

---

# **PIC PORTABLE PROGRAMMER QUICK START GUIDE**



---

## **IMPORTANT INFORMATION**

1) Do not leave the programmer connected to the PC adapter or a target system, as this will drain the battery.

---

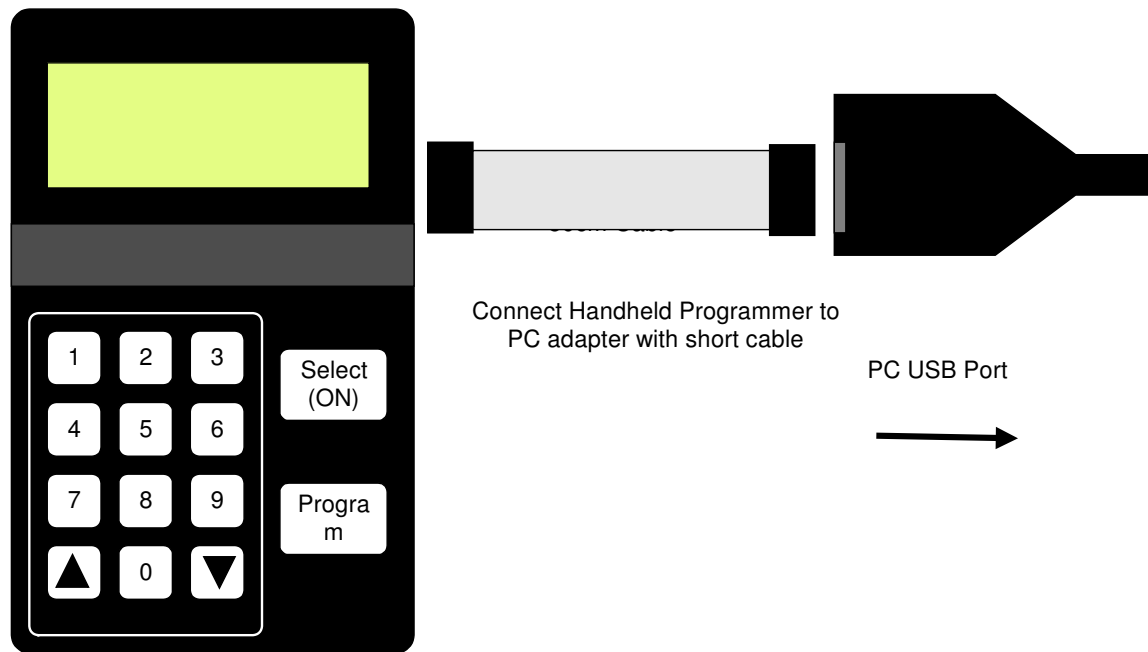
## **Installing Software**

- 1) Run the executable file on the CD. This will install software and pre-install USB drivers.
- 2) Plug in PC dongle to a USB port. The drivers should install automatically. If they do not, see Troubleshooting section
- 3) If you get message about Settings Files, see Troubleshooting Section.

