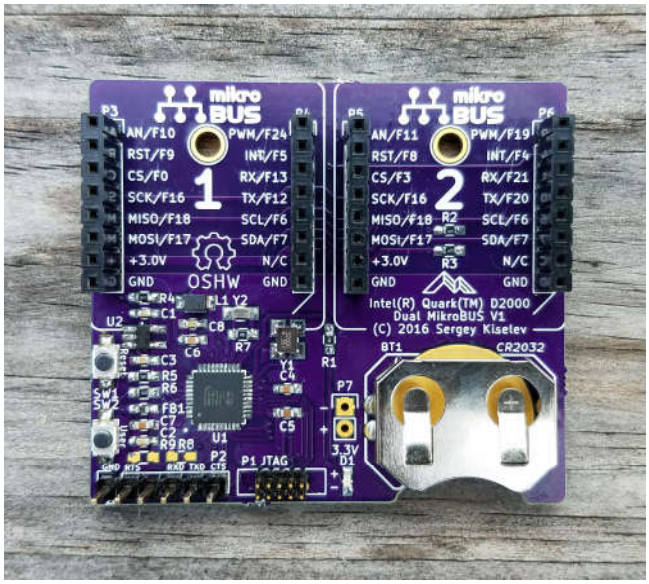
Intel® Quark™ Microcontroller D2000 Based Dual mikroBUS™ Board

Table of Contents

Introduction



This document describes the design, the implementation, and the programming of an Intel® Quark™ D2000 microcontroller based board. The board features two mikroBUS™ compatible sockets that allow connecting various peripherals, such as sensors, radio modules, displays, and memory. The mikroBUS™ is an open standard created by MikroElektronika*. A mikroBUS™ socket has a small footprint, and yet it features all essential MCU interfaces: I^{2}C, SPI, UART, PWM, GPIO, and analog input. The board additionally provides a 6-pin FTDI-compatible UART header, that can be used for connecting a serial console, and for upgrading the MCU firmware using boot loader ROM's Device Management capability. It also includes a 10-pin JTAG interface for the MCU programming and debugging purposes.

Design Overview

Microcontroller

The board uses the Intel® Quark™ D2000 microcontroller U1. This microcontroller contains a 32-bit x86 processor core, 25 GPIOs, I^{2}C and SPI interfaces, JTAG interface, two UART interfaces, PWM module, timers, and so on. It includes 32 KiB of the instruction Flash ROM, 8 KiB of the SRAM, and 8 KiB of the OTP memory (implemented as a write-protected flash memory).

Clocks

The Intel® Quark™ D2000 microcontroller runs on a 32 MHz clock provided by the quartz resonator Y1. It also uses the 32768 Hz quartz resonator Y2 for the RTC clock.

Power Supply

The Intel® Quark™ D2000 microcontroller includes an on-chip 1.8V buck converter voltage regulator to supply voltage for the CPU core. The 47 uH inductor L1 and the 4.7 uF capacitor C8 are the external components used by the voltage regulator. The ferrite EMI filter FB1 and the capacitor C7 form an LC filter for filtering the analog supply voltage AVDD. The I/O power supply IOVDD rail and the mikroBUS™ power signals are connected directly to the CR2032 battery BT1 (nominally 3V) and pin header P7. All the power supply rails are bypassed using 1 uF capacitors.

Reset Circuit

The board uses the Texas Instruments TPS3836E18 nanopower supervisory circuit U2 for generating a proper reset signal on power-on or after the power supply voltage drops below 1.7V. The manual reset can be accomplished using the reset switch SW1.

