## **GPIO\_INPUT**

Main.c

|  |
| --- |
|  |

BML\_GPIO.c

(Double click to open)

|  |
| --- |
|  |

BML\_GPIO.h

(Double click to open)

|  |
| --- |
|  |

***GPIO Functions:-***

**gpio\_read(port, pinNumber)**

***How to use Functions:-***

Mostly all functions are same from previous document except here, we only have used the INPUT\_MODE to configure the PIN as input as to read if the pin is high or low but as we are choosing input then selecting PIN speed does not make sense so keep those default. There is one gpio read function. It returns a ‘1’ if the pin is high and ‘0’ if low.

***How the function works?***

We are using a 8 bit variable which stores the bit we get from this – ‘(uint8\_t)((port->IDR >> pinNumber) & 0x00000001)’. What this line of code do is that is shifts the IDR register to right side and ‘pinNumber’ bits are shifted to right. Example:- If the Input register on PIN 5 got an input, it will be High. The register will be 0000000000100000 and if we shift it by 5 bits then it will be 00...000001.If the PIN is 6 then again right shifting it 6 times will again result in 1 at same 0th position. Now we AND it with 0x00000001 then 1 & 1 will be ‘1’ and our flag will be ‘1’ or True. But if the input pin gets a low signal or ‘0’ then 0 & 1 will be ‘0’ and flag will be False and it will return the value.