Please note:

This information is Copyright (C) 1987 by the author, and may not be published without the permission of the author. You may copy and pass this on to anyone who is interested as long as it is not used for commercial gain, and you don't make a profit from it (you can, of course, charge for your photocopying, postage etc). Any new or relevant information should be passed back to the author so that it can be shared with other people.

The author cannot guarantee that all the facts contained herein are correct and up to date, please tell me of any errors that you find. The author does not guarantee that you can successfully implement a Winchester Disk on your computer system - it is a reasonably complex project.

Winchester disk systems - background

Unlike a floppy disk a Winchester disk drive is much faster (it spins at 3600rpm instead of 360rpm) and can store from ten to hundreds/thousands of times more data. In addition to being faster is also has more heads (from 2 to 16 heads), the data transfer rate from the disk to the controller is 5Mb/s. As a trade off a Winchester disk is permanently sealed in it's housing and cannot be removed, extra controller circuitry is also needed to interface it.

Most Winchester drives are the same physical size as a 5.25" full height or half height floppy drive and come in numerous storage capacities ranging from 5Mb to around 140Mb (10Mb is most common and usually affordable). The drive will have three connectors, power, data and control. The SASI (Shugart Associates Standard Interface) ST406/506 applies to the data and control connectors, basically the control connector is a 34 way edge connector carrying similar control functions to those found on a standard floppy disk, and the data connector is a 20 way edge connector that carries the data to/from the head(s).

The Winchester disk cannot be connected to a micro computer directly, instead an intelligent (microprocessor based) controller has to used. The Winchester controller is a vital part of the system, it is operated under software control of the host and performs all the read/write/seek/format operations on the Winchester - the host has only to send high level "commands" to the controller to be able to use the winchester.

About controllers and all that...

To the host, the Winchester just looks like a large number of sectors (or blocks), the host is not concerned with the track/cylinder/head/sector number, it simply uses a Disk Address (DA) which uniquely referes to each sector (the Disk Address ranges from zero to the maximum which is number of cylinders * number of heads * number of sectors per track - 1). The Disk Address is represented by a 24 bit number (3 hex bytes), so drive Ø head Ø track Ø is DA ØØØØØØ, sector 1 is ØØØØØ1 etc. to the end of the disk. This notation is very convenient as it allows media independence in the filing system, for example a floppy disk can be addressed in the same way, it just has less sectors so it has a correspondingly lower maximum DA.

Again there is a SASI standard for controllers, originally it defined only the electrical interface signals, but a defacto standard emerged for the majority of high level commands, so in most applications controllers are interchangeable. Most controllers can support two winchester drives (connected by the 34 way and 2Ø way sockets) and provide a 5Ø way socket to the host. This socket provides a bi directional 8 bit parallel data port and a number of input and output handshaking signals, these TTL level signals have to be interfaced to the host and software provided for data I/O.

To be more precise the Disk Address is actually a 21 bit number which represents the sector number, and a 3 bit number which represents the device on the controller (or Logical Unit Number, the LUN has a range from Ø-7). Most SASI controllers support more than one device, for example SASI controllers to support 2 Winchesters only, 2 winchesters + 2 floppies, or 1 Winchester and 1 tape streamer are common.

Each connection port on a controller has a LUN related to it such that requests for I/O to a LUN are channeled to a particular port. Some controllers can be daisy chained on the SASI bus to make use of all 8 LUNs (the controller can be configured, by links or software, to map particular LUNs to particular devices). As an example, the DTC 52ØA controller board supports 2 Winchesters and 2 floppies, accesses are as follows:

LUN = Ø	First Winchester (Winchester port Ø)
LUN = 1	Second Winchester (Winchester port 1)
LUN = 2	First floppy (Floppy port Ø)
LUN = 3	Second floppy (Floppy port 1)

Usually a controller can support two Winchesters (all the ones I have met do), but some only allow this if the two devices are of the same type and capacity others allow the size/capacity of each drive to be specified to the controller - it's safer to use one Winchester or two of the same size (a friend nearly ruined a 2ØMb Rodime when using it in conjunction with a 1ØMb Rodime).

