
WinCortex Emulator and Cortex<->PC Data Transfer Utility

WinCortex Emulator Version: 2.0
PC Data Transfer Utility Version: 1.3
Software by: Dave Hunter
Documentation by: Stuart Conner

D

R

A

F

T

Contents

D	1 Introduction.....3
	2 WinCortex Emulator4
	2.1 Introduction4
	2.2 Emulated Hardware4
	2.3 PC System Requirements4
	2.4 Downloading and Installing the Emulator5
	2.5 Running the Emulator.....5
	2.5.1 Front Panel LEDs.....5
	2.5.2 Virtual Drives6
	2.5.3 Menu6
	2.5.4 WinCortex Emulator Key Substitutions8
	2.5.5 Summary of Keyboard Shortcuts9
	2.6 Using Disk Images10
	2.7 Using Cassette Images11
	2.7.1 Understanding Image and Raw Cassette Files11
	2.7.2 Saving and Loading a Program from BASIC12
	2.7.3 Saving and Loading a Machine Code Program (or Block of Data) from the Monitor.....12
	2.7.4 Recording and Playing Back in Raw Mode13
	2.8 Using the CPU Debug Window13
	2.9 Using Alternative EPROM Images14
	3 Cortex<->PC Data Transfer Utility15
	3.1 Introduction15
	3.2 How Does It Work?15
	3.3 Data Transfer Cable15
	3.4 Downloading and Installing the Data Transfer Utility16
	3.5 Patching the Cassette Routines on the Cortex16
	3.6 Saving and Loading a Program from BASIC18
	3.7 Saving and Loading a Machine Code Program (or Block of Data) from the Monitor.....19
	3.8 Making a CDOS Boot Floppy Disk20
	3.9 Transferring Disk Images21

List of Tables

Table 1	WinCortex Emulator Key Substitutions9
Table 2	WinCortex Emulator Keyboard Shortcut9

List of Figures

Figure 1	WinCortex Emulator Main Window5
Figure 2	Cortex<->PC Data Transfer Cable Wiring Diagram16
Figure 3	Cortex<->PC Data Transfer Utility Application Window17

1 Introduction

D

This document describes the use of the WinCortex software application which emulates the Powertran Cortex computer.

R

The Powertran Cortex is a 16-bit TMS 9995-based computer that was available in the UK between late 1982 and early 1985. The computer could be bought in kit or ready-built forms, and its design was serialised in the Electronics Today International (ETI) magazine between November 1982 and January 1983.

Further details on the history of the Powertran Cortex are available [here](#). The technical specification of the Powertran Cortex hardware is available [here](#).

A

This document also describes the use of a data transfer utility for transferring cassette and disk image files between a PC and the Powertran Cortex computer over the RS-232 serial port.

Note that this document does not describe how to program/use the Powertran Cortex – that information is given in the Cortex BASIC manual, which is available [here](#).

F

The WinCortex emulator and data transfer utility were written by Dave Hunter, who can be contacted at dth8765@gmail.com. This document was written by Stuart Conner, who can be contacted at ti99@stuartconner.me.uk.

T

2 WinCortex Emulator

2.1

Introduction

The WinCortex emulator is a Windows application developed under Borland (now Embarcadero) C++ Builder 5 (Standard Edition). It consists of a single executable file, with a copy of the original Powertran Cortex EPROMs in a separate file.

R

In addition, there are many disk images available that are binary copies of physical Powertran Cortex disks, allowing you to explore the CDOS¹ and MDEX² disk operating systems.

2.2

Emulated Hardware

A

The WinCortex emulator is based on a standard Powertran Cortex with 64KB RAM and disk interface. Four virtual disk drives are available, each of which supports any valid physical disk drive configuration (40/80 tracks, single/double sided, single/double density).

F

The emulated video hardware is based on a TMS 9928/9 video display processor (VDP) with 16KB of RAM. All the VDP text and graphics modes are supported with the exception of the 64×48 Multicolor Mode. Sprites are also supported. Additionally, the 80×26 Text Mode II from the Yamaha V9938 MSX chip is also emulated to support the MDEX native 80 column environment.

2.3

PC System Requirements

The WinCortex application requires the following features on the host PC:

- Windows 98 or later, 32- or 64-bit;
- DirectX4 or later;
- video resolution of 1024×768 or higher;
- 32-bit colour;
- 4MB disk space.

¹ CDOS – Cortex Disk Operating System. Adds ability to load and save to disk to the standard Cortex BASIC. The CDOS User's Guide is available [here](#).