GPIO Pins Allocation

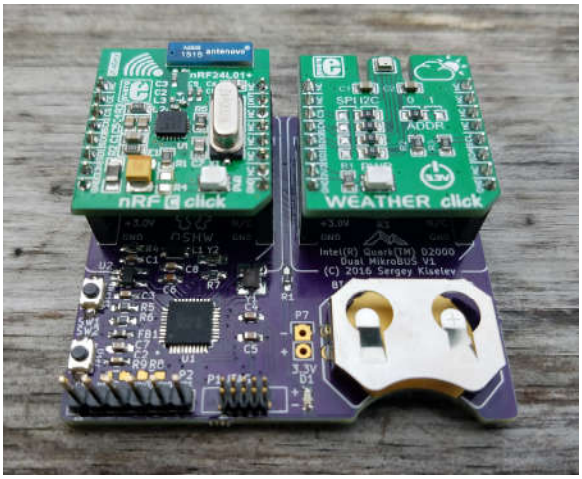
The GPIO pins of the Intel® Quark™ D2000 MCU are allocated on this board as shown in the table below. Note that GPIO pins have multiple functions (modes). The modes need to be configured in the software according to the function used by this board. The suggested pin modes are marked with ***italic bold***.

Quark™ D2000 pin number	Quark D2000 pin name	Function used by the board	Mode 0	Mode 1	Mode 2
31	F0_SPI_M_SS0	mikroBUS 1: SPI chip select – CS1	GPIO0	AI0	***SPI_M_SS0***
32	F1_SPI_M_SS1	Battery monitor (analog input)	GPIO1	***AI1***	SPI_M_SS1
33	F2_SPI_M_SS2	User switch SW2, 560k pull-up – 0: switch pressed 1: switch not pressed	***GPIO2***	AI2	SPI_M_SS2
34	F3_SPI_M_SS3	mikroBUS 1: SPI chip select – CS2	GPIO3	AI3	***SPI_M_SS3***
35	F4_RTC_CLK_OUT	mikroBUS 2: interrupt input – INT2	***GPIO4***	AI4	RTC_CLK_OUT
36	F5_SYS_CLK_OUT	mikroBUS 1: interrupt input – INT1	***GPIO5***	AI5	SYS_CLK_OUT
37	F6_I2C_SCL	mikroBUS 1, 2: I^2C clock – SCL	GPIO6	AI6	***I2C_SCL***
38	F7_I2C_SDA	mikroBUS 1, 2: I^2C data – SDA	GPIO7	AI7	***I2C_SDA***
39	F8_SPI_S_SCLK	mikroBUS 2: reset output – RST2	***GPIO8***	AI8	SPI_S_SCLK
11	F9_SPI_S_SDIN	mikroBUS 1: reset output – RST1	***GPIO9***	AI9	SPI_S_SDIN
2	F10_SPI_S_SDOUT	mikroBUS 1: analog input – AIN1	GPIO10	***AI10***	SPI_S_SDOUT
3	F11_SPI_S_SCS	mikroBUS 2: analog input – AIN2	GPIO11	***AI11***	SPI_S_SCS
4	F12_UART_A_TXD	mikroBUS 1, UART header: serial transmit – TXD	GPIO12	AI12	***UART_A_TXD***

5	F13_UART_A_RXD	mikroBUS 1, UART header: serial receive – RXD	GPIO13	AI13	***UART_A_RXD***
6	F14_UART_A_RTS_DE	UART header: serial request to send – RTS	GPIO14	AI14	***UART_A_RTS***/UART_A_DE
7	F15_UART_A_CTS_RE	UART header: serial clear to send – CTS	GPIO15	AI15	***UART_A_CTS***/UART_A_RE
8	F16_SPI_M_SCLK	mikroBUS 1, 2: SPI clock – SCLK	GPIO16	AI16	***SPI_M_SCLK***
9	F17_SPI_M_TXD	mikroBUS 1, 2: SPI master out slave in – MOSI	GPIO17	AI17	***SPI_M_TXD***
10	F18_SPI_M_RXD	mikroBUS 1, 2: SPI master in slave out – MISO	GPIO18	AI18	***SPI_M_RXD***
18	F19_PWM0_TDO	JTAG header: test data output – TDO mikroBUS 2: pulse width modulation output – PWM2	***TDO***	GPIO19	PWM0
13	F20_UART_B_TXD_TRST_N	JTAG header: test reset – TRST mikroBUS 2: serial transmit – TXD	***TRST***	GPIO20	UART_B_TXD
14	F21_UART_B_RXD_TCK	JTAG header: test clock – TCK mikroBUS 2: serial receive – RXD	***TCK***	GPIO21	UART_B_RXD
15	F22_UART_B_RTS_DE_TMS	JTAG header: test mode select – TMS	***TMS***	GPIO22	UART_B_RTS/UART_B_DE
16	F23_UART_B_CTS_RE_TDI	JTAG header: test data input – TDI	***TDI***	GPIO23	UART_B_CTS/UART_B_RE
21	F24_PWM1	mikroBUS 1, on-board LED D1: pulse width modulation output – PWM1	***GPIO24***	-	***PWM1***

