DataMover

Unless otherwise noted, this document and the information herein disclosed are proprietary to JK microsystems, Inc. Any person or entity to whom this document is furnished or having possession thereof, by acceptance, assumes custody thereof and agrees that the document is given in confidence and will not be copied or reproduced in whole or in part to meet the purposes for which it was delivered.

The information in this document is subject to change without notice, and should not be construed as a commitment by JK microsystems, Inc. JK microsystems, Inc. will make every effort to inform users of substantive errors.

JK microsystems, Inc. disclaims all liability for any loss or damage resulting from the use of this manual or any software described herein, including without limitation contingent, special, or incidental liability.

JK microsystems, Inc. recognizes our customer's need for a consistent product and will make every effort to provide one. In order to provide the best possible product for all of our customers, we reserve the right to make incremental improvements in our product designs.

DataMover is a trademark of JK microsystems, Inc. All other brand and product names are trademarks or registered trademarks of their respective companies.

DataMover User's Manual Version 1.0

Printed in U.S.A.
Document Part No. 94-0035
Published April 2007

Limited Warranty

JK microsystems, Inc. warrants each DataMover to be free from defects in material and workmanship for a period of 90 days from the date of purchase. This warranty shall not apply to any unit which has been subject to misuse, neglect, accident, or abnormal conditions of operation.

JK microsystems' obligation under this warranty is limited to repairing or replacing, at JK microsystems' option, any unit returned to the factory within 90 days of the date of purchase, provided that JK microsystems determines that the unit is defective and has been used in compliance with the terms of this warranty. If the failure has been caused by misuse, neglect, accident, or abnormal conditions of operation, repairs will be billed at a nominal cost.

The foregoing warranty is exclusive and in lieu of all other warranties, expressed or implied, including, but not limited to, any warranty of merchantability or fitness for any particular purpose. JK microsystems shall not be liable for any special, incidental or consequential damages, whether in contract, tort, or otherwise.

Important Notice

Life Support / Mission Critical Applications

This product is not fault-tolerant and is not designed, manufactured or intended for use or resale as on-line control equipment in hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which the failure of our hardware or software could lead directly to death, personal injury, or severe physical or environmental damage.

Table of Contents

Overview

The DataMover single board computer is based on the RC R8820LV processor. The R8820LV is a low power 16-bit processor that is software compatible with the 80C186 family of microprocessors. DOS compatibility allows development in a familiar environment with a wide range of tools. Ample flash memory allows for convenient and stable storage of application programs for execution. A variety of accessories add unique capabilities, including GPRS cell modem technology and GPS capabilities.

Software development for the DataMover is remarkably simple and quick. Programs are written on a PC compatible computer in the language of your choice. After your application has been compiled or assembled and lined into .EXE or .COM form, it is uploaded to the DataMover's flash disk with your favorite telecommunications program using the X-Modem protocol. The application can then be tested and debugged through the serial console. When the application is running to your satisfaction, the startup batch file can be modified so that the application will load and execute upon reset or power-up.

Features

20 MHZ RDC R8820LV Processor (x186 compatible)
1 MByte Static RAM (SRAM) Memory
512 KBytes Flash Memory capable of hardware write protection
High Speed Serial Ports (2 External):
    1 RS232 (TxD, RxD, GND)
    1 Jumper Configurable as RS-232 with modem control, or RS-485
Daughterboard Serial Ports (2 Internal)"
    1 TTL Level for Universal Socket Modem (GPRS)
    1 TTL Level for GPS
12-bit A/D, Up to four channels
    Buffered and conditioned
12 Digital I/O lines in groups of four
    4 Individually jumper selectable for A/D
    4 Individually jumper selectable for driver output
    4 Individually jumper selectable for Opto-isolated input
Battery backed clock calendar
Watchdog Timer
Compact Size, 4.6" x 4.3" (116.8mm x 109.2mm),
Rugged Die-Cast Aluminium Enclosure

Operation

The DataMover is configured with two ‘disk drives’. A: and B:. Drive A: contains DOS, the BIOS, and utility programs essential to the operation of the DataMover. Drive A: is read-only. Drive B: is read/write and contains optional utility programs and is available for user files and applications.

The default serial console port for the DataMover is the Serial 1 port on the R8820LV, wired to J5. The port is configured for 9600 baud, 8 data bits, 1 stop bit and no parity. This is the primary mode of communicating with the DataMover. DOS and the BIOS treat this port as the logical device console in place of a keyboard and monitor.