² MDEX – Marinchip Disk Executive. A disk operating system for computers based on the TMS 99xx(x) range of microprocessors. Converted to run on the Powertran Cortex by MicroProcessor Engineering Ltd of Southampton, UK. Once loaded, it completely replaces Cortex BASIC and provides a rich programming and development environment. Several languages are available (QBASIC, SPL, Pascal, Forth, Assembler), along with a professional Word Processor. The MDEX documentation is available [here](#).

2.4 Downloading and Installing the Emulator

D

The WinCortex emulator can be downloaded [here](#). The download is in the form of a .zip file. To install the emulator, simply extract the two files from the .zip file into a folder. The emulator executable is named **WinCortex.exe**. The file **CORTEX.BIN** contains a copy of the Powertran Cortex BASIC EPROMs.

2.5 Running the Emulator

R

To run the WinCortex emulator, double-click the **WinCortex.exe** file. The emulator main window is shown in figure 1. Once loaded, the emulator is ready to accept input from the keyboard. The Cortex BASIC manual is available [here](#).

A

F

T



Figure 1 WinCortex Emulator Main Window

2.5.1 Front Panel LEDs

The RUN, IDLE, MEM and BASIC graphics below the screen image emulate the four LEDs on the Powertran Cortex front panel. These LEDs provide the following indications:

- RUN lights to indicate when the CPU is executing instructions.
- IDLE lights to indicate when the CPU is in an internal loop waiting for interrupts (including keyboard input).



- MEM lights to indicate when external memory is being accessed through the memory mapper device (which is not emulated).
- BASIC flashes approximately once per second while the BASIC interpreter is running.

When the emulator is waiting for keyboard input, the IDLE LED is lit and the BASIC LED is flashing.



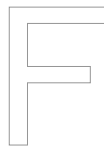
2.5.2 Virtual Drives

The four controls in the **Drives** group show the name of the disk image loaded in each virtual drive, and the adjacent graphic flashes when each drive is accessed. Clicking a drive label display a file list dialog to select the .dsk disk image file to load in the selected virtual drive (the same action as clicking the drive in the Floppy menu).



2.5.3 Menu

The menu structure, shortcut keys and a description of each command are given below.



Floppy

The commands in the Floppy menu are for working with disk image files. See section 2.6 for more information on using disk images.

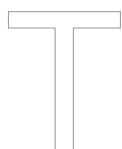
- **Drive 0...** (F1) **Drive 1...** (F2)
- **Drive 2...** (F3) **Drive 3...** (F4)

Displays a file list dialog to select the .dsk disk image file to load in the selected virtual drive.

- **Double Density Boot**

Toggles the disk system between double density and single density.

This setting is read and used to configure the disk system only when the **BOOT** command is executed. Changing this setting after the **BOOT** command has no effect³.



Cassette

The commands in the Cassette menu are for working with cassette image files. See section 2.7 for more information on using cassette images.

- **Play...** (F5)

Displays a file list dialog to select a .cas cassette image file to start playing.

³ On the Powertran Cortex, the density is set by a jumper on the PCB. The jumper position is read and used to configure the disk system only when the **BOOT** command is executed. The jumper has no direct connection to the disk system hardware so changing this setting after the **BOOT** command has no effect.

D

- **Record Image... (F6)**

Displays a file save dialog to specify a .cas cassette image file to record to in Image mode.

- **Abort Image Record**

Recording to a cassette image file normally stops automatically after saving a program. Use this menu command to abort a recording if you have selected an image to record to but no longer wish to save the program.

R

- **Record Raw... (F7)**

Displays a file save dialog to specify a .cas cassette image file to record to in Raw mode.

- **Stop Raw Record**

Stops the Raw mode recording and closes the file.

A

- **Overwrite Mode**

Toggles cassette image file overwrite mode on or off.

- ◆ If overwrite mode is off, trying to record over an existing cassette image file displays an "Overwrite file?" confirmation dialog.
- ◆ If overwrite mode is on, an existing cassette image file is overwritten without any warning.

F

Terminal

- **Serial Terminal (F10)**

Displays a Serial Terminal window with an emulated connection to the WinCortex emulator serial RS-232 port.

Use the **UNIT 2** command to enable output to the serial port. See the Cortex BASIC manual for further details.

- **Start Terminal Log...**

Displays a file save dialog to specify a .trm terminal log file, and starts saving all output sent to the serial port to the log file.