There are two distinct types of formatting a controller can use on a Winchester: MFM (Modified Frequency Modulation) and RLL (Run Length Limiting - sometimes known as 2,7 RLL). Basically RLL is the newer technique which uses a type of data compression to get more data onto the disk (a bit like using double density on floppy disks) but requires a disk capable of supporting RLL (the transfer rate is increased to approximately 7.5Mb/s). Also the controller has to be able to support RLL (eg. Adaptec ACB4Ø7Ø). Most controllers on the second hand market support only MFM and most drives were intended for use with MFM. My advice is to stick with MFM unless you really need the extra space - also it's probably cheaper to use a bigger drive with MFM than to purchase a RLL controller.

Controllers are not, unfortunately, interchangeable. That is a Winchester that has been formatted by one controller cannot be read or written to by a different make of controller. This is because the different controller manufacturers use slightly different methods of putting the data onto the disk, different numbers of sectors per track etc.

Another compatibility "problem" with controllers is how they are informed about the size/shape of the Winchester attached. All controllers have a "default" drive configuration at power up (usually a Shugart 5Mb drive) which is not much use unless you happen to be using this type of drive... 3 ways around this problem are used:

1. After power up the host computer sends a command block to the controller telling it about the drive connected (this remains in use until power is lost or the RESET line is pulled low) eg. WD1002. The problem is the host computer needs to know when to send the info to the controller (problem with the BBC as controller reverts to power up default parameters every time BREAK is pressed).

2. If the controller is microprocessor based then the default parameter values are stored as part of the program in Eprom on the controller card. These default parameters can be edited so that they match your drive, eg. DTC controllers.

3. Truly intelligent controllers are "told" about the drive when it is formatted, and a special sector is written to the Winchester (on track 0/head 0) automatically containing drive parameter information. On subsequent power ups (or after BREAK is pressed on the BBC) the controller reads the special sector to find out about the drive, eg. Adaptec ACB4000 and Xebec S1410 series controllers.

Interfacing to a host computer

Generally, to interface a controller to a host, a pair of 8 bit parallel read ports and an 8 bit parallel latched write port are required with some extra logic to handle the control signals (handshaking etc.) - all data into/out of the controller is negative logic, so for the majority of machines, it has to be inverted. The interface between a standard SASI controller and a particular host computer is usually known as a "Host Interface Adaptor" or "Winchester Host Adaptor", further sections of this article describe host adaptors for various computers.

Once a host adaptor is implemented a power supply is required for the drive and controller, typically a Winchester will require +5v at 1A and +12v at 2.5A when running (peak of 4A at switch on), and the controller will require around 3A at 5v, these supplies should be "computer grade", ie. well regulated, probably from a switched mode power supply unit.

Software to "drive" a winchester varies greatly with the host in use, for the BBC micro the routines required are already inside Acorn's ADFS ROM, for a Z80 based CP/M machine a driver routine is required for the BIOS (similar too for IBM PC type machines with MSDOS). If you cannot find software to handle a Winchester on your computer you will have to write routines from scratch - the controller manuals usually give detailed flow charts to make this easier...

Buying winchester disk drives/controllers/hardware

Drives

There is a reasonable supply of cheap drive units available on the second hand market, but prices and quality vary greatly. Points of note:

1. All Winchesters (except the very large units with removable media) are sealed in a clean room, most manufacturers put a seal across the top plate - don't buy a drive that has the seals broken. (Some drives have been back to the manufacturer and been re furbished and may have two sets of seals, this is quite probably okay as long as the top set are not broken).

2. All winchesters have control logic attached to them to operate the stepper motor, regulate drive speed etc. (should have a 4 pin power connector, 20 way and 34 way edge connector) - don't but a lump of die cast aluminium that's no use other than a door stop!

3. Sometimes, if you're lucky, the disk controller is mounted on the winchester and already (still) connected up (by 20 way and 34 way IDC ribbon cable) - if the drive looks okay you might have a bargain, it might even be worth buying the drive just to get the controller board depending on the price!

4. If possible try out the drive on someone's system before parting with cash, or at least try the motor, put power on the drive (+5v and +12v) and watch/listen, most drives (all the ones I've met) are up to speed after about 30 seconds, the end of a successful power up is indicated by the drive re calibrating the heads (stepping them out and back to track zero) - you should hear the stepper motor move the heads. If, after 30 seconds, the drive speed has not settled down (motor speed hunting), or nothing's happened, or if you hear a loud clattering noise, then switch off - choose another drive!