mikroBUS™ Sockets

The board provides two mikroBUS™ compatible sockets. These sockets can be used for connecting peripheral devices, sensors, and actuators. MikroElektronika* provides a variety of Click* add-on boards (https://shop.mikroe.com/click) that can be used in mikroBUS™ sockets. The picture on the right shows the board with two mikroBUS™ Click* add-on boards installed on it.



Each mikroBUS™ socket provides the following signals:

- I^2C bus
- SPI bus. Each mikroBUS™ socket has an individual SPI chip select signal
- UART – TXD and RXD signals. The mikroBUS™ socket 1 uses UART interface A signals. The mikroBUS™ socket 2 uses UART B signals. Note that UART B signals are multiplexed with JTAG interface
- Miscellaneous GPIOs: Analog input, PWM output, reset input, interrupt input
- Power and the ground signals. Note, that the board only provides 3V power supply to mikroBUS™ sockets. It is compatible with most 3.3V devices, but will not work with mikroBUS™ add-on boards that require 5V power supply.

The mapping of the Intel® Quark™ D2000 microcontroller GPIO pins to the mikroBUS™ sockets is shown in the GPIO pin allocation table above.

UART Connector

The UART interface A of the Intel® Quark™ D2000 is exposed on the FTDI compatible header P2. Normally this header is used to connect an FTDI USB to serial cable to provide a debug output. It also can be used for firmware update capability using Device Management capability.

JTAG Connector

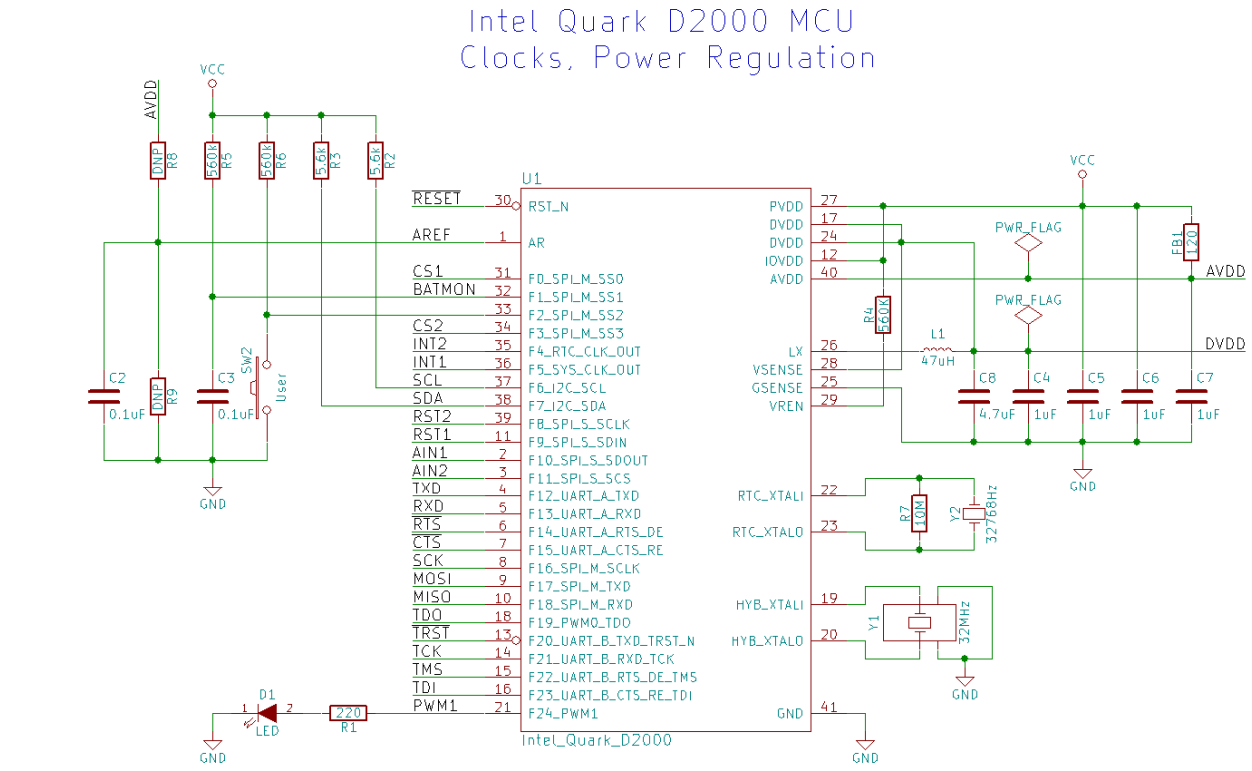
The board uses the standard ARM 10-pin JTAG connector P1. The board has been tested with Tin Can Tools* Flyswatter2 JTAG adapter and Olimex* ARM JTAG 20 pin to 10 pin adapter.