- **Stop Terminal Log**

Stops recording to the terminal log file and closes the file.

T

View

- **Control Panel (F9)**

Displays the CPU Debug window. See section 2.8 for further details.

- **Large Display (F8)**

Toggles the WinCortex emulator main window between normal and large display sizes.

- **LED Panel**

Toggles the display of the front panel LED graphics beneath the screen image in the main window.

D

- **Floppy Drives**

Toggles the display of the virtual drives beneath the screen image in the main window.

R

Keyboard

- **Smart Caps**

Toggles smart caps mode on or off.

- ◆ If smart caps mode is off, keyboard input to the WinCortex emulator uses the same lower/upper case setting as other Windows applications.
- ◆ If smart caps mode is on, the lower/upper case setting is inverted – characters typed as lower case in another Windows application are input as upper case in the WinCortex emulator. Smart caps mode is on by default as input to the Cortex is normally in upper case.

A

- **Game Mode**

*** NO IDEA WHAT THIS DOES ***

F

CPU Speed

- **x1** (Ctrl+F1) **x2** (Ctrl+F2)
x5 (Ctrl+F5) **Max** (Ctrl+F12)

Sets the speed of the WinCortex emulator. When **x1** is selected, the emulator runs at approximately the same speed as the Powertran Cortex hardware. The **x2** and **x5** settings run the emulator at approximately two and five times as fast as the hardware. When **Max** is selected, the emulator runs as fast as the PC can support.

T

Reset

- **Warm Start** (F11)

Performs a warm start of the WinCortex emulator. This has the same effect as pressing the Break button on the Powertran Cortex rear panel. This normally has the effect of running the built-in Monitor program.

- **Reset** (Ctrl+Del)

Resets the WinCortex emulator. This has the same effect as pressing the Reset button on the Powertran Cortex rear panel.

2.5.4

WinCortex Emulator Key Substitutions

The Powertran Cortex keyboard has several keys that do not exist on a standard PC keyboard, and are therefore substituted by the keys listed in table 1.

Table 1 WinCortex Emulator Key Substitutions

Powertran Cortex Key	PC Keyboard Key
EDIT	End
CLEAR	Page Up
GRAPH	Page Down
REPEAT	(Not applicable – keys auto-repeat)

D

2.5.5 R

Summary of Keyboard Shortcuts

A summary of the keyboard shortcuts defined is given in table 2.

Table 2 WinCortex Emulator Keyboard Shortcut

Key	Function
F1	Load drive 0 image file
F2	Load drive 1 image file
F3	Load drive 2 image file
F4	Load drive 3 image file
F5	Select and play cassette image file
F6	Save cassette image file in Image mode
F7	Save cassette image file in Raw mode
F8	Toggles WinCortex emulator main window between normal and large display sizes
F9	Display CPU Debug window
F10	Display Serial Terminal window
F11	Perform system warm start
Ctrl+Del	Perform system reset
Ctrl+F1	Set emulator speed to ×1 (same speed as Powertran Cortex hardware)
Ctrl+F2	Set emulator speed to ×2
Ctrl+F5	Set emulator speed to ×5
Ctrl+F12	Set emulator speed to maximum supported by PC

A

F

T

2.6 Using Disk Images

D

A number of disk images are available in the .zip file [here](#). The disk images are byte-for-byte binary copies of actual floppy disks, with a few additional bytes appended to the end to specify their original physical attributes (number of tracks, sides, density) and disk name to the WinCortex emulator. The .zip file includes a **README.txt** file which provides information about the disk subsystem (some of which is reproduced here) and a short description of each disk image.

R

When a disk image is 'loaded' in a virtual floppy drive, that drive assumes the physical attributes of the image (for example, 80 tracks, double sided, double density – '80TDSDD'). It is important to ensure the WinCortex emulator is configured to be compatible with that image. For instance, if a single density copy of CDOS is placed in drive 0, but the **Floppy > Double Density Boot** menu option is selected to boot from a double density disk, the disk will not boot. Another example would be loading an 80TDSDD disk in drive 1 when CDOS is configured to expect a 40TSSDD disk. Both these scenarios would cause errors in the WinCortex emulator as they would on a real Powertran Cortex. The physical attributes of an image can be determined by looking at the disk image file size:

A

- 81K = 40 track, single sided, single density (40TSSDD);
- 321K = 80 track, double sided, single density (80TSSDD);
- 641K = 80 track, double sided, double density (80TDSDD).

