

2.3.5 Exceptions and Interrupts

The Cortex-M4F processor supports interrupts and system exceptions. The processor and the Nested Vectored Interrupt Controller (NVIC) prioritize and handle all exceptions. An exception changes the normal flow of software control. The processor uses Handler mode to handle all exceptions except for reset. See “Exception Entry and Return” on page 101 for more information.

The NVIC registers control interrupt handling. See “Nested Vectored Interrupt Controller (NVIC)” on page 117 for more information.

2.3.6 Data Types

The Cortex-M4F supports 32-bit words, 16-bit halfwords, and 8-bit bytes. The processor also supports 64-bit data transfer instructions. All instruction and data memory accesses are little endian. See “Memory Regions, Types and Attributes” on page 87 for more information.

2.4 Memory Model

This section describes the processor memory map, the behavior of memory accesses, and the bit-banding features. The processor has a fixed memory map that provides up to 4 GB of addressable memory.

The memory map for the TM4C123BH6ZRB controller is provided in Table 2-4 on page 84. In this manual, register addresses are given as a hexadecimal increment, relative to the module's base address as shown in the memory map.

The regions for SRAM and peripherals include bit-band regions. Bit-banding provides atomic operations to bit data (see “Bit-Banding” on page 89).

The processor reserves regions of the Private peripheral bus (PPB) address range for core peripheral registers (see “Cortex-M4 Peripherals” on page 115).

Note: Within the memory map, attempts to read or write addresses in reserved spaces result in a bus fault. In addition, attempts to write addresses in the flash range also result in a bus fault.

Table 2-4. Memory Map

Start	End	Description	For details, see page ...
Memory			
0x0000.0000	0x0003.FFFF	On-chip Flash	536
0x0004.0000	0x00FF.FFFF	Reserved	-
0x0100.0000	0x1FFF.FFFF	Reserved for ROM	522
0x2000.0000	0x2000.7FFF	Bit-banded on-chip SRAM	521
0x2000.8000	0x21FF.FFFF	Reserved	-
0x2200.0000	0x220F.FFFF	Bit-band alias of bit-banded on-chip SRAM starting at 0x2000.0000	521
0x2210.0000	0x3FFF.FFFF	Reserved	-
Peripherals			
0x4000.0000	0x4000.0FFF	Watchdog timer 0	781
0x4000.1000	0x4000.1FFF	Watchdog timer 1	781
0x4000.2000	0x4000.3FFF	Reserved	-
0x4000.4000	0x4000.4FFF	GPIO Port A	658

Table 2-4. Memory Map (continued)

Start	End	Description	For details, see page ...
0x4000.5000	0x4000.5FFF	GPIO Port B	658
0x4000.6000	0x4000.6FFF	GPIO Port C	658
0x4000.7000	0x4000.7FFF	GPIO Port D	658
0x4000.8000	0x4000.8FFF	SSI0	977
0x4000.9000	0x4000.9FFF	SSI1	977
0x4000.A000	0x4000.AFFF	SSI2	977
0x4000.B000	0x4000.BFFF	SSI3	977
0x4000.C000	0x4000.CFFF	UART0	914
0x4000.D000	0x4000.DFFF	UART1	914
0x4000.E000	0x4000.EFFF	UART2	914
0x4000.F000	0x4000.FFFF	UART3	914
0x4001.0000	0x4001.0FFF	UART4	914
0x4001.1000	0x4001.1FFF	UART5	914
0x4001.2000	0x4001.2FFF	UART6	914
0x4001.3000	0x4001.3FFF	UART7	914
0x4001.4000	0x4001.FFFF	Reserved	-
Peripherals			
0x4002.0000	0x4002.0FFF	I ² C 0	1027
0x4002.1000	0x4002.1FFF	I ² C 1	1027
0x4002.2000	0x4002.2FFF	I ² C 2	1027
0x4002.3000	0x4002.3FFF	I ² C 3	1027
0x4002.4000	0x4002.4FFF	GPIO Port E	658
0x4002.5000	0x4002.5FFF	GPIO Port F	658
0x4002.6000	0x4002.6FFF	GPIO Port G	658
0x4002.7000	0x4002.7FFF	GPIO Port H	658
0x4002.8000	0x4002.8FFF	PWM 0	1138
0x4002.9000	0x4002.9FFF	PWM 1	1138
0x4002.A000	0x4002.BFFF	Reserved	-
0x4002.C000	0x4002.CFFF	QE10	1209
0x4002.D000	0x4002.DFFF	QE11	1209
0x4002.E000	0x4002.FFFF	Reserved	-
0x4003.0000	0x4003.0FFF	16/32-bit Timer 0	730
0x4003.1000	0x4003.1FFF	16/32-bit Timer 1	730
0x4003.2000	0x4003.2FFF	16/32-bit Timer 2	730
0x4003.3000	0x4003.3FFF	16/32-bit Timer 3	730
0x4003.4000	0x4003.4FFF	16/32-bit Timer 4	730
0x4003.5000	0x4003.5FFF	16/32-bit Timer 5	730
0x4003.6000	0x4003.6FFF	32/64-bit Timer 0	730
0x4003.7000	0x4003.7FFF	32/64-bit Timer 1	730
0x4003.8000	0x4003.8FFF	ADC0	824
0x4003.9000	0x4003.9FFF	ADC1	824

Table 2-4. Memory Map (*continued*)