5. Look on the side of the drive for a sticker called "Bad Block Map", "Illegal Address Map" or similar, this details any areas on the winchester that were found to be defective at manufacture - if possible find a drive that has no map entries (a "physically perfect" disk). Many disks have bad blocks, this is not a problem because these can be overcome during formatting making a "logically perfect" disk.

6. Always get technical information on a drive you are buying, you will need to know a number of parameters to configure your controller card/host system. You will need to know such things as the number of cylinders, number of heads, head step time, cylinder number after which reduced write current is used, etc.

There's not too much point in going into who makes what in the way of drives - basically there are two types: head positioned by stepper motor or by voice coil, the later is much more expensive and usually reserved for multi user systems where speed is really critical.

Older 5, 7 and 10Mb drives usually have a fixed head step speed 3mS, whereas newer drives support buffered seek. Buffered seek means that the controller sends all the head step pulses to the drive, which counts them, and then the drive works out the best way to move the head quickly to the required cylinder. Various manufacturers have implemented their own ways of moving the heads quickly, and may call them such things as "fast step algorithms" etc. basically it doesn't matter since it is transparent to the controller.

There's a good chance that you'll see more Rodime drives around than some others as they are a BRITISH (!!!) company (and they are a pretty good make!). The RO200 series is most popular and goes something like: RO201 = 5Mb, RO202 = 10Mb, RO203 = 15Mb, RO204 = 20Mb, RO203E = 30Mb and RO204E = 40Mb. The RO200 series are all 5.25" full height, quiet and reliable.

Controllers

There are a number of different SASI controllers around on the market made by Shugart, Adaptec, Xebec, DTC, Western Digital, NCL etc., most of these are software compatible as far as the command structure is concerned except some of the Western Digital controllers. The list below is not complete or exhaustive, there are other makes of controller that may be suitable.

Shugart

A new Shugart controller is available in the U.S.A. which can emulate Xebec S1410 and DTC 510B controllers, new price is around $79 from companies advertising in BYTE - we just need someone to import them!!

Adaptec (model ACB4000 and ACB4070)

The ACB4000 is used by Acorn in their "official" systems with Rodime or Seagate drives, it works in MFM only. As the ACB4000 is used by Acorn it is fully supported and will work with their SuperForm (Winchester formatter program). Viglen use the ACB4070 which supports MFM and RLL.

Advantages of Adaptec controller(s) are that they are used by Acorn, and store the drive parameters on the drive. The Adaptec controllers have a few opcodes not found on other controllers such as $2F - read drive parameters. This is not a problem as these are only used during formatting, and a different format program for a different controller solves this.

Tame Micro Systems sell Adaptec controllers (ACB4000 = £155, ACB4070 = £175 both +VAT)

DTC (models 510A/B and 520A/B)

Disk Technology Corp. One of the more profuse series of controllers - as DTC have been in the market a long time their controllers have been used with a lot of different machines.

The 510 series controllers support two Winchesters, while the 520 series support two Winchesters and also two floppy disks (using an NEC765 FDC). Basically the B versions of both are newer than the A versions, the firmware in the A versions is stored on a 2732 and in the B versions a 2764 is used. The B version of the controllers have some extra opcodes, but these are not needed for the BBC.

The importer/distributor of DTC controllers is not known o prices can't be found, but I believe that the current new price for a DTC510A is around £110+VAT.

Most hobbiests use the DTC controllers as they have been found on the second hand market at affordable prices (circa £50-£60). With the BBC micro the drive parameters are put into the controller's Eprom as the default parameters.

Xebec (model S1410)

Another profuse controller - available for around £167+VAT new (Farnell). Believed to be used in some third party Acorn compatible products for the BBC, also used in various CP/M and IBM PC type machines.

Advantages of this controller are it's ability to support up to 65535 cylinders and 16 heads for very large drives, and it stores the drive parameters on the drive.

We will be experimenting with getting the S1410A controller going on the BBC shortly when some 67Mb drives become available.

Western Digital (models WD1001, WD1002 and WD1003)

The WD1001 controller series are believed to be somewhat non standard and will not work with the Beeb.