F

The following example shows how to boot the CDOS Cortex Disk Operating System:

T

1. Extract the **CDOS40 Release.dsk** disk image from the zip file linked to above and save it into a convenient folder (the same folder as the emulator **WinCortex.exe** file is fine). The '40' in the file name indicates that this version of CDOS is for a 40-column display.
2. Run the WinCortex emulator.
3. The disk image file size is 641K, indicating an 80TDSDD disk image. Click the **Floppy** menu and ensure that the **Double Density Boot** option is selected.
4. Click the **Floppy > Drive 0** menu command and select the **CDOS40 Release.dsk** disk image.
5. Type **BOOT<Return>** to boot from the floppy. CDOS is loaded and the command prompt is redisplayed.
6. Type ***DIR<Return>** to load and run the DIR program to list the files on the disk. Press 0 (for drive 0) at the prompt to list the files on drive 0.

To use the 80-column version of CDOS, reset the WinCortex emulator and repeat the steps above but use the **CDOS80 Release.dsk** disk image.

User guides for CDOS and many of the other disk images are available [here](#).

2.7 Using Cassette Images

2.7.1

Understanding Image and Raw Cassette Files

The Cortex emulator supports two types of cassette files: Image and Raw. This section explains the differences and uses of these two formats.

In the Powertran Cortex, the cassette interface is formed by a TMS 9902 serial interface controller IC connected to circuitry that generates and decodes the audio tones used by a cassette recorder. When saving a program, the Cortex first repeatedly sends a specific byte to the TMS 9902 for a period of 2 seconds which forms a steady tone 'preamble' while the tape speed stabilises, followed by sending the bytes for a header that includes the program name and other information, followed by the bytes that form the program itself.

With the Cortex emulator, when recording a file in Raw mode, the emulator records the exact stream of bytes sent to the (emulated) TMS 9902, and so recorded cassette files include the preamble described above. When recording in Image mode, the recorded file is processed to remove the preamble (and other padding bytes that might be added) to produce a recorded cassette file that is compatible with the PC communications utilities described in section 3.

When using the emulator to save and load a program in BASIC or the Monitor, Image mode should be used. More details on loading and saving a program in BASIC or the Monitor are given in sections 2.7.2 and 2.7.3.

Raw mode is used to capture characters or data sent directly to a cassette file, rather than saving a program. For example, using the UNIT command, all the characters typed or displayed on screen can also be sent to the (emulated) TMS 9902 and so recorded by the virtual cassette recorder, and these recorded characters can then be played back as if they being typed at the keyboard. This is particularly useful in the emulation environment as a program can be 'listed' to a text file on the PC, the program listing modified using a text editor, then the program 'played back' from the PC into the emulator as if being retyped from the keyboard. More details on using Raw mode with the emulator are given in section 2.7.4.

When a cassette file is 'played back', the virtual cassette recorder checks whether the file contains a valid header. If it does then the playback is treated as an Image file, otherwise it is treated as a Raw file.

2.7.2 Saving and Loading a Program from BASIC

Saving

D

To save a program on cassette from BASIC:

1. Click the **Cassette > Record Image** menu command and specify the name of the .cas cassette image file to record.
2. Type **SAVE ""** or **SAVE "PROG_NAME"<Return>**.
3. Type **Y** or **N** at the "Auto-Run ?" prompt as required.
4. Type **Y** at the "Cassette ready ?" prompt.
5. The program is recorded to the cassette image file then recording stops.

R

If necessary, the recording session can be aborted before saving the program by clicking the **Cassette > Abort Image Record** menu command. The recording session will time out and abort automatically after one minute if no SAVE is performed.

A

The **Cassette > Overwrite Mode** menu option provides the ability to automatically overwrite existing .cas cassette image files without being prompted.

F

Loading

To load a program from cassette in BASIC:

1. Type **LOAD ""** or **LOAD "PROG_NAME"<Return>**.
2. Type **Y** at the "Cassette ready ?" prompt.
3. Click the **Cassette > Play** menu command and select the .cas cassette image file to play.
4. The program is loaded.

T

2.7.3 Saving and Loading a Machine Code Program (or Block of Data) from the Monitor

Saving

To save a machine code program (or block of data) from the Monitor:

1. Click the **Cassette > Record Image** menu command and specify the name of the .cas cassette image file to record.
2. Type **D SSSS EEEE AAAA** where SSSS is the start address, EEEE is the end address and AAAA is the optional auto-run address.
3. Type the optional program name at the "IDT=" prompt, followed by <Return>.
4. Type **Y** or **N** at the "Auto-Run ?" prompt as required.
5. Type **Y** at the "Cassette ready ?" prompt.
6. The program is recorded to the cassette image file then recording stops.

