



# CIFER

## **HD** Standards Converter

## Update Instructions

© Pro-Bel Ltd

Issue: 1.0  
Date: April 07

**Pro-Bel Ltd  
Hartman House  
Danehill  
Lower Earley  
Reading  
Berks, RG6 4PB  
ENGLAND**

Web: [www.pro-bel.com](http://www.pro-bel.com)



## Contents

---

<b>Contents</b>	<b>2</b>
<b>1 Introduction</b>	<b>3</b>
<b>2 Firmware Principles</b>	<b>4</b>
2.1 V6481 Standards Converter (SC)	4
2.1.1 Operating code for the H8 Microcontroller	4
2.1.2 Object Code for the FPGAs	4
2.1.3 Data Tables	5
2.2 V6482 Motion Estimator (ME)	5
2.3 V6482 ME Preparation	6
<b>3 Updating</b>	<b>7</b>
3.1 V6481 Standards Converter	7
3.1.1 Memory Stick – New Version	7
3.1.2 Memory Stick – Old Version	8
3.2 V6482 Motion Estimator	8
3.2.1 Board Switches	8
3.2.2 Connection	9
3.2.3 Upgrade	9
<b>4 Testing</b>	<b>11</b>
4.1 Version Numbers	11
4.2 Pictures	11



## 1 Introduction

---

CIFER is a very high quality television Standards Converter. It operates with both High Definition and Standard Definition pictures and uses Motion Compensation to remove the blurring or judder that is usually associated with Standards Converters. Cifer converts between video signals operating on different field rates, usually 50Hz as used in Europe and 59.94Hz as used in the USA. Other parts of the world use one or other of these field rates. There is a multitude of HD standards and Cifer will also convert between them, but the field rate conversion is a much more complex operation than converting between 1080i and 720p operating at the same field rate.

Both the performance and feature set of Cifer are being continually improved, so it is necessary to upgrade the system in the field. This document describes how this is done.

Cifer has two main processing boards – V6481 Standards Converter and V6482 Motion Estimator. These are referred to as the SC and ME (or sometimes MEC) boards. Apart from the titles on the front panels they can be easily identified since the SC has a front panel display and several push buttons while the ME has only a couple of LEDs and a USB socket.

This document has three parts. The first part describes in some detail how the units use their software and firmware, and how and where the data is stored. By having a full understanding of the operation of each board the user will find it easier follow the upgrade procedure. The second part describes how to set up a PC or Laptop In preparation for updating the ME board. Finally the procedure for actually upgrading both the SC and ME boards is shown in some detail.

Since there are two separate boards needing to be upgraded there may be a problem with incompatibility between the firmware versions on each board. For this reason every release will be accompanied by a record of released versions showing compatibility.



## 2 Firmware Principles

---

### 2.1 V6481 Standards Converter (SC)

The SC board has three programmable parts:

- Operating code for the H8 Microcontroller
- Object code for the FPGAs
- Data Tables for conversion details.

All these data blocks are stored in a memory stick which is fitted to the main board of the V6481 but they are used in different ways

If there is a change to any or all part of any of these data blocks, then a new Memory Stick with a printed version number will be issued. The fact of a new memory stick does indicate how significant the changes are.

#### 2.1.1 Operating code for the H8 Microcontroller

Normally the H8 actually operates from code stored in its own internal PROM memory. However it is also able to re-programme itself from the H8 code in the memory stick.

After power-up or a reset the H8 looks at the H8 code on the memory stick and reads the version number. If it is newer than the code inside the H8 it will start the re-programming sequence, but if it is the same version or earlier it will ignore the memory stick and carry on with its own code.

While re-programming itself a simple progress bar will be shown on the front panel display.

It is possible to force the H8 to re-programme itself, which may be needed to return to an earlier version, for example. This is done using the front panel buttons and forcing a reset.

One consequence of the H8 always running its internal code is that it can report errors on the front display, even if it is unable to read the memory stick or the stick's contents have been badly corrupted.

The H8 will only re-programme itself if the H8 data itself has a newer version number. It does not care if the other data types on the memory stick have changed or not.

#### 2.1.2 Object Code for the FPGAs

Modern processing equipment is heavily dependent on FPGAs (Field Programmable gate Arrays). These are general purpose devices that are programmed after power up to perform

# CIFER Update Instructions



a particular function. The working programme is stored inside them in volatile RAM so the contents are lost after a power down. Because the data is stored in RAM it is also possible to re-programme them 'on the fly', perhaps to do a slightly different function. This feature is sometimes used when operating video equipment in different standards.

Because the storage in the FPGAs is volatile the data must be stored somewhere else so that it is not lost through a power down. As with the H8 data it is also stored in the memory stick.

