



JOYSTICK

+



SPI



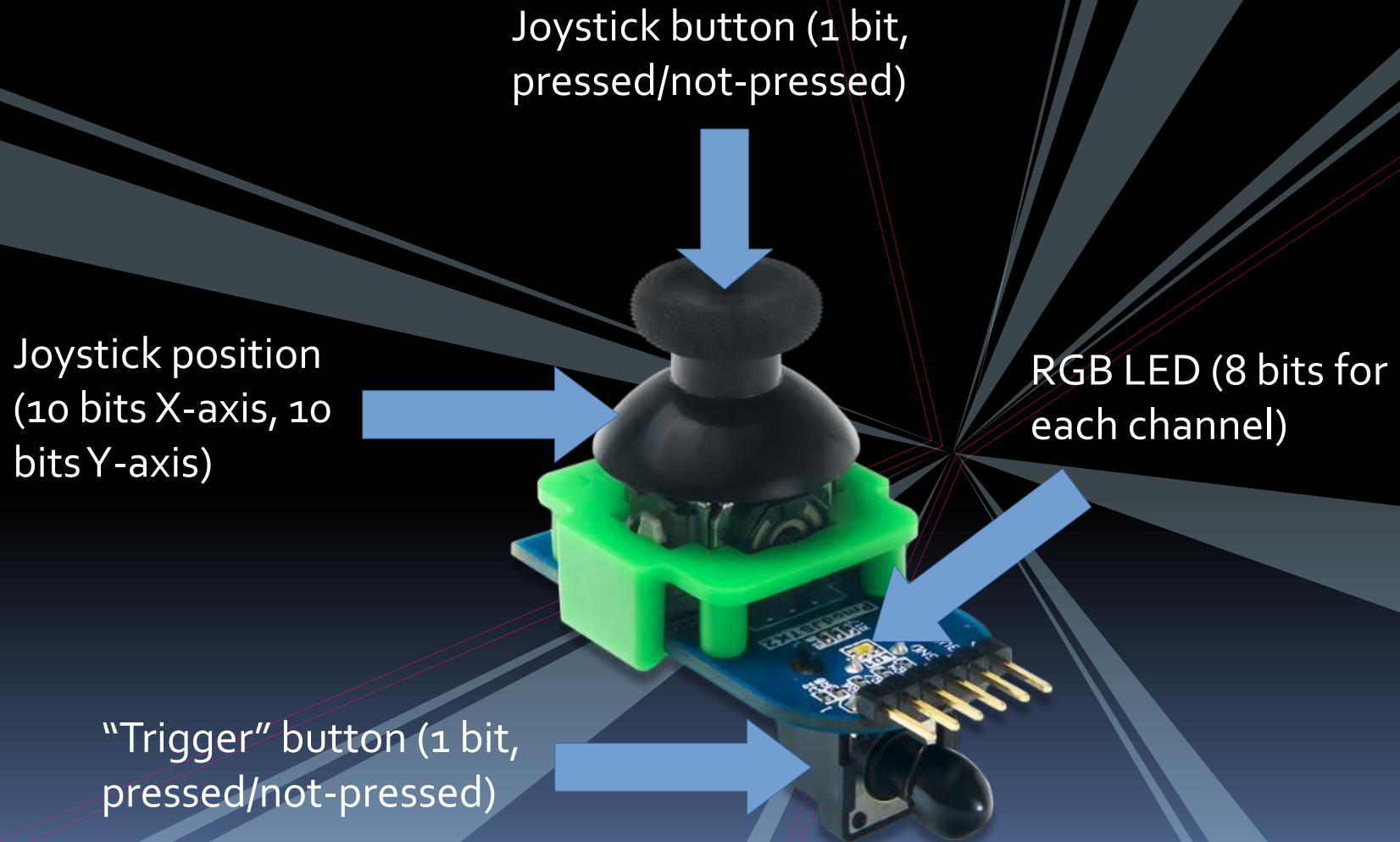
Digilent Pmod JSTK2

At the beginning of LAB2, we will give each one of you a Digilent Pmod JSTK2 module. This can be connected to your Basys3 board through the Pmod connectors.