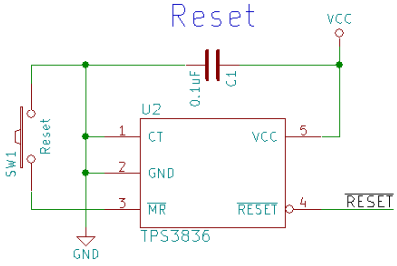
PCB Design Considerations

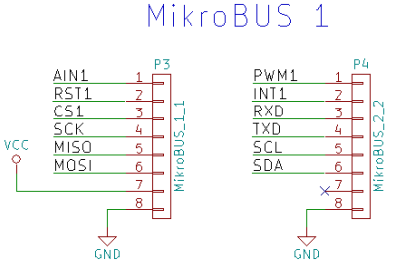
The cost and the ease of manual PCB assembly were the main objectives of the PCB design. Therefore the design uses 2-layer PCB, and the PCB size is only 2.25” x 2”. Since 2-layer PCB does not allow for the separate planes for the ground and the power supplies, special attention has been paid to ensure that the power supply traces are as wide and as short as possible. The SMD components footprints were modified to allow manual soldering – they have larger pads than regular reflow footprints.

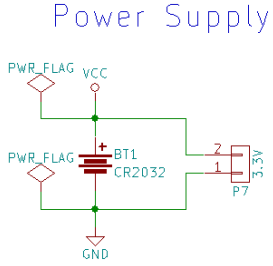
The schematic and the PCB were designed using the open-source KiCad EDA.