Once the H8 has performed its own internal boot up and diagnostics it will load up the FPGAs from data stored on the memory stick. A suitable message will be shown on the display while this is happening.

In fact the Standards Converter board has two FPGAs and they are programmed sequentially by the H8, but this is transparent to the user.

The FPGA code has a version number embedded with itself. Once the FPGA has been programmed this can be read by the H8 and displayed on the front panel.

## 2.1.3 Data Tables

Various data tables are stored on the memory stick. These carry specific information for each conversion type and direction so the hardware knows all the various line and field lengths, aspect ratios etc. This data is simply read by the H8 which then sends the proper information to the FPGAs.

## 2.2 V6482 Motion Estimator (ME)

The ME board has only one programmable part. This is an FPGA but it is programmed in a completely different way to the SC board. There is a non-volatile memory on the ME that holds the data, and the FPGA itself is programmed by dedicated hardware at power up.

The non-volatile memory itself can be re-programmed through the front panel USB port with a PC fitted with suitable software.

Before actually updating the ME it is important to have a PC or Laptop prepared. This preparation has three parts:

- One of two proprietary programmes which communicates with the ME board.
- A driver for the for the USB link onto the ME board.
- The actual software file that will be sent to the ME board.



# CIFER Update Instructions

## 2.3 V6482 ME Preparation

If this is the first time that a particular computer has been used for an upgrade you must follow this procedure.

- Load the MecmonLite executable code onto the PC or Laptop. There is an alternative programme, MecMon, but this is more complex and is usually used for development work. This document only describes how to use MecMonLite.
- Connect the PC to the USB port on the front panel of the MEC board with a USB type A-B cable
- The normal Found New Hardware installation process will begin.
- It is important that you load the driver from the files supplied. Do not allow the computer to search the Internet or Microsoft websites to find a driver. For XP SP2 onwards there is a check box in the wizard for "Do Not Search the Internet". For earlier OS version you may have to actually disconnect the PC from the Internet.
- During the installation process Browse to the directory which contains the driver supplied with each Cifer Upgrade. Although the driver is supplied with each upgrade it only needs to be installed on first use.
- Full installation information on the driver can be found at:

[http://www.ftdichip.com/Documents/InstallGuides/Windows\\_XP\\_Installation\\_Guide.pdf](http://www.ftdichip.com/Documents/InstallGuides/Windows_XP_Installation_Guide.pdf)

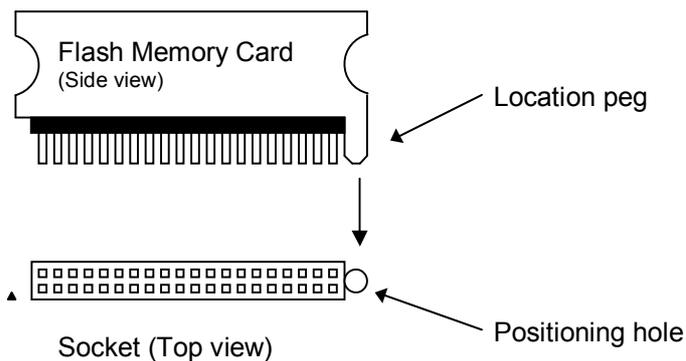
- Once the installation wizard has completed you can run the MecMonLite executable programme and follow the instructions in section 3.2.

## 3 Updating

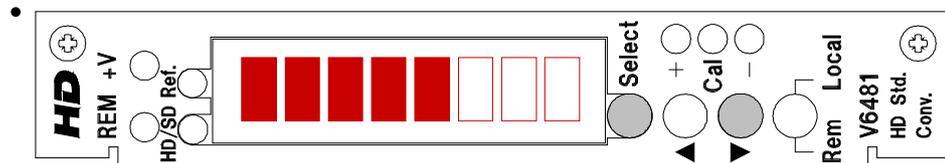
### 3.1 V6481 Standards Converter

#### 3.1.1 Memory Stick – New Version

- Remove the V6481 from its rack.
- Carefully remove the existing memory stick from M1.
- Check that the replacement stick is the correct one for the V6481. It will have a label with markings like:
- V6481 V1.1
- Carefully insert it into the socket with the correct alignment. The locating peg should be nearest to the edge connector. Do not force it. If there is significant resistance then check that the pins are well aligned and not bent.



- Replace the V6481 into its rack. Observe the front panel display which should show a progress bar (see below). When the re-programming is complete the H8 will go through its normal start-up routine.



It is possible that the upgrade only includes changes to the FPGA code, and not the H8 code. In this case the boot up routine will not include the progress bar but will go through its normal start up sequence.



