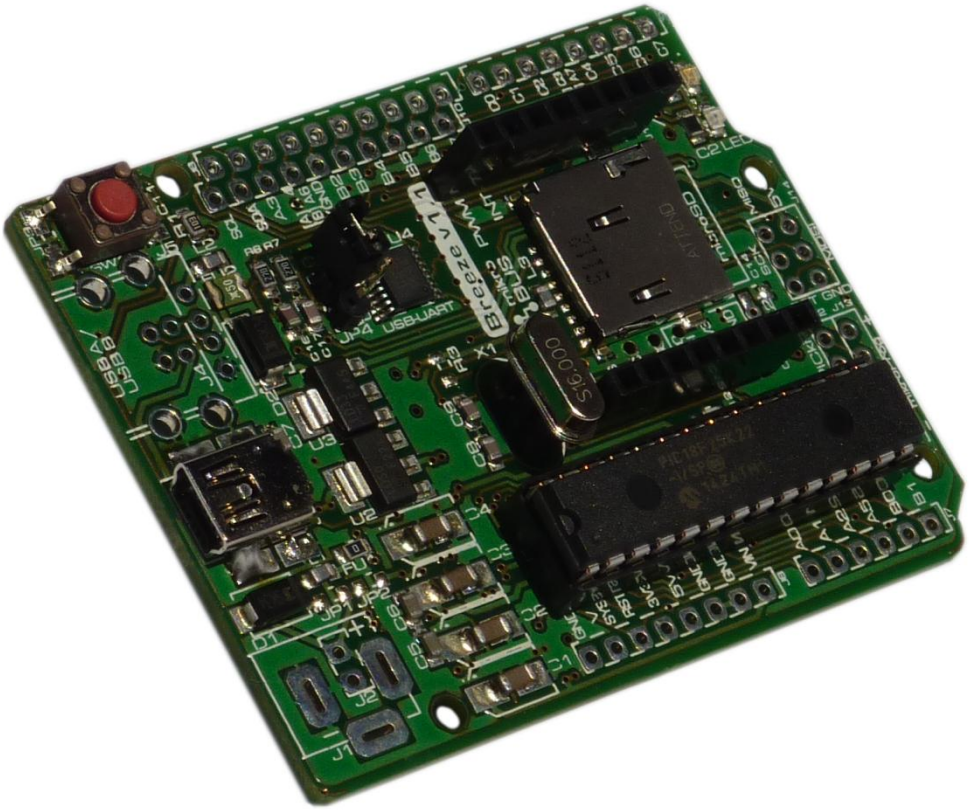
Breeze Board

v1.1 Type A



User Manual

DIZZY enterprises

www.dizzy.co.za

Contents

Introduction 3

Overview – Top 4

Overview – Bottom 5

Getting Started (Proton Compiler) 6

Power Circuitry 7

USB 8

Microcontroller 9

- Crystal Oscillator 9
- Reset Button 9
- Schematic 10

Programming 10

Inputs and Outputs 11

- mikroBus Socket 11
- microSD Socket 12
- "ISP" Header 12
- PortC.2 LED 13
- Offset PortB Socket 13

Drawing 13

Microcontroller I/O Map 13

Links 14

Disclaimer 14

Introduction

The Breeze is a compact and cost-effective microcontroller platform, drawing inspiration from the Arduino (www.arduino.cc) and Amicus (www.myamicus.co.uk) microcontroller platforms. It also features a mikroElektronika (www.mikroe.com) mikroBus socket.

At the most basic level, the Breeze board takes care of providing power to the microcontroller and conveniently exposing its I/O (input/output) pins. It also provides a socket for connecting an external crystal oscillator, includes a convenient reset button, and provides connections for two different types of external programmers – the mikroProg and the PICkit (programmers are not needed to use the board however, as the microcontroller comes pre-loaded with a bootloader – more on that later). The board also includes a power LED, so that you can easily see when the board is powered or not.

Besides for the above-mentioned basic features, the Breeze also includes a USB connection interface, a microSD socket for storing data, a mikroBus socket for easily connecting a variety of accessory boards and a LED connected to one of the PIC's I/O pins. The Breeze is also the same size and shape as the Arduino, and every effort has been made to match the pinout of the Arduino as closely as possible, which means that the board is compatible with most Arduino shields/enclosures/etc.

The USB connection is routed through a USB-UART interface, which means that the USB connection behaves identically to how a serial "COM" port would. This greatly simplifies communication with a computer.

