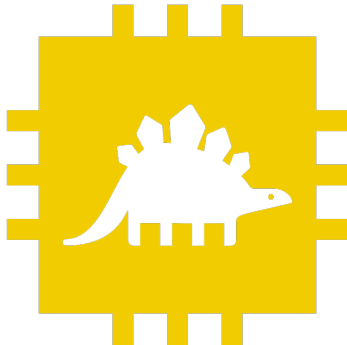


---

# **eZ80 for RC User Manual**

Dean Netherton



Contents

<b>eZ80 for RC - User Manual</b>	<b>2</b>
Minimum System Configurations . . . . .	2
Optional Extras in the Green Machine Series . . . . .	2
Memory Layout . . . . .	2
Bootup process . . . . .	3

## eZ80 for RC - User Manual

This document provides a high level overview of the eZ80 for RCBus system and how to operate it.

It assumes you have successfully assembled and booted up a working system. It also assume you have at least a basic understanding of CP/M

### Minimum System Configurations

A minimum configuration to enable a booting system, requires: 1. the [eZ80 for RC](#) module 2. a 512K banked RAM/ROM module (with the eZ80 configured build of RomWBW flashed to the ROM). 3. An RC2014 or RCBus compatible backplane 4. A serial terminal connected to the [eZ80 for RC](#) serial connector

### Optional Extras in the Green Machine Series

1. [USB for RC](#) enabling external flash & HDD storage, Floppy Disks, Keyboard and any other supported devices.
2. [2MB Linear SRAM](#) module to give your eZ80 up to 2 Megabytes of continuous memory.

### Memory Layout

The eZ80 can address up to 16MB of continuous memory and address up to 64KB of I/O ports. And supports 4 Chip Select signals that can be independently configured for specific address ranges and specific access timings.

The following table details how the memory has been configured for the system:

Address Range	Description
000000 -> 01FFFF	ON-CHIP ROM

Address Range	Description
02E000 -> 02FFFF	ON-CHIP RAM
030000 -> 03FFFF	EXTERNAL BANKED ROM/RAM (CS3)
200000 -> 3FFFFFF	EXTERNAL FLAT RAM (CS0) 2MB
400000 -> FFFFFFF	RESERVED

CS2 is wired for external I/O access onto appropriate RC2014/RC Bus signals. The following table details how the I/O addressing is mapped:"

Address Range	Description
0000 -> 00FF	Reserved ON-CHIP IO
FF00 -> FFFF	External IO (CS2)

## Bootup process

On power up, the onboard 'firmware' of the eZ80, stored in its on-chip ROM, starts. This firmware code will initialise the CPU and on-chip services, including configuring the on-chip UART. After this initialising of the on-chip hardware, it will delegate control the external bank ROM/RAM module - typically this will contain a copy of RomWBW software allowing for the system to boot into an operating system such as CP/M.

The firmware's startup initialisation will perform the following tasks:

1. Identify if the on-chip Real-Time-Clock (RTC) is powered.
2. If RTC is available, use it to measure and determine the CPU's actual clock frequency (eg: 20Mhz or 25Mhz)
3. Configure the on-chip UART serial baud rate to 115200. \*
4. Configure the Chip Select lines (CS0 to CS3) to appropriate wait states for reliable access to memory and I/O devices.