

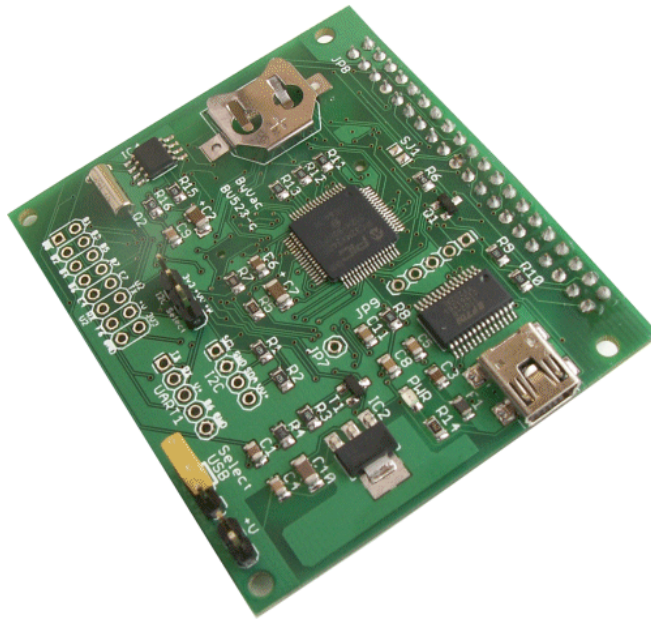
---

**Microcontroller**

---

**BV523**

---

**BV523****32bit Microcontroller**

Product specification

Jun 2011 V0.a

---

**Microcontroller**

---

---

**BV523**

---

## Contents

1.	Introduction .....	3
2.	Features .....	3
3.	Physical Specification .....	3
3.1.	PIC32.....	3
3.2.	USB Interface.....	3
3.3.	Power Supply .....	4
3.4.	Power Off .....	4
3.5.	Real Time Clock.....	4
3.6.	I2C Interface .....	4
3.7.	Serial Interface .....	5
3.8.	SD Card .....	5
3.9.	General I/O.....	5
3.10.	Display.....	5
3.11.	Programming Socket.....	6
4.	Firmware .....	7
4.1.	Application Loader .....	7
4.2.	BV-Basic.....	7
4.2.1.	Power Control .....	7
5.	Circuit Diagram .....	7

# Microcontroller

# BV523

Rev	Change
Jun 2011	Preliminary
July 2011	Minor typing errors
Nov 2011	Display on board regulator
Dec 2011	Corrected display connector table
Nov 2012	General I/O pinouts corrected and Display pin 16 corrected.
Nov 2012	Circuit diagram added

## 1. Introduction

The BV523 is a 32bit microcontroller designed to provide a complete user interface and control when used with the display. It can be used with or without the display and has a unique power control that allows the board to be shut down completely.

When used without the display most of the I/O is available for general purpose use. When used with the display there is a second SD card interface.

This text describes the hardware only. The BV523 comes complete with a comprehensive BASIC that makes using the device very easy, this is described elsewhere. It can also be programmed in C without the need of a programmer as there is a built in boot loader that operated through the USB interface.

For getting started and examples see the online resource:

[http://doc.byvac.com/index.php5?title=Product\\_BV523](http://doc.byvac.com/index.php5?title=Product_BV523)

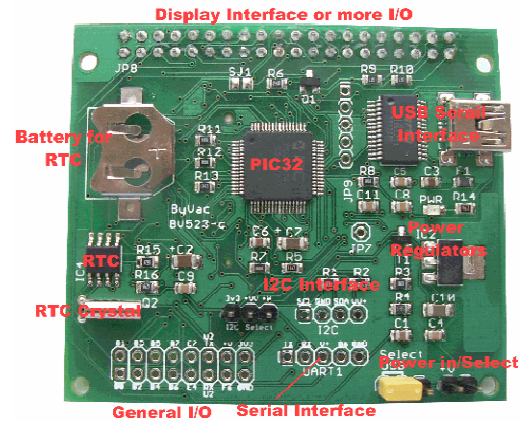
## 2. Features

- 32bit, PIC32
- BV-Basic Firmware
- SD Card
- RTC with battery backup
- USB serial interface
- Power control
- Watchdog
- Zero power standby
- 320x240 Colour display option

## 3. Physical Specification

The BV523 is a complete microcontroller based on the PIC32MX340 with 512k Flash and 32k RAM. It is 55mm x 63mm and designed to fit on the back of a full colour

display although it can quite happily work without the display.



The following deals with the various sections in turn.

