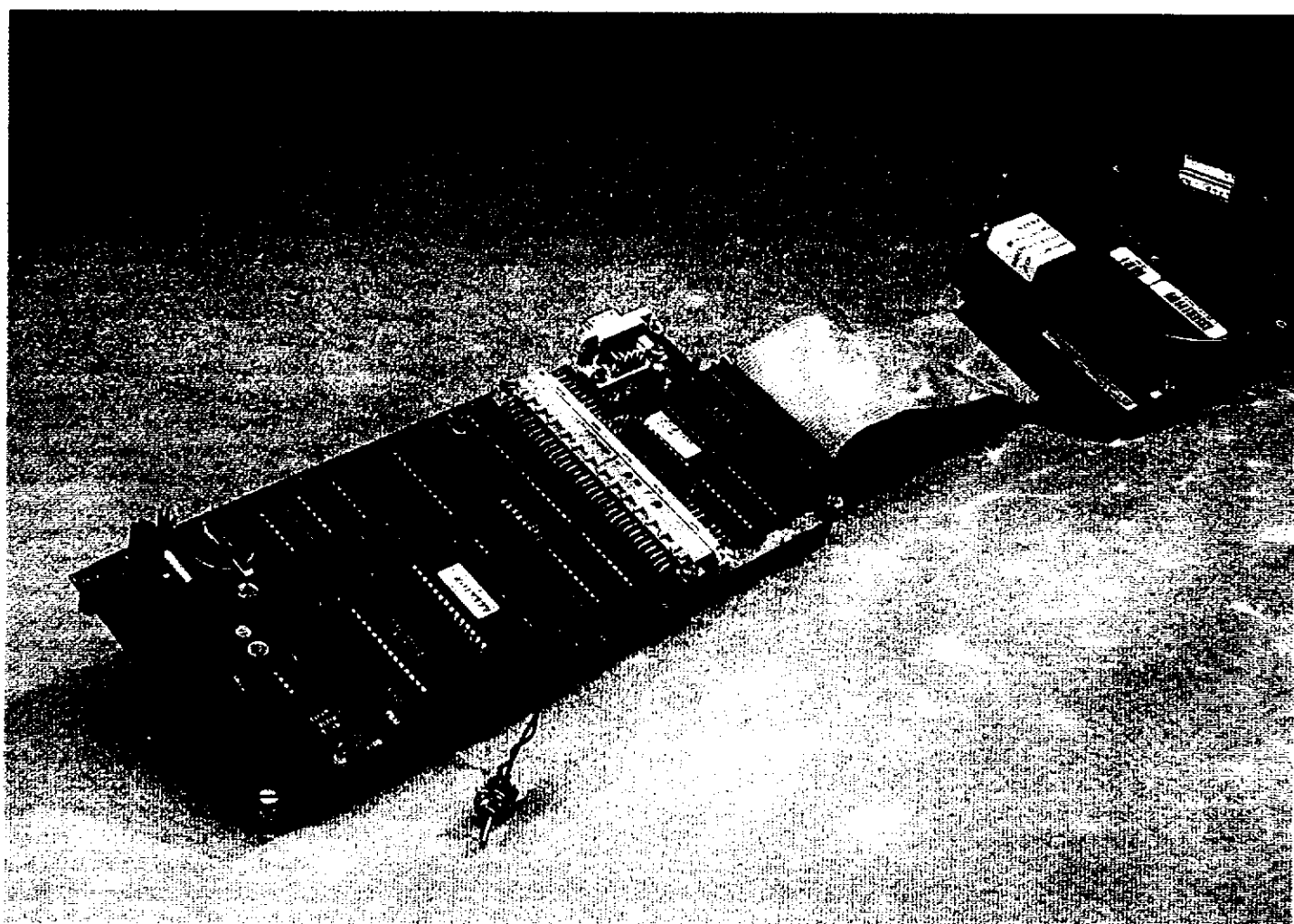


Interface disque dur pour μ C 8 bits

Microcontrôleur cherche relation avec DD

Stefan Schwark

Bien que l'interface IDE soit également dotée d'un mode 8 bits, la plupart des disques durs ne reconnaissent pas ce mode. Il sera difficile, si l'on tient en dépit de tout cela, à entreprendre la commande d'un disque dur par le biais d'un microcontrôleur 8 bits, d'ignorer l'interface décrite dans le présent article.