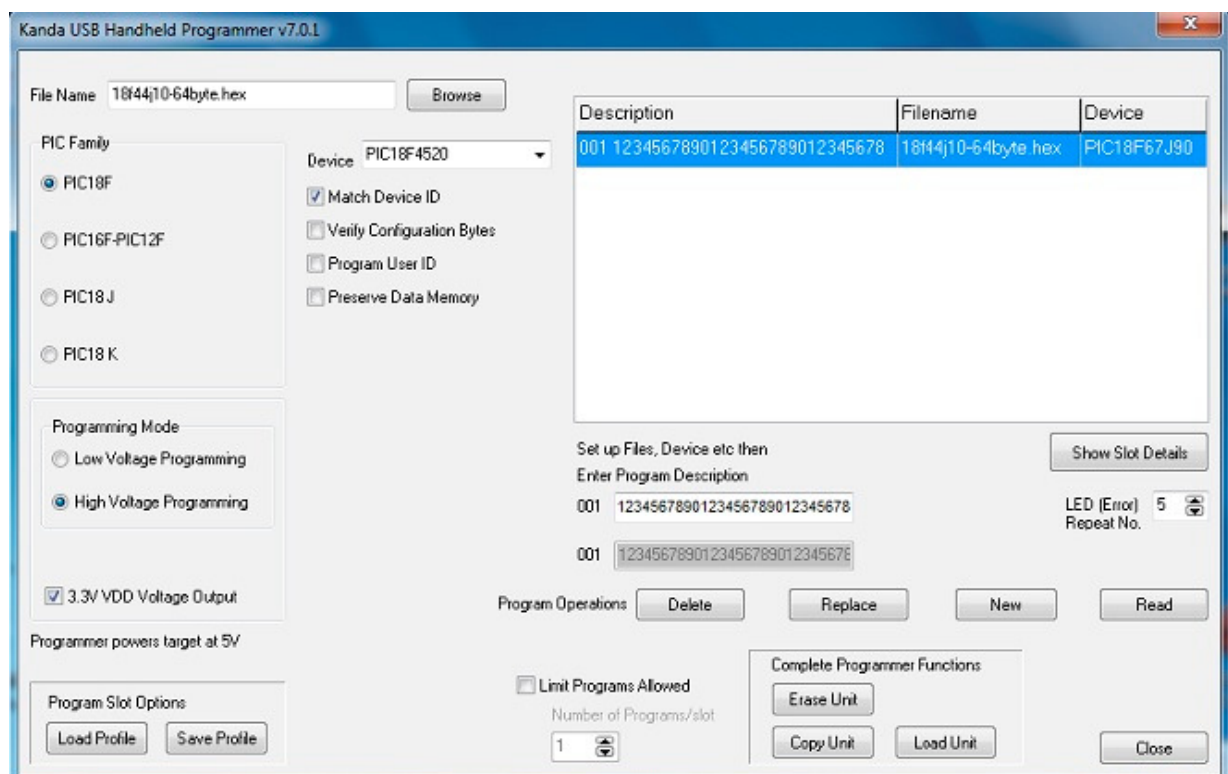
---

## **LOADING FROM THE PC**

- 1) Connect the PC adapter (dongle) to the PC USB port.
- 2) Connect the PC adapter directly to the programmer using the short ribbon cable.



### 3) Run the Portable Programmer software



The first stage is the software reads the programmer. If a programmer is not found, it will prompt for Abort or Retry or Ignore

- **Ignore.** Opens programmer in Demo mode. All operations that do not require a connected programmer, for example load Profile will work. Connect programmer and click Read button to exit demo mode
- **Abort.** Closes programmer software

- **Retry.** Tries to connect to programmer again

If the programmer needs a firmware update this will be carried out.

Then the screen shown above will appear and the programmer LCD will say "Loader Connected", unless in demo mode.

- Select the file to use in the filename box.

**Notes:**

The programmer expects a Microchip type hex file (as produced by MPLAB). The file should contain

- Program Code
- EEPROM data – if PIC device has EEPROM and data is required
- Configuration Bytes including Security information
- User ID if required

***See section on Hex File formats for more details***

- Now select the device family required (16F or 18F) and the device type
- Choose the Programming Mode, Low voltage (LVP) or High Voltage (HVP). If you choose LVP, you must connect LVP pin – see section on ***Low Voltage Programming.***

**Note:** Not all PIC devices have LVP mode – check device datasheet

- There are also four check boxes.

1. **Match Device ID.** Checking this box means the programmer will read the device ID and compare it with the ID of the selected device to ensure they match. Some newer device where there is incomplete information could fail this test. ***See section on Error Codes.***
2. **Verify Configuration Bytes.** Allows configuration byte programming to be verified. ***See section on Error Codes.***
3. **Program User ID.** User ID bytes are programmed – PIC18F only. The User ID bytes must be included in the Hex file or all locations will be programmed with 0xFF. ***See section on PIC18F Hex File formats for more detail***
4. **Preserve Data Memory.** Existing contents of device Data EEPROM are preserved.

- **Program Options.** Some PIC16F devices can have different options set. Program the whole device (Program All), program just the Code or just the data. Program Data only will not work if CPD bit is set – Protect Data Memory.
- **3.3VDD Voltage Output:** For LF devices or circuits where 5V could cause damage. 3.3V is default for PIC18FxxJ and PIC18FxxKxx devices

- Now give it a description in the **Program Description** Box
- The final step is to click the **Load Program** button to load the Portable programmer.

**Your Programmer is now loaded with the first program.**

***If you get errors with connecting to the programmer, see the troubleshooting section at the end of this guide.***

## Profiles

Profiles will save program settings, including family, device, filename, program options and description. File extension is **.PRJ**. Use **Save Profile** to create a profile and **Load Profile** to load a profile.