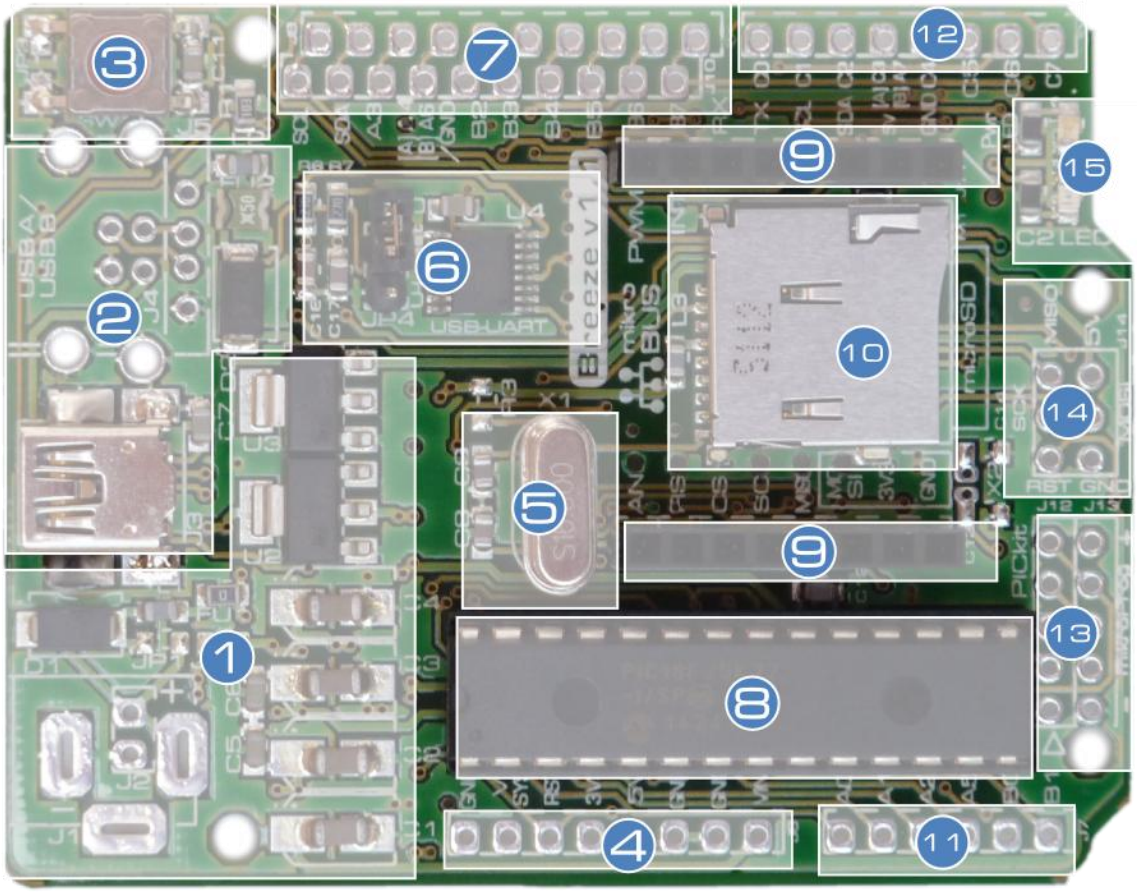
The mikroBus socket allows "click" boards from mikroElektronika to be easily plugged into the system. This means that the functionality of the system can be easily extended with features such as Ethernet, WiFi, an MP3 decoder, GPS, and much, much more. There are currently over 30 "click" boards, and counting.

The LED, connected to PortC.2 of the PIC, is extremely useful for debugging, or whatever else you may need it for.

Finally, the PIC microcontroller comes pre-programmed with a USB bootloader, meaning that no external programmer is needed to program the board. The bootloader is compatible with the free Amicus compiler from Crownhill Associates (www.crownhill.co.uk).