Qui pourrait s'imaginer un ordinateur moderne qui ne disposerait pas, en tant que mémoire de masse, d'un disque dur. Si, voici quelques années à peine, il s'agissait de périphériques dont le prix était à la mesure de la taille - rapport qui a, depuis, basculé, plus un disque dur est petit (à capacité égale), plus il est cher - il est possible de trouver aujourd'hui (si l'on fait abstraction de la génération la plus récente) des disques durs de très forte capacité et de dimensions « amusantes » à des prix « ridicules ». La plupart des ordinateurs utilisent aujourd'hui des disques durs à interface IDE. Vu que cette interface, également connue sous la dénomination de Interface ATA, tire son origine du PC, et travaille partant avec un système de bus à 16 bits, il est envisageable d'en effectuer la commande par le biais d'un microcontrôleur 8 bits mais les choses sont moins évidentes qu'il paraît à première vue.

En effet, bien qu'il soit possible de basculer l'interface en mode 8 bits, la pratique est bien moins rose, pour la simple et bonne raison que de nombreux disques durs ne reconnaissent pas cette caractéristique. On se trouve de ce fait forcé de faire en sorte que l'électronique au niveau du microcontrôleur mette à disposition un bus de données d'une largeur de 16 bits.

Et c'est très précisément la tâche de l'interface décrite ici. Elle a été développée de manière à permettre à SIMCAD, la mono-carte à 8051/8032 décrite le siècle dernier (en 1991) dans *Elektor*, de commander des disques durs IDE utilisés partout actuellement. Vu la complexité relativement faible de cette interface il ne devrait pas y avoir de gros problèmes pour l'adapter à d'autres types de microprocesseurs.