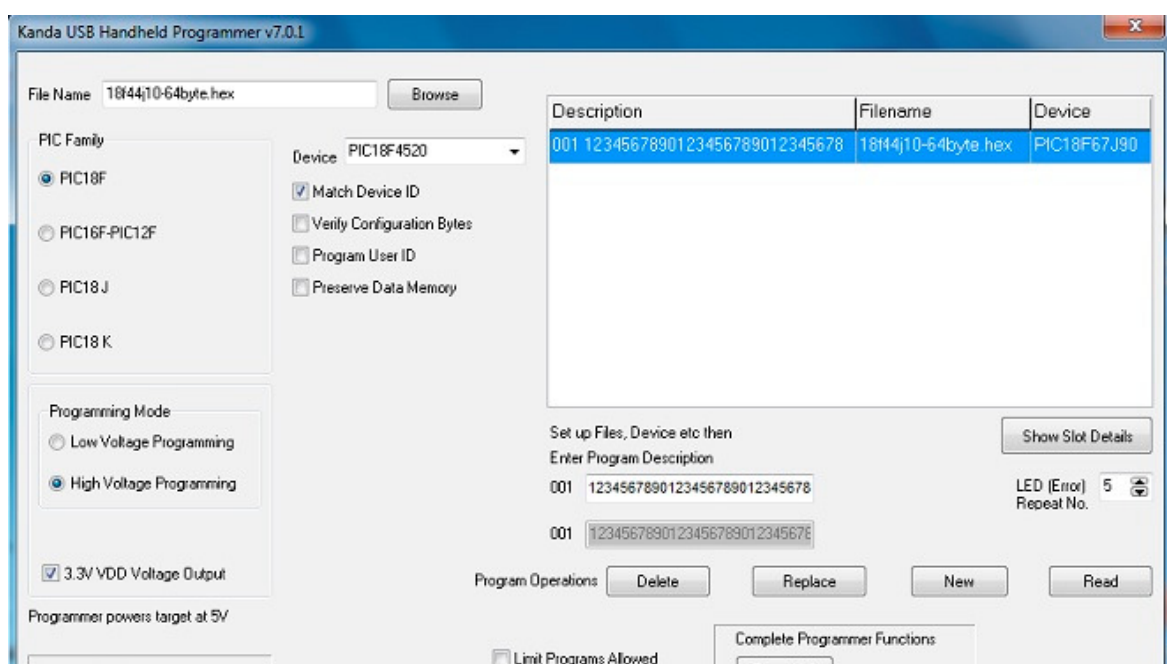
## Adding further programs, deleting and replacing programs

To add more programs, select the device type, filename, description and all settings as listed for first program above, or load a Profile. Then click **Load button**.

To replace an existing program, select it in the program list. Then change filename and whatever else needs changing and click **Replace** Button.

To delete an existing program, select it in the program list and click **Delete** Button. It will now appear in list as **Empty**

**Note:** When PC is connected, the keypad and select button on programmer are disabled and slot selection is done on program list on screen.



When the programmer is fully loaded – 32 programs, then the **Load Program** button is disabled. Existing programs can be edited by selecting them in program list and using **Replace Button** or removed using **Delete Program Button**. An Empty slot can only be replaced not loaded.

The existing program description of the program selected in the list is shown in the read-only shadow box, or Empty appears if slot is unused. Enter new description in Program Description box.

## Duplicate Names

When you load a new program or replace a program, a check is done for existing filename or description. If a duplicate is found, you are prompted on whether you want to continue or not.

## Reading Programmer

When software is run, it reads the programmer and displays the programs in it in the program list. It also updates the program list after an edit or new program.

If you change programmer hardware without closing software or want to change from Demo mode, then use the **Read Button** to refresh the program list. This will also check if the new programmer needs a firmware update.

## Saving Program List

Right click on Program list and choose Copy to Clipboard, then paste it into a text editor. This only saves descriptions, filenames and devices, not programmer. See Copy Unit for copying programmer.

## Show Slot Details

Clicking the Show Slot Details button will update all the settings (filename, device, etc.) with the details in the currently selected slot. To show details of another slot, click on it and click Show Slot Details button again. If programmer is empty, button is disabled.

It can also be accessed by a right mouse click on programmer display box, which gives a Popup menu. This menu also has Copy functions.

---

## Advanced Functions

**Erase Unit.** All programs are erased after prompt.

**Copy Unit.** The whole contents of the programmer are copied to a programmer file, with .PRG extension. This can be a long process, especially if programmer is full, and can take half an hour.

**Load Unit.** The programmer is loaded with an existing programmer file (.PRG). Do not load ordinary hex files as these will not work.

**Find Name.** A Find Dialog appears where you can enter a filename or description. It searches from the beginning of the program list and if a match is found, it scrolls to that position in the list. Options are to search for a partial match and case sensitivity.