The recording session can be aborted and overwrite mode set as described in the previous section.

D

Loading

To load a machine code program (or block of data) in the Monitor:

1. Type **L** followed by the optional program name at the "IDT=" prompt then <Return>.
2. Type **Y** at the "Cassette ready?" prompt.
3. Click the **Cassette > Play** menu command and select the .cas cassette image file to play.
4. The program or block of data is loaded.

R

2.7.4

Recording and Playing Back in Raw Mode

A

Recording

To list a BASIC program to a text file using Raw mode:

1. Click the **Cassette > Record Raw** menu command and specify the name of the .cas cassette image file to record.
2. In BASIC, type **UNIT 3**<Return> to turn on character output to the cassette.
3. Type **LIST**<Return> to list the program.
4. Click the **Cassette > Stop Raw Record** menu command to stop recording.

F

T

The recorded file can be edited using a text editor on the PC.

Any data printed to the screen is output to the cassette file. Instead of listing a program, data could for example be generated from a BASIC or machine code program.

Playing Back

To 'play back' the contents of a Raw mode recording as if the contents was being typed at the keyboard:

1. In BASIC, type **BAUD 3,9600**<Return> to enable input from the cassette (any supported Baud rate can be specified; the value is effectively ignored by the WinCortex emulator).
2. Click the **Cassette > Play** menu command and select the .cas cassette image file to play.
3. The contents of the selected file is played back.

2.8

Using the CPU Debug Window

*** TO DO ***

2.9 Using Alternative EPROM Images

The 'operating system' in the Powertran Cortex is stored in EPROMs (electrically-erasable programmable read only memories). Two EPROM images are currently available:

D

R

- The EPROM image for Cortex BASIC, supplied as standard with the Powertran Cortex and the WinCortex emulator. With the emulator, the EPROM image is contained in the file **CORTEX.BIN**.
- An EPROM image for a Cortex implementation of the [fig-FORTH](#) language. A copy of this EPROM image can be downloaded [here](#) (under the heading "Cortex EPROMs"). Open the .zip file, extract the file **FORTH.BIN** and save it in the same folder as the **CORTEX.BIN** file.

To swap between the two EPROM images:

A

F

T

1. Ensure that the WinCortex emulator is not running.
2. Rename the **CORTEX.BIN** and **FORTH.BIN** files to **CORTEX.BIN backup** and **FORTH.BIN backup**.
3. Make a copy of the EPROM image file to use and rename that file to **CORTEX.BIN**.
4. Run the emulator, which loads the EPROM image to use from the **CORTEX.BIN** file.

3 Cortex<->PC Data Transfer Utility

3.1



Introduction

The Cortex<->PC Data Transfer Utility serves two purposes:

- It enables cassette and disk images to be transferred between a Powertran Cortex and a PC, where they can be used with the WinCortex emulator.
- It enables a new boot floppy disk to be created on a Powertran Cortex, and other disk images to be transferred to the Cortex and written to floppy disks.



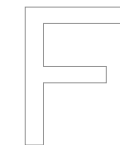
A simple data transfer cable is required to connect the serial port on the Cortex with a serial port on the PC.

3.2



How Does It Work?

The Powertran Cortex contains two TMS 9902 serial interface controller ICs – one connected to the cassette interface circuitry and the other connected to the serial port. The Cortex<->PC Data Transfer Utility makes use of the ability to patch (modify) the cassette routines on the Cortex so that they send/receive data over the serial interface instead of the cassette interface, with the Data Transfer Utility running on the PC and acting as a virtual cassette recorder. Data is transferred at 19200 Baud, so the transfers are reasonably quick.



3.3



Data Transfer Cable

The Cortex<->PC Data Transfer Utility requires the Cortex and PC to be connected using a cable wired as shown in figure 2. At the PC end, the cable can connect to a serial port on the motherboard or to a USB-serial converter. On the PC, you will need to know the COM port number assigned to the serial port you are using – the Windows Device Manager can be useful in determining this.