### 3.1. PIC32

This is a PIC32MX340 with 512k Flash. The specification for this processor is as follows:

- Speed MHz 80
- Program Memory Size (KB) 512
- RAM (KB) 32
- DMA Channels 4
- SPITM 2
- I2CTM Compatible 2
- A/D channels 16
- Max A/D Sample Rate 1000
- Input Capture 5
- Output Compare/Std. PWM 5
- 16-bit Digital Timers 5
- Parallel Port PMP
- Comparators 2
- Internal Oscillator 8 MHz x 10 via PLL

The processor is configured to use the internal oscillator and normally runs at 80MHz. The in built RTC is not used in favour of the external battery backed one. This does not use any of the available I2C channels.

### 3.2. USB Interface

This is based around the FTDI chip (<http://www.ftdichip.com/>) virtual COM port driver and has up to date drivers for nearly all operation systems including Linux, Microsoft and Apple.

When connected it presents itself as a COM port and communication with the BV523 is via UART 2. Programming and main communication is done via this interface. The firmware as shipped contains BV-Basic and so it will work straight away. There is also a boot loader (application loader) so that C programs can also be developed without the need of an external programmer.

# Microcontroller

# BV523

## 3.3. Power Supply

The power can be supplied either via the USB or an external source and is selected via a jumper. The power is regulated by an on board 3v3 regulator capable of supplying up to 1A. The absolute maximum input voltage is 18V and so would be suitable for an in-car application.

S1	S2	S3		V+	GND
----	----	----	--	----	-----

Power Select and connector.

S1-S2 Selects power from the USB

S2-S3 Selected power from the external source

V+ and Gnd are the input for the external source (max 18V).

**\*\* NOTE:** The display power is connected directly to +V, this is because it has an on board 3v3 regulator of its own. It however cannot take a large input voltage so when a display is connected limit the input voltage to 5.5V

## 3.4. Power Off

There is a unique power off circuit that is used in conjunction with the Real Time Clock (RTC). All power from the USB or external source goes through this switch which is controlled by RG9 and the Alarm output from the RTC.

It is possible therefore to set an alarm to wake the processor up and then power down completely.

*BV\_Basic firmware takes care of power control for an example see 'Getting Started' this description is provided for those who want to program this in C*

If RG9 is high then no power will get to the processor and it is effectively off, consuming no power at all. If RG9 is taken low then power is supplied to the processor.

At reset RG9 would normally be high and so the application loader, under normal circumstances, will set this low so as to keep power to the microcontroller. This action can be prevented by placing 0x55 in address 0xa0003000 and 0xa4 in 0xa0003001. This will prevent any spurious reset activation the microcontroller until the power is actually removed.

When power is removed and reapplied the bytes will of course not have that information and RG9 will be held low once again by the application loader.

An application is responsible for setting RG9 to an output and holding it low within about 500ms after switch on, otherwise power will be lost.

The main advantage of this kind of power control is that the processor will consume zero

power whilst asleep. The only power being consumed is by the RTC.

## 3.5. Real Time Clock

Although the PIC32 has a built in RTC there is no real provision for holding the time with a battery and hence the external RTC is provided.

It uses the MT41T81 RTC that also has alarms that can be set. As previously mentioned the alarm output is connected to RG9 and is used for the power down circuit.

The battery is a CR1220 and a holder is provided for it.

## 3.6. I2C Interface

The I2C interface uses the PIC channel 1 I2C circuit.

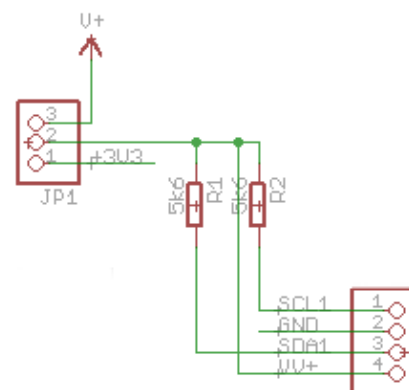
Pin 1 (square)	Pin 2	Pin 3	Pin 4
SCL	GND	SDA	V+

I2C Connector

3V3	VV+	V+
-----	-----	----

I2C Power Selector

The I2C interface has built in pull up resistors and these can either be powered from the 3V3 supply or from the +V supply.



V+ is the raw input voltage, if this is supplied via USB this will be 5V. 3V3 is the regulated 3.3V output from the regulator.