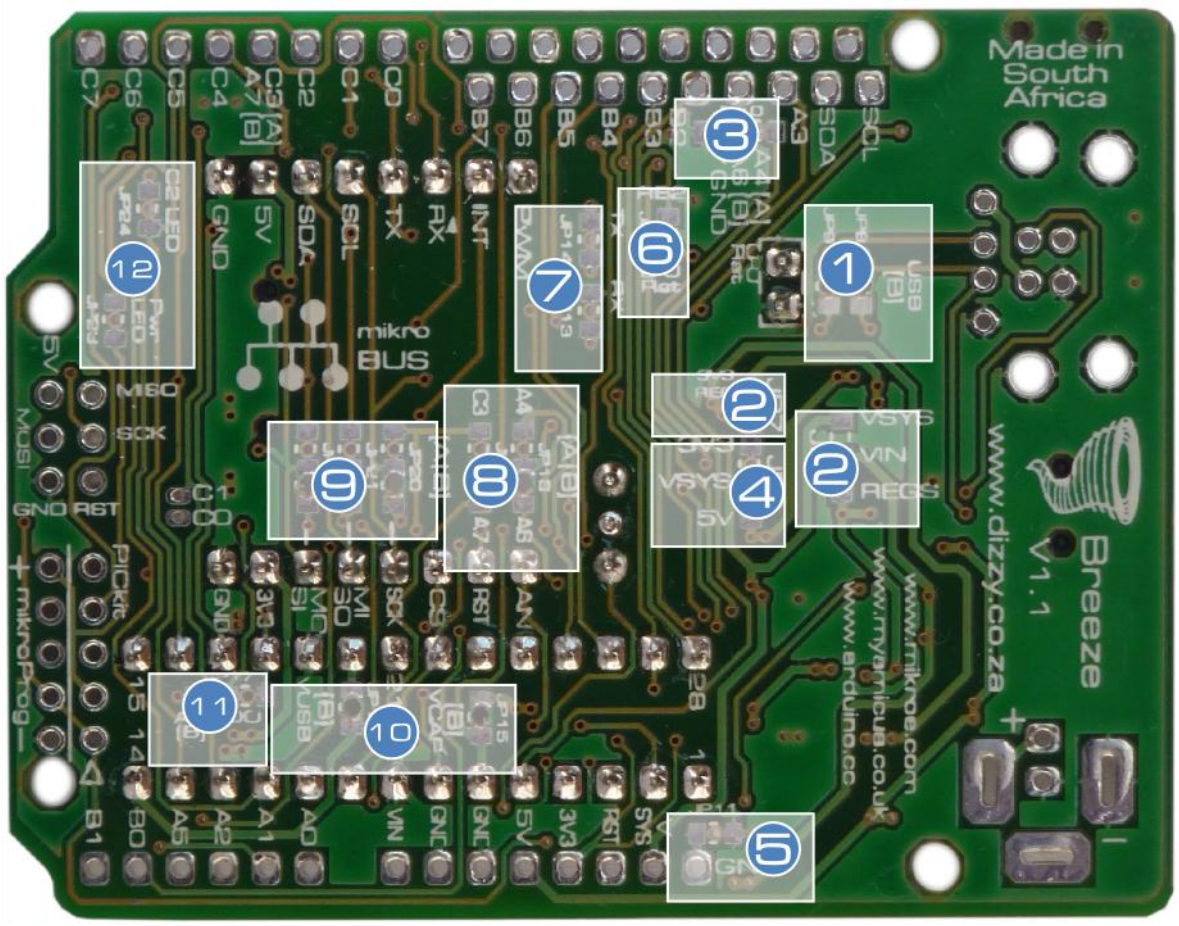
Whether you're a beginner who is just getting started with microcontrollers, or a pro that has been using them for ages, we hope that the Breeze will be exactly what you need to continue your journey.

Overview – Top



1. Power-supply circuitry.
2. USB.
3. Reset button.
4. Power socket (3.3V, 5V, Ground, VIN, VSYS).
5. 16MHz crystal oscillator.
6. USB-UART (USB to serial) converter.
7. PortB socket (inputs and outputs).
8. PIC18F25K22 microcontroller.
9. mikroBus socket.
10. microSD socket.
11. PortA/analogue socket (inputs and outputs).
12. PortC socket (inputs and outputs).
13. Programmer connectors (mikroProg/PICkit).
14. Arduino compatibility "ISP" connections.
15. Power and PortC.2 LEDs.

Overview – Bottom



Although there are many jumpers on the Breeze, most (if not all) of them can be ignored – so if you're a beginner then don't feel overwhelmed, you really don't need to pay any attention to any of these!

The most relevant jumpers are the VSYS jumper (4), which is necessary if you wish to use a 5V microcontroller with the system, and the Arduino compatibility jumpers (3) and (5), which may be necessary if you wish to use an Arduino shield with the system.

1. [Advanced users only] Breeze type A/B USB-UART/direct-USB jumpers.
2. [Advanced users only] Power circuitry reconfiguration.
3. PortA.4/Ground jumper: For compatibility with Arduino shields, this jumper can be placed in the "GND" position.
4. VSYS jumper: This jumper is used to select whether VSYS (which is used to power the microcontroller) is connected to 3.3V or 5V. I.e. if you wish to use a 5V microcontroller with the system, then place this jumper in the 5V position (also make sure that the 3.3V jumper has been disconnected).
5. For maximum compatibility with Arduino shields, this jumper can be disconnected (this connection is currently unused on the Arduino).

6. USB-UART / PortB.2 jumper: This jumper can be used to connect PortB.2 of the PIC to the USB-UART's reset connection, enabling you to disable the USB-UART from the PIC if desired.
7. USB-UART Rx/Tx jumpers: These jumpers can be used to disconnect the USB-UART from the PIC's Rx and Tx connections.
8. [Advanced users only] Breeze type A/B PortA.4/PortA.6 and PortC.3/PortA.7 selection jumpers.
9. [Advanced users only] Breeze type A/B SPI/I²C connection selection jumpers.
10. [Advanced users only] Breeze type B VCAP and VUSB connection jumpers.
11. [Advanced users only] Breeze type B alternative SDO connection jumper.
12. LED disconnect jumpers: These jumpers can be used to easily disconnect the LEDs.

Getting Started (Proton Compiler)

This quick-start guide is designed to get you up-and-running with your Breeze board as quickly as possible.