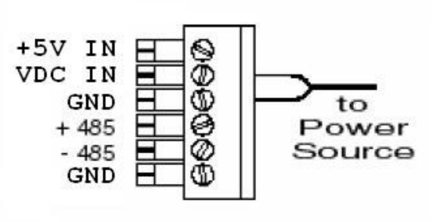
When power is applied to the DataMover or when it is reset, the board goes through its initialization procedure and then starts DOS. A simple (read-only) AUTOEXEC.BAT file is executed and then the board is ready to use. The batch file performs several functions before the user is given control. The DOS search path is set, the DOS prompt is set, the CNTL-C flag (discussed later in this manual) is checked and finally, an attempt is made to execute a file named STARTUP on the B: drive. This provides a convenient way for user applications to execute immediately after initialization of the DataMover. If you wish to have your application start automatically, create a batch file named STARTUP.BAT that invokes the program. Renaming your application STARTUP.EXE or STARTUP.COM is not recommended. If this is done and the program locks up, typing CNTL-C at bootup may not break the program and exit to the DOS prompt.

Warning
Although the flash memory devices used have a guaranteed lifetime of over 10,000 write cycles, it is possible for an application to quickly wear them out. The on board flash memory is intended to store programs, and setup data which is normally not changed. Avoid storing data or frequently changed information on the flash disk. Limiting file write intervals to greater than 15 minutes should insure reliable flash operation for the lifetime of the product.

Warning
To reduce the possibility of an unexpected powerdown corrupting the flash disk, it is strongly recommended that flash file writes be done as quickly as possible. After the file write, close the file and issue a Disk Reset, Int 21h function 0Dh. For more information, see the Application Note on Flash Disk Reliability,
http://www.jkmicro.com/documentation#app_notes

Getting Started

To begin development with the DataMover, you will need a PC compatible computer with a telecommunications program and a free serial port. Connect the DataMover's Console Port to the PC's serial port with a 9-pin serial cable. Run the telecommunications program and configure the serial port for 9600 baud, 8 data bits, 1 stop bit and no parity. Apply power to the DataMover, using our A/C adapter PN 70-0005 or a source of unregulated DC between 8 and 40 Volts (if using the optional Power Auxiliary board). If the Power Auxiliary board is not necessary, 5V regulated DC can be applied to pin 6 of the connector. Observe the polarity indicated on the panel silkscreen.



The DataMover should respond with a welcome message and a B: prompt. Enter `DIR` to look at the directory of drive B:. If you do not get a welcome message or echo of the characters that you type, you need to check your serial port setup. To test everything but the DataMover, remove the serial cable from J2 and jumper pins 2 and 3. If characters typed on the keyboard are not echoed on the screen, the problem is with your setup. You must resolve the problem before you can continue.

If you were able to do a `DIR`, take a few minutes to explore the contents of the DataMover's file system. You will find all of the essential utilities on drive A: and some optional programs on drive B:. Drive A: is write-protected and cannot be altered. Drive B: is read/write and can be changed or reformatted. After you have looked at the programs on the DataMover, the next step is to try to upload a file. This is the procedure for getting a file from your PC to the DataMover. On the DataMover, type the command `UP` followed by the name of the file you wish to upload. The DataMover will begin sending characters to your PC polling it for the file.

On your PC, start the transfer, usually by pressing the PgUp key. The telecomm program should respond by requesting the file name and protocol. Enter the file name and select X-Modem for the protocol. The transfer should start and when it is complete, you should get a new B: prompt on the screen. If the transfer does not work, the problem is most likely the Carrier Detect signal (pin 1 on the DB-9 connector) into the PC being sensed as low or false. Make sure that the signal is at least +3 volts into your PC if you are not able to transfer files.

If the transfer terminated without problems, you have a working development environment for the DataMover controller.

Time and date can be set with the following commands:
`B:\>TIME 13:30:00` Sets the time to 1:30 pm
`B:\>DATE 2-29-2000` Sets the date to February 29, 2000

When power is applied to the DataMover, one of the first things the BIOS initialization code does is check for a CNTL-C character typed at the console. If this character is typed as soon as the board is powered up or reset, a flag is set which overrides the usual boot procedure. When DOS runs its `AUTOEXEC.BAT` file on drive A:, the state of the CNTL-C flag is checked and any user application set to run on drive B: is not loaded. This insures that a hung application or quiet console can always be interrupted. The board will only recognize the CNTL-C at 9600/8/n/1, even if you have changed the console baudrate.
If the CNTL-C flag is not set, the `AUTOEXEC` file will attempt to transfer control to a file named `STARTUP` on drive B:. DOS also looks for and, if present, loads `CONFIG.SYS` from drive B:.

Hardware

Memory Configuration

All but the topmost 32kbytes of memory is used for static ram.

If the boot_sel jumper is installed, of the topmost 32kbytes, the upper half is a 16kbyte window into the onboard flash and the lower half is a 16kbyte window into any device installed in the DIP socket.

If the boot_sel jumper is removed, the window assignments are reversed from the above.

Since the windows only allow access to 16kbytes of the 51k address space available in the flash memory device, a map register is used to select the desired 16k block from the flash that is mapped into the window.

