Sunbeam Institute of Information Technology
Pune and Karad

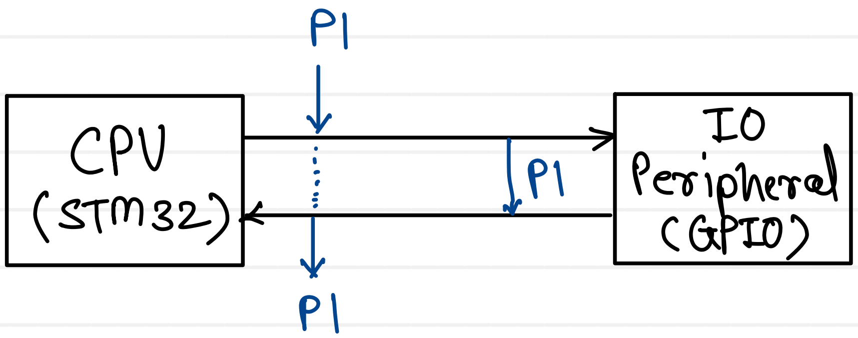
Module - Micro controller Programming and Interfacing

Trainer - Devendra Dhande

Email – devendra.dhande@sunbeaminfo.com

Types of I/O

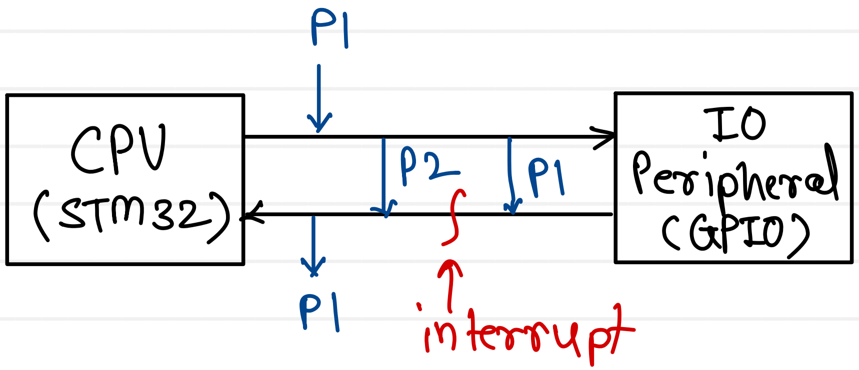
Synchronous



- CPU will wait for the completion of IO
- CPU will continuously check whether IO is completed or not.

- Hardware technique used is called as "polling".