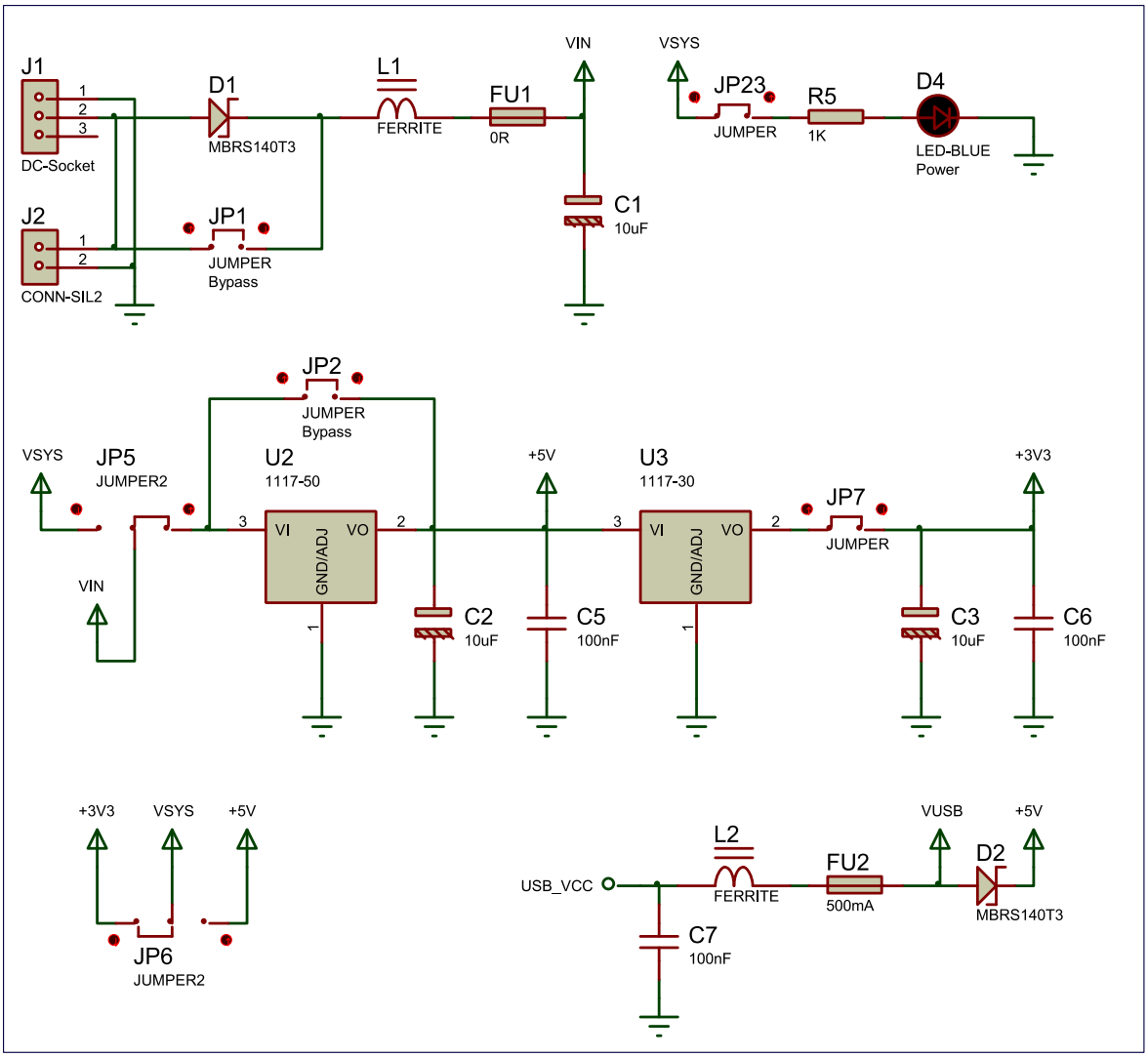
1. Download and install the Proton compiler from www.protonbasic.co.uk.
2. Download and install the Breeze Bootloader and USB drivers from www.dizzy.co.za – follow the included guide to install the Breeze Bootloader into the Proton IDE.
3. Make sure that the USB-UART (disable jumper (JP4) is in the "off" position (disconnected) and connect your Breeze board to the computer via a USB cable.
4. Check that the power LED on the Breeze board lights and that the computer detects "USB Serial Port" (this is normally displayed in the taskbar when the board is first connected – alternatively you can check whether the board has been detected properly by running Device Manager (Start -> Run -> devmgmt.msc) and checking that "USB Serial Port" shows up under Ports and that "USB Serial Converter" shows up under USB Controllers).
5. Run the Proton compiler (Start -> Programs -> Crownhill Associates Ltd -> Proton Compiler).
6. Load our "LED Blink" example program, or alternatively load Crownhill's "LED_Flash" example program and change RB0 to RC2 (in two places). Also add "Device 18F25K22" (without quotes) to the top of the program, if using the PIC18F25K22.
7. Click "Comile and Program" (press the reset button on the Breeze if prompted to do so).

That's it! The LED connected to PortC.2 of the PIC on your Breeze board should now be flashing at 1 second intervals.