The three options therefore for the pull up resistors is:

1. No jumper – make other arrangements to supply pull up resistors
2. Jumper to 3V3 – this will be used on 3V3 I2C devices
3. Jumper to V+ - normally used on 5V devices but care as it is the raw input from the power input and so it is up to the user to make sure that it is a suitable voltage.

# Microcontroller

# BV523

The I2C pins on the processor are:

SCL RG2  
SDA RG3

Note that this interface is the hardware channel 1 for the PIC. For more information about this consult the PIC32 family data sheet.

## 3.7. Serial Interface

A serial interface is provided for easy access to UART 1, UART 2 is connected to the USB COM port but is still provided on the general I/O connector.

Pin 1 (Square)	Pin 2	Pin 3	Pin 4	Pin 5
TX	RX	V+	RB6	GND

UART 1 Connector

TX This is the TX pin of UART 1 which is also RF3

RX This is the RX pin of UART 1 which is also RF2

V+ Is intended to supply power to the serial device (output), this will be +5V if the USB is being used but will be whatever is used for the input if external power is being used.

RB6 Is a port from the processor and can be used for anything – probably RTS or reset perhaps.

Gnd Ground

When using BV-Basic this can be used in various ways, simply open the communication to COM 1. RB6 can be specified as RTS or CTS by using the open command for example:

**opencom 1 115200 0,,"rb6"**

## 3.8. SD Card

At the back of the PCB is an SD Card holder which is designed to take the standard sized SD cards. It is wired for the SPI interface as follows.

SD Card (pin)	BV523/PIC
+V (4)	3V3
Insert detect (11)	n/c
Write Protect (10)	n/c
Data Out (7)	SDI2 (RG7)
Data In (2)	SDO2 (RG8)
Clock (5)	SCK2 (RG6)
Chip Select (1)	RF0

### SD Card wiring

The SD card interface uses the I2C/SPI channel 2 and so if this is required for other uses as well then this should be taken into

account. Several SPI devices will happily work together so this should not be a problem.

This is a hardware driven SPI interface and so can easily achieve 20MHz, however most SD Cards will only work at 10MHz.

This interface is also brought out at the display connector as that also has an SD Card when using BV-Basic this is card 'a' and the display is card 'b'.

RF1 is used for the chip select on the second SD Cards and is presented on pin 36 of the display connector.

## 3.9. General I/O

Nearly all of the I/O pins from the PIC32 are either brought out to the I/O interface or the display interface.

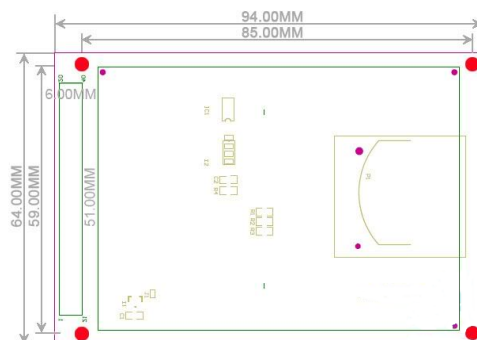
Pin	2	4	6	8	10	12	14	16
	B1	B3	B5	B7	E7	U2 TX	V+	3v3
	B0	B2	B4	B6	E4	U2 RX	F6	G
Pin	1	3	5	7	9	11	13	15

General I/O connector

NOTE: that V+ and 3v3 are outputs, input power should only be supplied through the power connector described previously.

## 3.10. Display

The display is optional, without it the BV523 is a small powerful microcontroller, with it, it becomes a complete user interface as the display also incorporates a touch screen.

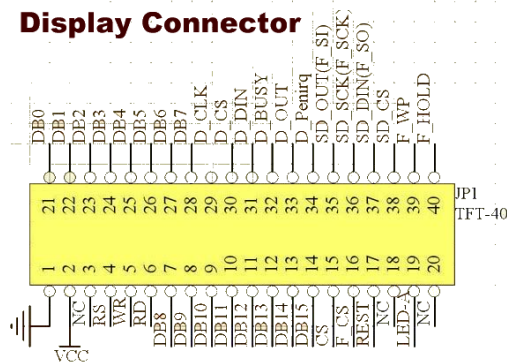


The display is based on the SSD1289 controller, uses 65k Colour mode, has a touch screen and is 3.2". The display resolution is 240 x 320.