## Using Programmer Keypad and LCD

Switch programmer on using Select (ON) button

If the PC loader is connected, LCD says "Loader Connected" and keypad is disabled.

If programmer is empty, LCD says "No programs" and beeps.

Normal operation is the first program description appears on LCD with a 3-digit number eg 001 Unit 1 Assembly.

## Selecting programs

**Method 1:** Scroll through program list using arrow keys until you get to the one you want and then press **Select** button. Scrolling will only go as far as last used slot and then return to start. Scrolling down will go from first entry to last entry.

**Method 2:** Enter a three digit number for the program slot you need and press **Select**. All numbers must be 3-digit eg 001 or 031. If you enter a number for an empty program slot, LCD message will be "Not available"

You must press **Select** before pressing **Program**. If you do not, programmer will give an error beep and display a message.

## Switching programmer Off

Press the two arrow keys simultaneously and hold them down until power goes off. The programmer will automatically switch off after 5 minutes if no keys are pressed and it is not connected to PC or target.

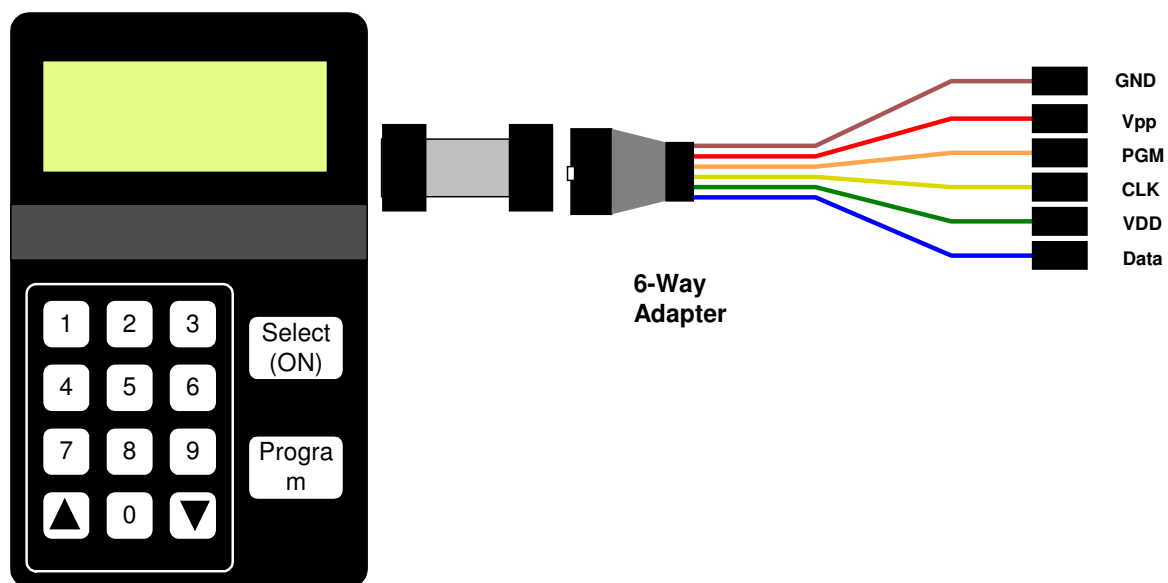
## CONNECTING TO A TARGET

### Target Power

The target circuit can be powered or not, it does not matter. The only problem is if you select 5V VDD in software and power circuit at 3.3V. Otherwise, it does not matter. You must connect target GND to programming header but VDD can be omitted.


### 6-way connector

The programmer is supplied with a 10-way to 6-way flying lead adapter. Only 5 of these connections need to be made for HVP programming, as PGM pin is only used for LVP programming. Connect the adapter to the programmer as shown in the diagram



**Note :** The 10 to 6-way flying lead adapter connects all ground pins on 10-way connector. If you make your own adapter, it should also connect all the GND pins on 10-way connector.

### Adapter 10-way connector



N/C	9	10	VDD
PGM	7	8	GND
MCLR/Vpp	5	6	GND *
Clock - PGC	3	4	GND *
Data - PGD	1	2	GND

This diagram shows 10-way connector on adapter viewed as shown in the picture. If you make your own adapter instead of using the supplied one, you will need to

- Match this pin-out
- **GND\*** These pins must be connected to target ground
- **GND** One or both must be connected to Target ground
- **VDD** must be connected to target circuit Vdd
- **PGM** pin is only connected for LVP programming

### Connecting Directly to programmer

If you want to make your own lead that connects directly to programmer instead of using the 10-way lead supplied, then please be aware that the supplied lead is NOT wired pin 1 to pin 1 so the connector on the programmer is not the same as the adapter, and is shown below.