It is recommended that the map register be modified only by the following bios functions:

Load Map -

```
AH = 0A0h
AL = page (bits 0-4 only, rest ignored)
int 15h
```

Inc Map -

```
AH = 0A1h
int 15h
```

Bios Map -

```
AH = 0A2h
int 15h
```

The Map register must never be changed inside an interrupt routine while a disk I/O operation might be in progress. After changing the map register, the BIOS map function should be called to restore the bios image to the map window.

Analog to Digital Converter (A/D)

To read the value of the analog channel, first make sure the jumpers are properly configured to accept A/D inputs. Within the software, do an 8-bit write to the A/D Channel register (604h) containing the channel value you wish to read (0-3).

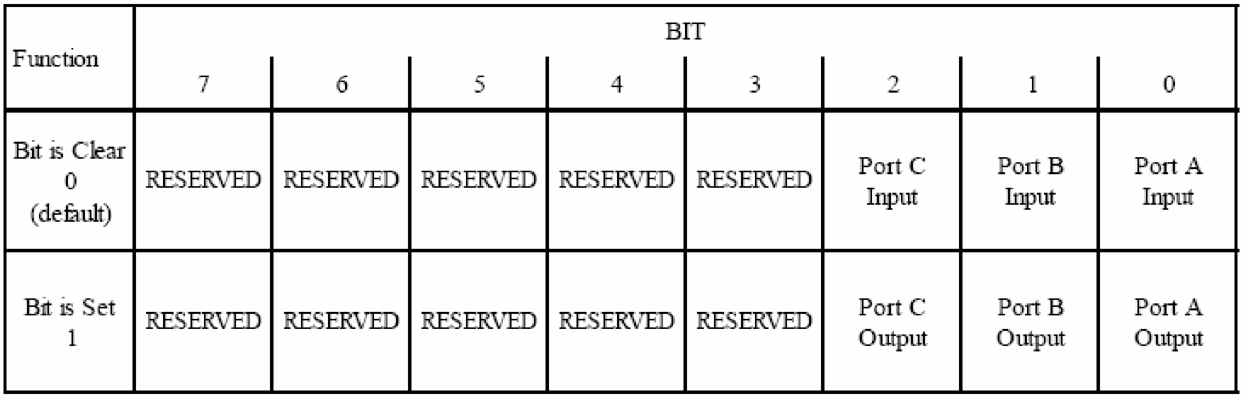
Next do a 16-bit read of the A/D Channel register (604h). If bit 15 is set, the data is not yet ready. If it is clear, the data will be in the lower 12 bits of the value returned. The register may be continuously polled until the data is ready.

Digital I/O Ports

The DataMover has 12 digital I/O lines organized in 4-bit ports. Each of the three ports may be configured as either inputs or outputs. The following tables show the port registers, addresses, and configuration bits.

Address	Register
600h	Port A Data
601h	Port B Data
602h	Port C Data
603h	Reserved
604h	A/D Channel
605h	A/D Data
606h	Config Register

I/O Configuration Register, READ/WRITE (I/O Port 606 hex)



Function	BIT							
	7	6	5	4	3	2	1	0
Bit is Clear 0 (default)	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	Port C Input	Port B Input	Port A Input
Bit is Set 1	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	Port C Output	Port B Output	Port A Output

Table 1: I/O Configuration Register

Ports A through C each have a single data register that is read/write. Data read from the port represents the current state of the port, data written to the port will be present on the port pins, if the port is configured as an output. Each port may be configured as either an input or an output. Write a 0 to the appropriate bit in the direction register to configure the port as an input, write a 1 to configure it as an output. All ports default to inputs.

Port A A/D:

In addition to the 4 bits of digital I/O, port A is jumper selectable for up to 4 channels of 12-bit A/D in the range of 0 to 5V. JP9-12 may be individually moved to position 2-3 to select A/D on the appropriate pin.

Port B I/O

In addition to the 4 bits of digital I/O, port B is jumper selectable for up to 4 driver outputs. JP4-7 may be individually moved to position 2-3 to select driver output on the appropriate pin.

The driver outputs are controlled in the same way that the port B output is controlled, i.e. the port MUST be configured for output in order for the driver to function. The Port B Data register may then be written to as normal.

JP8 may be used to select a power supply for the driver, either 5V from the DataMover or an external source. When using an external source for the driver, make certain to observe proper wiring techniques and ensure that the grounding is appropriately connected.

Port C I/O

In addition to the 4 bits of digital I/O, port C is jumper selectable for up to 4 opto-isolated inputs. JP14-17 may be individually moved to position 2-3 to select opto-isolated input on the appropriate pin.

The opto-isolated inputs are controlled in the same way as the port C inputs are read, i.e. the port MUST be configured for input in order for the opto-isolated inputs to function. The Port C Data register may then be read as normal.

LED

The green onboard LED is to indicate power status. If illuminated, the DataMover is receiving power.

The orange onboard LED is available for user applications, and is connected to PIO5 of the R8820 processor. The LED is located at bit 5 of the PIO Data 0 register at FF74h. For more information on the PIO ports, please see the R8820LV manual.