D

R

A

F

T

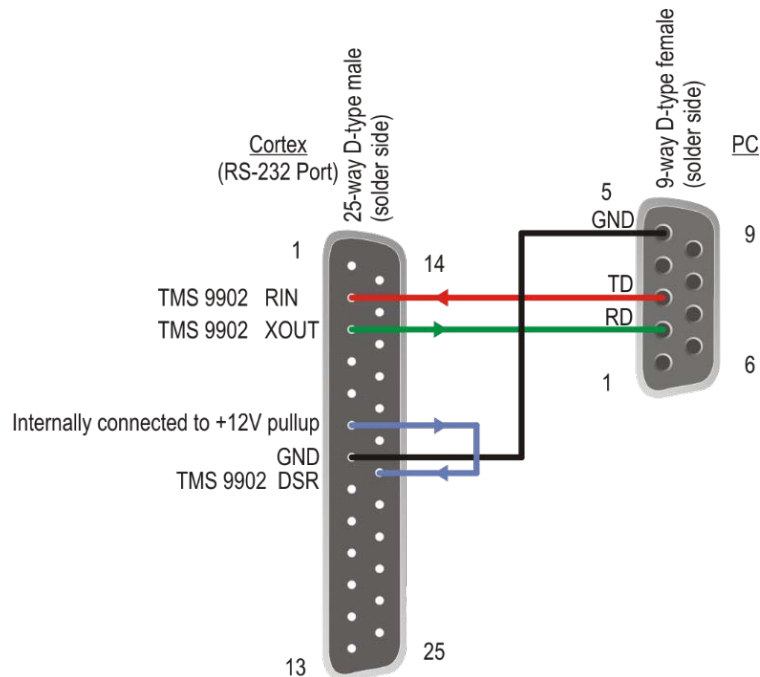


Figure 2 Cortex<->PC Data Transfer Cable Wiring Diagram

3.4

Downloading and Installing the Data Transfer Utility

The Cortex<->PC Data Transfer Utility can be downloaded [here](#) (under the heading “PC Comms Utility”). The download is in the form of a .zip file. To install the Data Transfer Utility, simply extract the files from the .zip file into a folder.

3.5

Patching the Cassette Routines on the Cortex

Before using the Cortex<->PC Data Transfer Utility, the cassette routines on the Cortex have to be patched so that they send/receive data over the serial interface. You need to do this each time the Cortex is powered on or reset. To patch the cassette routines:

1. Ensure that the Cortex is displaying the “*Ready” prompt.
2. On the PC, run the Data Transfer Utility **CommsUtil.exe**. The application window is shown in figure 3. Select the COM port, leave the **Select Baud Rate** and **Byte Delay** fields at their default values, then click the **Init** button to initialise the COM port on the PC.

3. Click the **Send Command File** button. The following commands⁴ should appear on the Cortex display to patch the cassette routines:

D

```
MWD[001C8H] = 08300H ! 8, N, 1
```

```
MWD[018B2H] = 01AH ! 19200
```

```
MWD[006CEH] = 080H
```

```
MWD[05B30H] = 080H
```

```
MWD[004A8H] = 080H
```

```
MWD[004B2H] = 080H
```

```
MWD[018A0H] = 080H
```

```
MWD[05520H] = 002H
```

R

4. Click the **Reset** button on the Data Transfer Utility, change the **Select Baud Rate** field to 19200, then click the **Init** button to re-initialise the COM port on the PC. The utility is now ready to send and receive cassette and disk images, and the **SEND** and **RECEIVE** buttons should be enabled.

A

As an alternative to the procedure above, if you have already created a CDOS boot disk, the patch commands are included on the disk in the file **CASS232**. So you can boot CDOS, type **LOAD 0,"CASS232"<Return>** to load and run the patch commands, then perform just step 4 of the procedure above to prepare the Data Transfer Utility.

