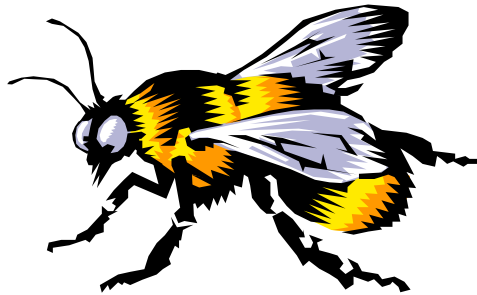


MAXI-BEE

Automation Adaptor

Installation and Users Manual
(Including Bee-Step64 Software)



Available exclusively from
PC Control Ltd.

www.pc-control.co.uk

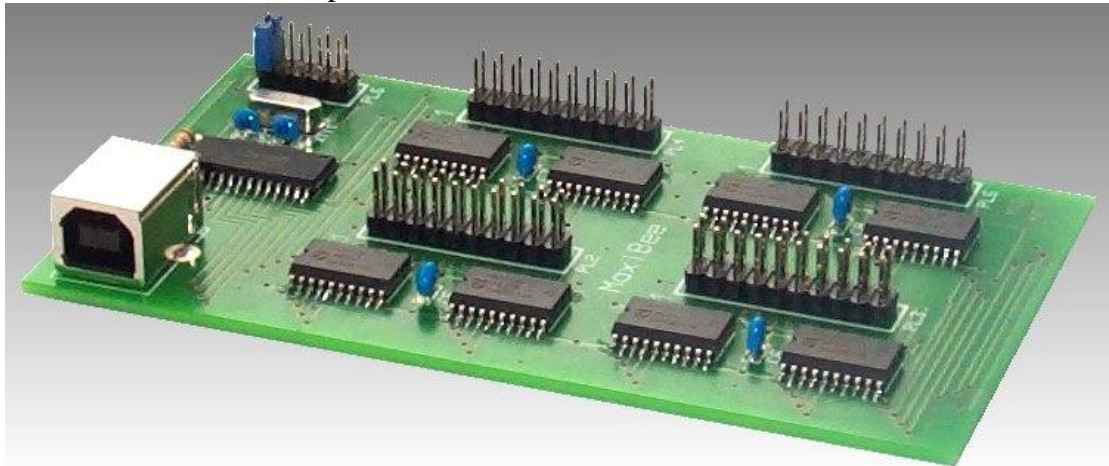
© 2009 Copyright PC Control Ltd.

Contents

- 1 Introduction
- 2 Hardware Installation
- 3 Bee-Step64 Application Software
 - 3.1 Overview
 - 3.2 Bee-Step64 Main Screen
 - 3.2.1 Main Sequence Display
 - 3.2.2 Step Editing
 - 3.2.3 Run Control
- 4 Connecting Devices to Maxi-Bee
- 5 Connector Pinouts
- 6 Writing Your Own Software to Use Maxi-Bee
- 7 Minimum PC System Requirements

1. Introduction

The MAXI-BEE is a versatile USB adaptor, which allows the PC User to explore the world of real time control and automation. It is a tool, which is attractive to both the novice and experienced user.



For the absolute beginner it can be used straight from the box as a flexible controller for a wide range of projects. The beginner can take advantage of the ease of connectivity of USB, making connection to the PC simple. The included Bee-Step64 application software allows the beginner to quickly create elaborate control sequences without any need for prior programming knowledge or PLC techniques.

For the intermediate user a DLL is provided to allow the programmer to construct their own software applications to take advantage of the MAXI-BEE hardware without having to know the details of USB communication protocols etc..

2 Hardware Installation

Simply connect the MAXI-BEE to any available USB port (*This will require a standard USB cable*). Although it will operate from bus powered hubs it is recommended that you connect it to a primary USB socket or a self powered hub. This allows MAXI-BEE to take full advantage of the available 500mA from such a connection. Bus powered hubs are limited to 100mA. Windows operating system will automatically detect and install the appropriate device drivers. The MAXI-BEE is regarded by Windows as a standard HID (Human Interface Device) which makes it very easy to install.