Programming the I/O Ports

The I/O ports are located in the processor I/O space. Using the ports requires the use of functions unique to the x86 family of processors. Creating a pointer to the location may seem logical, but that reference would be in memory space, not I/O space. The Borland C functions `inport(port)` and `outport(port,value)` are 16 bit (word) instructions, `inportb(port)` and `outportb(port,value)` are 8 bit (byte) instructions. These functions are part of the `dos.h` header file. Similar functions (and header files) are available for other C compilers and languages. The following code illustrates the use of `inportb()` and `outportb()`.

```
unsigned char port;
port = inportb(PORT_DIR); // get value of dir. reg
port |= PORT_DIR_MASK; // set dir. bit for input
outportb(PORT_DIR, port); // write value to dir. reg
printf("PORT: %X\n", (int)inportb(PORT)); // read & print port value
```

Driving Loads with the I/O Ports

The digital I/O ports on the DataMover are capable of driving small loads or interfacing to TTL logic devices. These ports can only source/sink a few milliamps. In order to interface with many loads, the drivers, opto-isolators, or additional circuitry such as a transistor or relay, will be required. Designing the interface between an output pin and a higher current load can present a challenge, especially if high speed is required or the load is inductive in nature. For more information, see Appendix A.

Asynchronous Serial Ports

The DataMover has 2 external serial ports, Serial 0 and Serial 1. Both ports are internal to the R8820. These UARTs are not compatible with the 16450 UARTs on a PC. The maximum data rate is 115k Baud at RS-232 levels.

Serial 0 is wired as Data Communications Equipment (DCE) for direct connection to a computer or terminal. This port is jumper configurable to be either RS-232, or half-duplex RS-485. When configured to use RS-232 levels, this is port implements the RTS, and CTS handshaking control lines.

Serial 1 is wired as Data Communications Equipment (DCE) for direct connection to a computer or terminal. This port is configured as a 3 wire RS-232 port implementing RxD and TxD.

Additionally, the DataMover has 2 internal serial ports. These ports are intended for use with a Socket Modem or a GPS device, and as such, have no connectors for access outside of the enclosure.

The Socket serial port is wired for connection to a Universal Socket Modem. Only available at 5V TTL level, this port implements RTS, CTS, and CD modem control signals. The port is located at address 100h, and the registers are accessible as per a 16550 serial port. The reset pin of the Universal Socket is connected to PIO11 of the R8820 processor, or bit 11 of FF74h. Please see the R8820 manual for more information on the registers.

The GPS port is intended for connection to a GPS device such as the JKGPS. The header implements only RX, and TX signals. The port is located at address 200h, and the registers are accessible as per a 16550 serial port.

RS-485 Configuration

The Serial 0 port of the DataMover can be configured and used for RS-485 communications. To configure Serial 0 as RS-485, move the jumper on JP2 to the 1-2 location. RS-485 signals are present on J4 (the power connector). The RTS line on Serial 0 is used to control the RS-485 transmitter. Library functions are available for use with RS-485 in the supplied driver.
The RS-485 driver is internally looped back. Characters transmitted will appear in the UART receiver. This condition is inerrant to the RS-485 implementation and may present user application implications.

Console Serial Port

As shipped, the default console port is Serial 1 at 9600 baud. To allow maximum flexibility, the console can be located on either of the 2 hardware serial ports at a number of baudrates.To relocate the Console or change the baudrate, use the `console.com` utility and specify the desired port, and the baudrate. After a power cycle (or reset), the BIOS will use the new console settings. See the examples below.

```
B:\>console 1 19200
Changes will take effect when board is rebooted
```

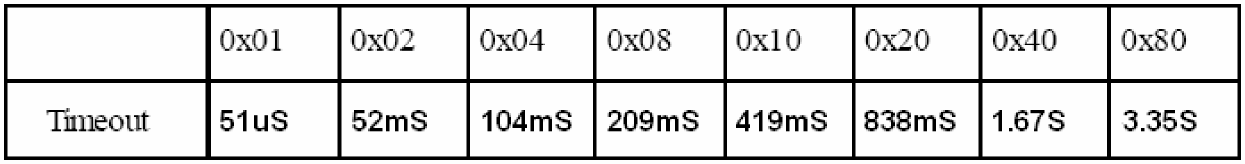
To see a usage message, do not specify any arguments.

```
B:\>console

DataMover Console Configuration v1.00

Usage: console port baudrate
 Valid port ------> 0 1
 Valid baudrates -> 300 600 1200 1800 2400 4800
                    7200 9600 19200 28800 38400
                    57600 76800 115200 153600

Changes will take effect when board is rebooted

B:\>
```