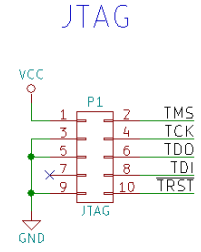
Schematic

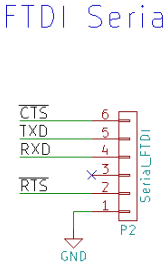

Intel Quark D2000 MCU
Clocks, Power Regulation
U1
Intel_Quark_D2000
RESET
AREF
CS1
BATMON
CS2
INT2
INT1
SCL
SDA
RST2
RST1
AIN1
AIN2
TXD
RXD
RTS
CTS
SCK
MOSI
MISO
TDO
TRST
TCK
TMS
TDI
PWM1
RST_N
AR
F0_SPI_M_SS0
F1_SPI_M_SS1
F2_SPI_M_SS2
F3_SPI_M_SS3
F4_RTC_CLK_OUT
F5_SYS_CLK_OUT
F6_I2C_SCL
F7_I2C_SDA
F8_SPI_S_SCLK
F9_SPI_S_SDIN
F10_SPI_S_SDOUT
F11_SPI_S_SCS
F12_UART_A_TXD
F13_UART_A_RXD
F14_UART_A_RTS_DE
F15_UART_A_CTS_RE
F16_SPI_M_SCLK
F17_SPI_M_TXD
F18_SPI_M_RXD
F19_PWM0_TDO
F20_UART_B_TXD_TRST_N
F21_UART_B_RXD_TCK
F22_UART_B_RTS_DE_TMS
F23_UART_B_CTS_RE_TDI
F24_PWM1
PVDD
DVDD
DVDD
IOVDD
AVDD
LX
VSENSE
GSENSE
VREN
RTC_XTALI
RTC_XTALO
HYB_XTALI
HYB_XTALO
GND
VCC
AVDD
DVDD
PWR_FLAG
FB1
120
L1
47uH
R4
660K
R7
10M
Y2
32768Hz
Y1
32MHz
C2
0.1uF
C3
0.1uF
C8
4.7uF
C4
1uF
C5
1uF
C6
1uF
C7
1uF
DNP
R8
660K
R5
660K
R6
5.6K
R3
5.6K
R2
DNP
R9
SW2
User
D1
LED
220
R1

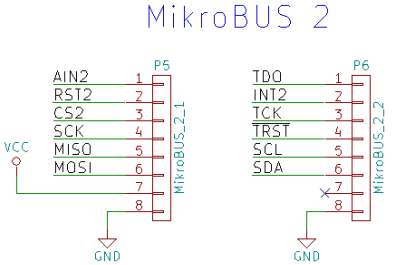

Reset
U2
TPS3836
SW1
Reset
C1
0.1uF
CT
GND
MR
VCC
RESET
RESET
VCC
GND

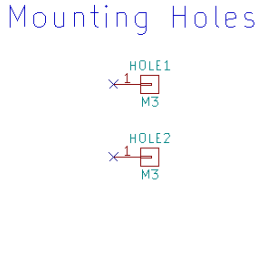

MikroBUS 1
P3
MikroBUS_1_1
AIN1
RST1
CS1
SCK
MISO
MOSI
P4
MikroBUS_2_2
PWM1
INT1
RXD
TXD
SCL
SDA
VCC
GND


Power Supply
PWR_FLAG
VCC
PWR_FLAG
BT1
CR2032
P7
3.3V
GND


JTAG
P1
JTAG
VCC
TMS
TCK
TDO
TDI
TRST
GND


FTDI Serial
P2
Serial_FTDI
CTS
TXD
RXD
RTS
GND


MikroBUS 2
P5
MikroBUS_2_1
AIN2
RST2
CS2
SCK
MISO
MOSI
P6
MikroBUS_2_2
TDO
INT2
TCK
TRST
SCL
SDA
VCC
GND


Mounting Holes
HOLE1
M3
HOLE2
M3

Bill of Materials

Use the following link to view and order all components except of PCB:
http://www.mouser.com/ProjectManager/ProjectDetail.aspx?AccessID=8269bf21e3

Use the following link to view and order the PCB from OSH Park:
https://oshpark.com/shared_projects/WnBaVjZa