3.2.1 Main Sequence Display Window Sequence Summary List

The sequence summary window shows each individual step in a programmed sequence in summary form. Each line in the window corresponds to a single step in the sequencer. On the left of each line is the line number. This will always be consecutive from 1 to your last entered line. When editing the sequence list using insert or delete (discussed later), the line numbers will be automatically adjusted so that they remain in strict numerical sequence. This line number is used as the reference for the “next” facility (also discussed later) to control sequence loops and jumps. Immediately to the right of the line number is the duration. This is split into Hours, Minutes, Seconds and Milliseconds. The smallest duration is 100msec (0.1sec). The next 64 columns show the on / off status that the MAXI-Bee board outputs will use during this step. On is shown as ‘1’ and off as ‘0’. On the right hand end of the line is the “next” selection. This determines which line will be used after this one. A value of 0 in the next position indicates that the next line immediately below the current one will be used. For sequences larger than the sequence summary window can display, the window may be scrolled up and down using the scrollbar on the right

3.2.2 Step Editing

The step editing section immediately below the summary window allows individual steps to be added, deleted and edited. Each of the elements of a step is available in this area in editable form. The row of boxes along the top corresponds to the step’s on/off pattern of outputs. Green is off and red is on. Clicking on the box will toggle it between on and off. The duration of the step has individual edit boxes for hours, minutes, seconds and milliseconds. Using the adjacent arrows allows new values to be specified. In a similar way the value of the next step to be run can be edited. Note that the last step in a sequence will always cause the sequence to stop even if the “next” box is set to another location (eg to loop back). To avoid this add a dummy step after the one in which you wish to loop back.

To edit a step that already exists in the summary window simply click on that line in the window. This will cause a red underline to appear at that line and a copy of the step elements to appear in the edit area. Once editing is complete you can update the step in the sequence list by pressing the “Update” button. Alternatively you can check the “Auto Update” box and see the changes you make updated immediately.

To create a new step to be added to the end of the current sequence make sure the “Auto Update” box is unchecked, construct your step in the edit area and then click on the “Add” button.

To insert a new line somewhere in the middle of the current sequence, again make sure the “Auto Update” box is unchecked, click on the line BELOW the intended point of insertion and, when your editing of the step is complete, click on the “Insert” button.

To delete a line currently in the sequence list, simply click on the line and press the “Delete” button. To preserve the consecutive line numbering, all lines below the deleted line will be automatically moved up one and correspondingly re-numbered. Note that step 0 in the sequence list cannot be deleted or have other steps inserted above it.

Once a control sequence has been created to your specifications it can be saved to hard drive using the “Save” button. Files created in this way can later be restored using the “Restore” button.

3.2.3 Run Control

On the right of the main screen there are two buttons for run control, which are virtually self-explanatory. Clicking “Run” starts execution at the first step (0) in the sequence. It will continue to “run” until either the last step is reached or the “Stop” button is pressed. If the stop button is pressed, the current state of the outputs will be maintained. Care should be taken when arbitrarily stopping the running sequence since it may stop with an output permanently on that shouldn't be (for example it may leave a motor running continuously).

While running it is possible to see the current state of the outputs changing on the screen in real time mimicking the real outputs. This is selected by checking the “Mimic Display On” box. Normally there is no problem in running the mimic concurrently with the real time control sequence, however, if the control sequence involves very short duration steps and/or your PC is relatively slow it is better to turn off the mimic in favour of the real time control outputs.

It is good practice to test your sequence before actually applying it to the MAXI-Bee outputs. To do this, simply uncheck the “Use Maxi-Bee” box. The mimic will run but no outputs will change on the MAXI-Bee board.

4. Connecting Devices to MAXI-Bee

Four 20 way headers are provided for making connections to the Maxi-Bee. These each have 16 digital outputs and 4 grounds (0v). Since the outputs are standard CMOS digital they only require a good ground connection and a compatible standard digital logic input. The use to which these outputs are put is, obviously beyond the scope of this manual, but to assist in determining electrical compatibility and suitability to application, the data sheet for the output devices on the MAXI bee is included on the installation disk. These are 74HC573D devices. The output characteristics such as current source/sink capacity, switching speed etc.. etc.. are fully detailed in the data sheet.

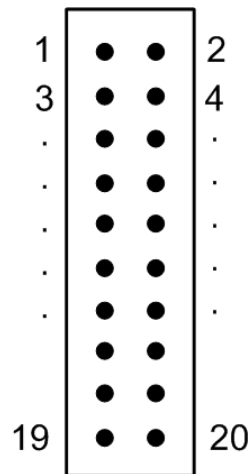
However, the following guidelines may be useful as a set of tips for the beginner.