L'interface IDE (*Integrated Disk Electronic*) a vu le jour en 1984 sous les

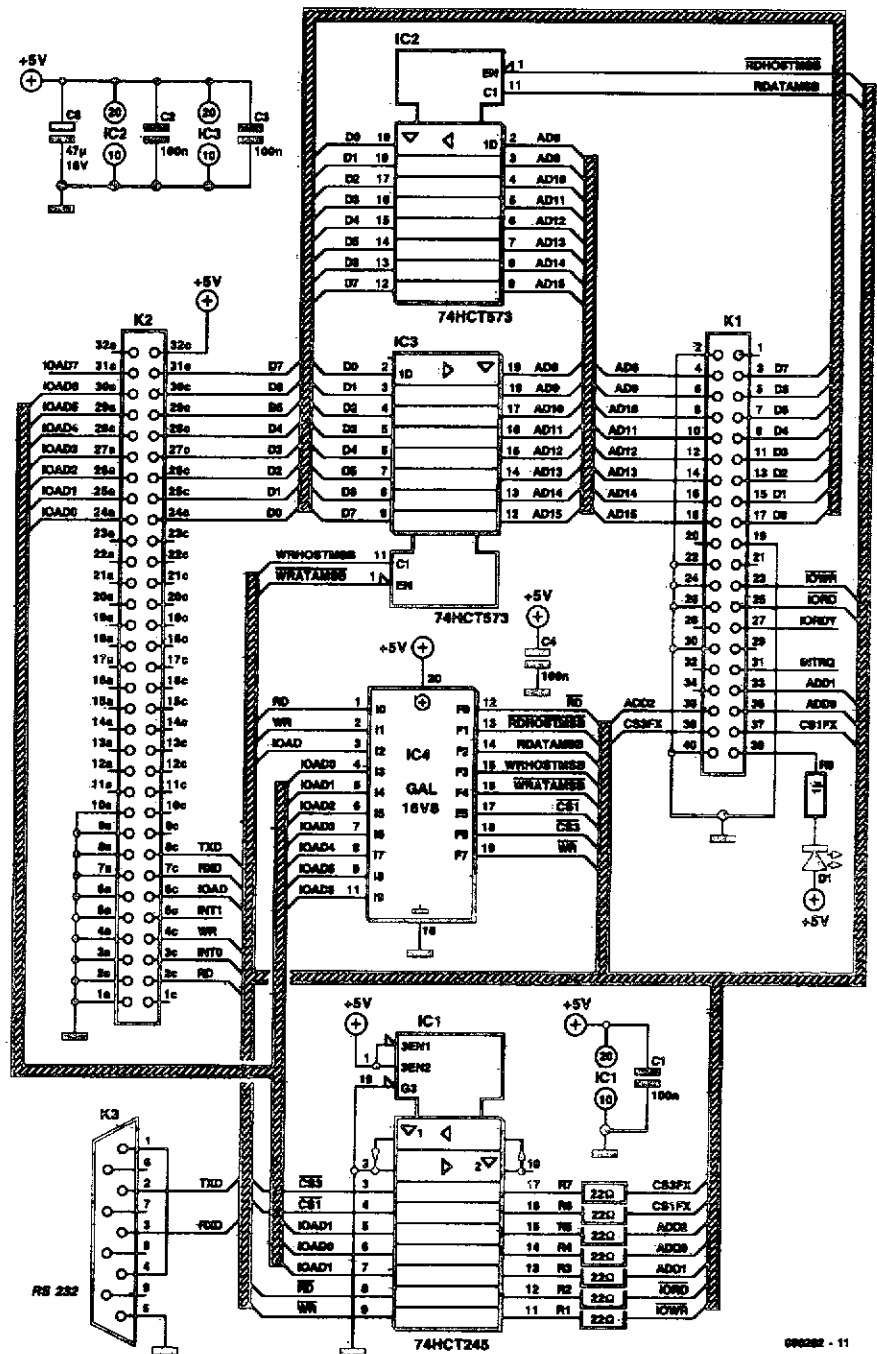


Figure 1. Le cœur de l'électronique de cette interface est une GAL, IC4.

égides de Compaq, l'un des grands fabricants de micro-ordinateurs. Le but de l'affaire était d'intégrer le contrôleur de disque dur, une carte enfichable de bonnes dimensions à cette époque, au disque dur lui-même. La carte enfichable dans le PC pouvait alors être sensiblement simplifiée vu que sa fonction se résumait à l'établissement d'une liaison avec le bus-système. Le connecteur IDE n'est en fait rien de plus qu'un

connecteur AT ramené à sa plus simple expression sachant qu'il est limité aux lignes requises par la commande du disque dur. Ceci explique d'ailleurs l'autre dénomination sous laquelle est connue cette interface, interface ATA (pour AT-Attachment).

Au niveau du PC l'adressage du disque dur se fait par le biais des domaines d'adresses allant de 1F0 à 1F7_{HEX} et 3F0 à 3F7_{HEX}. Une paire de lignes de sélection CS (*Card Select* cette fois) permet de choisir l'un de ces 2 domaines. Le choix de registre proprement

Liste des composants

Résistances :

R1 à R7 = 22 Ω

R8 = 1 k Ω

Condensateurs :

C1 à C4 = 100 nF

C5 = 47 μ F/16 V

Semi-conducteurs :

D1 = LED rouge à haut rendement

IC1 = 74HCT245

IC2, IC3 = 74HCT573

IC4 = 16V8 (programmée EPS 000202-31)

Divers :

K1 = embase HE-10 à 2 rangées de
20 contacts avec encoche de détrompage

K2 = embase femelle à 64 contacts en
équerre selon DIN41612-C

dit se fait par le biais de 3 lignes d'adresses. Nous avons, sur la présente interface, opté pour d'autres adresses que celles mentionnées plus haut. Les 2 domaines de registres se trouvent dans le domaine d'adresses allant de C000_{HEX} à C00F_{HEX}. Nous entrerons dans le détail des registres et de leur objet dans le paragraphe consacré au logiciel.

On utilise, pour la commande du disque dur, le set d'instructions ATA. Il y est défini les codes attribués aux différentes fonctions à réaliser (cf. **tableau 1**). Les instructions destinées au disque dur sont écrites dans le registre de commande à l'adresse 1F7_{HEX} (qui correspond à C00F_{HEX}). Les spécifications du set d'instructions ATA définit les différentes instructions disponibles. Il existe cependant des variantes d'un fabricant à l'autre de sorte qu'un disque dur quelconque ne comprend pas la totalité des instructions.