Note: The Breeze Bootloader application can be used with other compilers as well, by simply running the Breeze Bootloader application and then browsing to the .hex file generated by the compiler.

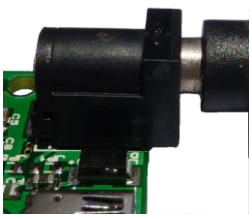
Power Circuitry

The schematic for the Breeze's power circuitry is detailed below:

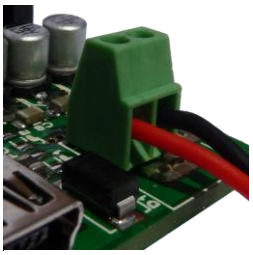


The main points are:

- DC socket or terminal-block power-input connectors.
- Reverse-polarity protection diode.
- Optional fuse.
- 5V and 3.3V regulators.
- USB power-source option, with fuse and reverse-current protection diode.
- VSYS selection jumper, between 3.3V and 5V (change this to 5V if you plan to use a 5V PIC with the system).
- Power LED.



DC socket



Terminal Block

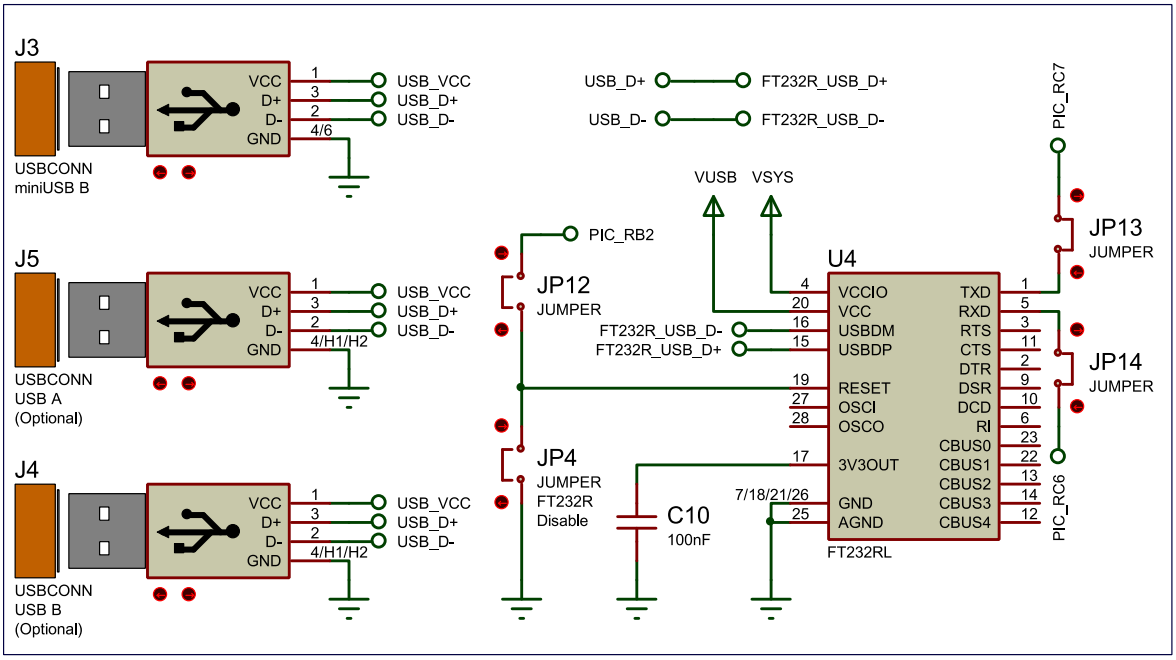
For advanced users, the power circuitry can be reconfigured:

- JP1 can be used to bypass the polarity protection diode, thus eliminating the volt-drop caused by the diode.
- JP2 can be used to bypass the 5V regulator (if, for example, a 5V power supply is used).
- JP5 can be used to bypass both regulators. This is intended to be used when the system is powered by a 3.6V lithium battery. Either JP7 or JP6 should also be disconnected if this is used.

For more information on the power socket, please see the "Inputs and Outputs" section.

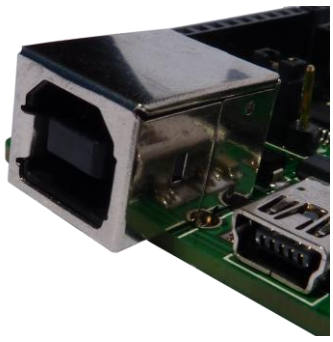
USB

The schematic for the Breeze A's USB circuitry is detailed below:



The Breeze comes with a miniUSB connector, and has the option to install a USB-B or USB-A connector as well.

miniUSB Connector

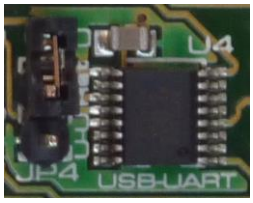


USB-B Connector

USB-A Connector

If you wish to disable the USB-UART (USB to serial) converter, in case you are using your Rx and Tx lines for something else and are afraid that it might interfere, then this can be done in one of two ways:

1. Connect JP4 (JP4 is disconnected in the picture to the right).
2. Connect JP12 (preferably using a 10kΩ resistor rather than a jumper in order to eliminate the possibility of a short-circuit), thus connecting the USB-UART's reset line to the PIC's PortB.2 pin, and drive the PIC's PortB.2 pin low.