# Microcontroller

# BV523

## Display Connector



The display is wired for a 16 bit interface and this is translated to the BV523 as follows

Pin	Display	BV523	Notes
1	Gnd	Gnd	
2	VCC	3V3	
3	n/c	n/c	
4	RS	RC12	Display command/data
5	WR	RC13	Write when low
6	RD	RC14	Read
7	Data 8	RB8	16 bit data bus
8	Data 9	RB9	
9	Data 10	RB10	
10	Data 11	RB11	
11	Data 12	RB12	
12	Data 13	RB13	
13	Data 14	RB14	
14	Data 15	RB15	
15	LCD_CS	RC15	
16	Flash_CS	*RF4	See text
17	RST	RE1	Reset
18	n/c	n/c	
19	BL	RE2	Via Transistor
20	n/c		

### Top Row

The display has pads for a serial (SPI) flash device but the flash is not fitted. If this were to be implemented then RE4 could be used as the Chip select. It does NOT however by default connect to this pin but goes via SJ1 which is near the display connector. To use RF4 for this pin then short out the two pads on SJ1.

Pin	Display	BV523	Notes
-----	---------	-------	-------

21	Data 0	RD0	lower half of 16 bit data bus.
22	Data 1	RD1	
23	Data 2	RD2	
24	Data 3	RD3	
25	Data 4	RD4	
26	Data 5	RD5	
27	Data 6	RD6	
28	Data 7	RD7	
29	T_CLK	RD8	Touch control
30	T_CS	RD9	
31	T_IN	RD10	
32	T_BUSY	n/c	
33	T_OUT	RD11	
34	T_PER	n/c	
35	SD_OUT	SDI2	See text
36	SD_CLK	SCK2	
37	SD_IN	SDO2	
38	SD_CS	RF1	
39	F_WP	n/c	
40	F_hold	n/c	

### Lower Row

The SD connector uses the same interface as the SD card on the BV523. The only difference is in the CS lines. This means that two SD cards may be implemented at any one time.

The above is all that is required to operate the display efficiently and using this arrangement is possible to refresh the whole screen in about 30ms which is approximately 30 frames per second, providing you can get data to it fast enough of course.

When the display is not connected the lines can be used for general purpose I/O or whatever the PIC32 has designated for them (ADC, PWM etc.)

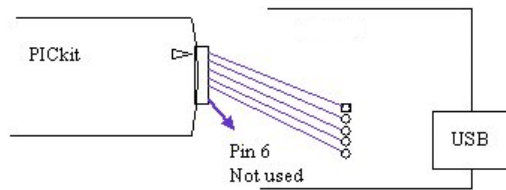
When used with BV-Basic the display will be automatically detected.

### 3.11. Programming Socket

The programming socket is provided for installing the firmware at the factory. There is no need for a programmer when using this device as there is a built in application loader that can load any binary files required. This information is only provided for completeness of for those who wish to program from scratch.

## Microcontroller

## BV523



There is a row of 5 pads near the USB connector as shown. The square pad is pin 1. A PICKit can be connected directly to this connector in a 1:1 arrangement, pin 6 is not used. The square pad is pin 1.

### 4. Firmware

The following refers to the firmware that is specific to this microcontroller and therefore may not be presented anywhere else. This also applies to the section on BV-Basic.

All BV523 devices come with firmware that may or may not be used as the user chooses.

#### 4.1. Application Loader

This is also called a Boot Loader. This firmware resides in protected memory and can be used with BV-Tools which has the necessary protocols to handle the download.

The BV523 uses **version 2** boot loader and using BV-Tools the 'upgrade' menu option is chosen. A slightly more detailed description can be found here:

[http://doc.byvac.com/index.php5?title=BV-Basic Installation#Version 2 Boot Loader](http://doc.byvac.com/index.php5?title=BV-Basic%20Installation#Version%20Boot%20Loader)

The application Loader will load any binary file, a binary file can be created from a hex file by using the hex2bin utility, this is also contained within the above link. when using BV\_COM2 this will automatically download a HEX file and so the conversion tool is not needed.

#### 4.2. BV-Basic

In general the common BV-Basic functions are described here:

<http://doc.byvac.com/index.php5?title=BV-Basic>

This section describes some of the BASIC functions that are unique to this device.

##### 4.2.1. Power Control

The device can be powered down completely using 'poff'. There are two methods of restoring power:

1. Reset by using the reset pin or cycling the power
2. Using the alarm from the RTC

When the device is powered off it consumes zero current and so using the alarm method could provide some very interesting applications. For example waking up once a

month to provide some service or monitoring, the device could quite easily last on small batteries.

The technique for using power control is to set an alarm - repeating or otherwise, make sure that the alarm is on 'alon' and then issue the poff command. The device will immediately power off and the power will be automatically be restored at the time and date set by the alarm.