F

T

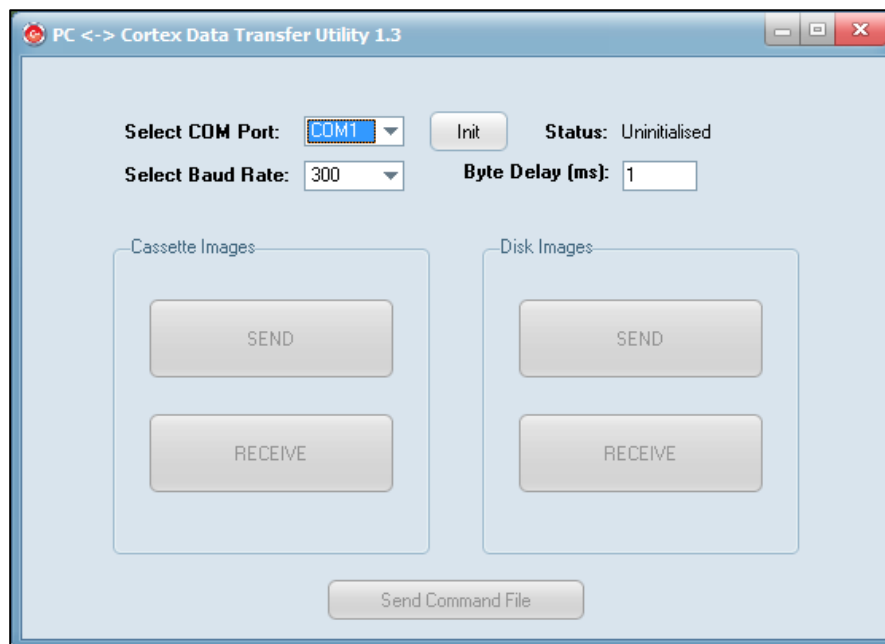


Figure 3 Cortex<->PC Data Transfer Utility Application Window

⁴ These commands are stored in the **commands.txt** file, which is in the same folder as the Data Transfer Utility program executable. The commands set the default RS-232 port configuration to 8 data bits, no parity, 1 stop bit, 19200 Baud, and replace references to the TMS 9902 for the cassette interface at CRU address >0180 with the address for the TMS 9902 for the serial port at >0080.

3.6 Saving and Loading a Program from BASIC

Saving

D

To save a program from BASIC (receive a file using the Data Transfer Utility):

1. Ensure that the cassette routines on the Cortex have been patched as described in section 3.5.
2. Click the **Cassette Images > RECEIVE** button on the Data Transfer Utility and specify the name of the .cas cassette image file to record. A window is displayed showing the progress of the receive operation:

```
Waiting for SYN & STX preamble...
```

R

3. On the Cortex, type **SAVE ""** or **SAVE "PROG_NAME"**<Return>.

4. Type **Y** or **N** at the "Auto-Run ?" prompt as required.

5. Type **Y** at the "Cassette ready ?" prompt.

A

6. The program is saved on the Cortex and stored in the specified file. The progress window is updated as the program is received:

```
Waiting for SYN & STX preamble...
```

```
Receiving SYN preamble...
```

```
STX received, processing header...
```

```
Auto-Run = N
```

```
Name: PROG_NAME
```

```
File Type: BASIC
```

```
Load Length: 60(0x003C) bytes
```

```
Program data read
```

```
Cassette image successfully transfered to PC.
```

F

7. When the transfer is complete, click the **Quit** button on the progress window to close it.

T

The saved file is compatible with the WinCortex emulator, and can be loaded as described in section 2.7.2.

Loading

To load a program in BASIC (send a file using the Data Transfer Utility):

1. Ensure that the cassette routines on the Cortex have been patched as described in section 3.5.
2. On the Cortex, type **LOAD ""** or **LOAD "PROG_NAME"**<Return>.
3. Type **Y** at the "Cassette ready ?" prompt.
4. Click the **Cassette Images > SEND** button on the Data Transfer Utility and select the .cas cassette image file to play. A window is displayed showing the progress of the send operation:

```
Sending preamble...
```

```
Sending STX...
```

```
Sending Header...
```

```
Sending program of Load Length: 60 (0x003C) bytes
```

```
Transfer complete...No Errors!
```

```
Cassette image successfully transfered to Cortex.
```

- When the transfer is complete, click the **Quit** button on the progress window to close it.

D

If the progress window shows that an error occurred, reset the Cortex and repeat the operation, but before sending the cassette image, increase the **Byte Delay** field in the Data Transfer Utility – try a value of 5 first and if that works, try again with successively lower values to find the minimum value for a successful transfer.

3.7 R

Saving and Loading a Machine Code Program (or Block of Data) from the Monitor

Saving

A

To save a machine code program (or block of data) from the Monitor (receive a file using the Data Transfer Utility):

- Ensure that the cassette routines on the Cortex have been patched as described in section 3.5.
- Click the **Cassette Images > RECEIVE** button on the Data Transfer Utility and specify the name of the .cas cassette image file to record. A window is displayed showing the progress of the receive operation:
Waiting for SYN & STX preamble...
- On the Cortex, type **D SSSS EEEE AAAA** where SSSS is the start address, EEEE is the end address and AAAA is the optional auto-run address.
- Type the optional program name at the “IDT=” prompt, followed by <Return>.
- Type **Y** or **N** at the “Auto-Run ?” prompt as required.
- Type **Y** at the “Cassette ready ?” prompt.
- The program (or block of data) is saved on the Cortex and stored in the specified file. The progress window is updated as the program is received:

```
Waiting for SYN & STX preamble...
Receiving SYN preamble...
STX received, processing header...
Auto-Run = N
Name: PROG_NAME
File Type: Machine Code
Load Length: 256(0x0100) bytes
Program data read
Cassette image successfully transferred to PC.
```

- When the transfer is complete, click the **Quit** button on the progress window to close it.