Depuis sa naissance voici près de 4 lustres, l'interface ATA a fait son petit bonhomme de chemin au point d'être aujourd'hui, sur la grande majorité des PC, l'interface utilisée par le disque dur et le lecteur de CD-ROM (sans oublier les Zip et autres lecteurs de super-disquettes en tous genres). Vu que les spécifications d'origine n'étaient pas prévues pour des périphériques du type lecteur de CD-ROM, elles se sont vues dotées, ces dernières années, d'une extension. Cette interface ATAPI (ATA Packet Interface) reprend, au niveau du matériel, la totalité des spécifications de l'interface ATA. Les modes de transfert et le modèle défini pour les registres ont été conservés tels quels.

Si les instructions ATA normales ne comportent, pour la plupart, qu'un unique code de

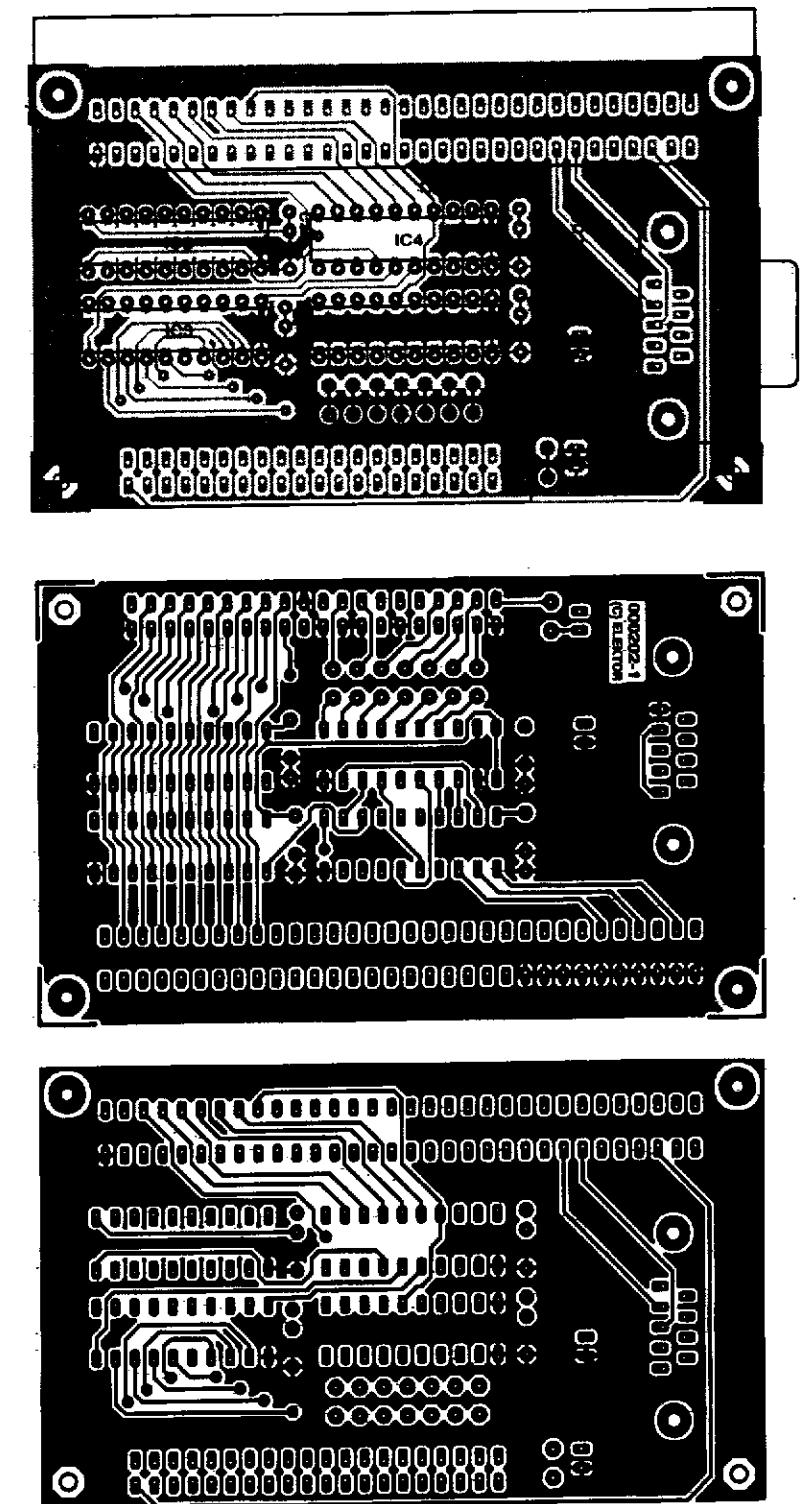
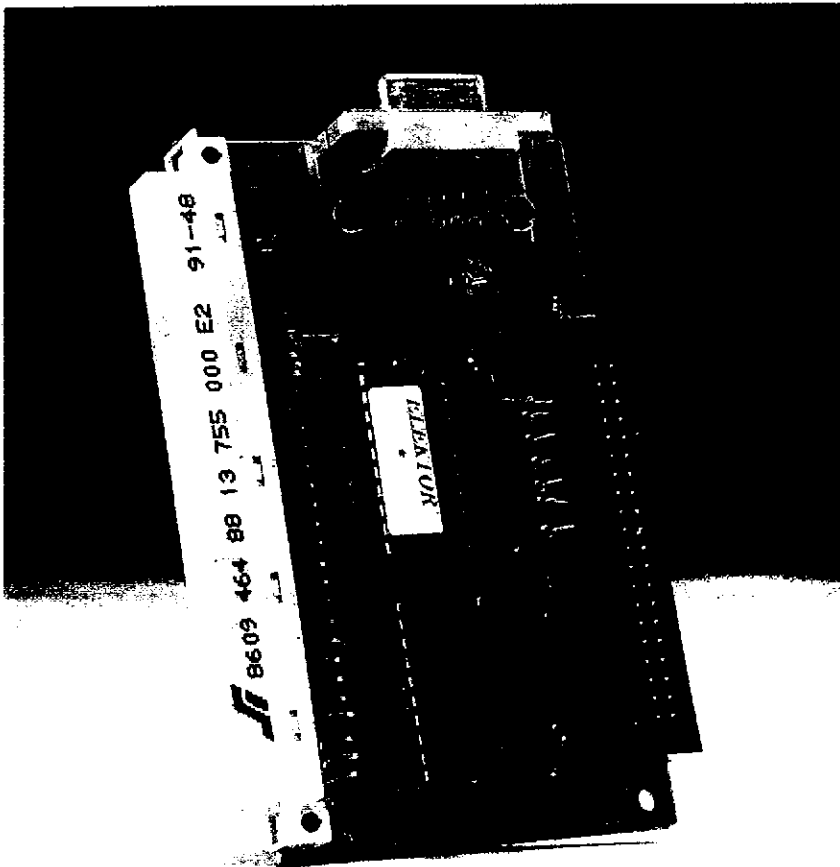