The following code if stored on an SD Card using the file name "auto.bas" will make the device come on for 2 seconds every minute.

```
2 function setal
3     alset "8/6 12:01:00" // any time
4     alrepeat 4 // every minute
5     alon
6 endf
7 function auto
8     print #3 "hello"
9     wait 2000
10    print #3 "godbye"
11    poff
12 endf
```

Run setal first and then reset.

Refer to [www.doc.byvac.com](http://www.doc.byvac.com) for all of the other BV-Basic firmware commands including the display commands.

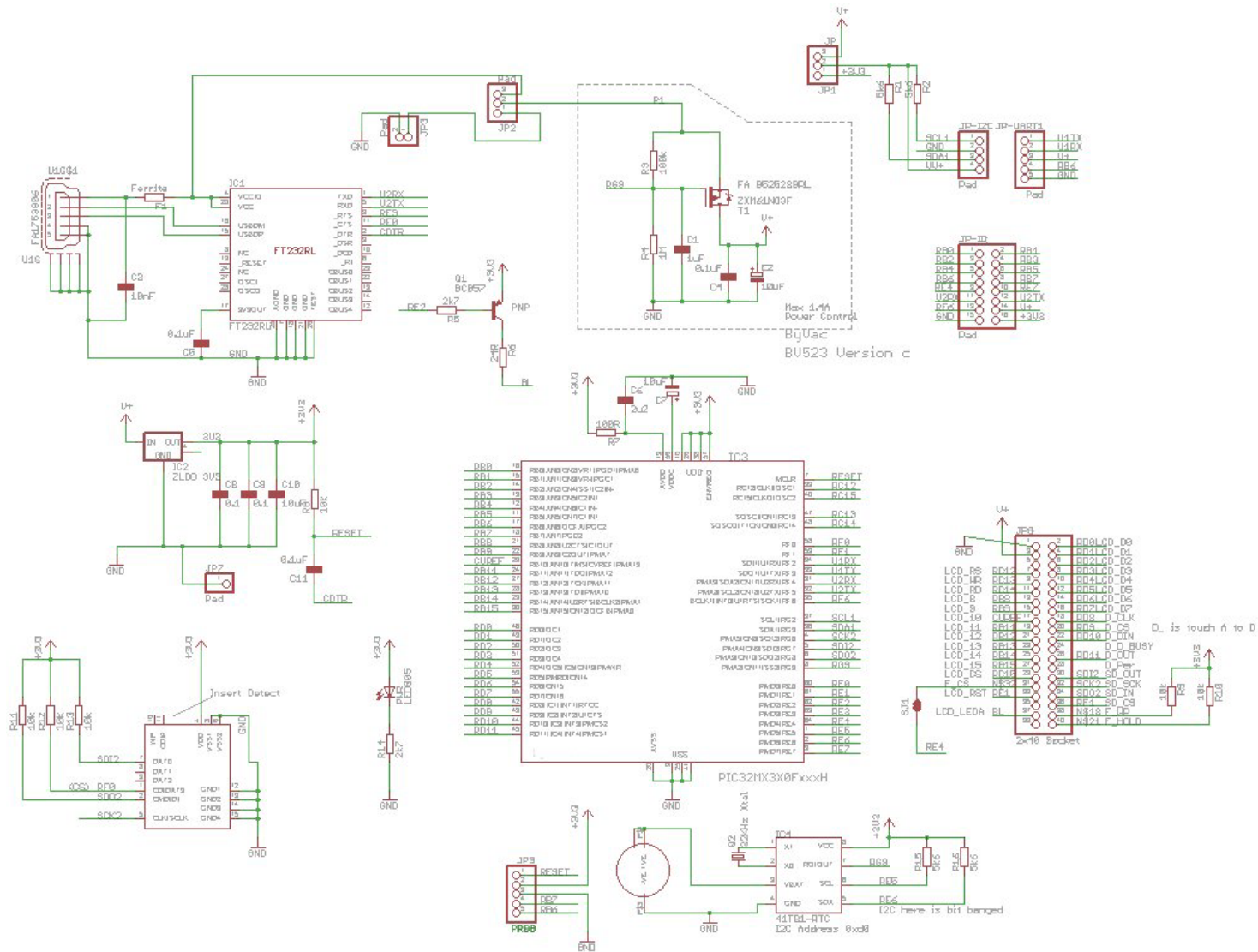
### 5. Circuit Diagram

The circuit diagram for PCB version c is shown on the next page



# Microcontroller

# BV523



Circ