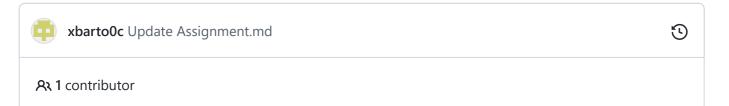xbarto0c / **Digital-electronics-2**   Public

Code   Issues   Pull requests   Actions   Projects   Wiki   Security   Insights   Settings

⌁ main ▾   ⋯

Digital-electronics-2 / **Labs** / **03-gpio** / **Assignment.md**

xbarto0c Update Assignment.md   ↺

⧑ **1** contributor

☰ 65 lines (47 sloc) │ 2.58 KB   ⋯

Lab 3: Jan Bartoň

Link to your `Digital-electronics-2` GitHub repository:

https://github.com/xbarto0c/Digital-electronics-2

Data types in C

1. Complete table.

Data type	Number of bits	Range	Description
uint8_t	8	0, 1, ..., 255	Unsigned 8-bit integer
int8_t	8	-128, ..., 127	Signed 8-bit integer
uint16_t	16	0, ..., 65 535	Unsigned 16-bit integer
int16_t	16	-32 768, ..., 32 767	Signed 16-bit integer

Data type	Number of bits	Range	Description
float	32	-3.4e+38, ..., 3.4e+38	Single-precision floating-point
void	0	0	A data type that has no values or operators and is used to represent nothing

GPIO library

1. In your words, describe the difference between the declaration and the definition of the function in C.
   When we are declaring a function, we are telling the compiler, that in the program, there will be a function that's going to have certain inputs and outputs. When we then define the function, we tell the compiler, what the function is supposed to do with the inputs to get the desired outputs.

2. Part of the C code listing with syntax highlighting, which toggles LEDs only if push button is pressed. Otherwise, the value of the LEDs does not change. Use function from your GPIO library. Let the push button is connected to port D:

```c
// Green LED at port B
GPIO_config_output(&DDRB, LED_GREEN);
GPIO_write_low(&PORTB, LED_GREEN);

// Configure the second LED at port C
GPIO_config_output(&DDRC, LED_BLUE);
GPIO_write_high(&PORTC, LED_BLUE);

// Configure Push button at port D and enable internal pull-up resistor
GPIO_config_input_pullup(&DDRD, PUSHBUTTON);

// Infinite loop
while (1)
{
    if (GPIO_read(&PIND, PUSHBUTTON))
        {
            // Pause several milliseconds
            _delay_ms(BLINK_DELAY);
            // WRITE YOUR CODE HERE
            GPIO_write_low(&PORTC, LED_BLUE );
            _delay_ms(BLINK_DELAY);
            GPIO_write_high(&PORTC, LED_BLUE);
            _delay_ms(BLINK_DELAY);
            GPIO_write_high(&PORTB, LED_GREEN);
            _delay_ms(BLINK_DELAY);
            GPIO_write_low(&PORTB, LED_GREEN);
```

```
            }
        }
```

Traffic light

1. Scheme of traffic light application with one red/yellow/green light for cars and one red/green light for pedestrians. Connect AVR device, LEDs, resistors, one push button (for pedestrians), and supply voltage. The image can be drawn on a computer or by hand. Always name all components and their values!