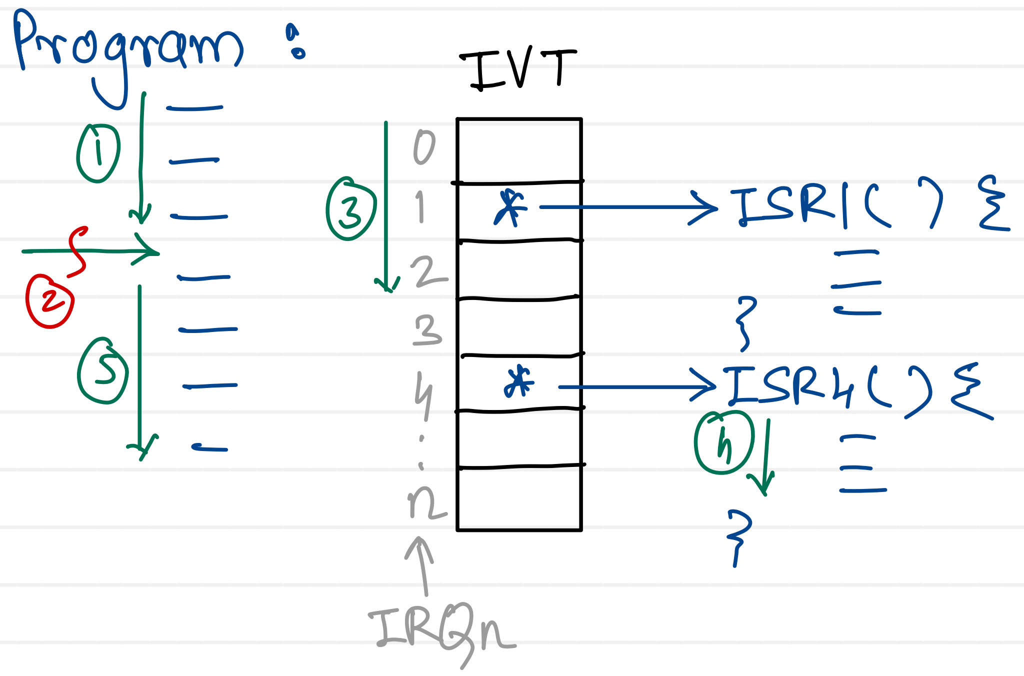
Asynchronous

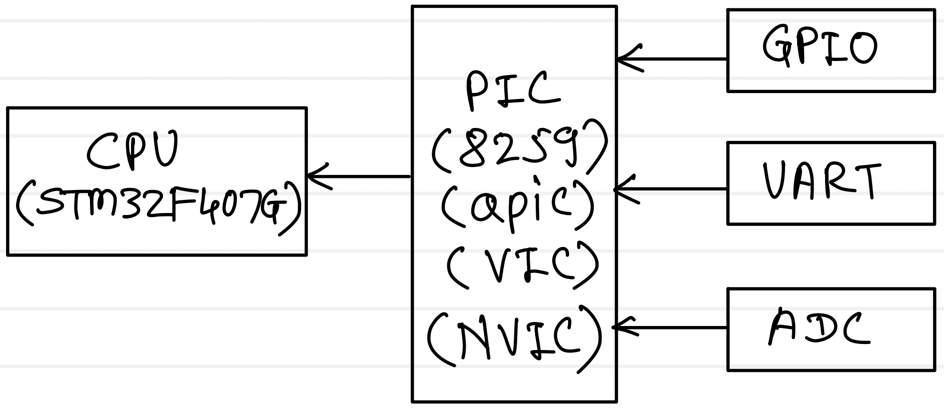


- CPU will not wait for completion of IO
- CPU will start executing another program / context.
- when IO will be completed, it will be informed to CPU by sending interrupt.
- on interrupt CPU will resume program

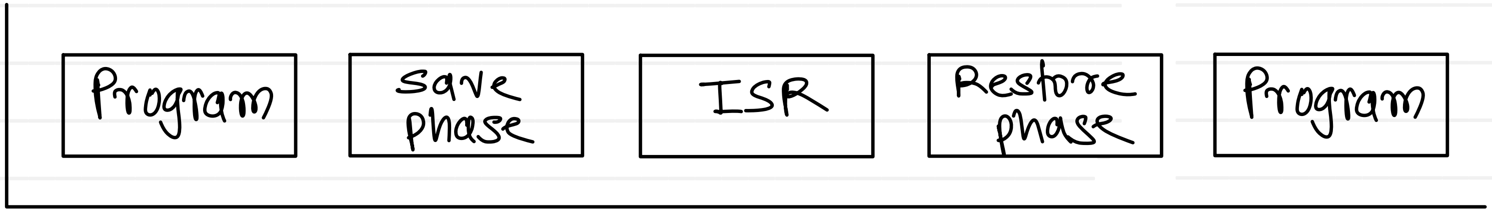
- Hardware technique used is called as "interrupt."

