RomWBW

Introduction

Version 3.5

Updated 23 Jan 2025

RetroBrew Computers Group
www.retrobrewcomputers.org

Wayne Warthen
wwarthen@gmail.com

Contents

1 Overview **1**

1.1 Conventions Used 2

2 Description **3**

2.1 Primary Features 3

2.2 Included Software 4

2.3 ROM Distribution 5

2.4 Documentation 6

3 Support **7**

3.1 Getting Assistance 7

4 Contributions **8**

4.1 Acknowledgments 8

4.2 Related Projects 11

5 Licensing **12**

5.1 License Terms 12

Chapter 1

Overview

RomWBW software provides a complete, commercial quality implementation of CP/M (and workalike) operating systems and applications for modern Z80/180/280 retro-computing hardware systems.

A wide variety of platforms are supported including those produced by these developer communities:

- RetroBrew Computers (https://www.retrobrewcomputers.org)
- RC2014 (https://rc2014.co.uk),
  RC2014-Z80 (https://groups.google.com/g/rc2014-z80)
- Retro Computing (https://groups.google.com/g/retro-comp)
- Small Computer Central (https://smallcomputercentral.com/)

A complete list of the currently supported platforms is found in RomWBW Hardware .

1.1 Conventions Used

Size Suffixes

Within the documentation and in RomWBW in general, the use of size suffixes KB, MB, GB, and TB refer to the binary variant as shown below. The modern suffixes (KiB, MiB, etc.) are not used here because they were not prevalent during the time that the RomWBW OSes were used. This keeps all of RomWBW and associated applications consistent.

Suffix	Value	Meaning
KB	1024	1,024 bytes
MB	1024^2	1,048,576 bytes
GB	1024^3	1,073,741,824 bytes
TB	1024^4	1,099,511,627,776 bytes

Links and URLs

Many of the references in the documentation to Internet addresses (URLs) do not provide the address in the text. However, these links are embedded and “clickable” within the documents. Your PDF viewer should highlight these links in some manner (typically an alternate color or an underline).

Chapter 2

Description

2.1 Primary Features

By design, RomWBW isolates all of the hardware specific functions in the ROM chip itself. The ROM provides a hardware abstraction layer such that all of the operating systems and applications on a disk will run on any RomWBW-based system. To put it simply, you can take a disk (or CF/SD/USB Card) and move it between systems transparently.

Supported hardware features of RomWBW include:

- Z80 Family CPUs including Z80, Z180, and Z280
- Banked memory services for several banking designs
- Disk drivers for RAM, ROM, Floppy, IDE ATA/ATAPI, CF, SD, USB, Zip, Iomega
- Serial drivers including UART (16550-like), ASCI, ACIA, SIO
- Video drivers including TMS9918, SY6545, MOS8563, HD6445
- Keyboard (PS/2) drivers via VT8242 or PPI interfaces
- Real time clock drivers including DS1302, BQ4845
- Built-in VT-100 terminal emulation support

A dynamic disk drive letter assignment mechanism allows mapping operating system drive letters to any available disk media. Additionally, mass storage devices (IDE Disk, CF Card, SD Card, etc.) support the use of multiple slices (up to 256 per device). Each slice contains a complete CP/M filesystem and can be mapped independently to any drive letter. This overcomes the inherent size limitations in legacy OSes and allows up to 2GB of accessible storage on a single device, with up to 128MB accessible at any one time.

2.2 Included Software

Multiple disk images are provided in the distribution. Most disk images contain a complete, bootable, ready-to-run implementation of a specific operating system. A “combo” disk image contains multiple slices, each with a full operating system implementation. If you use this disk image, you can easily pick whichever operating system you want to boot without changing media.

Some of the included software

- Operating Systems (CP/M 2.2, ZSDOS, NZ-COM, CP/M 3, ZPM3, Z3PLUS, QPM)
- Additional operating systems, p-System, FreeRTOS, and FUZIX.
- Programming Tools (Z80ASM, Turbo Pascal, Forth, Cowgol)
- C Compilers including Aztec, and HI-TECK
- Microsoft Basic Compiler, and Microsoft Fortran
- Some games such as Colossal Cave, Zork, etc
- Wordstar

Some of the provided software can be launched directly from the ROM firmware itself:

- System Monitor
- Operating Systems (CP/M 2.2, ZSDOS)
- ROM BASIC (Nascom BASIC and Tasty BASIC)
- ROM Forth

A tool is provided that allows you to access a FAT-12/16/32 filesystem. The FAT filesystem may be coresident on the same disk media as RomWBW slices or on stand-alone media. This makes exchanging files with modern OSes such as Windows, MacOS, and Linux very easy.