Figure 2. Dessin de la platine et sérigraphie de l'implantation des composants de cette minuscule (tout est relatif) platine d'interface.

commande qui sera écrit dans le registre de commande, les normes ATAPI utilisent un bloc de commande d'une longueur de 12 octets. La structure de ces paquets de données rappelle beaucoup les instructions SCSI, ce qui explique que les

lecteurs de CD-ROM IDE et les lecteurs de bandes (ZIP) par exemple soient en fait des périphériques SCSI qui communiquent avec le PC par le biais de ce que l'on pourrait appeler une sorte d'étage de conversion.



Le matériel

Le matériel de notre interface se résume en fait à un quarteron de circuits intégrés. Le coeur de l'électronique dont le schéma est représenté en **figure 1** est IC4, une GAL16V8 chargée de la commande des verrous (*latch*) IC2 et IC3. Le code-source du programme grillé dans la GAL est disponible sur le site Internet d'Elektor mais également sous la forme d'une disquette (**EPS000202-11**) disponible auprès des adresses habituelles. Cette GAL fournit un certain nombre de signaux de commande indispensables au fonctionnement de l'interface ATA, dont les 2 signaux de sélection de carte CS1Fx et CS3Fx.

Ces 2 signaux servent à la sélection des 2 domaines de registres dont dispose le disque dur. La dénomination de ces 2 signaux réfère au domaine du PC vu que sur le PC les registres se trouvent respectivement aux adresses 1F0 à 1F7_{HEX} et 3F0 à 3F7_{HEX}. La GAL de l'interface utilise le domaine d'adresses allant de C000 à C00F_{HEX}, domaine dans lequel sont incrustés tous les registres. La sélection de chacun des registres se fait par le biais des lignes d'adresses A0 à A2. On trouvera dans le paragraphe « Le logiciel » une description plus approfondie des différents registres concernés.

La GAL génère en outre les signaux d'écriture, WR (Write), et de lecture, RD (Read). On procède, pour cela, à une inversion des signaux WR et RD en provenance du microcontrôleur. Les sorties restantes pilotent les circuits intégrés IC2 et IC3. IC2 sert de mémoire-tampon pour l'octet de poids fort (MSB = *Most Significant Byte*) lorsque le processeur procède à la lecture d'un mot de donnée présent sur le bus.

Dès que l'hôte envoie une instruction de lecture par le biais de l'interface ATA, IC2 transfère l'octet de poids fort présent sur les lignes de données D8 à D15 vers sa mémoire. Lorsque le processeur lit ensuite l'adresse C000_{HEX} IC2 transfère au processeur le dernier octet de poids fort qu'il a pris en compte par la voie du bus de données. IC3 relaie cette fonction dans le cas d'une instruction d'écriture par le biais de l'interface ATA. L'hôte écrit l'octet de poids fort correspondant à l'adresse C000_{HEX}, adresse d'où IC3 transfère cet octet vers sa mémoire. Lorsque le disque dur veut lire ces données, le verrou place l'octet de poids fort sur les lignes de données D8 à D15. Le dernier circuit intégré, IC1, fait office de tampon intermédiaire pour les signaux de commande qui relient et IC4 et le processeur à l'interface ATA.

On trouve, sur la platine, outre le connecteur ATA à 40 contacts K1 et le connecteur

Tableau 1. Instructions ATA.

Instruction	Code	FR	SC	SN	CY	DH
CHECK POWER MODE	E5 _{HEX}		x			D
EXECUTE DEVICE DIAGNOSTIC	90 _{HEX}					D
IDENTIFY DEVICE	E _{HEX}					D
IDLE	E3 _{HEX}		x			D
IDLE IMMEDIATE	E1 _{HEX}					D
INITIALIZE DEVICE PARAMETERS	91 _{HEX}		x			x
READ MULTIPLE	C4 _{HEX}		x	x	x	x
READ SECTOR(S)	20 _{HEX} 21 _{HEX}		x	x	x	x
READ VERIFY SECTOR(S)	40 _{HEX} 41 _{HEX}		x	x	x	x
SEEK	70 _{HEX}			x	x	x
SET FEATURES	E _{HEX}	x				D
SET MULTIPLE MODE	C6 _{HEX}		x			D
SLEEP	E6 _{HEX}					D
STANDBY	E2 _{HEX}		x			D
STANDBY IMMEDIATE	E0 _{HEX}					D
WRITE DMA	CA _{HEX} CB _{HEX}		x	x	x	x
WRITE MULTIPLE	C5 _{HEX}		x	x	x	x
WRITE SECTOR(S)	30 _{HEX} 31 _{HEX}		x	x	x	x

SC = Compteur de secteurs (Sector Counter) C00A_{HEX}

SN = Registre du numéro de secteur (Sector Number Register) C00B_{HEX}

CY = Registre de cylindre (CYlinder Register) C00C_{HEX} et C00D_{HEX}

DH = Registre du numéro de tête (Head Number Register) C00E_{HEX}

La présence à ce niveau d'un D signifie que seuls les paramètres Composant (Device) du registre sont à prendre en compte.