1. Always ensure you have a good 0v (ground) connection between the MAXI bee and your own electronics. This is why 4 pins have been devoted to GND on each of the connectors.
2. Make sure the current requirements of the devices you are connecting to can be provided by the MAXI-Bee 74HC573 output devices.
3. A standard 20 way ribbon cable assembly with IDC sockets is the recommended way of connecting to MAXI-Bee. If you are connecting to your own electronics board then why not fit an identical 20 way header to your board allowing you to use one of the cheap and plentiful ribbon cable assemblies available from most electronics stores.
4. If you need to switch inductive loads such as motors or solenoids consider purchasing the “Bee-Driver” from www.pc-user.co.uk which converts a MAXI-Bee connector into 14 high voltage switching outputs with voltage spike protection. Or simply construct your own using devices such as the ULN2003 transistor array driver available from most electronics suppliers. With this type of conversion it is then very easy to scale up the power handling potential of the MAXI-Bee to unlimited amounts using relays which can be “driven” directly.

5 Connector Pinouts

Pinout of Connectors (PL2 – PL5)

Pin	PL2	PL3	PL4	PL5
1	Digital Output 1	Digital Output 17	Digital Output 33	Digital Output 49
2	Digital Output 2	Digital Output 18	Digital Output 34	Digital Output 50
3	Digital Output 3	Digital Output 19	Digital Output 35	Digital Output 51
4	Digital Output 4	Digital Output 20	Digital Output 36	Digital Output 52
5	Digital Output 5	Digital Output 21	Digital Output 37	Digital Output 53
6	Digital Output 6	Digital Output 22	Digital Output 38	Digital Output 54
7	Digital Output 7	Digital Output 23	Digital Output 39	Digital Output 55
8	Digital Output 8	Digital Output 24	Digital Output 40	Digital Output 56
9	Digital Output 9	Digital Output 25	Digital Output 41	Digital Output 57
10	Digital Output 10	Digital Output 26	Digital Output 42	Digital Output 58
11	Digital Output 11	Digital Output 27	Digital Output 43	Digital Output 59
12	Digital Output 12	Digital Output 28	Digital Output 44	Digital Output 60
13	Digital Output 13	Digital Output 29	Digital Output 45	Digital Output 61
14	Digital Output 14	Digital Output 30	Digital Output 46	Digital Output 62
15	Digital Output 15	Digital Output 31	Digital Output 47	Digital Output 63
16	Digital Output 16	Digital Output 32	Digital Output 48	Digital Output 64
17	GND	GND	GND	GND
18	GND	GND	GND	GND
19	GND	GND	GND	GND
20	GND	GND	GND	GND



PL2 - PL5
Pin Arrangement

6. Writing your own software for MAXI-Bee

To use MAXI-Bee straight from the box does not require any programming other than entering the step details into BeeStep64. However, if you prefer to design your own software then the following information will be of use.

Provided with MAXI-Bee is a DLL (dynamic link library) called “mxb.dll”. This encapsulates the functions used by BeeStep64 in communicating with MAXI-Bee across the USB interface into two simple functions easily understood and used in custom software. Although the DLL was written in ‘C’ it can be used (called) by programs written in a number of popular languages including BASIC (visual BASIC etc.). For the purposes of explanation the example usage below uses the ‘C++’ language, but additional information on using the DLL from a BASIC program is contained in the document “Using a ‘C’ DLL with VB4” which is included on the installation disk.

Ignoring some of the formalities in the construction of a Visual C++ program for the windows environment the techniques in using “mxb.dll” consists of four main tasks....

Loading the DLL into memory

Before any functions within the DLL can be used it is necessary to instruct windows to load it into memory. This is done by calling the LoadLibrary() function. i.e.

```
.....  
HINSTANCE BeeHandle; // declaration of variable to hold the handle to the dll  
.....  
BeeHandle = LoadLibrary("mxb.dll"); // load the dll into memory and return handle
```

The declaration of the variable BeeHandle used to store the handle to a DLL , uses a built in type definition which is called HINSTANCE in this particular 'C' compiler, but you should use the appropriate one defined in your own compiler for this purpose.

The LoadLibrary() function return a handle to the DLL if the load is successful otherwise NULL. Ideally your own program should check for a NULL returned and give an error message. Make sure the function parameter is the full pathlist to where you copied the bee.dll file from the installation CD.

