Freescale MQX RTOS Example Guide
IIC example

This document explains the IIC driver example, what to expect from the example and a brief introduction to the IIC driver API.

The Example

The example shows the usage of the IIC driver as a master using both polling or interruption drivers and an EEPROM 24LC16 as slave device. However any 24LCxx device can be implemented. For writing this document, a 24LC04 was used.

Running the example

The connections needed for running this example are:

- Serial cable connected to the UART used, this may vary between targets. And a terminal set to 115200 baud, no parity, 8 bits.
- Wire SDA and SCL with the corresponding pull-up resistors from your target to the EEPROM device.
- If necessary provide Vdd and GND to the EEPROM from your board.

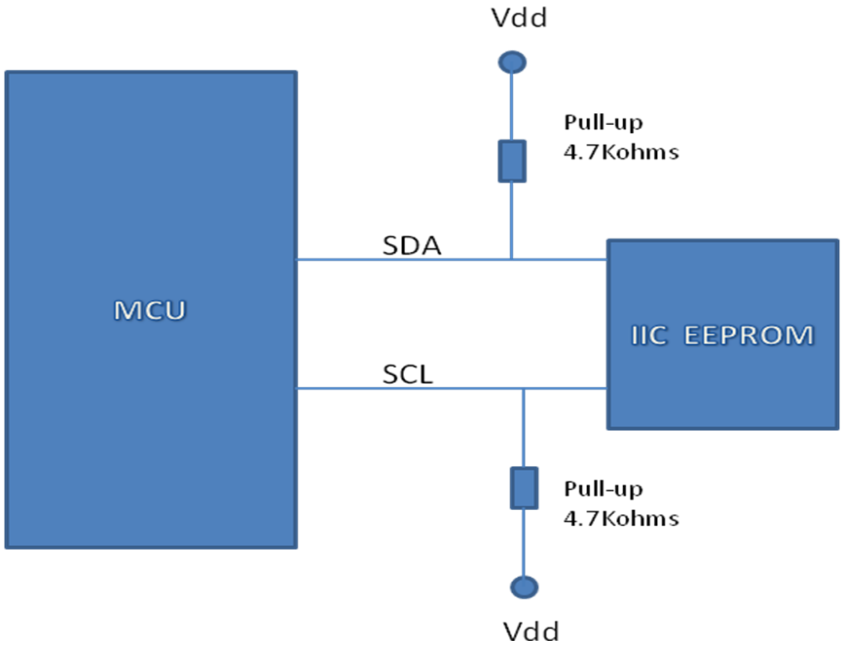

Figure 1
Communications lines

After the connections are set, now the application can be executed. verify that the target BSP has the IIC driver installed (either polled or interruption), if not, please add the proper macro and rebuild the libraries.

Explaining the example

The driver example will open the IIC driver and will test the different IOCTL commands that are available on the driver, such as:

- IO_IOCTL_I2C_GET_BAUD
- IO_IOCTL_I2C_SET_MASTER_MODE
- IO_IOCTL_I2C_GET_MODE
- IO_IOCTL_I2C_SET_STATION_ADDRESS
- IO_IOCTL_I2C_GET_STATISTICS