Watchdog Timer

The R8820 is equipped with a watchdog timer that can be configured to generate either an NMI or a processor reset. When enabled, software must keep the watchdog from timing out, indicating proper operation. If the watchdog timer expires, the configured action will be taken. The watchdog signal is not available external to the R8822, so a full board reset is not possible. The watchdog control register is a word located at FFE6h. To Initialize the watchdog, first reset the counter register by writing two words to the configuration register. Write an AAAAh followed by a 5555h to the control register. Next, unlock the register by writing a 3333h followed by a CCCCh to the control register. Finally, write the enable command to the watchdog configuration register. The enable command is one word, with the high byte of C0h (enable watchdog to generate system reset) and a low byte from the table below that represents the timeout interval. Software will then periodically reset the timer by writing two words to the configuration register: AAAAh followed by 5555h. More details are available in the RDC R8820 Users Manual.

```
/* Watchdog Configuration Example */
#define WDTCON 0xFFE6
outport(WDTCON, 0xAAAA); /* reset the watchdog timer */
outport(WDTCON, 0x5555);
outport(WDTCON, 0x3333); /* unlock the control register */
outport(WDTCON, 0xCCCC);
outport(WDTCON, 0xC080); /* set timeout of 3.35 seconds */
```



	0x01	0x02	0x04	0x08	0x10	0x20	0x40	0x80
Timeout	51uS	52mS	104mS	209mS	419mS	838mS	1.67S	3.35S

Table 2: Watchdog Timeout Configuration

Battery Backed Clock Calendar

The DataMover can keep track of the time and date through power loss. On a normal boot-up, the BIOS will read the value of the clock calendar and update the system time appropriately. Additionally, any standard BIOS or C function calls done from user space programs will appropriately change the value stored in the clock calendar. Note that the clock calendar has a resolution of seconds, and thus cannot be set to the millisecond.

DiskOnChip

M-Systems. DiskOnChip is a high performance single-chip Flash Disk. The DiskOnChip has become the standard Flash Disk module for Embedded Single Board Computers. The DiskOnChip is a Flash Disk in a standard 32-pin DIP package that has built-in TrueFFS (True Flash File System) technology, allowing full read/write disk emulation.

The DataMover currently supports the 8Meg Millennium DiskOnChip (MD2800-D08) with version 4.2 (or earlier) firmware. Other DiskOnChip capacities will function if they are loaded with the correct firmware, however M-Systems will not guarantee that the older firmware will continue to be supported on parts other than the MD2800.

Install the DiskOnChip module in the memory expansion socket U6. Note the location of pin 1. If the DiskOnChip is installed and functioning, there will be an installation message that is displayed during the boot process and a C: drive will be available to DOS. For troubleshooting information, please see Appendix B.

Jumpers

***JP1 :* <u>RS-485 Termination</u>**

This jumper selects the termination for the RS485 transmitter. To enable termination for the RS-485 communication, install jumpers across pins 1-2 and 3-4. If no termination is desired, remove all jumpers from JP1.

Default: 1-2, 3-4 jumpered

***JP2 :* <u>Serial 0 Level Select</u>**

This jumper selects Serial 0 as RS-232 or RS-485. Install jumpers across pins 3-4 for RS232, 1-2 for RS-485.

Default: 3-4 jumpered, UART 1 RS-232 level.

***JP3 :* <u>Flash Write Protect</u>**

This jumper selects whether the flash memory should be write protected. This is useful in applications where security of the system is vital, and should not be changed. In order to write protect the flash, move this jumper to position 1-2.

Default: 2-3 jumpered, Flash memory not write protected.

***JP4-7 :* <u>PortB J12 I/O Selector</u>**

These jumpers select the I/O connections on J12. Jumper pins 2-3 for driver output, or pins 1-2 for TTL I/O.

Default: 2-3 jumpered, Driver output selected.

***JP8 :* <u>Driver Power Select</u>**

This jumper connects J12.1 and the driver power supply to either 5V (position 1-2) or to no connection(2-3).
Default: 2-3 jumpered, powered externally.

***JP9-12 :* <u>PortA J11 I/O Selector</u>**

These jumpers select the I/O connections on J11. Jumper pins 2-3 for A/D input, or pins 1-2 for TTL I/O.

Default: 2-3 Jumpered, A/D input selected

***JP13 :* <u>Boot Select</u>**

This jumper selects the location to boot from. Jumper pins 1-2 to boot from the DIP socket, leave open to boot from onboard flash.

Default: Open, boot from onboard flash

***JP14-17* - <u>PortB J12 I/O Selector</u>**

These jumpers select the I/O connections on J13. Jumper pins 2-3 for opto-isolated inputs, or pins 1-2 for TTL I/O.

Default: 2-3 jumpered, Opto-isolated inputs selected.

***JP18* - <u>J4 +5V Power Enable</u>**

This jumper allows for the DataMover to be powered by an external +5V. Jumper pins 1-2 to connect J4.6 to the +5V node. Leave open to prevent accidental connection.

Default: Open.

Cables and Connectors

The following drawings show the signal name for each connector pin.