Component type	Reference	Description	Quantity	Possible sources and notes
PCB		D2000 Dual MikroBUS V1.0	1	Order from OSH Park or another PCB manufacturer using provided Gerber files.
Battery	BT1	CR2032 lithium battery	1	Mouser 658-CR2032 Note: Due to shipping restrictions battery is not included in Mouser's project BOM
Battery Holder	BT1	CR2032/CR2025 battery holder, THM	1	Mouser 712-BAT-HLD-001-THM
Capacitor	C1 - C3	0.1 uF multilayer ceramic capacitor, 0603 SMT	3	Mouser 77-VJ0603Y104JXJPBC
Capacitor	C4 - C7	1 uF multilayer ceramic capacitor, 0603 SMT	4	Mouser 810-C1608X7R1E105K
Capacitor	C8	4.7 uF multilayer ceramic capacitor, 0603 SMT	1	Mouser 81-GRM188R61A475KE5D
LED	D1	Green LED, 0603 SMT	1	Mouser 78-VLMG1300-GS08
Ferrite Bead	FB1	Ferrite EMI filter, 120 ohm, 0603 SMT	1	Mouser 81-BLM18PG121SN1D
Inductor	L1	47 uH, 1008 SMT	1	Mouser 810-NLV25T-470J-EF
Connector	P1	2x5 pin header, 1.27 mm pitch, THM	1	Mouser 649-221111-00010T4LF
Connector	P2	1x6 pin header, 2.54 mm pitch, THM	1	Mouser 855-M20-9990645
Connector	P3 – P6	1x8 socket, 2.54 mm pitch, THM	4	Mouser 517-929850-01-08-RB

Connector	P7	1x2 pin header, 2.54 mm pitch, THM	1	Mouser 855-M20-9990245
Resistor	R1	220 ohm resistor, 0603 SMT	1	Mouser 71-CRCW0603220RFKEB
Resistor	R2, R3	5.6 kohm resistor, 0603 SMT	2	Mouser 71-CRCW0603-5.6K-E3
Resistor	R4 - R6	560 kohm resistor, 0603 SMT	3	Mouser 71-CRCW0603-560K-E3
Resistor	R7	10 Mohm resistor, 0603 SMT	1	Mouser 71-CRCW060310M0FKEB
Resistor	R8, R9	Do not populate	2	Note: These resistors can be used to set up voltage divider for analog reference voltage.
Tactile Switch	SW1, SW2	Tactile Switch, C&K Components PTS820 series	2	Mouser 611-PTS820J25MSMTR
Integrated Circuit	U1	Intel Quark D2000 MCU	1	Mouser 607-FND2000SR2KF
Integrated Circuit	U2	Texas Instruments TPS3836E18 nanopower supervisory circuit	1	Mouser 595-TPS3836E18DBVR
Crystal	Y1	32 MHz crystal oscillator	1	Mouser 581-CX3225GB32PHPQCC
Crystal	Y2	32768 Hz crystal oscillator	1	Mouser 581-ST2012SB32H5HPWA

Firmware Development for the Board

The board can be programmed using Intel® System Studio for Microcontrollers tool suite. A JTAG adapter, such as Tin Can Tools* Flyswatter2 (http://www.tincantools.com/JTAG/Flyswatter2.html) is required to program the boot loader ROM. The JTAG adapter can also be used for debugging and flashing the firmware. Alternatively it is possible to compile and program a boot loader ROM with the DFU-based Device Management capability enabled. Such boot loader ROM allows updating the firmware using the serial port and an FTDI USB to serial cable, without need for the JTAG adapter.

Compiling and Flashing the Boot ROM with the Device Management Capability

The steps below assume using a Linux-based host system, and that the Intel® System Studio for Microcontrollers Update 1 (version 2016.1.057) is installed on the host system.

Step 1. Set the ISSM environment variables

```
$ source /opt/intel/issm_2016.1.057/issm_env.sh
```

Step 2. Copy the QMSI source code, and change directory to the QMSI root

```
$ cp -a $ISSM_BSP_ROOT qmsi-1.1
```

```
$ cd qmsi-1.1
```

Step 3. Compile the Boot ROM with Device Management capability enabled

```
$ make rom SOC=quark_d2000 ENABLE_DM=1
```