USB-UART with Reset Jumper (JP4)

Microcontroller

The microcontroller used in the Breeze A is the PIC18F25K22.
The PIC18F25K22 features:

- Operating voltage range from 2.3V to 5.5V (PIC18F25K22)
- 32KB of program memory, 1536B of SRAM and 256B of EEPROM. (Approximately 200B of program memory are used by the Amicus bootloader).
- Operating frequency up to 64MHz (16MIPS).

For more information on the PIC18F25K22 please download its datasheet from http://ww1.microchip.com/downloads/en/DeviceDoc/41412F.pdf

It is of course possible to replace the PIC18F25K22 with almost any other 28-pin 8-bit PIC microcontroller (adjustment of some jumpers on the board may be necessary).

Crystal Oscillator

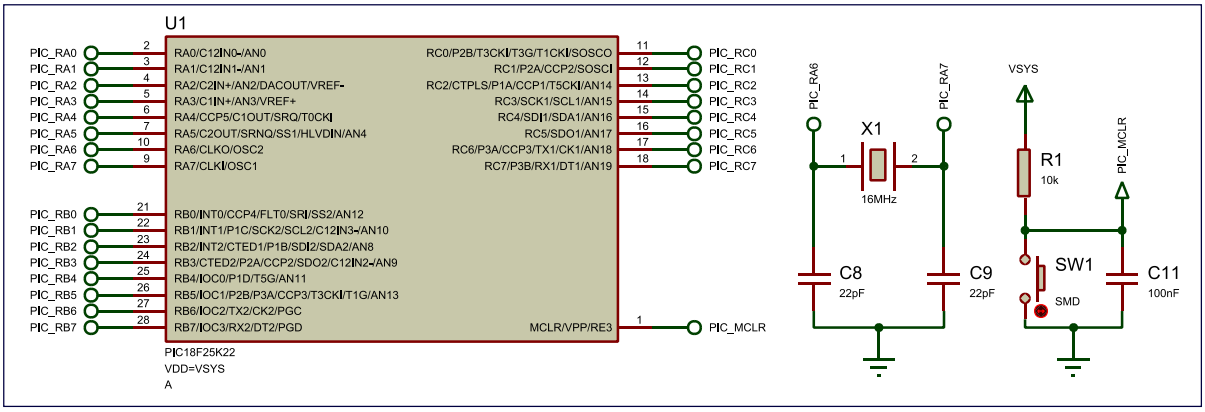
A 16MHz crystal oscillator (X1) is connected to the microcontroller. This oscillator is placed in a socket, meaning that it is easily replaceable with a different oscillator. Please note however that the Amicus bootloader will only work with a 16MHz oscillator.

Reset Button

A tactile switch (SW1) is connected to the MCLR pin of the microcontroller. Depressing this button will pull the MCLR pin low and place the microcontroller in a state of reset. Releasing the button will allow the R1 resistor to pull the MCLR pin high again, and program execution will restart from the beginning. (Note: The reset button will not work if the MCLR function has been disabled on the PIC by programming it with the relevant configuration setting).

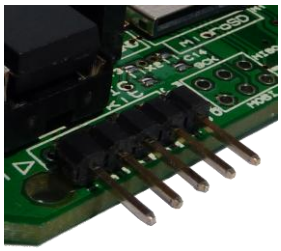
Schematic

The schematic for the microcontroller, oscillator and reset button is detailed below:



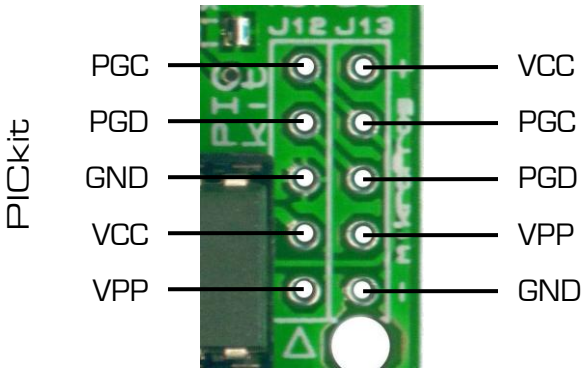
Programming

The Breeze A comes pre-loaded with a bootloader, meaning that no external programmer is needed in order to use the board. It is however possible to use an external programmer if desired, with connectors provided for both the mikroProg and PICkit programmers.