Start	End	Description	For details, see page ...
0x4003.A000	0x4003.BFFF	Reserved	-
0x4003.C000	0x4003.CFFF	Analog Comparators	1108
0x4003.D000	0x4003.DFFF	GPIO Port J	658
0x4003.E000	0x4003.FFFF	Reserved	-
0x4004.0000	0x4004.0FFF	CAN0 Controller	1077
0x4004.1000	0x4004.1FFF	CAN1 Controller	1077
0x4004.2000	0x4004.BFFF	Reserved	-
0x4004.C000	0x4004.CFFF	32/64-bit Timer 2	730
0x4004.D000	0x4004.DFFF	32/64-bit Timer 3	730
0x4004.E000	0x4004.EFFF	32/64-bit Timer 4	730
0x4004.F000	0x4004.FFFF	32/64-bit Timer 5	730
0x4005.0000	0x4005.7FFF	Reserved	-
0x4005.8000	0x4005.8FFF	GPIO Port A (AHB aperture)	658
0x4005.9000	0x4005.9FFF	GPIO Port B (AHB aperture)	658
0x4005.A000	0x4005.AFFF	GPIO Port C (AHB aperture)	658
0x4005.B000	0x4005.BFFF	GPIO Port D (AHB aperture)	658
0x4005.C000	0x4005.CFFF	GPIO Port E (AHB aperture)	658
0x4005.D000	0x4005.DFFF	GPIO Port F (AHB aperture)	658
0x4005.E000	0x4005.EFFF	GPIO Port G (AHB aperture)	658
0x4005.F000	0x4005.FFFF	GPIO Port H (AHB aperture)	658
0x4006.0000	0x4006.0FFF	GPIO Port J (AHB aperture)	658
0x4006.1000	0x4006.1FFF	GPIO Port K (AHB aperture)	658
0x4006.2000	0x4006.2FFF	GPIO Port L (AHB aperture)	658
0x4006.3000	0x4006.3FFF	GPIO Port M (AHB aperture)	658
0x4006.4000	0x4006.4FFF	GPIO Port N (AHB aperture)	658
0x4006.5000	0x4006.5FFF	GPIO Port P (AHB aperture)	658
0x4006.6000	0x4006.6FFF	GPIO Port Q (AHB aperture)	658
0x4006.7000	0x400A.EFFF	Reserved	-
0x400A.F000	0x400A.FFFF	EEPROM and Key Locker	554
0x400B.0000	0x400B.FFFF	Reserved	-
0x400C.0000	0x400C.0FFF	I ² C 4	1027
0x400C.1000	0x400C.1FFF	I ² C 5	1027
0x400C.2000	0x400F.8FFF	Reserved	-
0x400F.9000	0x400F.9FFF	System Exception Module	481
0x400F.A000	0x400F.BFFF	Reserved	-
0x400F.C000	0x400F.CFFF	Hibernation Module	502
0x400F.D000	0x400F.DFFF	Flash memory control	536
0x400F.E000	0x400F.EFFF	System control	229
0x400F.F000	0x400F.FFFF	μDMA	601
0x4010.0000	0x41FF.FFFF	Reserved	-
0x4200.0000	0x43FF.FFFF	Bit-banded alias of 0x4000.0000 through 0x400F.FFFF	-

Table 2-4. Memory Map (continued)

Start	End	Description	For details, see page ...
0x4400.0000	0xDFFF.FFFF	Reserved	-
Private Peripheral Bus			
0xE000.0000	0xE000.0FFF	Instrumentation Trace Macrocell (ITM)	63
0xE000.1000	0xE000.1FFF	Data Watchpoint and Trace (DWT)	63
0xE000.2000	0xE000.2FFF	Flash Patch and Breakpoint (FPB)	63
0xE000.3000	0xE000.DFFF	Reserved	-
0xE000.E000	0xE000.EFFF	Cortex-M4F Peripherals (SysTick, NVIC, MPU, FPU and SCB)	127
0xE000.F000	0xE003.FFFF	Reserved	-
0xE004.0000	0xE004.0FFF	Trace Port Interface Unit (TPIU)	64
0xE004.1000	0xE004.1FFF	Embedded Trace Macrocell (ETM)	63
0xE004.2000	0xFFFF.FFFF	Reserved	-

2.4.1 Memory Regions, Types and Attributes

The memory map and the programming of the MPU split the memory map into regions. Each region has a defined memory type, and some regions have additional memory attributes. The memory type and attributes determine the behavior of accesses to the region.

The memory types are:

- Normal: The processor can re-order transactions for efficiency and perform speculative reads.
- Device: The processor preserves transaction order relative to other transactions to Device or Strongly Ordered memory.
- Strongly Ordered: The processor preserves transaction order relative to all other transactions.

The different ordering requirements for Device and Strongly Ordered memory mean that the memory system can buffer a write to Device memory but must not buffer a write to Strongly Ordered memory.

An additional memory attribute is Execute Never (XN), which means the processor prevents instruction accesses. A fault exception is generated only on execution of an instruction executed from an XN region.

2.4.2 Memory System Ordering of Memory Accesses

For most memory accesses caused by explicit memory access instructions, the memory system does not guarantee that the order in which the accesses complete matches the program order of the instructions, providing the order does not affect the behavior of the instruction sequence. Normally, if correct program execution depends on two memory accesses completing in program order, software must insert a memory barrier instruction between the memory access instructions (see “Software Ordering of Memory Accesses” on page 88).

However, the memory system does guarantee ordering of accesses to Device and Strongly Ordered memory. For two memory access instructions A1 and A2, if both A1 and A2 are accesses to either Device or Strongly Ordered memory, and if A1 occurs before A2 in program order, A1 is always observed before A2.