The WD1002 is available in two very different versions:

-WX1 suffix, which is a half length plug in card for the IBM PC using SMD devices (costs around £9Ø) and not suitable for the BBC, and

-SHD suffix, conventional card. The WD1ØØ2-SHD will work with the BBC but there are problems with storing the drive parameters (Solidisk use the 1ØØ2 with their Winchester system but you have to use their host adaptor and their ADFS - I have not found their ADFS particularly reliable!)

The WD1ØØ3 is a newer controller, and apparently has a link to make it Adaptec ACB4ØØØ compatible, but we don't have any more info on it...

WD1ØØ2-SHD disadvantages: the drive parameters cannot be stored on the drive and cannot be programmed into Eprom since the controller does not have an Eprom (it uses custom chips instead of being microprocessor based).

Solidisk have got around the drive parameters problem by storing them inside the ADFS ROM, this means that the ADFS ROM has to be supplied by Solidisk to match the drive. An Eprom could be written for the Beeb that detected power up or pressing BREAK and then sent the parameters to the controller if it was connected - this is being investigated.

You are probably best avoiding WD1ØØ1/1ØØ2 controllers for a BBC application because of the problem storing the drive parameters.

NCL

NCL make a small controller that looks like it should work okay with the BBC, it is sold by Tame MicroSystems for around £8Ø-1ØØ, but as yet no one has tried one.

To implement a winchester disk system on the BBC micro (model B or B+) or Master you need the following:

1. A suitable drive (any size upto 4 Giga bytes is supported!!!).

2. Acorn's ADFS (Advanced Disk Filing System) version 1.30 for the BBC B/B+, or version 1.50 for the Master (already fitted in the Master 128).

3. A suitable controller card: DTC, Adaptec or Xebec recommended.

4. Host adaptor (detailed later).

5. Power supply - you cannot run this of the internal power supply!!! (around +5V at 5A and +12V at 5A should suffice)

6. Cables, you will need the follwing cables:

    1. 50 way to 50 way IDC to go from the host adaptor to the disk controller (typically 6")

    2. 34 way to 34 way IDC to go from the BBC's 1MHz bus to host adaptor (typically 3')

    3. 34 way IDC to 34 way edge connector for drive control signals to controller card (typically 12")

    4. 20 way IDC to 20 way edge connector for drive data signals to controller card (typically 12")

Using a Winchester on the Beeb has absolutely nothing to do with any floppy disk controller that you may have fitted be it 8271, 1770 or any other. If you have something other than a 1770 controller then you might consider changing to one but it is not necessary. The BBC B+ and Master have the 1770 fitted (or the 1772 which is similar), for the BBC B you can fit the Acorn 1770 board or the Solidisk DDFS or DFDC boards (Issue 2). The advantage of using the 1770 is that ADFS can work with it on floppies as well as with the Winchester controller. For compatibility with the older DFS's Acorn's 1770-DFS, or Solidisk's STL-DFS 2.2 can be used.

The layout of "drive numbers" under ADFS is as follows (when a Winchester is fitted):

Drive Number	Drive Letter	Handled by	Description
0	A	Controller	1st SASI device (LUN=0), 1st Winny
1	B	Controller	2nd SASI device (LUN=1), 2nd Winny
2	C	Controller	3rd SASI device
3	D	Controller	4th SASI device

4	E	1770	First floppy
5	F	1770	Second floppy
6	G	-	
7	H	-	

(note a floppy drive under ADFS is a unit, not a surface like DFS, a drive can have 1 or two sides and 40 or 80 tracks).

Basically references to "drive numbers" of 4 or higher are passed to code within ADFS which uses the 1770/1772 controller chip in the Beeb/Master and pretends to be a SASI compatible controller, while "drive numbers" of 0 to 3 are used to generate requests for I/O to a SASI device responding to the equivalent LUN on one of the controller(s) connected (see discussion of LUNs above).

The BBC micro/Master computers support Acorn's Winchester disk units, which are connected via the 1MHz bus. The Advanced User Guide shows that memory locations $FC40 to $FC47 are set aside for Winchester disk units (these are located in page FRED).

A host adaptor on these machines requires 4 bytes of memory space, as 8 bytes are reserved it is believed that two controllers can be mapped on to the system.

