LLC-I2C Click

PID: MIKROE-3276
Weight: 24 g

LLC I2C click can be utilized as the level converter for logic signals, which makes it a very useful Click board™. The topology of this logic level conversion (LLC) circuit is perfectly suited for the bi-directional I2C communication. Although there are some specialized integrated circuits on the market, sometimes it is more convenient to have a simple solution made of just a few passive elements and four MOSFETs. Besides this, MOSFETs can withstand up to 20V across their terminals (12V $V_{CC}$), and they support far greater currents than typical ICs, making them less likely to fail. The LLC circuitry, also known as the level translator or level shifter, can be used in virtually any application that utilizes the I2C communication protocol and needs matching between the logic voltage levels.

It comes in the package which also includes the mikroSDK™ software and a library with all the functions. The Click board™ comes as a fully tested and approved prototype, making it a reliable device ready to use on the development board.

Table of contents

1. How does it work?
2. Specifications
3. Pinout diagram
4. Onboard settings and indicators
5. Software support
6. mikroSDK
7. Downloads

How does it work?

LLC I2C click does not use an integrated circuit, as already mentioned. With ICs avoided completely, there are some benefits gained: the overall cost of the LLC circuit is greatly reduced, a more robust MOSFET solution reduces the failure rate, and when powered off, both the low voltage and the high voltage sides are isolated from one another (by non-conductive MOSFETs). This type of circuit is sometimes referred to as a level shifting or level translating circuit and it is often necessary when the I2C slave device (typically a sensor IC) uses different logic voltage levels for the I2C communication than the master device, which is a host microcontroller in most cases. The conversion of this circuitry is bi-directional, which makes it suitable to be used with the I2C communication protocol.
The I2C protocol was first introduced by NXP Semiconductors (formerly Philips Semiconductors), in 1982. They also introduced an application note that explains the operation of the LLC circuit in more details. The circuit is divided into low-side and high-side sections for future reference, although the circuit is symmetrical and can be used in both directions.

When there is no communication going on, both gates and sources of the MOSFETs are pulled up to their specific reference voltage levels. This will turn OFF both MOSFETs since there is no gate-source voltage difference (e.g., $V_G = V_S = V_S$).

Since the I2C is operated by asserting its bus lines to a LOW logic level, when the source terminal of the MOSFET on one side (e.g., high-side) is driven to a LOW logic level, its $V_{GS}$ potential will rise since the gate voltage is fixed. When the $V_{GS}$ reaches the threshold voltage (1.2V typically for the used transistors), the MOSFET will turn ON, conducting current through the body diode of the opposite-side (low-side) MOSFET, which will become directly polarized. This mechanism can be used to convert signal levels in both directions, within the whole operating range of the used MOSFETs.

The reference voltage for the high-side can be selected by using the SMD jumper labeled as VCC SEL. The pull-up voltage for the high-side can be selected from the mikroBUS™ power rails, so it can be either 3.3V or 5V. For the low-side, an arbitrary reference voltage can be applied to the VSL pin of the J1 header, respecting the maximum voltage rating. J1 is the standard, 2.54mm pin-header. The low-side I2C bus pins are also routed to the J1 header, allowing an external device to be connected (using the standard wire-jumpers). As already mentioned, the low-side can actually use higher voltage levels than the master, but in most usage scenarios, it will be lower than the master, thus the terminology.

Specifications

<table>
<thead>
<tr>
<th>Type</th>
<th>Port expander</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applications</td>
<td>I2C logic level conversion circuit has its use in many different projects. It is used for the I2C logic voltage level matching between the slave and the master device</td>
</tr>
<tr>
<td>Key Features</td>
<td>Very low cost achieved using only passive elements, and four MOSFETs, ability to use an arbitrary voltage level conversion in the range</td>
</tr>
</tbody>
</table>

Downloaded from Arrow.com.
<table>
<thead>
<tr>
<th>Notes</th>
<th>Pin</th>
<th>AN</th>
<th>PWM</th>
<th>16</th>
<th>NC</th>
</tr>
</thead>
<tbody>
<tr>
<td>NC</td>
<td>2</td>
<td>RST</td>
<td>INT</td>
<td>15</td>
<td>NC</td>
</tr>
<tr>
<td>NC</td>
<td>3</td>
<td>CS</td>
<td>RX</td>
<td>14</td>
<td>NC</td>
</tr>
<tr>
<td>NC</td>
<td>4</td>
<td>SCK</td>
<td>TX</td>
<td>13</td>
<td>NC</td>
</tr>
<tr>
<td>NC</td>
<td>5</td>
<td>MISO</td>
<td>SCL</td>
<td>12</td>
<td>SCL I2C Clock</td>
</tr>
<tr>
<td>NC</td>
<td>6</td>
<td>MOSI</td>
<td>SDA</td>
<td>11</td>
<td>SDA I2C Data</td>
</tr>
<tr>
<td>Power supply</td>
<td>3V3</td>
<td>7</td>
<td>3.3V</td>
<td>10</td>
<td>5V Power supply</td>
</tr>
<tr>
<td>Ground</td>
<td>GND</td>
<td>8</td>
<td>GND</td>
<td>9</td>
<td>GND Ground</td>
</tr>
</tbody>
</table>

