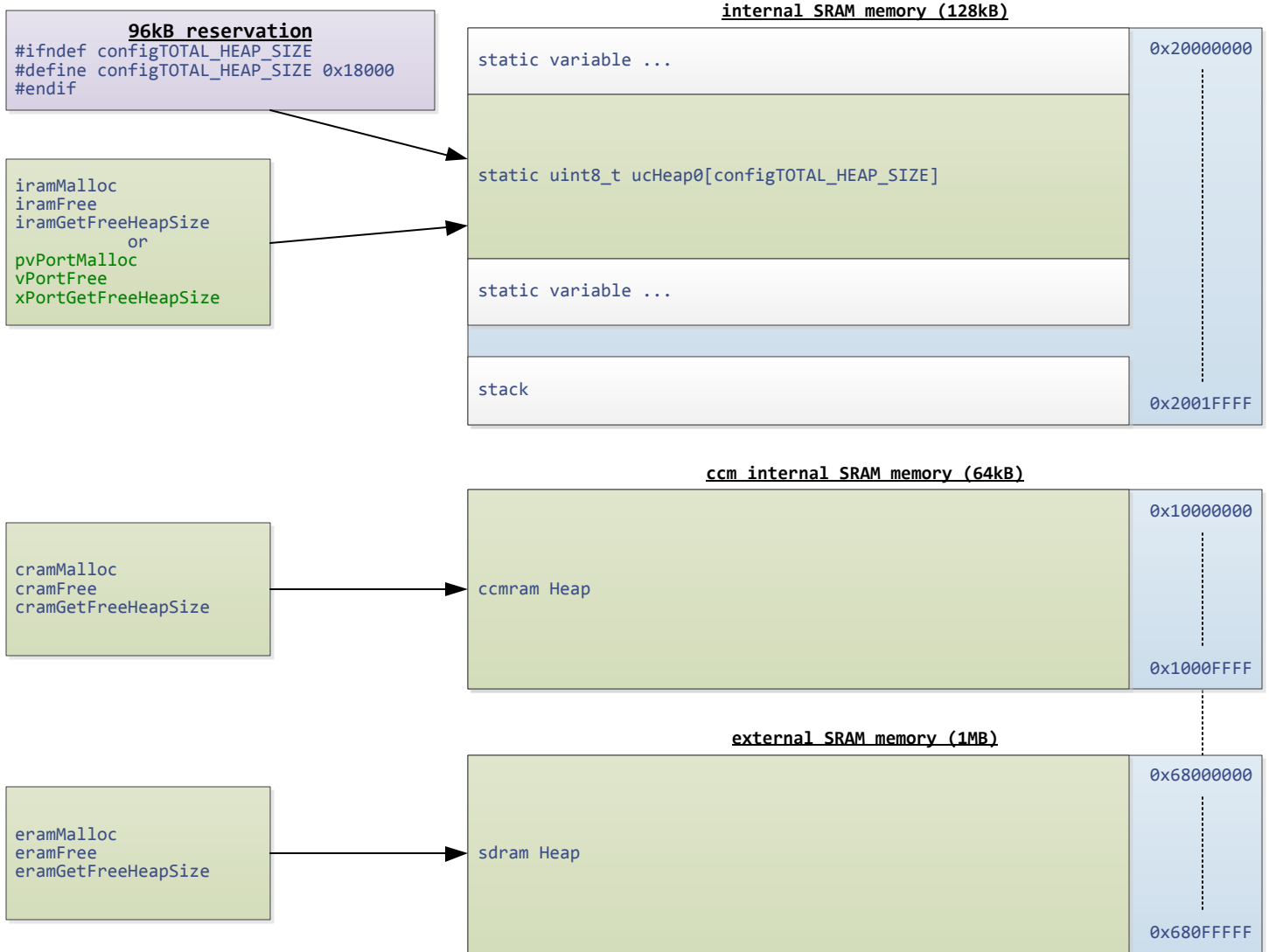


## Multi region heap memory divide example (stm32f407 black board with 1MB external sram)



### multi heap 4.h

```

/* Heap region number (1..6) */
#define HEAP_NUM 3

/* region 0 heap static reservation (if not used freertos -> check the free RAM size for setting) */
#ifndef configTOTAL_HEAP_SIZE
#define configTOTAL_HEAP_SIZE 0x18000
#endif

#define HEAP_0 ucHeap0[configTOTAL_HEAP_SIZE] // internal sram heap reservation if freertos used

/* regions table: address and size (internal sram region (0), ccmram region (1), external ram region (2) ) */
#define HEAP_REGIONS {{ (uint8_t *) &ucHeap0, sizeof(ucHeap0) }, \
                      { (uint8_t *) 0x10000000, 0x10000 }, \
                      { (uint8_t *) 0x68000000, 0x100000 } };

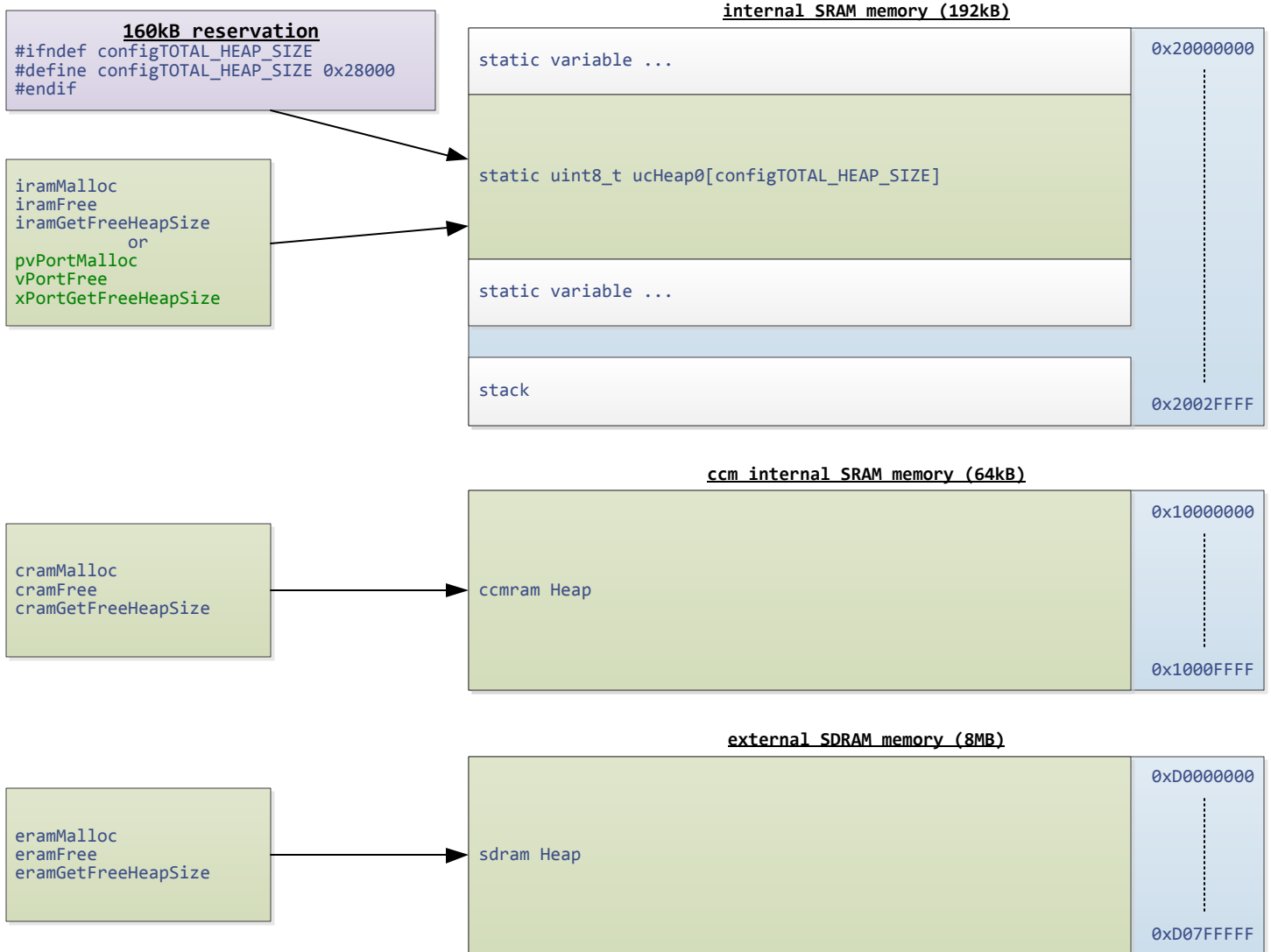
/* internal sram memory (region 0) procedures (*iramMalloc, iramFree, iramGetFreeHeapSize) */
#define iramMalloc(a) multiPortMalloc(0, a)
#define iramFree(a) multiPortFree(0, a)
#define iramGetFreeHeapSize(a) multiPortGetFreeHeapSize(0)
#define iramGetMinimumEverFreeHeapSize multiPortGetMinimumEverFreeHeapSize(0)

/* ccm ram memory (region 1) procedures (*cramMalloc, cramFree, cramGetFreeHeapSize) */
#define cramMalloc(a) multiPortMalloc(1, a)
#define cramFree(a) multiPortFree(1, a)
#define cramGetFreeHeapSize(a) multiPortGetFreeHeapSize(1)
#define cramGetMinimumEverFreeHeapSize multiPortGetMinimumEverFreeHeapSize(1)

/* external ram memory (region 2) procedures (*eramMalloc, eramFree, eramGetFreeHeapSize) */
#define eramMalloc(a) multiPortMalloc(2, a)
#define eramFree(a) multiPortFree(2, a)
#define eramGetFreeHeapSize(a) multiPortGetFreeHeapSize(2)
#define eramGetMinimumEverFreeHeapSize multiPortGetMinimumEverFreeHeapSize(2)