Two winchester drives have been used successfully on one controller, and as expected they appear as drive 0 and 1. Unfortunately ADFS 1.30 on the BBC B/B+ doesn't support a second host adaptor, probably because there was not sufficient space in the ROM for the extra code, we can only hope that Acorn will implement the extra code one day???

The memory map looks like this:

Controller	Address	Read	Write
0	FC40	Data	Data
	FC41	Status	-
	FC42	-	Select
	FC43	-	Interrupt latch
1	FC44	Data	Data
	FC45	Status	-
	FC46	-	Select
	FC47	-	Interrupt latch

Providing you have your power supply, winchester drive, ADFS ROM, controller card and cables, the sticking point is the host adaptor - the host adaptor connects the standard SASI controller to the BBC, it sorts out the addressing, buffering and inverts the data bus (SASI uses negative logic).

Acorn do not sell parts of their Winchester disk system - you cannot buy just the host adaptor. A number of other companies now sell complete Winchester disk systems (Amcon, Viglen, Technomatic, Midwich etc.) they have had to design and manufacture their own host adaptors, it is just possible that they might sell the host adaptor seperately. If they do then it should cost you around, say, £50 maximum - any more would be scandalous.

If you can't buy a host adaptor then you have to resort to building one... the next part of this note deals with building a Host Adaptor. Fig. 1 is the circuit diagram of a modified version of Acorn's host adaptor, it can be built for around £35. Double sided PCBs are available from the author for £15.00 plus £1.00 P&P, these boards are being made in small quantities only, by hand, hence the price.

How it works

IC5 (74LS245), IC6 (74LS138) and IC7 (74LS138) perform the address line buffering and decoding, S1 provides a link selectable setting for controller 0 or controller 1 (base address FC40 or FC44). The output of the address decoder provides eight lines (active low), four for read and four for write (shown for controller 0 on the diagram).

IC1 (74LS245) buffers the data bus continuously, IC2 (73LS373) is used to latch data that is written to the controller. Write data is inverted by IC13/14 when enabled by the I/O control signal from the controller card. IC13/14 (7438) are open collector output devices that drive the SASI data port (pins 2 to 16).

Data is read from the controller card, inverted and buffered by IC3 (74LS240). Controller status is inverted by individual Schmitt inverters in IC15 (7414) and latched and buffered by IC4 (74LS373). Acknowledge (ACK) signals are provided automatically after a Request from the controller when a read or write to the data port is performed. Select (SEL) is driven automatically by the Busy (BSY) status from the controller.

An unusual interrupt generating arrangement is also incorporated, a set reset flip flop, IC12, is set every time a Request (REQ) is received from the controller, and is cleared when a "1" is written to bit of the interrupt latch, half of IC10. At certain times this system interrupts the CPU (via the IRQ line). The precise operation of this interrupt system is not known, if it is omitted from the host adaptor then all OSFILE (load/save) operations work correctly, but writes in random access mode (OPENUP, *SPOOL) fail and leave the controller hung in a selected state, we can only presume that it forms a hardware "bodge" to fix a software problem...

The host adaptor connects to the 1MHz bus via a length of 34 way ribbon cable and to the host adaptor via a length of 50 ribbon cable. IDC connectors or cheaper pin headers (available from Farnell) may be used. A power connector (similar to those used on floppy drives) is provided for connection of +5v (no 12v supply is required). The power rails are decoupled at every chip by small 0.1uF capacitors.

The Acorn application note on interfacing to the 1MHz bus recommends that all output and data lines should be pulled up to +5V and down to ground on the LAST DEVICE connected to the 1MHz bus. If the Winchester disk unit is the last device (or the only device) then fit the eight 2K2 resistors and RP1 to RP4.

If the Winchester is not the last device in the chain then a 1MHz bus in and 1MHz bus out socket should be provided to allow daisy chaining - this is most easily done by putting a 34 way IDC socket (one with pins) onto the cable at the back of the Winchester unit. Note that the Teletext Adaptor has the special status of last device in the 1MHz bus chain (it has the terminating resistors fitted).

Fig. 2 to Fig. 4 are the art work for the double sided PCB to build the host adaptor, as follows:

Fig. 2 - Bottom side - track layout
Fig. 3 - Component side - track layout
Fig. 4 - Component placing diagram.