Interrupt Handling





- convey interrupts to CPU
- check priorities of interrupts



IRQn - interrupt number
↳ interrupt priority

ISR - Interrupt Service Routine
IVT - Interrupt Vector Table

Bit banding

Program :

① —
—

GPIOD->ODR |= BV(5); -> regr = 0000 0000 - r

② BV(5) = 0010 0000

③ f -> | = 0010 0000 - m

⑤ ODR = 0010 0000 - w

⑥ —
—

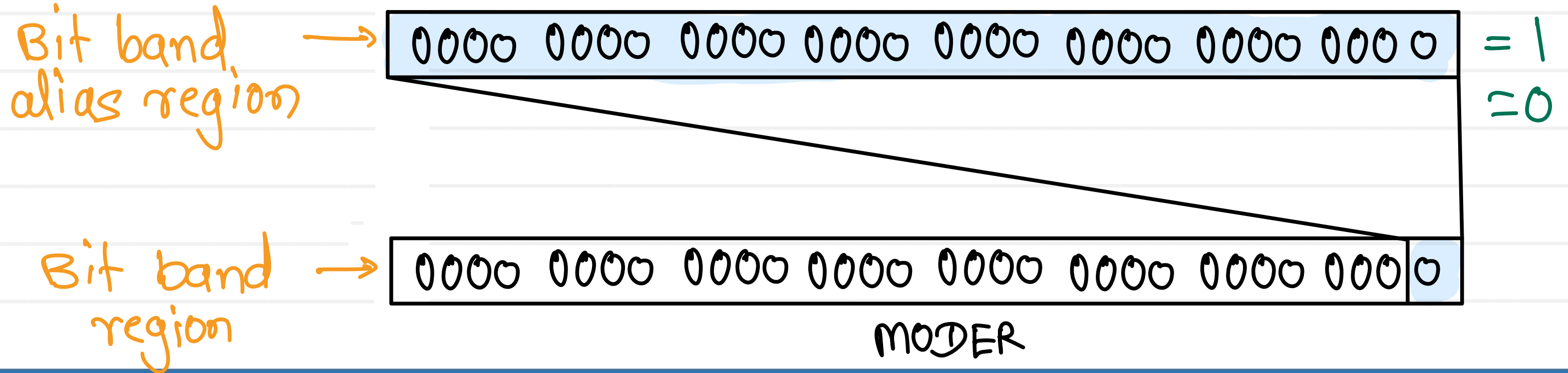
ISR :