J5 Power

+RAW	1	2	+5V
+RAW	3	4	+5V
+RAW	5	6	+5V
GND	7	8	GND
GND	9	10	GND

J6 PowerAux

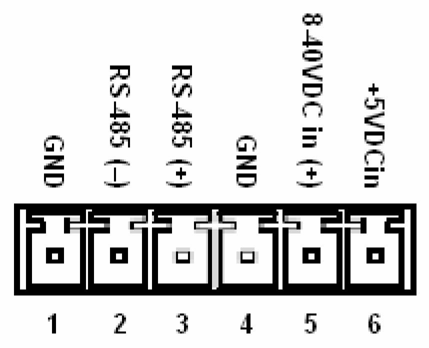
+3.3V	1	2	PIO10
+3.3V	3	4	PIO0
+3.3V	5	6	PIO4
+5V	7	8	GND
+5V	9	10	GND

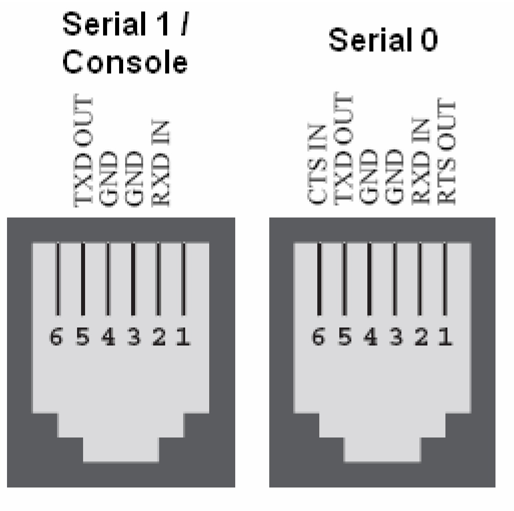
J13 JTAG

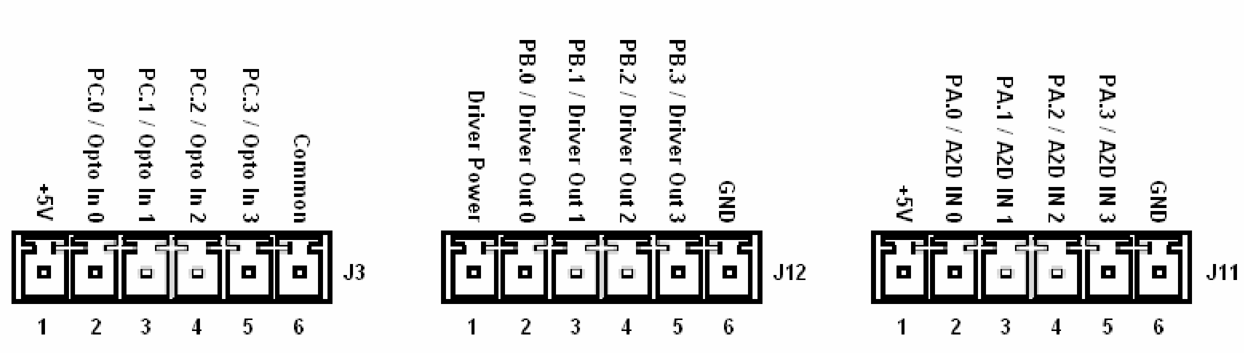
1	+ 3.3V
2	JTAG_I
3	JTAG_O
4	JTAG_C
5	JTAG_R
6	GND

J13 GPS

+3.3V	1	2	GND
TXTTL	3	4	GND
RXTTL	5	6	BATT
GND	7	8	P-UP
GND	9	10	+5V







To locate pin one of a connector, look for the following identifiers. Pin one has a square PCB pad and the others are round. This should be visible on the bottom of the PCB. Pin one will also be identified on the board silkscreen with a ‘1’ and/or a dot. Dual row headers have ODD numbered pins on one side and EVEN numbered pins on the other.

Software

Supported PC BIOS Functions

The DataMover BIOS supports the following functions (software interrupts) common to PC compatible computers. Please refer to a DOS/PC reference for more information on DOS and BIOS software interrupts.

Int 10h, Video Driver, functions 9 and 0Eh
Int 11h, Get Equipment Configuration
Int 12h, Get Memory Size
Int 13h, Disk Driver, Functions 0-4
Int 14h, Serial Port Driver, Functions 0-3

Int 16h, Keyboard Driver, Functions 0 and 1
Int 19h, Boot System
Int 1Ch, Hook Timer Tick Interrupt

Programming Languages

The DataMover can be programmed in the language of your choice, as long as the compiler will generate 16-bit DOS executables (.exe or .com).
A tutorial for getting started with the Borland 4.52 IDE is available on the development CD (see \QuickStart.html) or from our website (http://www.jkmicro.com/documentation).

Utilities

The DataMover comes preloaded with several utilities to aid system development. These utilities are located on drive A: of the DataMover and/or the development CD.