# CIFER Update Instructions

- The release notes will give the current version numbers and these can be checked on the STATUS menu.

## 3.1.2 Memory Stick – Old Version

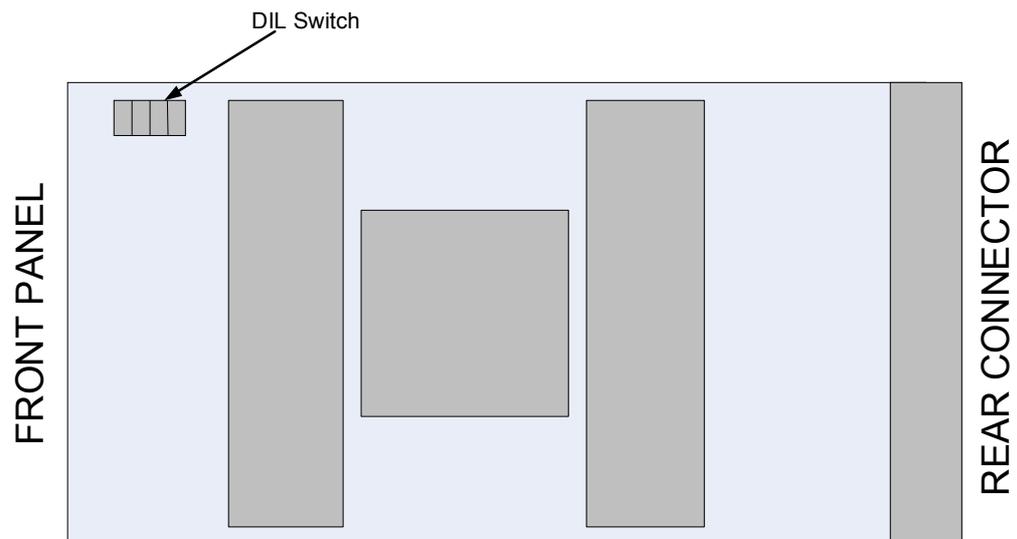
Follow the same procedure described above until replacing the module into the rack:

- Press and hold the Select and Down-Arrow buttons while replacing the V6481 into its rack. These buttons are shown shaded in the drawing above. The front panel display will again show the progress bar shown above. (Once the progress bar is being displayed you can release the two shaded buttons.) When the re-programming is complete the H8 will go through its normal start-up routine.

## 3.2 V6482 Motion Estimator

### 3.2.1 Board Switches

Remove the V6482 board and check the small DIL switch as shown:



To use MecMonLite all four switches should be set to **ON**.

The only way to know which direction is ON is to look at the marking on the switch itself.

# CIFER Update Instructions

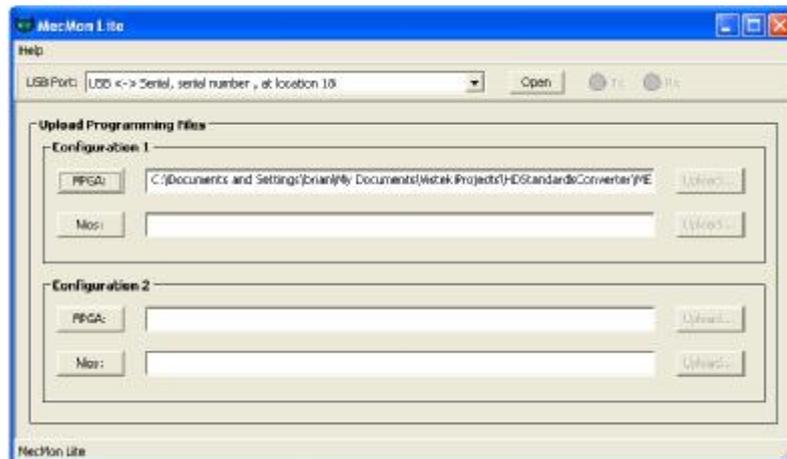


## 3.2.2 Connection

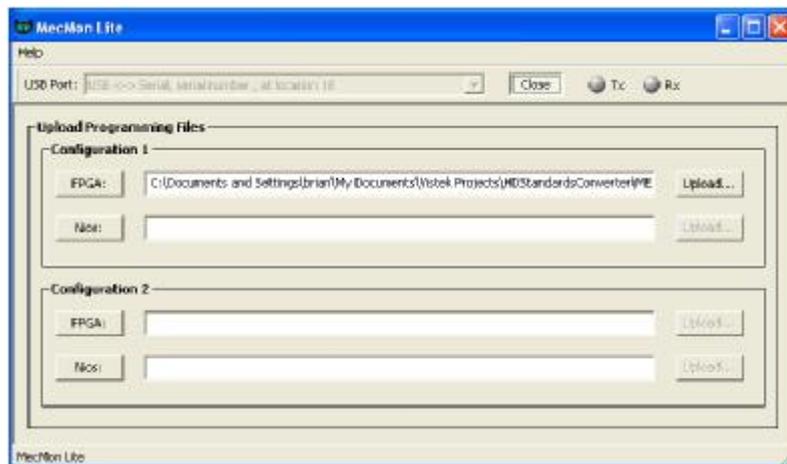
If the PC or Laptop has not been used with a ME board before then you must install a suitable driver first. Follow the instructions in section 2.3.