Get the addresses of the functions within the DLL

Using the DLL handle returned above you can now obtain pointers to the functions within the DLL. Using the following

```
TypeInitMbee      InitMbee;  
TypeSetOutputs    SetOutputs;  
  
.....  
InitMbee = (Type InitMbee)GetProcAddress( BeeHandle, " InitMbee ");  
SetOutputs=(TypeSetOutputs)GetProcAddress(BeeHandle, "SetOutputs");  
.....
```

The TypeInitMbee and TypeSetOutputs type definitions are contained in the header file "mxbdll.h" and defines the correct type of function pointer to reference the DLL function. mxbdll.h should be included in your own source file eg.

```
.....  
#include "mxbdll.h"  
.....
```

The call to GetProcAddress() returns a pointer to this function if found within the DLL otherwise NULL. Once the functions pointers have been obtained in this way the internal functions within the DLL are simply accessed like ordinary function calls e.g.

```
.....  
InitMbee ();  
SetOutputs(0x00000000, 0x00001234);  
.....
```

Initialising The DLL

Once the addresses of the DLL functions are obtained as above the remaining functions required to use them are very simple. The first step is to initialise the DLL using....

```
int status;  
.....  
status = InitMbee();
```

Your program should check to see if a value of zero has been returned by `InitMbee ()`. Any other value indicates an error. E.g. MAXI-Bee not connected etc...

Using the SetOutputs() Function

The majority of your programming will use the `SetOutputs()` function. This simply applies the pattern 1's and 0's of the two 32 bit parameters directly to the outputs. For example: to create a pattern of alternate on and off over all outputs use..

```
SetOutputs(0x55555555, 0x55555555);           // hexadecimal number
```

Or to turn on just output 1 only use ...

```
SetOutputs(0x00000001, 0x00000000);           // hexadecimal number
```

Etc.....

More generally.....

```
unsigned long bits1, bits2;  
.....  
bits1 = 0x00001234;           // hexadecimal number  
bits2 = 0x00000000;  
SetOutputs(bits1, bits2);  
.....
```

This example will turn on outputs 13, 10, 6, 5 and 3.

i.e. the first parameter (bits1) corresponds to outputs 1 to 32 and the second parameter (bits2) to outputs 33 to 64. The least significant bit of each parameter corresponding to the lowest numbered output of each.

Although this only gives a glimpse of the possibilities of writing your own programs, it should be apparent that the use of the DLL functions greatly simplifies this process. It frees the programmer from the task of getting to know the fine details of programming USB interface communications and lets him concentrate on the main function of controlling switching outputs.

7. Minimum PC System Requirements

Maxi-Bee and BeeStep64 software do not require a high spec PC for correct operation, but the following system is suggested as a sensible Minimum

Processor	500MHz Pentium
Memory	64MB
HDD	10MB free space required
Screen Resolution	1024x768 (256 colours)
Interface	One free USB socket (1.0 or 2.0)
Operating System	Windows 2000, XP or Vista

WARNING: The Maxi-Bee adaptor board is intended for standard 5 volt digital signals. It should not be connected directly to mains voltages under any circumstances.

Terms of Use for all Goods Supplied

Definitions

'Supplier' shall mean PC Control Ltd.

'Buyer' shall mean the person, company or any other body that purchases or agrees to purchase Goods.

'Goods' shall mean all goods and services which the Buyer agrees to buy from the Supplier including replacements for defective Goods, hardware, documentation and software products licensed for use by the Buyer.

Use of the Goods in any way by the Buyer constitutes acceptance of these terms and conditions.

Terms and Conditions

1. The Goods are intended to be part of the buyer's own design of apparatus and not a finished product in their own right.
2. The Goods supplied are not to be used in any design where there is a risk, however small, either directly or indirectly, of death or personal injury.
3. The Buyer will be responsible for ensuring the fitness for purpose of the Goods for the Buyer's application.
4. To the extent permitted by law, the Supplier accepts no liability whatsoever or howsoever arising in respect of loss, damage or expense arising from errors in information or advice provided whether or not due to the Supplier's negligence or that of its employees, agents or sub-contractors save for any loss or damage arising from death or personal injury.
5. To the extent permitted by law, the Supplier shall not be liable to the Buyer by reason of any representation (unless fraudulent), or any implied warranty, condition or other term, or any duty at common law, or under the express terms of any Contract with the Buyer, for any indirect, special or unforeseen loss or damage (whether for loss of profit or otherwise), costs, expenses or other claims for compensation whatsoever (whether caused by the negligence of the Supplier, its employees or agents or otherwise) which arise out of or in connection with the supply of the Goods or their use or resale by the Buyer.
6. The entire liability of the Supplier under or in connection with the Contract with the Buyer shall not exceed the price of the Goods except as expressly provided in these terms and conditions.
7. These terms are an important part of the full terms and conditions of business as published on the website at www.pc-control.co.uk/general-terms.htm which also apply.

If you cannot agree to the terms and conditions of use of the MaxiBee then you should return the MaxiBee to the supplier within 7 days of receipt to receive a refund. Your use of the board or the associated software in any way whatsoever will be regarded as an acceptance of these terms and conditions.