Right-Angle Connector (in PICkit socket)

Straight Connector (in PICkit socket)



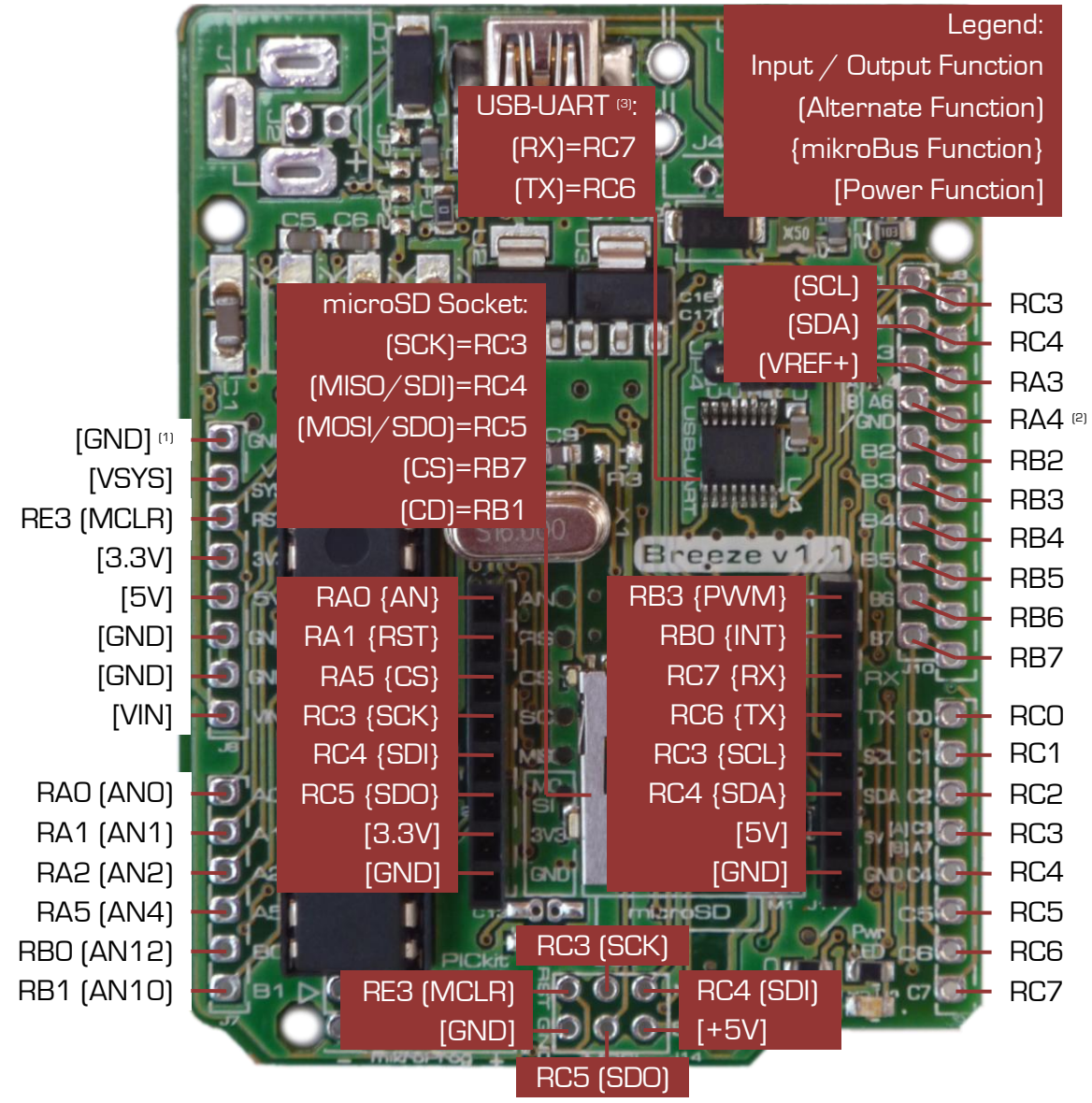
mikroProg

VCC: Power (VSYS)
GND: Ground (0V)
PGC: Program Clock (RB6)
PGD: Program Data (RB7)
VPP: Programming Voltage
(The VPP pin on the PIC is also its MCLR (reset) pin).

For a quick guide on how to program the board with the Amicus compiler / Breeze bootloader, please see the "Getting Started" section of this manual.

Inputs and Outputs

The inputs and outputs of the Breeze A board are detailed in the following diagram:



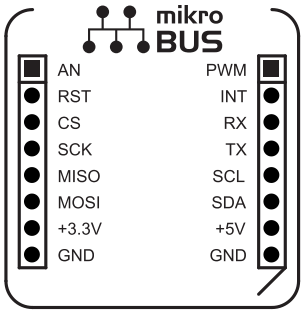
[1] Can be disconnected using jumper JP11

[2] Can be connected to ground instead using jumper JP10

[3] Please see the "USB" section of this manual for more information

mikroBus Socket

The mikroBus socket allows a huge variety of mikroElektkronika (www.mikroe.com) "click" add-on accessory boards to be connected to the system. There are currently over 30 "click" boards and counting, performing a vast variety of functions such as communication, storage, audio, etc.