## 3.2.3 Upgrade

- With the board plugged in connect a USB Type A-B cable from a USB port on the PC to the USB port on the front panel.
- Run the MecMonLite programme. You should get the following screen:



- Click on OPEN which will start communication with the ME board and you should get:

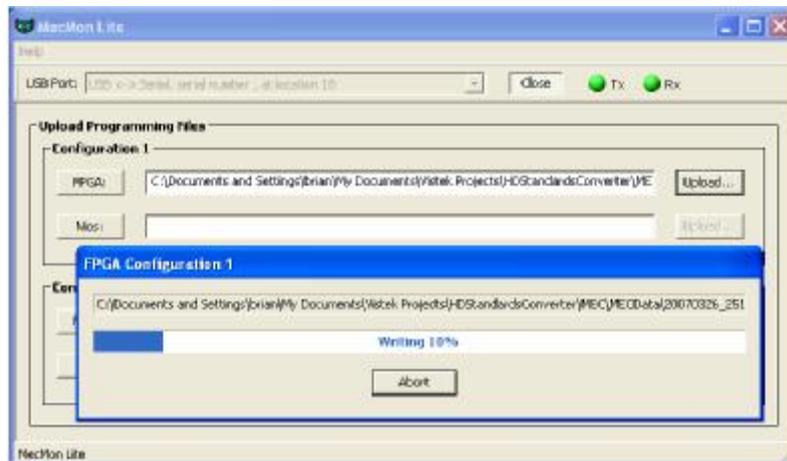


- Check the filename is OK or click on FPGA under Configuration 1 and find the correct file. The filename will be something like:

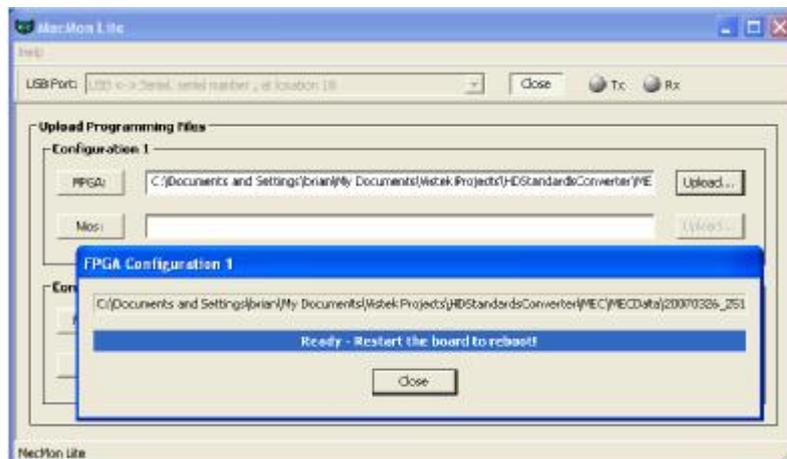
**v6482\_v00\_251.rbf**

# CIFER Update Instructions

- Click on UPLOAD to start the process, and you will get this screen:



- WAIT. The loading process takes many minutes, depending on the computer. The main variable is whether the computer has USB V2.0.
- There are three stages – Erasing, Writing and Verifying
- When the loading is complete this will be the screen. Click on close and disconnect the USB cable from the ME board.



- You **MUST** now reboot the ME board. Either Power down and up the rack, or remove and re-insert the ME board. This is essential for the new firmware to be loaded.
- Check that the correct ME version is being used by looking at the STATUS menu on the V6481.



## 4 Testing

---

### 4.1 Version Numbers

On the V6481 go to the Status menu and scroll down to check the version number of the various items.

This is a list, in order, of the various versions. The ones in bold are those most likely to change with an upgrade. The numbers here are examples, but you should check them against the instructions with the upgrade data.

Soft	1.07.00
<b>FPGA</b>	<b>00.03</b>
CPLD	00.01
<b>STRTX</b>	<b>0.7.4</b>
Boot	01.00.01
<b>MEC</b>	<b>1.0</b>

### 4.2 Pictures

If at all possible check the whole Cifer with a variety of picture standards.