GPIO as input

- Enable GPIO clock (AHB1ENR)
- Select GPIO mode as input
- Select GPIO speed
- Set pull-up or pull-down resistor
- Read pin state

GPIO BSRR

- LED_GPIO->ODR |= BV(pin);
    - LED_GPIO->BSRR = BV(pin);
- LED_GPIO->ODR &= ~BV(pin);
    - LED_GPIO->BSRR = BV(pin + 16);

ARM7

- Cortex-M have all GPIO pins as Fast GPIO.
- ARM-7 have GPIO pins as configurable i.e. Legacy or Fast GPIO.
- Older ARM have only Legacy GPIO pins.

Difference - Legacy GPIO vs Fast GPIO

- Legacy GPIO: Connected to Peripheral bus (APB).
- Fast GPIO: Connectect to ARM Core bus (AHB).
- Legacy GPIO: Work at lower speed i.e. peripheral speed.
- Fast GPIO: Work at higher speed i.e. ARM core speed.
- Legacy GPIO: Word (32-bit) accessible.
- Fast GPIO: Word (32-bit), Half-word (16-bit), or Byte (8-bit) accessible.
- Legacy GPIO: No mask/lock feature.
- Fast GPIO: Mask/Lock feature to avoid accidental changes in GPIO pins.

Bit Banding

- The System bus interface contains logic that controls bit-band accesses as follows:
    - It remaps bit-band alias addresses to the bit-band region.
    - For reads, it extracts the requested bit from the read byte, and returns this in the LSB of the read data returned to the core.
    - For writes, it converts the write to an atomic read-modify-write operation.
    - The processor does not stall during bit-band operations unless it attempts to access the System bus while the bit-band operation is being carried out.

Bit-banging (is not Bit-banding)

- Informally bit-banging is any method of data transmission that employs software as a substitute for dedicated hardware to generate transmitted signals or process received signals.
- Frequently switching GPIO pin state as per protocol requirement.

Bit-banding (Cortex-M architecture feature)

- Bit-banding maps each bit of a register to a word in the address space
- Typically two bit-band regions (each of 1 MB) are available on CM3/CM4 devices, which are mapped to corresponding bit-band alias regions (each of 32 MB).
- Region1 - SRAM:
  - 0x20000000 - 0x22000000 (alias)
- Region2 - Peripherals:
  - 0x40000000 - 0x42000000 (alias)
- bit_word_addr = bit_band_alias_base + (byte_offset * 32) + (bit_number * 4);

```
#define BB_BASE 0x40000000UL
#define BB_ALS_BASE 0x42000000UL

//#define BB_ALS_ADDR(regr,bit) (BB_ALS_BASE + ((regr-BB_BASE) * 32) + (bit * 4))
#define BB_ALS_ADDR(regr,bit) (BB_ALS_BASE + ((regr-BB_BASE) << 5) + (bit << 2))
#define BB_ALS(regr,bit) (*(uint32_t*)BB_ALS_ADDR((uint32_t)regr,bit))
```