**Onboard settings and indicators**

<table>
<thead>
<tr>
<th>Label</th>
<th>Name</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PWR</td>
<td>PWR</td>
<td>-</td>
<td>Power LED indicator</td>
</tr>
<tr>
<td>VCC SEL</td>
<td>VCC SEL</td>
<td>Left</td>
<td>Logic voltage level selection (master): left position 3.3V, right position 5V</td>
</tr>
<tr>
<td>J1</td>
<td>-</td>
<td>-</td>
<td>I2C bus with the converted voltage levels (I2C slave)</td>
</tr>
</tbody>
</table>

**Software support**

We provide a library for the LLC I2C click on our LibStock page, as well as a demo application (example), developed using MikroElektronika compilers. The demo can run on all the main MikroElektronika development boards.

**Library Description**

The library contains functions for starting, writing data to I2C wires and reading data from I2C wires.

Key functions:

- `void i2c_llc_start()` - The function executes start condition on I2C wires.
void i2clc_write(uint8_t slaveAddress, uint8_t *pBuf, uint16_t nBytes, uint8_t endMode) - The function should execute write sequence on I2C wires.

void i2clc_read(uint8_t slaveAddress, uint8_t *pBuf, uint16_t nBytes, uint8_t endMode) - The function should execute read sequence on I2C wires.

Examples description

The application is composed of the three sections:

- System Initialization - Initializes I2C module
- Application Initialization - Initialization driver init
- Application Task - (code snippet) - Reads the temperature from the Thermo 7 click board and logs data to UART.

Note: The start configuration chip is required at the beginning of each program so that the chip wakes up and prepares for operation and measurement. What is included and set in the start-up function can be viewed in the help file.

```c
void applicationTask()
{
    // Thermo 7 measurement Temperature
    uint8_t writeReg[1] = {0};
    uint8_t readReg[2] = {0};

    float Temp_msb;
    uint8_t Temp_lsb;
    char tempText[50];

    writeReg[0] = 0x08;
    I2CLC_start();
    I2CLC_write(0x48, &writeReg[0], 1, I2CLC_END_MODE_RESTART);
    I2CLC_read(0x48, &readReg[0], 2, I2CLC_END_MODE_STOP);

    Temp_msb = readReg[0];
    Temp_lsb = readReg[1] & 0x0F;

    if (Temp_lsb & 0x60) Temp_msb += 0.50;
    if (Temp_lsb & 0x40) Temp_msb += 0.25;
    if (Temp_lsb & 0x20) Temp_msb += 0.125;
    if (Temp_lsb & 0x10) Temp_msb += 0.0625;

    floatToStr(Temp_msb, tempText);
    mikrobus_logWrite(" Ambient temperature : ", _LOG_TEXT);
    mikrobus_logWrite(tempText, _LOG_TEXT);
    mikrobus_logWrite(" C", _LOG_LINE);
    Delay_ms(1000);
}
```

The full application code, and ready to use projects can be found on our LibStock page.

Other mikroE Libraries used in the example:

- I2C

Additional notes and information

Depending on the development board you are using, you may need USB UART click, USB UART 2 click or RS232 click to connect to your PC, for development systems with no UART to USB interface available on the board. The terminal available in all MikroElektronika compilers, or any other terminal application of your choice, can be used to read the message.
mikroSDK

This click board is supported with mikroSDK - MikroElektronika Software Development Kit. To ensure proper operation of mikroSDK compliant click board demo applications, mikroSDK should be downloaded from the LibStock and installed for the compiler you are using.

For more information about mikroSDK, visit the official page.

Downloads

- mikroBUS™ Standard specification
- LibStock: mikroSDK
- Click board catalog
- LLC-I2C Click Schematic
- LLC-I2C Click 2D and 3D files
- LLC-I2C Click Libstock

YOU MIGHT ALSO NEED

PRODUCTS IN THE SAME CATEGORY