<b>GND</b>			<b>PGD</b>
<b>MODE</b> (GND)			<b>PGC</b>
<b>BAT</b> (GND)			<b>MCLR/ VPP</b>
<b>GND</b>			<b>PGM</b> LVP Only
<b>VDD</b>			<b>N/C</b>



## **RJ11 Adapter**

An adapter is available on our website that connects to the end of the 10-way lead and has a RJ11 plug for connecting to RJ11 socket.

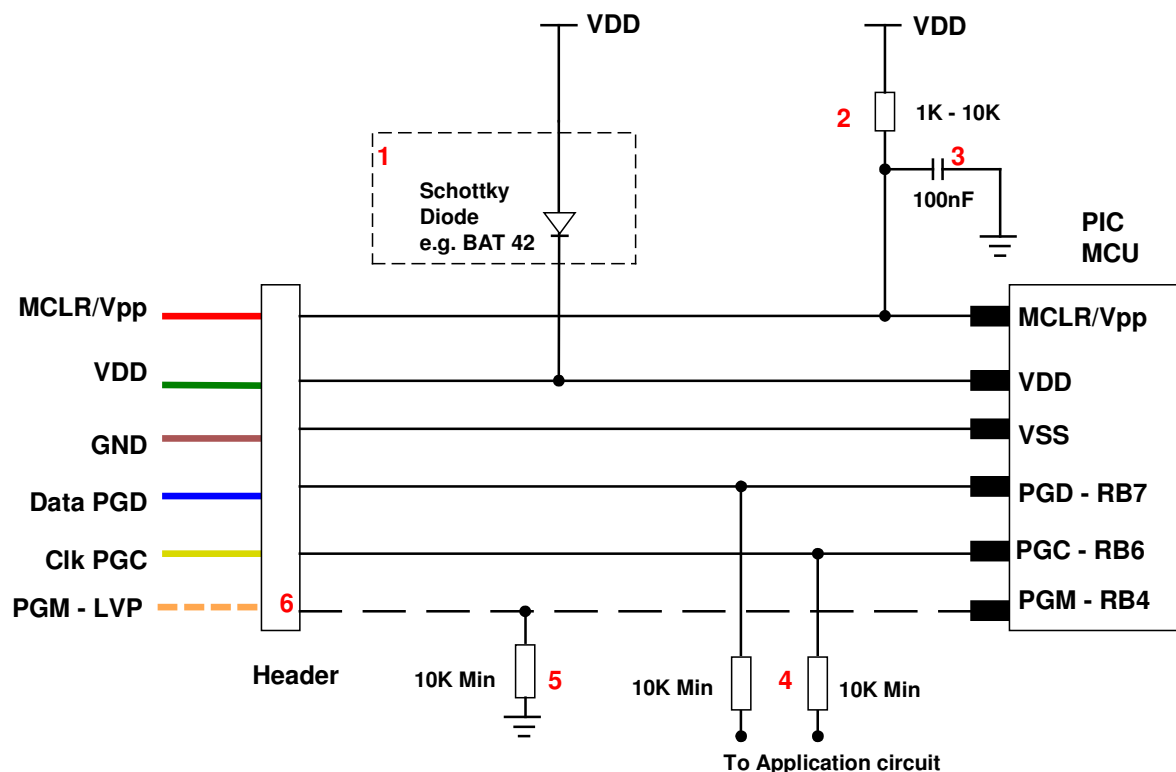
See <http://www.kanda.com/go/10RJ11>



## ICSP CIRCUIT REQUIREMENTS

Microchip do not recommend any particular circuit for ICSP programming. There are diagrams for different tools, such as Pro Mate and PICKit2 with similar circuitry but slight variations.

Kanda have produced a recommended circuit that will work effectively with our programmer. This circuit is shown in the diagram below. Please read the notes that explain the circuit and expand on the effect of extra components such as capacitors.



- 1) The programmer is designed to provide 5V to the target circuit. If your circuit operates at a lower voltage then the diode shown on VDD should be fitted to protect the rest of the circuit. A series resistor may be acceptable instead of the diode in some cases.