On Fig. 4 the crosses "+" indicate the 0.8mm holes used for the IDC connectors and through pins. Our PCBs are not through plated boards, instead through pins are used to connect the top layer to the bottom, a pin is used in every position that is not an IDC connector.

You may have difficulty making PCBs from this artwork as photocopiers have a tendency to slightly change the size of the artwork and add distortion, as an alternative you could wire wrap a board, or even build the circuit on Veroboard...

Host adaptor parts list

IC1	74LS245	C1	100uF 16V
IC2	74LS373	C2-17	0.1 Disc
IC3	74LS240		
IC4	74LS373	R1-8	2K2 0.5W
IC5	74LS245	RP1-4	2K2 8 resistor SIL
IC6	74LS138	RP5-6	330R 8 resistor SIL
IC7	74LS138		
IC8	74LS14	S1	Wire link
IC9	74LS04		
IC10	74LS74	SK1	Power connector (Farnell P/N 148-086)
IC11	74LS74	SK2	34 way pin header (Farnell P/N 148-195)
IC12	74LS00	SK3	50 way pin header (Farnell P/N 148-195)
IC13	7438		
IC14	7438		
IC15	7414		
IC16	7407		

Controller configuration

For the DTC510A/B controllers there is an ten byte parameter block stored in the controller software EPROM on the controller card. ADFS expects the power up default drive parameters to be correct or the controller to get the parameters from the drive (eg. Adaptec/Xebec).

The following table indicates the bytes that have to be altered and their addresses within the EPROM. <u>NOTE:</u> the addresses are for firmware version WD24 (DTC-51ØA controller) and WD3Ø (DTC-51ØB) if you have a different firmware version number the parameter block may be at a different address.

Address (hex)		Description	RMS	RODIME	RODIME
51ØA	51ØB		5Ø9X	2Ø2	2Ø4
Ø59B	Ø941	Step pulse width	Ø2	Ø1	Ø1
Ø59C	Ø942	Step period	3C	Ø1	Ø1
Ø59D	Ø943	Step mode	ØØ	ØØ	ØØ
Ø59E	Ø944	Max head address	Ø3	Ø3	Ø7
Ø59F	Ø945	Max Cyl. lo byte	D7	3F	3F
Ø5AØ	Ø946	Max Cyl. hi byte	ØØ	Ø1	Ø1
Ø5A1	Ø947	Reduce current cyl.	8Ø	84	84
Ø5A2	Ø948	Do not change	ØØ	ØØ	ØØ
Ø5A3	Ø9~~5~~49	Do not change	ØA	ØA	ØA
Ø5A4	Ø9~~5~~4A	Do not change	ØØ	ØØ	ØØ

NOTE:

1. Max head address is number of heads - 1
2. Max cylinder address is number of tracks -1 (16 bit number)
3. Rodime drives are operated in fast seek mode, buffered with ramped stepper speed controlled by the on board processor on the drive (see Rodime manual).

The format of the ten bytes is the same as detailed in the DTC manual for the "load drive parameters" opcode <u>except</u> that the 16 bit numbers are stored low byte first. For other drives you will have to calculate the parameters and blow an EPROM with the appropriate bytes. If you don't have the facility to edit/blow EPROMs then I can probably do it for you.

<u>Controller links (DTC-51ØA)</u>

Link W1 on the controller should be open to select 256 byte sectors (closed is 512 byte sectors), and link W2 should be across BC. Link W (by the crystal) should be on.

<u>Controller links (DTC-51ØB)</u>

Sect Select links W6 and W7 should both be open to select 256 byte sector operation. W1, W2, W3 & W5 should be open, parity select W8 should be open, and W4 should be connected between 1 & 2. Link IDØ should be connected (nearest to W7).

Once a host adaptor has been built or purchased, and the controller card has been configured for the Winchester the system can be put together, formatted and tested.

Power up the Winchester drive, controller, host adaptor and BBC micro at the same time, ADFS should be installed as the highest filing system ROM (preferably the highest priority ROM, socket 14 is okay is you have BASIC in 15) a high priority socket is needed as ADFS has to service interrupts from the host adaptor, and is reasonably timing dependant, a few ROMs can cause problems if they are in a higher priority socket (eg. IEEE filing system).

If you have a Master you should do a *CONFIGURE FILE 8 to make ADFS the default filing system.