2.3 ROM Distribution

The RomWBW Repository (https://github.com/wwarthen/RomWBW) on GitHub is the official distribution location for all project source and documentation.

RomWBW is distributed as both source code and pre-built ROM and disk images.

The pre-built ROM images distributed with RomWBW are based on the default system configurations as determined by the hardware provider/designer. The pre-built ROM firmware images are generally suitable for most users.

The fully-built distribution releases are available on the RomWBW Releases Page (https://github.com/wwarthen/RomWBW/releases) of the repository.

On this page, you will normally see a Development Snapshot as well as recent stable releases. Unless you have a specific reason, I suggest you stick to the most recent stable release.

The asset named RomWBW-vX.X.X-Package.zip includes all pre-built ROM and Disk images as well as full source code. The other assets contain only source code and do not have the pre-built ROM or disk images.

Distribution Directory Layout

The RomWBW distribution is a compressed zip archive file organized in a set of directories. Each of these directories has its own ReadMe.txt file describing the contents in detail. In summary, these directories are:

Directory	Description
Binary	The final output files of the build process are placed here. Most importantly, the ROM images with the file names ending in ".rom" and disk images ending in .img.
Doc	Contains various detailed documentation, both RomWBW specifically as well as the operating systems and applications.
Source	Contains the source code files used to build the software and ROM images.
Tools	Contains the programs that are used by the build process or that may be useful in setting up your system.

Building from Source

It is also very easy to modify and build custom ROM images that fully tailor the firmware to your specific preferences. All tools required to build custom ROM firmware under Windows are included – no need to install assemblers, etc. The firmware can also be built using Linux or MacOS after confirming a few standard tools have been installed.

2.4 Documentation

There are several documents that form the core of the RomWBW documentation:

- RomWBW User Guide is the main user guide for RomWBW, it covers the major topics of how to install, manage and use RomWBW, and includes additional guidance to the use of some of the operating systems supported by RomWBW
- RomWBW Hardware contains a description of all the hardware platforms, and devices supported by RomWBW.
- RomWBW Applications is a reference for the ROM-hosted and OS-hosted applications created or customized to enhance the operation of RomWBW.
- RomWBW Disk Catalog is a reference for the contents of the disk images provided with RomWBW, with a description of many of the files on each image
- RomWBW System Guide discusses much of the internal design and construction of RomWBW. It includes a reference for the RomWBW HBIOS API functions.

Each of the operating systems and ROM applications included with RomWBW are sophisticated tools in their own right. It is not reasonable to fully document their usage. However, you will find complete manuals in PDF format in the Doc directory of the distribution. The intention of this documentation is to describe the operation of RomWBW and the ways in which it enhances the operation of the included applications and operating systems.

Since RomWBW is purely a software product for many different platforms, the documentation does **not** cover hardware construction, configuration, or troubleshooting – please see your hardware provider for this information.

Chapter 3

Support

3.1 Getting Assistance

The best way to get assistance with RomWBW or any aspect of the RetroBrew Computers projects is via one of the community forums:

- RetroBrew Computers Forum
- RC2014 Google Group
- retro-comp Google Group

Submission of issues and bugs are welcome at the RomWBW GitHub Repository.

Also feel free to email Wayne Warthen at wwarthen@gmail.com. I am happy to provide support adapting RomWBW to new or modified systems

Chapter 4

Contributions

All source code and distributions are maintained on GitHub. Contributions of all kinds to RomWBW are very welcome.

4.1 Acknowledgments

I want to acknowledge that a great deal of the code and inspiration for RomWBW has been provided by or derived from the work of others in the RetroBrew Computers Community. I sincerely appreciate all of their contributions. The list below is probably missing many names – please let me know if I missed you!

- Andrew Lynch started it all when he created the N8VEM Z80 SBC which became the first platform RomWBW supported. Some of his original code can still be found in RomWBW.
- Dan Werner wrote much of the code from which RomWBW was originally derived and he has always been a great source of knowledge and advice.
- Douglas Goodall contributed code, time, testing, and advice in “the early days”. He created an entire suite of application programs to enhance the use of RomWBW. Unfortunately, they have become unusable due to internal changes within RomWBW. As of RomWBW 2.6, these applications are no longer provided.
- Sergey Kiselev created several hardware platforms for RomWBW including the very popular Zeta.
- David Giles created support for the Z180 CSIO which is now included SD Card driver.