The maximum current that the programmer can supply is limited to 150mA, so you should fit the diode on VDD if the programmer over-current circuit trips - see *error codes section*

- 2) MCLR/VPP pin needs the resistor to VDD. A minimum of 1K should work but 10K is the optimum. Devices with only VDD first ICSP entry (PIC16F8x/87x/7x/7x7) should have 4K7 minimum to avoid code run problems preventing ICSP entry, but 10K is safer. Supervisory circuits or push buttons on MCLR should be isolated from the VPP voltage, by placing them on the VDD side of the resistor or by fitting a Schottky diode on this line as per note 1.

- 3) The 100nF capacitor shown on this pin is optional for HVP but we do recommend that a capacitor is fitted if possible to avoid glitches on MCLR. This is the **maximum** value, and we recommend something smaller. Larger capacitors may prevent the PIC from entering HVP mode. Do not fit for LVP mode.
  - 4) If possible, the Clock and Data lines should be dedicated to ICSP but where this is undesirable, the application circuit should be isolated from the data and clock lines with series resistors, above 10K. This is especially important if either of these lines forces the pin as an input or output. In exceptional cases, series resistors may not be sufficient and a 4053 multiplexer or similar circuit should be used.  
  
**Capacitors** on these lines should be avoided if at all possible. If they are needed, for noise immunity for example, then the maximum capacitance the programmer can handle is 1nF.
  - 5) If LVP mode is used, this resistor must be fitted.
  - 6) This line is only needed for Low Voltage Programming – LVP mode, except **PIC16F62x and PIC16F62xA**, when it should be connected or PGM pin must be grounded for HVP mode.
  - 7) PIC18F J parts need a decoupling capacitor between VccCore pin and Ground, typically 100nF
- 

## High Voltage Programming

The High Voltage Programming method or HVP mode uses 9V or 12V on the MCLR/VPP pin to enter programming mode. The programmer includes a 9-12V generator circuit that supplies this voltage.

There are two methods of entering HVP mode, Vpp before Vdd and Vdd before Vpp. Both methods are available on some PIC devices, but others only have one of the methods. The programmer will use Vpp first if it is available as it has advantages such as being independent of code in the PIC. This method will only work if the target is NOT powered.

The PGM line does not need to be connected for HVP mode, but should be grounded for best practice.

The programming mode must be set to High Voltage Programming (HVP) in the software.

---

## Low Voltage Programming

Only some PIC devices have 5V programming – LVP – option. Consult the device data sheet if you are unsure.

Assuming the PIC device has LVP mode, then you need to connect the PGM pin on the PIC to the header and connect the orange programmer line to this header pin. The pull-down resistor on the PGM line must be fitted for LVP to function correctly – see note 5 on diagram.

### Notes:

- 1) The LVP enable fuse must be set – in CONFIG4L register. Factory default is LVP enabled. If you program this fuse off, then only HVP mode can set it again.
  - 2) The programmer software must be set to Low Voltage mode - LVP
- 

## Press the Program button to start programming.

Programming progress is shown on LCD. If programming was successful, LCD gives success message, otherwise an error message is displayed.

Press 0 key to clear message and set programmer for next programming cycle.

Do NOT leave the programmer connected to target circuits for long periods as this will flatten the battery.

---

## HEX FILE FORMAT FOR PIC18F DEVICES

The programmer loads a Microchip format Hex file. This is an extended Intel Hex file with code, data, configuration bytes and user ID included. All PIC tools should follow this format.

Each line in the file has this format:

**:BBAAAATT[DDDDDDDD]CC**

where

: is start of line marker

BB is number of data bytes on line

AAAA is address in bytes

TT is type. 00 means data, 01 means EOF, 02 means linear address and 04 means extended address

DD is data bytes, number depends on BB value

CC is checksum (2s-complement of number of bytes+address+data)

- Code:** This is at the top of the file and may be preceded by an extended address line - :020000040000FA, where 04 is the type for extended address. Each extra 64KB block (maximum address with standard Intel Hex is FFFF, 64KB) will have an extended address marker, like :020000040001F9, :020000040002F8 etc. Some compilers use linear addressing instead for each 64KB code block, which uses 02 type, eg :020000021000EC, :020000022000DC.  
 Some compilers include empty code lines (all FF) but others omit these lines to save space
- EEPROM Data:** It is preceded by the extended address line - :0200000400F00A. The EEPROM section is optional
- Configuration bytes:** These are stored at 300000h and are preceded by the extended address line - :020000040030CA  
 The correct format is 8 Fuse bytes and 6 Lock bytes all on the same line but different compilers and assemblers have different methods of displaying these bytes. Sometimes lock bytes are omitted if they are not set, sometimes the data is spread over multiple lines.  
 The standard format displays unused bits as 1 (e.g. FF for an unused byte) but on the PIC device they read as 0. The programmer masks unused bits to 0 so that the Configuration Byte verify will be correct.
- User ID:** These are bytes for the user to store data, such as code version numbers. They are stored at 200000h. Again they are preceded by the extended address line :020000040020DA  
 The standard format requires 8 bytes but again some compilers omit unused bytes.
- End of File:** The End Of File marker for all Intel Hex files is :00000001FF