UP.COM

This utility facilitates uploading files to the DataMover via the console port using the X-MODEM transfer protocol. The utility requires the user to supply the name of the incoming file. Unless otherwise specified, the file is placed in the active directory of the current drive. A write-protect error will occur if UP tries to write to the read-only A: drive.

```
B:\>up
Upload file with X-MODEM Protocol
Usage: up file...
Version 3.0 for JK microsystems 186 products
B:\>up test.exe
Ready, start X-modem upload now,
Press CNTL-C to abort...
CCCC
B:\>
```

--

DOWN.COM

This utility facilitates downloading files from the DataMover via the console port using the X-MODEM transfer protocol. The utility requires the user to supply the name of the file to transmit

```
B:\>down
Download file with X-MODEM Protocol
Usage: down file...
Version 3.0 for JK microsystems 186 products
B:\>down test.exe
Ready, start X-modem download now,
B:\>
```

--

FORMAT.COM

If it becomes necessary to reformat the B: drive, FORMAT provides this function. CAUTION, all information on the drive will be lost during the formatting process.

```
B:\>format
Flashlite 186 FLASH Drive Format Program - Version 5.0
System will reboot after successful format...
Press 1 to initialize Drive B as 418 KB disk
Press ESC to exit with no changes
>1
Flash Drive is now formatted
Rebooting system...
```

--

EDIT.COM

A simple line editor is included to allow quick creation and modification of batch files or other text files. EDIT is similar to Microsoft.s EDLIN provided in earlier versions of MS-DOS. It allows list, insert, delete, and modify. Upon exit, a backup of the original file is created (`filename.BAK`) and the edits are saved. If a backup file with the same name already exists, it is overwritten. A list of commands and their usage is available by entering ‘h’ at the edit prompt (>>). The name of the file to edit must be supplied following the command EDIT on the command line.

```
B:\>edit test.bat
FlashLite Line Editor v1.0
Enter h for help
New File: test.bat
>> iz
0: @echo Batch file being processed...
1: mytsr
2: myapp
3: ^Z
>> l
0: @echo Batch file being processed...
1: mytsr
-> 2: myapp
>> q
Save before exit (Y,n): y
File Saved
B:\>
```

--

DOS

JK microsystems. controllers use XDOS, a compact operating system for embedded applications. The XDOS command structure is nearly identical to MS/PC DOS version 3.3. The switches for the DIR command have been changed and expanded. XDOS does not support redirected input or output with the use of < and >, but does support pipes (|). None of the external DOS commands are available due to size constraints. XDOS does not support installable file system functions.

XDOS Command Reference

In the list below, XDOS commands are followed by a function description and their format including available parameters and switches. Items in boldface type must be entered. Capitals or lowercase letters may be used. Items in italics are parameters. Those in boldface italics must be entered, those in [ ] are optional. All switches are optional. They are shown as [/X]. Spaces and punctuation are to be included. An ellipsis ... following items means that you may repeat the items as often as needed. Do not enter the ellipsis or the square brackets. Most XDOS commands allow the use of wildcards in filenames and extensions. When wildcards (?=one character, *=any character or characters) are used, the command is executed once for each matching file.

Common parameters are:

[*d:*] drive specification - a letter followed by a colon (:), e.g. A:, if no drive is specified, the default drive is used.

[*path*] the path DOS must take in traveling from one directory to another; directory names are separated by a backslash (\).

[*filename*] up to 8 characters used to name a file.

[*.ext*] a three character extension may be added to a filename; an extension is separated from a filename by a period.

CD / CHDIR

Function: Changes the current directory
Format: **CD** or **CHDIR** [[*d:*]*path*]

COPY

Function: Copies a file, combines two or more files into one file

Format: **COPY** [*d:*][*path*]*filename*[*.ext*][*switches*]
[+[*d:*][*path*]*filename*[*.ext*][*switches*]
[*d:*][*path*][*filename*[*.ext*]][*switches*]

Switches: /V - verify the contents of new file
/A - copy file in ASCII format
/B - copy file in binary format

DATE

Function: Displays or changes the current DOS date.
Format: **DATE** [*mm-dd-yyyy*]

DEL / ERASE

Function: Deletes (erases) one or more files from a disk
Format: **DEL** or **ERASE** [*d:*][*path*][*filename*[*.ext*]]

DIR

Function: Lists directory entries
Format: **DIR** [*d:*][*path*][*filename*[*.ext*]][*switches*]
Switches: /a - display file attributes
/b - sort by file size (in bytes)
/d - sort entries by date and time
/f - display entries by alphabetic file name order
/n - display entries in directory order (do not sort)
/s - include system and hidden files in output
/p - stop at end of each page
/w - display only the file name
/h - display Help screen (any invalid key)

MD / MKDIR

Function: Creates a subdirectory
Format: **MD** or **MKDIR** [*d:*]*path*

PATH

Function: Specifies directories DOS will search when trying to locate executable files
Format: PATH [[*d:*]*path*[;[*d:*]*path* ...]]