- Phil Summers contributed the Forth and BASIC adaptations in ROM, the AY-3-8910 sound driver, DMA support, and a long list of general code and documentation enhancements.
- Ed Brindley contributed some of the code that supports the RCBus platform.
- Spencer Owen created the RC2014 series of hobbyist kit computers which has exponentially increased RomWBW usage. Some of his kits include RomWBW.
- Stephen Cousins has likewise created a series of hobbyist kit computers at Small Computer Central and is distributing RomWBW with many of them.
- Alan Cox has contributed some driver code and has provided a great deal of advice.
- The CP/NET client files were developed by Douglas Miller.
- Phillip Stevens contributed support for FreeRTOS.
- Curt Mayer contributed the original Linux / MacOS build process.
- UNA BIOS and FDISK80 are the products of John Coffman.
- FLASH4 is a product of Will Sowerbutts.
- CLRDIR is a product of Max Scane.
- Tasty Basic is a product of Dimitri Theulings.
- Dean Netherton contributed eZ80 CPU support, the sound driver interface, and the SN76489 sound driver.
- The RomWBW Disk Catalog document was produced by Mykl Orders.
- Rob Prouse has created many of the supplemental disk images including Aztec C, HiTech C, SLR Z80ASM, Turbo Pascal, Microsoft BASIC Compiler, Microsoft Fortran Compiler, and a Games compendium.
- Martin R has provided substantial help reviewing and improving the User Guide and Applications documents.
- Mark Pruden has made a wide variety of contributions including:
  - significant content in the Disk Catalog and User Guide
  - creation of the Introduction and Hardware documents
  - Z3PLUS operating system disk image
  - COPYSL utility
  - a feature for RomWBW configuration by NVRAM
  - the /B bulk mode of disk assignment to the ASSIGN utility

- Jacques Pelletier has contributed the DS1501 RTC driver code.
- Jose Collado has contributed enhancements to the TMS driver including compatibility with standard TMS register configuration.
- Kevin Boone has contributed a generic HBIOS date/time utility (WDATE).
- Matt Carroll has contributed a fix to XM.COM that corrects the port specification when doing a send.
- Dean Jenkins enhanced the build process to accommodate the Raspberry Pi 4.
- Tom Plano has contributed a new utility (HTALK) to allow talking directly to HBIOS COM ports.
- Lars Nelson has contributed several generic utilities such as a universal (OS agnostic) UNARC application.
- Dylan Hall added support for specifying a secondary console.
- Bill Shen has contributed boot loaders for several of his systems.
- Laszlo Szolnoki has contributed an EF9345 video display controller driver.
- Ladislau Szilagyi has contributed an enhanced version of CP/M Cowgol that leverages RomWBW memory banking.
- Les Bird has contributed support for the NABU w/ Option Board

4.2 Related Projects

Outside of the hardware platforms adapted to RomWBW, there are a variety of projects that either target RomWBW specifically or provide a RomWBW-specific variation. These efforts are greatly appreciated and are listed below. Please contact the author if there are any other such projects that are not listed.

Z88DK

Z88DK is a software powerful development kit for Z80 computers supporting both C and assembly language. This kit now provides specific library support for RomWBW HBIOS. The Z88DK project is hosted at https://github.com/z88dk/z88dk.

Paleo Editor

Steve Garcia has created a Windows-hosted IDE that is tailored to development of RomWBW. The project can be found at https://github.com/alloidian/PaleoEditor.

Z80 fig-FORTH

Dimitri Theulings' implementation of fig-FORTH for the Z80 has a RomWBW-specific variant. This fig-FORTH is built into the RomWBW ROM. However, the project itself is hosted at https://github.com/dimitrit/figforth.

Assembly Language Programming for the RC2014 Zed

Bruce Hall has written a very nice document that describes how to develop assembly language applications on RomWBW. It begins with the setup and configuration of a new RC2014 Zed system running RomWBW. It describes not only generic CP/M application development, but also RomWBW HBIOS programming and bare metal programming. The latest copy of this document is hosted at http://w8bh.net/Assembly for RC2014Z.pdf.

Chapter 5

Licensing

5.1 License Terms

RomWBW is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

RomWBW is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with RomWBW. If not, see https://www.gnu.org/licenses/.

Portions of RomWBW were created by, contributed by, or derived from the work of others. It is believed that these works are being used in accordance with the intentions and/or licensing of their creators.

If anyone feels their work is being used outside of its intended licensing, please notify:

> Wayne Warthen
> wwarthen@gmail.com

RomWBW is an aggregate work. It is composed of many individual, standalone programs that are distributed as a whole to function as a cohesive system. Each program may have its own licensing which may be different from other programs within the aggregate.

In some cases, a single program (e.g., CP/M Operating System) is composed of multiple components with different licenses. It is believed that in all such cases the licenses are

compatible with GPL version 3.

RomWBW encourages code contributions from others. Contributors may assert their own copyright in their contributions by annotating the contributed source code appropriately. Contributors are further encouraged to submit their contributions via the RomWBW source code control system to ensure their contributions are clearly documented.

All contributions to RomWBW are subject to this license.