"ETH click" Ethernet board connected

"WiFi PLUS click" WiFi board connected

The mikroBus connections are explained in greater detail below:

AN	Analogue	Pulse Width Modulation	PWM
RST	Reset (different from the PIC's reset)	Interrupt	INT
CS	SPI Chip Select	UART (Serial) Receive	RX
SCK	SPI* Serial Clock	UART (Serial) Transmit	TX
MISO	SPI* Master-Out Slave-In (SDI on the PIC)	I²C** Serial Clock	SCL
MOSI	SPI* Master-In Slave-Out (SDO on the PIC)	I²C** Serial Data	SDA
+3.3V	3.3V Power Supply	5V Power Supply	+5V
GND	Ground (0V Power Supply)	Ground (0V Power Supply)	GND

*Serial Peripheral Interface (communication interface)

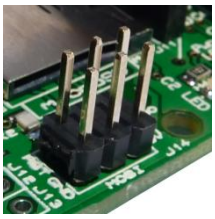
**Inter-Integrated Circuit (communication interface)

microSD Socket

A microSD socket is included on the Breeze board. It is connected to the standard SPI pins on the board, with the CS (chip select) pin being connected to PortB.7 and the CD (card detect) pin being connected to PortB.1. The card detect function is intended to be used with the internal "weak pull-up" available on the PIC's PortB.1 pin, however an external pull-up resistor (R3) can also be connected if desired.

"ISP" Header

"ISP" connections are included on the Breeze board for maximum compatibility with Arduino shields. "ISP" is the programming standard for Atmel AVR ATmega microcontrollers, which the Arduino uses. Note that soldering this header onto the board will block access to the microSD socket.



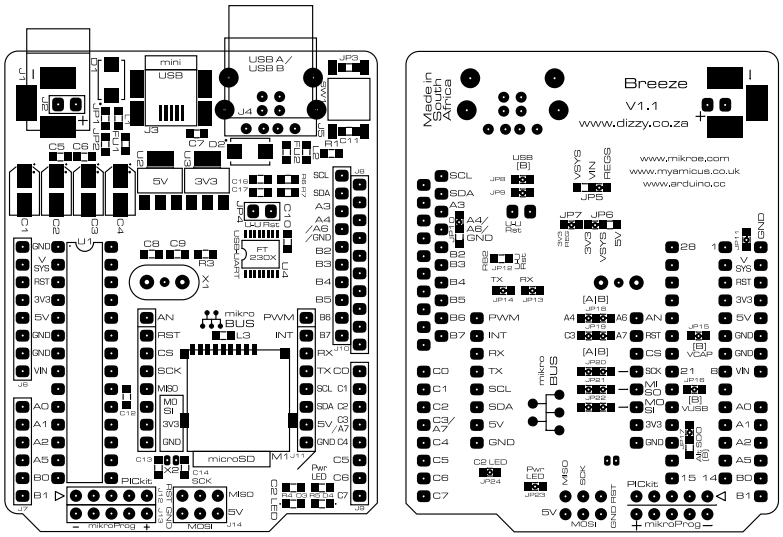
PortC.2 LED

An LED is connected to PortC.2 of the PIC, and is very useful for debugging, or whatever other purpose you may find for it. It can be easily disconnected if desired using jumper JP24.

Offset PortB Socket

An offset PortB socket is provided on the Breeze board. This was done so that all connections on the board are exposed in a 2.54mm (0.1in) grid format, making the board compatible with various prototyping tools such as veroboard.

Drawing



For more information please see the "Arduino Template" file, available from the Breeze page on our website.

Microcontroller I/O Map

RA0	mB AN	**RB0**	mB INT		**RC0**	
RA1	mB RST	**RB1**	(SD CD[1])		**RC1**	
RA2		**RB2**	(USB-UART Rst[2])		**RC2**	LED
RA3		**RB3**	mB PWM		**RC3**	SCK/SCL
RA4		**RB4**			**RC4**	SDI/SDA
RA5	mB CS	**RB5**			**RC5**	SDO
RA6	OSC2	**RB6**		(PGC)	**RC6**	USB-UART TX[2] mB RX
RA7	OSC1	**RB7**	(SD CS[1])	(PGD)	**RC7**	USB-UART RX[2] mB TX

1) Only applicable if an SD card is inserted. SD Card Detect switch is normally open – R3 can be used to pull-up the input, if the PIC's internal weak pull-up is not used.
2) USB-UART Reset (Disable) can be connected to RB2 with JP12 (active low, disconnected by default). JP4 can also be used to disable the USB-UART.

Links

The Breeze A board:

- Is designed by Dizzy Enterprises (www.dizzy.co.za).
- Is designed in the Proteus Design Suite (www.labcenter.com).
- Is compatible with (compiler), and draws inspiration from, the Amicus (www.myamicus.co.uk).
- Features a mikroBus socket (www.mikroe.com).
- Is compatible with (most shields), and draws inspiration from, the Arduino (www.arduino.cc).

Disclaimer

This part says that you cannot sue us because we accept no responsibility for any damages whatsoever that may be caused in connection with our products. We've designed them the best we can, but please, use your common sense.