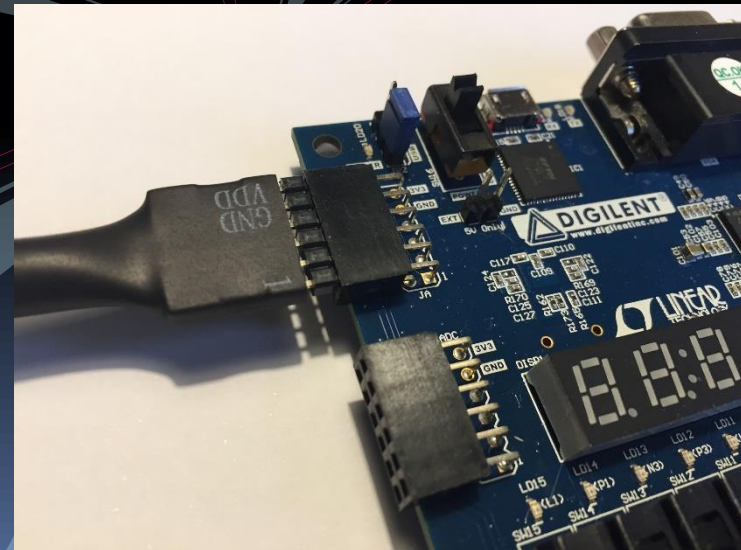
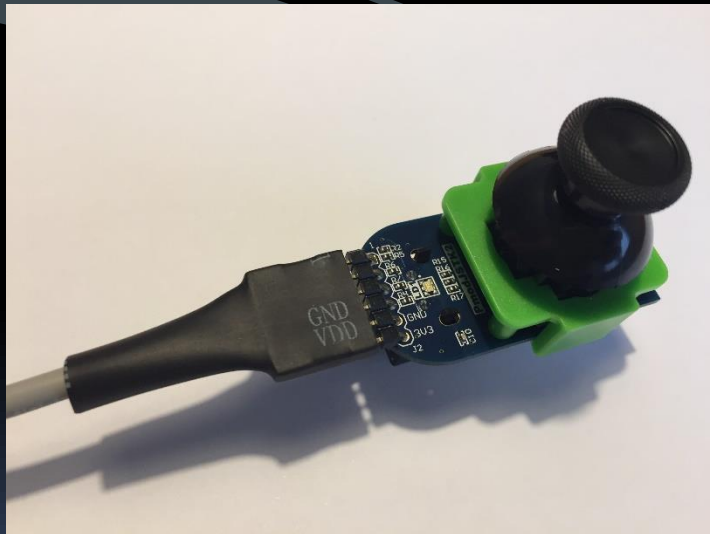
Digilent Pmod JSTK2 components





How to connect the Joystick

Connect the Joystick with the provided cable to "JA", top row, paying attention to the VCC and GND position.





JSTK2 interfacing

The Digilent Pmod JSTK2 module protocol is fully described in its [reference manual](#).

In short, it uses the SPI protocol to receive “commands” and send back the “readings” of the Joystick position and the buttons state.



SPI protocol

The Serial Peripheral Interface (SPI) is a very popular synchronous protocol for off-chip communication.

In its basic form it is composed by 4 signals:

- Serial CLock (SCLK)
- Master-Out Slave-In (MOSI)
- Master-In Slave-Out (MISO)
- Chip Select (CS)



SPI IP-Core

While SPI is a simple protocol, describing it in VHDL is not immediate.

To ease your work, we will give you a “AXI4-Stream SPI” IP-Core, similar to the UART one.





SPI IP-Core

You will find the details of its behavior in the README file. In short:

- Whatever you send to S_AXIS is sent to the SPI “slave”.
- Whatever is received from the SPI “slave” is sent to you through the M_AXIS interface.





SPI IP-Core

To respect the timing of the Digilent Pmod JSTK2 module (see [here](#)), use these parameters in the SPI IP-Core.

AXI4-Stream SPI Master (1.0)

Documentation IP Location

☐ Show disabled ports

Component Name

ack Frequency (Hz)

SPI parameters

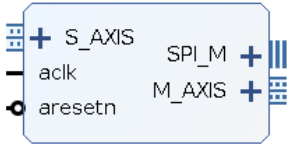
Desired SCLK frequency

CPOL

☒ 0
☐ 1

CPHA

☒ 0
☐ 1



```
graph LR
    S_AXIS[+] --- SPI_M[+]
    M_AXIS[+] --- SPI_M
    aclk[+] --- SPI_M
    aresetn[+] --- SPI_M
```



JSTK2 commands

Using this IP-Core, you can easily “talk” to the module, following the protocol described in the [reference manual](#).

We suggest you to use the `cmdSetLedRGB` command so that, in a single command, you can get the position of the joystick and the state of the button, and control the LED color.



JSTK2 commands - cmdSetLedRGB

cmdSetLedRGB

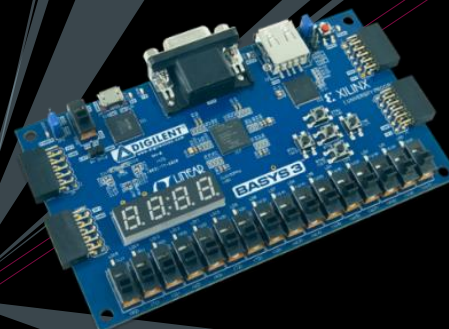
(0x84)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	0	0	0	0	1	0	0

Parameters

PARAM1 – Red LED duty cycle
PARAM2 – Green LED duty cycle
PARAM3 – Blue LED duty cycle
PARAM4 – ignored

Set the duty cycles for the Red, Green, and Blue LEDs.



	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
MOSI	COMMAND / 0	PARAM1 / DUMMY	PARAM2 / DUMMY	PARAM3 / DUMMY	PARAM4 / DUMMY
MISO	smpX (Low Byte)	smpX (High Byte)	smpY (Low Byte)	smpY (High Byte)	fsButtons

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
fsButtons	EXTPKT	0	0	0	0	0	TRIGGER	JOYSTICK

EXTPKT: Extended Packet Status Bit

1 = additional data corresponding to the command byte is available and may be retrieved after this byte
0 = standard response packet, no additional data follows this byte

TRIGGER: Trigger Button Status Bit

1 = trigger button is currently pressed
0 = trigger button is not being pressed

JOYSTICK: Joystick Center Button Status Bit

1 = joystick center button is currently pressed
0 = joystick center button is not being pressed