## Example

```
:10000000DEAE5A2336A27D0FF6C85C30E50DD29DD8
:1000100060B4243B43C9ECB7FC642523C2AFE43B86
:100020002CD2C7AAAA41F7D198FA51DCE5CC3039D5
:0200000400F00A
:10000000DDDDDDDDDDDDDDDD0DFFFFFFFFFFFFFFFFFE0
:10001000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF0
:020000040030CA
:10000000FFFBFDFFFFFFFFFFFFFFFFFFFFFFFFF07
:020000040020DA
:0800000011111111FFFFFFFFFB8
:00000001FF
```

Not always present  
Code line  
Code Line  
Code Line  
EEPROM Marker  
EEPROM Line  
EEPROM Line  
Config Marker  
Config Bytes  
User ID marker  
User ID bytes  
End of File

## HEX FILE FORMAT FOR PIC16F DEVICES

The programmer loads a Microchip format Hex file. This has a similar format to the PIC18F file but does not use extended addressing as the PIC16F devices are smaller and EEPROM and Configuration Word data is stored below 64KB, FFFFh.

- **Code:** Code is always at the top of the .hex file. The layout varies with different compilers and assemblers e.g. amount of data per line, whether blank lines are included etc. Note that the PIC16F devices use 14-bit instructions, so code is stored as Words with low byte first. Therefore, an unused location appears as FF3F. Addressing is in bytes though.
  - **EEPROM Data:** If the device has EEPROM, the data is stored at address 0x4200 upwards in the HEX file. It is stored in word format but only the lower byte contains data – the high byte is always 0 and is discarded.
  - **Configuration Word:** There is only one 14-bit configuration word, stored at address 0x400E. It is stored high byte first.
  - **End of File:** The End Of File marker for all Intel Hex files is :00000001FF
- 

## BATTERY, POWER SUPPLY AND LO-BAT LED

The Portable Programmer uses a rechargeable **9V Ni-Metal-Hydrade PP3 battery**. These are commonly available.

The Portable Programmer can also be powered from an external Power Supply, which will also power battery charging circuit. The power supply included with the programmer is

- 2.1mm barrel connector (coaxial plug), centre positive.
- 15V DC regulated
- 500 mA plus

**ONLY USE A RECHARGEABLE NI-MH BATTERY AS PROGRAMMER INCLUDES RAPID CHARGING CIRCUIT**

Spare batteries are available on our website, order code is BAT-PP

### Charging Time

A flat battery will fully charge in approximately 2 hours. The unit can still be used if it is charging, either for loading from PC or programming a target. It can also be used with the power supply plugged in.

**Note:** Ni-Metal-Hydride batteries will self discharge over a period of 2-3 weeks, so the programmer will need charging if left for this period.

## LED Behaviour

- Red Flash – Battery needs charging.
- Green Flash – Battery is charging from Power Supply
- Solid Green – Power Supply plugged in, battery fully charged.
- LED off – Normal operation, battery ok, no power supply

---

## ERROR Codes

Message	Meaning
Programming OK	Programmed successfully
Program FAILED	General failure, usually means device did not enter ICSP. Check connections and for faulty target board
Data Verify FAIL	Data EEPROM failed to verify
Code Verify FAIL	Code memory failed to verify, see <i>note 1</i>
User verify FAIL	User ID Bytes failed to verify
Cfg verify FAIL	Configuration Bytes failed to verify
Device ID WRONG	Device ID of target does not match device set, see <i>note 1</i>
Vcc Over-Current	Vcc (Vdd) current limit exceeds 150mA, see <i>note 2</i>

**Note 1:** PIC microcontrollers do not have any way of indicating that they are in ICSP mode, so a failure to enter ICSP mode (because of bad connections, faulty board, low voltage etc) will only be shown by first programming operation. If Match Device ID is set, then message will be Device ID WRONG. If you are sure device is correct, uncheck Match device ID and retry and fault message will be Code Verify FAIL. So, both of these can actually mean connection or other problems.