GPIOD->ODR |= BV(7); -> regr = 0000 0000 - r

④ BV(7) = 1000 0000

| = 1000 0000 - m

ODR = 1000 0000 - w

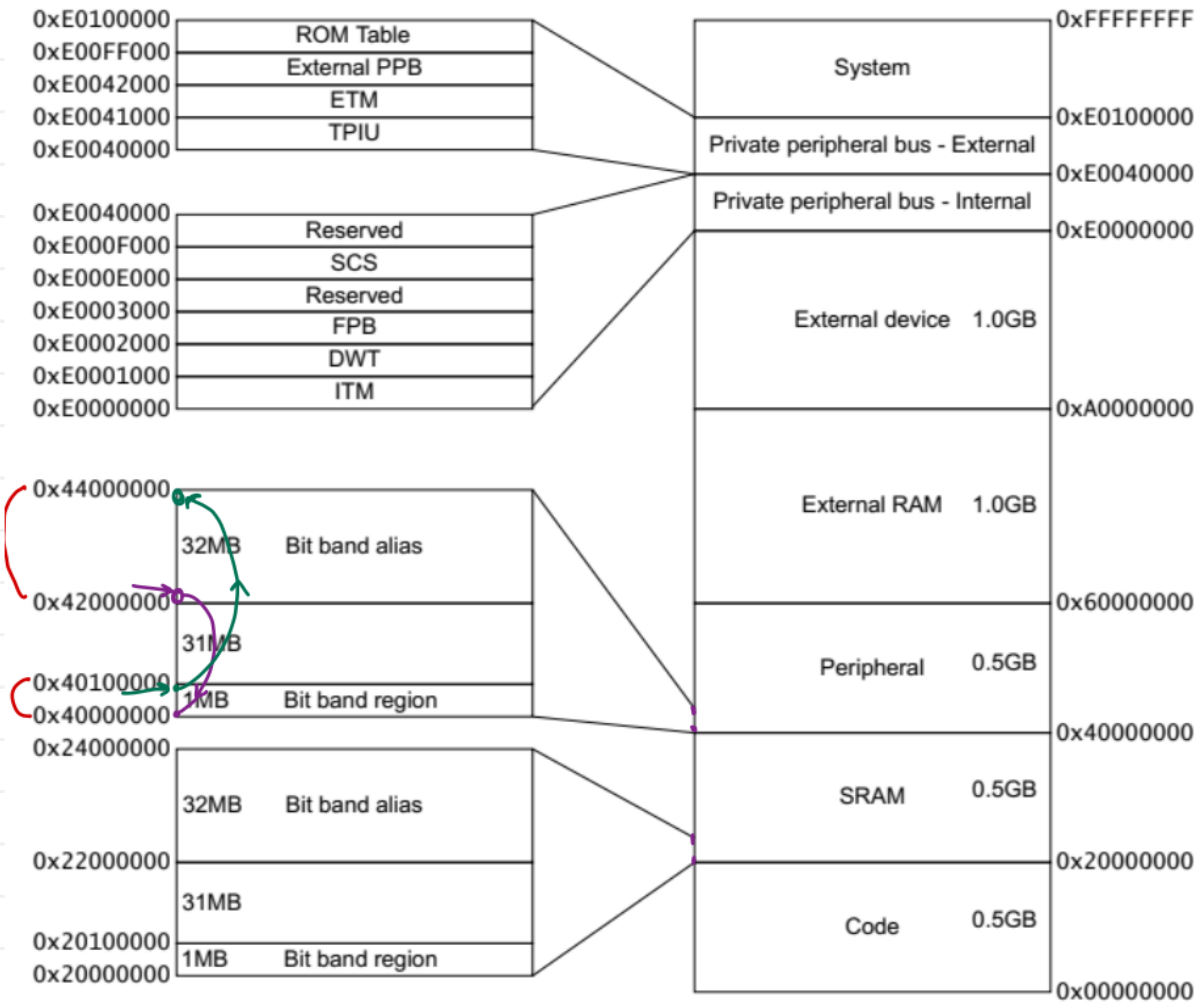


Bit banding

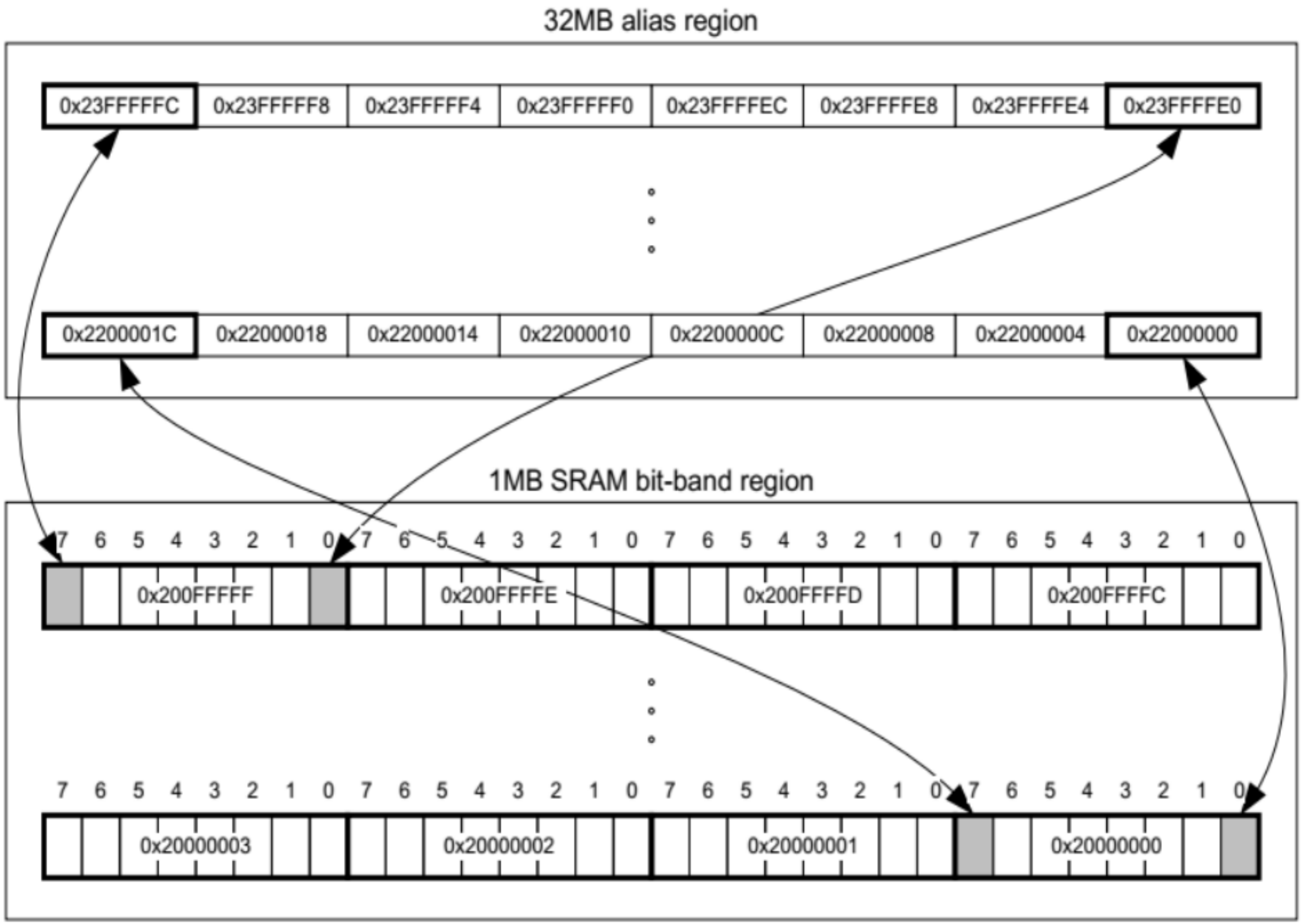
In Cortex M4,
There are two bit band regions
1. For peripheral memory (1Mb)
2. For SRAM (1Mb)

- 1 bit of bit band region is mapped with 32 bits (4 bytes) of bit band alias region.