PROMPT

Function: Sets the DOS system prompt
Format: PROMPT [*text*]
Text: Resulting Character(s):
$t The current time stored by DOS
$d The current date stored by DOS
$p The current directory
$v The version of DOS being used
$n The default drive
$g The character >
$l The character <
$b The character |
$q The character =
$$ The character $
$_ Carriage return plus line feed

REN

Function: Renames a file
Format: **REN** [*d:*][*path*]*filename*[*.ext*] *filename*[*.ext*]

RD / RMDIR

Function: Deletes a subdirectory
Format: **RD** or **RMDIR** [*d:*]*path*

TIME

Function: Displays or changes the current DOS time
Format: **TIME** [*hh:mm:ss.xx*]

TYPE

Function: Display the contents of a file
Format: **TYPE** [*d:*][*path*]*filename*[.*ext*]

VER

Function: Displays the DOS version number
Format: **VER**

VOL

Function: Displays the volume label of specified drive
Format: **VOL** [*d:*]

Specifications

Power Supply:	5 VDC +/- 5% regulated, 2W (nominal)
Operating Temperature:	-40 to +70 °C

Port A, B, H:

Symbol	Parameter	MIN	MAX	Units	Condition
V_{IL}	Input Low	-0.3	0.8	V	
V_{IH}	Input High	2.1	3.6	V	
V_{OL}	Output Low		1.2	V	I_{OL}= 10mA
V_{OH}	Output High	2.8		V	I_{OH}= -4mA

Mating Connectors:

Connector	Mfg	MFG P/N	Mfg	MFG P/N	JK micro P/N
2x5 Housing (J2, J4, J7)	Molex	22-55-2101	Oupiin	4072-2X05H	28-0030
Pins	Molex	16-02-0096	Oupiin	404-PIN-10K	28-0033
1x3 Housing, Friction Lock (J1)	Molex	22-01-2031	Oupiin	4071-03H	28-0012
Pins, Friction Lock Housings	Molex	08-50-0114	Oupiin	4071-PIN-T	28-0013

Mechanical:

Dimensions	4.2” x 3.4" x 0.7” 105.6mm x 83.4mm x 17.8mm
Weight	0.138 lbs 62.6 grams

Contact Information

JK microsystems, Inc.
1403 Fifth Street, Suite D
Davis, CA 95616

Telephone:	(530) 297-6073
Fax:	(530) 297-6074
Email:	sales@jkmicro.com (sales inquiries) help@jkmicro.com (technical support)
Web:	http://www.jkmicro.com

Change log:
1.0 First release - April-5-2007 AY

Appendix A

Switching inductive loads such as relays, solenoids and motors can generate transient voltages many times larger than the steady-state operating voltage of the load. For example, turning off a 12 volt solenoid can easily create a negative spike of 200 volts. Worst case, these transients can destroy your controller. In milder cases, they can cause program failures and flash memory corruption. In the case of high current, high inductance devices, the spike need not even be directly connected to the controller to cause damage or program failure.

> **Controllers damaged by inductive spikes are considered to be abused and are not eligible for warranty repair.**

A detailed study of dealing with inductive spikes is beyond the scope of this manual. For more information, a good starting point is The Art Of Electronics, 2nd Ed. (Horwitz and Hill, 1989) pages 52-53.

The following items should be considered when driving inductive loads:

A) When driving a DC inductive load, place a diode in parallel with the load. In most cases, the diode can be a general purpose power diode such as a 1N4002. The cathode (banded end) of the diode should connect to the positive side of the load. Locate the diode as physically close to the load as possible. This applies to a small relay driven by a port pin, as well as a larger inductive load connected to the contacts of a relay.

B) If you are using a relay to switch an AC-powered inductive load, put a varistor in parallel with the load. The varistor voltage rating should be about 2 times the RMS (1.5 times the peak-to-peak) steady-state voltage of the load.

C) Relays switching an inductive load may require a capacitor placed across their contacts. 0.1µF to 1.0µF is a good starting point. If the relays are switching an AC load, place a 100 ohm resistor in series with the capacitor.

D) Do not use the controller’s ground or power conductors to carry current from switched inductive loads. Isolate these signals and route then directly to and from the power supply and as far away from the controller as possible. A separate power source for large inductive loads is strongly recommended. In the case of very large inductive loads, a separate enclosure for the controller may be required.

Appendix B

If, after the installation of a DiskOnChip, DOS fails to return a prompt, the DiskOnChip probably has incompatible firmware and must be reformatted and it.s firmware reloaded. Remove the DiskOnChip, reboot the board, and upload the file `DOC.EXE` from the utilities CD. Power down the board, reinstall the DiskOnChip, and apply power while holding CNTL-C down. Find the file `DOC.EXE` and execute it. It will extract 3 files. Execute the batch file `FMT_DOC.BAT` and answer the prompts. This will format your DiskOnChip and load compatible firmware. You may delete the 3 files after the operation is complete.