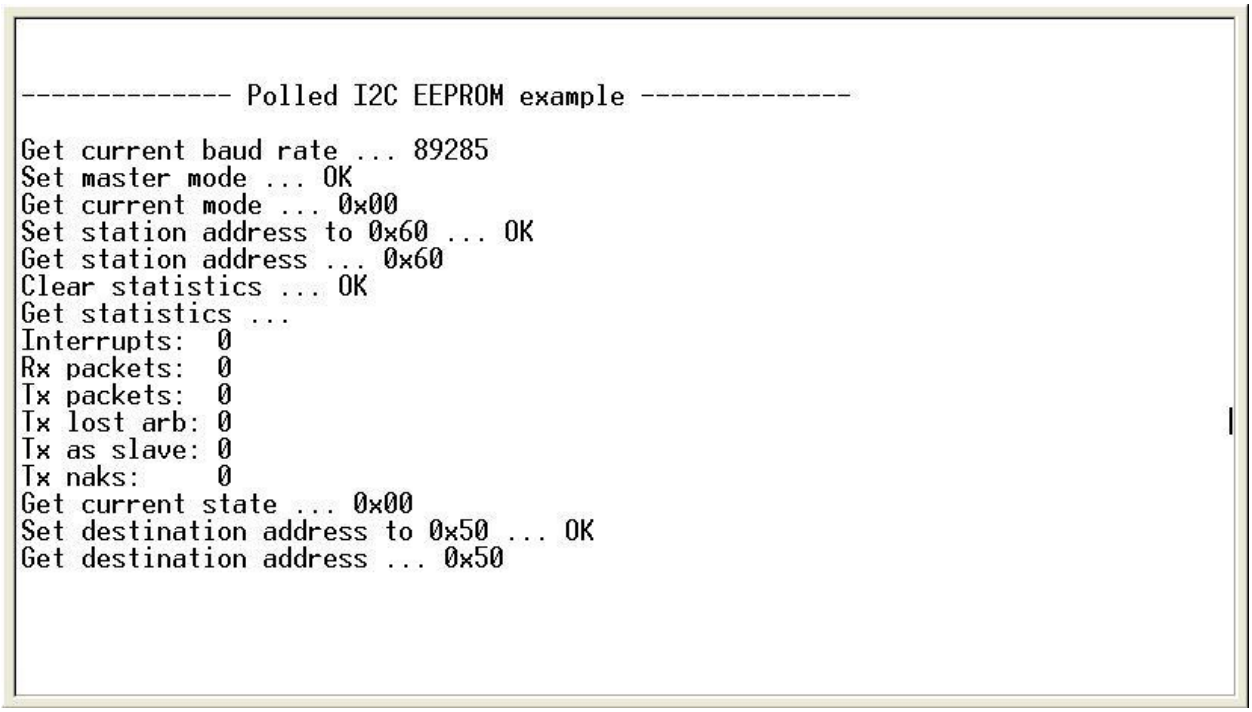

Figure 2
Example output before writing to EEPROM Addresses

After testing the IOCTL commands, the example will start different variations of read/write operations to the driver like:

```
printf ("Test write 0 bytes ... ");
   result = fwrite (&param, 1, 0, fd);
```

This will write "1" block of "0" bytes in the driver. These operations are just demonstrative of the usage of the driver.

The example implements a couple of functions that performs read/write to the EEPROM with the below functions:

- Polled Driver
  - i2c_write_eeprom_polled
  - i2c_read_eeprom_polled

- Interrupt Driver
  - i2c_write_eeprom_interrupt
  - i2c_read_eeprom_interrupt

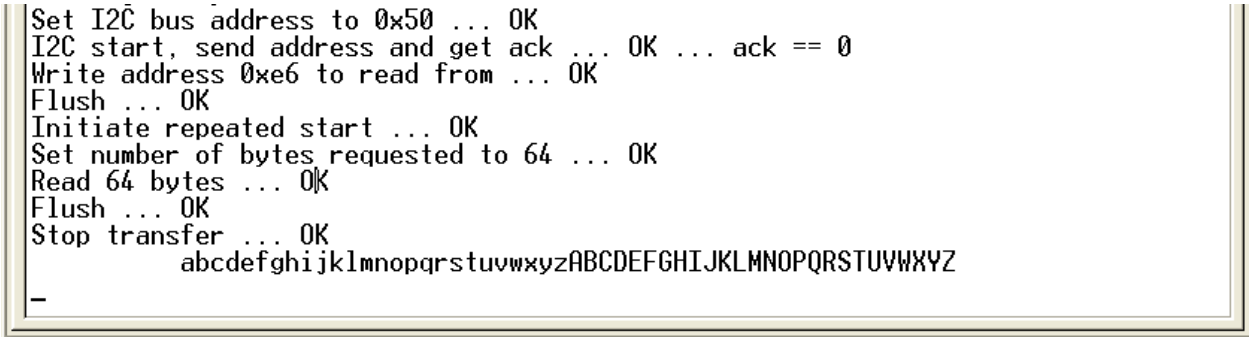

Figure 3
Extract of the example output on EEPROM read/write operations

The interrupts and polled drivers differ on how the read/write operations are made. The EEPROM functions are the same, however the read/write operation is different. For instance, a read with polled driver is as follows:

```
result = fread (buffer, 1, n, fd);
```

While the implementation of a read on the interrupt driver is:

```
do
{
   result += fread (buffer + result, 1, n - result, fd);
} while (result < n);
```

The read function needs to be called until the received data is the same as the requested.

Prior each read, the user should perform a RX request to the driver with the amount of data to be received:

```
param = n;
printf ("Set number of bytes requested to %d ... ", param);
if (I2C_OK == ioctl (fd, IO_IOCTL_I2C_SET_RX_REQUEST, &param))
{
   printf ("OK\n");
} else {
   printf ("ERROR\n");
}
```

After it performs the read/write operations, the example will report the statistics, close the driver and exit MQX.