```

## Multi region heap memory divide example (stm32f429 discovery board with 8MB external sdram)



### multi heap 4.h

```

/* Heap region number (1..6) */
#define HEAP_NUM 3

/* region 0 heap static reservation (if not used freertos -> check the free RAM size for setting) */
#ifndef configTOTAL_HEAP_SIZE
#define configTOTAL_HEAP_SIZE 0x28000
#endif

#define HEAP_0 ucHeap0[configTOTAL_HEAP_SIZE] // internal sram heap reservation if freertos used

/* regions table: address and size (internal sram region (0), ccmram region (1), external ram region (2) ) */
#define HEAP_REGIONS {{ (uint8_t *) &ucHeap0, sizeof(ucHeap0) }, \
                      { (uint8_t *) 0x10000000, 0x10000 }, \
                      { (uint8_t *) 0xD0000000, 0x800000 } };

/* internal sram memory (region 0) procedures (*iramMalloc, iramFree, iramGetFreeHeapSize) */
#define iramMalloc(a) multiPortMalloc(0, a)
#define iramFree(a) multiPortFree(0, a)
#define iramGetFreeHeapSize(a) multiPortGetFreeHeapSize(0)
#define iramGetMinimumEverFreeHeapSize multiPortGetMinimumEverFreeHeapSize(0)

/* ccm ram memory (region 1) procedures (*cramMalloc, cramFree, cramGetFreeHeapSize) */
#define cramMalloc(a) multiPortMalloc(1, a)
#define cramFree(a) multiPortFree(1, a)
#define cramGetFreeHeapSize(a) multiPortGetFreeHeapSize(1)
#define cramGetMinimumEverFreeHeapSize multiPortGetMinimumEverFreeHeapSize(1)

/* external ram memory (region 2) procedures (*eramMalloc, eramFree, eramGetFreeHeapSize) */
#define eramMalloc(a) multiPortMalloc(2, a)
#define eramFree(a) multiPortFree(2, a)
#define eramGetFreeHeapSize(a) multiPortGetFreeHeapSize(2)
#define eramGetMinimumEverFreeHeapSize multiPortGetMinimumEverFreeHeapSize(2)

```