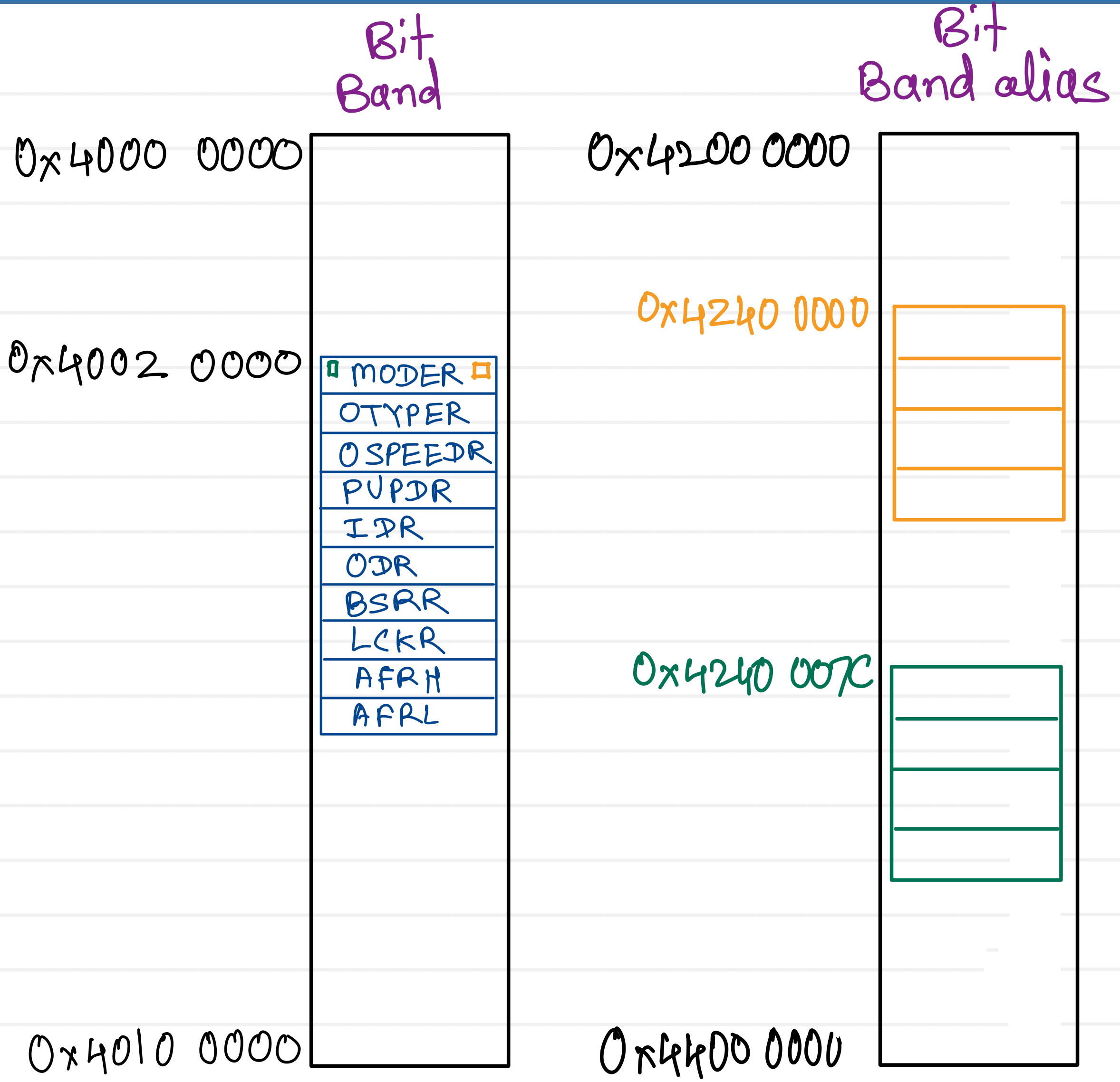
- Therefore, there are two bit band alias regions of 32 Mb



Bit banding


32MB alias region
0x23FFFFFC
0x23FFFFF8
0x23FFFFF4
0x23FFFFF0
0x23FFFFEC
0x23FFFFE8
0x23FFFFE4
0x23FFFFE0
0x2200001C
0x22000018
0x22000014
0x22000010
0x2200000C
0x22000008
0x22000004
0x22000000
1MB SRAM bit-band region
7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0
0x200FFFFF
0x200FFFFE
0x200FFFFD
0x200FFFFC
7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0 7 6 5 4 3 2 1 0
0x20000003
0x20000002
0x20000001
0x20000000

Bit banding



bit_word_addr = bit_band_base
+ (byte_offset * 32)
+ (bit_number * 4)

0x4200 0000		bit_band_base
0x0040 0000		byte_offset * 32
0x0000 0000		bit_number * 4
0x4240 0000		bit_word_addr

0x4200 0000		bit_band_base
0x0040 0060		byte_offset * 32
0x0000 001C		bit_number * 4
0x4240 007C		bit_word_addr

0x4000 0000

0x4002 0000	7 0
0x4002 0001	15 8
0x4002 0002	23 16
0x4002 0003	31 24

byte_offset = 0x4002 0000
  0x4000 0000
  ———————
  0x0002 0000
  <<5 => *32
  ———————
  0x0040 0000

byte_offset = 0x4002 0003
  0x4000 0000
  ———————
  0x0002 0003
  <<5
  ———————
  0x0040 0060

7*4 = 7<<2 = 0000 0111
0001 1100 = 0x1C

Thank you!!!

Devendra Dhande

devendra.dhande@sunbeaminfo.com