The system should be hanging after the "BBC Computer 32K", "Acorn ADFS" messages waiting for the Winchester to spin up to speed (takes up to 30 seconds) and become "ready". After this time ADFS will try to "mount" the drive, that is, read in the free space map and root directory. The DTC controllers have 8 status leds, they should be displaying 01000000, which is the idle condition, if they are 11111111 (all on) then you probably connected one of the cables the wrong way round or forgot the power on the host adaptor.

If you get "Disk error 5 at :0/000000" then hit BREAK, the system should now hang and wait for the drive, this seems to happen with some RODIME drives on the DTC controller.

Depending what is already on the disk (garbage as far as we are concerned on the first occasion), you will get a "Disk error", "Bad FS Map", "Compaction required" or "Broken directory" message and the system will then probably hang or drop into BASIC.

Press Control F and BREAK, ADFS will be entered without looking for the free space map/root directory. Load the Winchester formatter program from floppy disk (remember that under ADFS your floppy drives are now :4 and :5), example LOAD ":4.SYSROOT.WINCHESTER.DTCFORM"

Run the format program, it should take about 1 or 2 minutes to format the drive, depending on capacity. After the format is complete the drive should be exercised by the diagnostic routine, when finished successfully you can initialise the drive.

Like floppy disks Winchesters can be formatted with a sector skew to improve data transfer rate (allows time for the host to process sectors), we have done various experiments/timings and have found that a skew of 4 to be optimum.

NOTE: The formatter (DTCform) provided does not take care of bad blocks, if when you verify the disk, you have some errors you can note them down and "map them out" of the free space map with HARDERROR utility so that they are not used.

Initialising the drive writes an new (unused) free space map, and an empty root directory to the drive, making it usable to ADFS. The HDINIT program will initialise the drive, initialisation takes only a few seconds, when complete the drive can be used. Make sure that you have modified the variables cyl%, heads%, sect% to match the drive and controller that you are using before you use HDINIT otherwise it is possible to make the disk look bigger or smaller than it actually is.

After you've run HDINIT type *MOUNT Ø Control-Break, and the drive should be mounted, *FREE should reveal the free space on the drive, and *MAP the entries in the free space map. If you are new to ADFS you should read the ADFS manual thoroughly and then start to experiment...

Hints

A library for executable programs (* commands), called "LIBRARY" (or anything starting LIB...) will be selected automatically when ADFS boots, so put your machine code programs there.

A *EXEC file can be *RUN (or just *filename) if its load/exec address is set to $FFFFFFFF, this allows *EXEC files to work from the library (very useful!)

Acorn's utility to recover deleted files (called GETLOST), requires a top level directory called "LostFiles", with two sub-directories called "A" and "B", for this utility to work the directories must be on the disk BEFORE you need them - create them now even if you don't use them.

Many utilities written in Leicester require a top level directory called "Scratch" for storing temporary files. Also these utilities expect themselves to be running from a directory called "sysroot.utils" or "sysroot.winchester", in addition there is a directory called "sysroot.source" for machine code source files and "sysroot.exec" for EXEC files.

Use the following to create the directories (it could be an exec file):

```
*title  My New Big Winny  Disk
*cdir  Library
*cdir  SysRoot
*cdir Scratch
*cdir Lostfiles
*dir  sysroot
*cdir Exec
*cdir Source
*cdir Utils
*cdir Winchester
*dir $.lostfiles
*cdir A
```

```
*cdir B
*dir $
```

A utilities disk is available from the author with a number of useful programs for formatting/verifying/initialising/archiving Winchesters etc. with DTC and other controllers. The programs on the following pages are included along with:

HDarchive Fast Winchester to floppy archiver

PARK to take a copy of the first track on the Winchester and write it to the far end of the disk, this makes a second safe copy of the vital free space map/root directory, and leaves the heads in a better place than the middle of the disk for when you turn the power off.

DU An 80 column file dump routine

The utilities disk cost £5.00 including P&P.

The Advanced Disk Toolkit ROM from ACP has a number of useful commands and is probably a good investment. It includes a formatter for DFS/ADFS floppies (not the Winchester), a disk sector editor for floppy/Winchester DFS/ADFS, a utility to transfer files from one filing system to another etc.