F T

Loading

To load a machine code program (or block of data) in the Monitor (send a file using the Data Transfer Utility):

1. Ensure that the cassette routines on the Cortex have been patched as described in section 3.5.
2. On the Cortex, type **L** followed by the optional program name at the "IDT=" prompt then <Return>.
3. Type **Y** at the "Cassette ready ?" prompt.
4. Click the **Cassette Images** > **SEND** button on the Data Transfer Utility and select the .cas cassette image file to play. A window is displayed showing the progress of the send operation:

```

Sending preamble...
Sending STX...
Sending Header...
Sending program of Load Length: 60 (0x003C) bytes
Transfer complete... No Errors!
Cassette image successfully transferred to Cortex.

```

5. When the transfer is complete, click the **Quit** button on the progress window to close it.

If the progress window shows that an error occurred, reset the Cortex and repeat the operation, but before sending the cassette image, increase the **Byte Delay** field in the Data Transfer Utility – try a value of 5 first and if that works, try again with successively lower values to find the minimum value for a successful transfer.

3.8

Making a CDOS Boot Floppy Disk

To use the Cortex<->PC Data Transfer Utility to make a new CDOS boot floppy disk:

1. Ensure that the cassette routines on the Cortex have been patched as described in section 3.5.
2. On the Cortex, type **LOAD "SAFORMAT"**<Return> to start loading the Cortex Disk Formatter Utility.
3. Type **Y** at the "Cassette ready ?" prompt.
4. Use the Data Transfer Utility to send the cassette image file **SAFORMAT.cas** (which should be in the same folder as the Data Transfer Utility program executable).
5. The program auto-runs when transfer is complete. Insert a disk in drive 0 and use the program to format it as single sided, single density, 80 tracks (options 0, 1, 1, 2).
6. After the disk has formatted, type **LOAD "DISKIMG"**<Return> to start loading the Cortex Disk Image Transfer Utility.
7. Type **Y** at the "Cassette ready ?" prompt.

D

8. Use the Data Transfer Utility to send the cassette image file **DISKIMG.cas** (which should be in the same folder as the Data Transfer Utility program executable).

R

9. The program auto-runs when transfer is complete. Select target drive 0, 80 tracks, single sided, single density (options 0, 2, 1, 1). The program then resets and recalibrates the drive, then prompts to start transfer of the disk image.

A

10. Click the **Disk Images > SEND** button on the Data Transfer Utility and select the **Recovery Utils 80SSD 1.1.dsk** disk image file (which should be in the same folder as the Data Transfer Utility program executable), then press <Return> on the Cortex to start the transfer. Both the PC and the Cortex display the progress of the send operation. The transfer will take a couple of minutes.

11. When the transfer is complete, click the **Quit** button on the progress window to close it.

The created disk should be bootable. Type **BOOT** and check that the Cortex boots into CDOS.

3.9
F

Transferring Disk Images

To transfer further disk images, the created CDOS disk already contains the SAFORMAT program to format a disk (as well as CDOS's own FORMAT program) and the DISKIMG Data Transfer Utility. The process to transfer another disk image can therefore be rather shorter:

T

1. Ensure that the cassette routines on the Cortex have been patched as described in section 3.5.
2. Boot from the CDOS disk in drive 0.
3. Insert the disk that the disk image is to be written to in drive 1 and format it.
4. Run the DISKIMG program from the CDOS disk. Select target drive 1, 80 tracks, single sided, single density (options 1, 2, 1, 1). The program then resets and recalibrates the drive, then prompts to start transfer of the disk image.
5. Click the **Disk Images > SEND** button on the Data Transfer Utility, select the disk image file to transfer, then press <Return> on the Cortex to start the transfer. Both the PC and the Cortex display the progress of the send operation. The transfer will take a couple of minutes.
6. When the transfer is complete, click the **Quit** button on the progress window to close it.