**Note 2:** Target is drawing more than 150mA from programmer, so check target for shorts on Vdd. This can also be caused by faulty ICSP circuit design, See section on ICSP CIRCUIT REQUIREMENTS, page 9.

---

## TROUBLESHOOTING

### Windows driver problems

If you get a driver error, make sure that you have run the install software on CD and then plugged in programmer, NOT the other way round.

The driver should appear as a "USB Serial Converter" in USB section of Device Manager. Windows should do this automatically. If it does not, please follow this procedure.

- Plugin Programmer and ideally remove other USB devices
- Go to Control Panel > System > Hardware screen
- Click on Device Manager button
- Open USB section and select "USB Serial Converter"
- Right click on it, and select Update Driver
- Driver location is (default install path)  
C:\Program Files\embres\PICHPNEW\driver\driver

### **Error Message: "Programmer is not responding"**

- 1) Check dongle is attached to USB port
- 2) Check that programmer is connected to PC as shown on Page 1
- 3) Check battery or power supply to the programmer

### **Cannot Find Settings File**

From V4.9.6 onwards, application data including device support, firmware and program settings have been moved to Common Application Data folder. The location of this folder is different on various Windows Operating Systems.

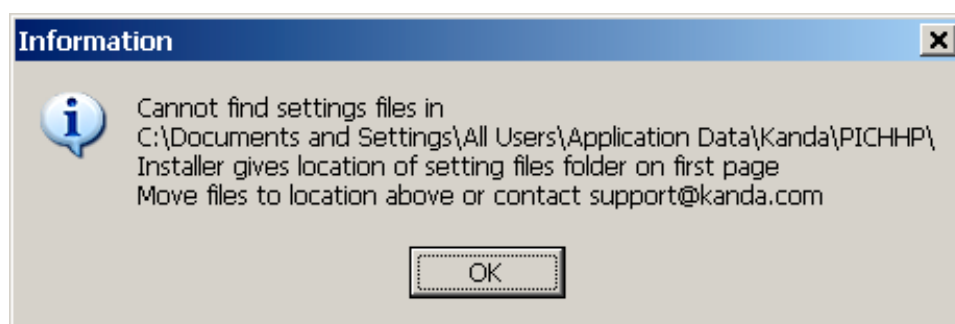
#### **Windows XP, Windows 2000 :**

C:\Documents and Settings\All Users\Application Data\Kanda\PICHP

**Windows 7 :** C:\Program Data

**Vista:** may be either of these

If there is a problem, you will see this message



Write down the path given in this dialog for later, and click OK. It will try and do a Firmware Update, so Cancel this.



This gives the path where program is looking for settings files. If you run the installer again from CD, on the first page it gives location where installer is saving these files. If they do not match, you will need to copy files to location shown above.

### **XP Note.**

On some XP systems, this path is hidden, typical Microsoft! In order to see it, you will need to go to **Tools** menu in My Computer or Explorer and select **Folder Options** and click on **View Tab**.

- Select Show Hidden Files and Folders
- Uncheck Hide protected operating system files

### **Programming Errors**

- 1) Ensure your target circuit is wired correctly as discussed in the *ICSP Circuit Requirements* section.
- 2) Check you have correct device selected, especially if ID match is turned on
- 3) Ensure you have selected LVP or HVP as appropriate
- 4) Check programmer is powering target PIC with 5V on VDD pin and 12V+ on VPP pin for HVP or 5V on PGM pin for LVP, during programming
- 5) Check lead connections on programming header are correct
- 6) PIC16F devices with only VDD first method of entering ICSP (PIC16F8x/87x/7x/7x7) can fail to enter ICSP (2 flashes) if resistor on MCLR is stronger than 4K7 eg 1K and user code carries out some operations before VPP can rise. On other PIC devices, 1K resistor on MCLR is fine.
- 7) If you have made your own adapter, check all ground pins on 10-way end are connected (see target layout diagram). Note programmer header has different pin out from end of lead as cable is not wired pin1 to pin1. It is mirrored about short axis.

---

## **FURTHER INFORMATION**

Please contact [support@kanda.com](mailto:support@kanda.com) for technical support or go to our website support pages for latest software.

See [www.kanda.com/support](http://www.kanda.com/support) > Software Downloads

### **Contact details**

**Website:** [www.kanda.com](http://www.kanda.com)

**email:** [sales@kanda.com](mailto:sales@kanda.com)

**Phone/Fax: +44 (0)1974 261 273**

---



*PICmicro, Microchip, ICSP are registered trade marks or trade marks of Microchip Corp, Az, USA*