Step 4. Flash the Boot ROM to the board

Power the board using a 3V power supply, for example using a CR2032 battery, or using an external power supply connected to the P7 header. Connect the board to the host system using Tin Can Tools* Flyswatter2 JTAG adapter. Type the commands below to flash the boot loader ROM to the board.

```
$ cd $ISSM_TOOLS_ROOT/debugger/openocd
```

```
$ bin/openocd -f scripts/interface/ftdi/flyswatter2.cfg -f scripts/board/quark_d2000.cfg \
```

```
-c "set QUARK_D2000_OTPC_DATA_WRITE_ENABLED 1" \
```

```
-c "flash_rom $HOME/qmsi-1.1/build/release/quark_d2000/rom/quark_d2000_rom_dm.bin" \
```

```
-c exit
```

In case of a programming error, it might be necessary to reset by pressing the **Reset** button and to repeat the last (**openocd**) command. This command can also be repeated to flash the boot loader ROM to the multiple boards.

Developing and Flashing the Firmware

Refer to the Intel® System Studio for Microcontrollers Reference Guide (https://software.intel.com/en-us/issm-2016-user-ref-guide) to get started with Intel® System Studio for Microcontrollers tool suite. When creating a new project in the Intel® System Studio for Microcontrollers IDE, select Intel® Quark™ D2000 Developer platform as the developer board, and USB-JTAG Flyswatter2 as the connection type.

Once firmware compiles successfully, it can be flashed to the board using either one of the two following methods:

Method 1. Using the Device Management Capability

1. Connect the board to the host system using an FTDI cable. Check the system log for the serial port name (usually /dev/ttyUSB0, adjust the device name in the command below if it is different).
2. Power the board using a 3V power supply.
3. Push and hold the **User** button. Push and release the **Reset** button. Release the **User** button. This will boot the board in to the Device Management mode.
   Note: It is possible to check that board enters the Device Management mode by running a serial terminal software, for example using **screen /dev/ttyUSB0 115200** command, and checking for "C" characters on the terminal. If using screen, push **Ctrl-A** and **'\'** to exit the terminal.
4. Run the following command from the project's directory (located in the Intel® System Studio for Microcontrollers workspace) to flash the firmware:
   ```
   $ make flash SERIAL_PORT=/dev/ttyUSB0 SOC=quark_d2000 TARGET=x86
   ```

Method 2. Using a JTAG Adapter

1. Connect Tin Can Tools* Flyswatter2 JTAG adapter to the board
2. Power the board using a 3V power supply.
3. Use **Debug** -> **<Project Name> (flashing)** or **Run** -> **<Project Name> (flashing)** commands from Intel® System Studio for Microcontrollers to flash the firmware.

References

General Information

- Intel® Quark™ D2000 Microcontroller: http://www.intel.com/content/www/us/en/embedded/products/quark/mcu/d2000/overview.html
- MikroElektronika* mikroBUS™ information: http://www.mikroe.com/mikrobus/

Software Development and Electronic Design Automation Tools

- Intel® System Studio for Microcontrollers: https://software.intel.com/en-us/intel-system-studio-microcontrollers
- Intel® System Studio for Microcontrollers Reference Guide: https://software.intel.com/en-us/issm-2016-user-ref-guide
- KiCad EDA: http://kicad-pcb.org/

Intel® Quark™ Microcontroller Firmware

- Intel® Quark™ Microcontroller Software Interface – Reference Guide: https://downloadcenter.intel.com/download/25619
- Intel® Quark™ Microcontroller Software Interface Bootloader – User Guide: http://www.intel.com/content/www/us/en/embedded/products/quark/mcu/software-interface-bootloader-user-guide.html
- Zephyr™ real-time operating system: https://www.zephyrproject.org/

Hardware Tools and Supplies

- Tin Can Tools* Flyswatter2 JTAG adapter: http://www.tincantools.com/JTAG/Flyswatter2.html
- Olimex JTAG 20-pin to 10-pin adapter: https://www.olimex.com/Products/ARM/JTAG/ARM-JTAG-20-10/
- MikroElektronika* Click* add-on board: https://shop.mikroe.com/click

Project Specific Information

- Bill of materials: http://www.mouser.com/ProjectManager/ProjectDetail.aspx?AccessID=8269bf21e3
- PCB design: https://oshpark.com/shared_projects/WnBaVjZa