64 contacts DIN4162-C destiné à être relié à la platine du microcontrôleur, une embase Sub-D à 9 contacts. Elle recevra les lignes sérielles en provenance de la carte du microcontrôleur. C1 à C5 sont les condensateurs de découplage classiques, la LED D1 signale les activités des disques durs lorsque le lecteur actif force cette ligne à la masse. Il n'est pas possible de réaliser une différenciation entre un disque dur-maître ou un disque dur-esclave sachant que les 2 signaux en provenance des disques durs utilisent la même broche.

La « construction » de la platine d'interface dont on retrouve le dessin des pistes en figure 2, ne présente pas de complication particulière. Les circuits intégrés pourront être mis sur support. Le seul point auquel il faudra faire attention est de respecter la polarité des composants en ayant une, à savoir le condensateur électrochimique C6, la LED et, bien entendu, les circuits intégrés.

Le logiciel

Le pilotage de l'interface ATA depuis un microcontrôleur implique, outre la présence de l'électronique (*hardware*) requise, l'utilisation du logiciel adéquat, programme chargé de l'écriture des différentes instructions dans les registres du disque dur. Nous allons, un peu plus loin, passer en revue ces différents registres. Aux adresses mentionnées à la suite des noms de registres l'interface ATA est commandée par la mono-carte à 8051/8032, SIM-CAD (Elektor n°154, avril 1991, page 21 et suivantes). Si le disque dur se trouve en mode LBA (*Logical Block Addressing*) certains des registres prennent une autre signification. L'adressage individuel de chacun des secteurs ne se fait plus par le biais des éléments cylindre, tête et secteur mais linéairement sachant que les différents secteurs du disque dur se sont vus numérotés dans leur ordre physique. On a introduit le mode LBA en vue de faire sauter la limite de capacité de 504 Moctets que connaissait l'interface ATA (il arrive également que l'on trouve la valeur de 528 Moctets sachant que l'on considère alors qu'un Koctet vaut 1 000 octets, auquel cas il faudrait d'ailleurs écrire koctets). Le premier secteur a l'adresse 0. Dans le cas d'un disque dur de 504 Moctets le dernier secteur physique aurait comme adresse le numéro 1 032 192. Cette technique d'adressage a permis de faire passer la limite à 7,8 Goctets. On en conclut partant qu'il faudra, si l'on veut connecter un disque dur d'une capacité supérieure à 504 Moctets, utiliser le mode de fonctionnement LBA pour le dit disque dur.

Tableau 2. Informations présentes dans le tableau de partition.

Adresse d'offset	Contenu
00 _{HEX}	Situation Partition (00h = inactive, 80 _{HEX} = Partition de Boot)
01 _{HEX}	Tête de lecture/écriture avec laquelle débute la partition
02 _{HEX}	Secteur et cylindre avec lesquels débute la partition
04 _{HEX}	Type de partition (par ex., 04 _{HEX} pour FAT 16 bits sous DOS)
05 _{HEX}	Tête de lecture/écriture avec laquelle se termine la partition
06 _{HEX}	Secteur et cylindre avec lesquels se termine la partition
08 _{HEX}	Écart entre le premier secteur de la partition (Boot-Sector) et le secteur de la partition [en secteurs]
0C _{HEX}	Nombre des secteurs dans cette partition

Registre de données C000_{HEX}

Le transfert, dans une direction ou dans l'autre, des bits de données 8 à 15 se fait par le biais de ce registre.

Registre de données C008_{HEX}

Ce registre sert au transfert des bits de données 0 à 7. Les données sont valides à condition que le bit DRQ du registre d'état (*status register*) soit positionné (soit à « 1 »).

Bits d'erreur/Feature C009_{HEX}

On trouve ici, en mode lecture, les bits d'erreur qui permettent de retrouver le type d'erreur ayant eu lieu. En mode écriture on transfère à cet endroit les paramètres pour, par exemple, Cache On/Off.

Compteur de secteurs C00A_{HEX}

Ce registre contient le nombre de secteurs à traiter lors du prochain accès au disque dur. Il garde la même signification qu'il s'agisse d'une opération de lecture ou d'écriture.

Secteur de départ C00B_{HEX}

On peut, à cette adresse, lire ou écrire le secteur de départ pour le prochain accès. En mode LBA on trouve à cet endroit les bits LBA 0 à 7.

LSB Cylindre C00C_{HEX}

On trouve à cette adresse les 8 bits de poids faible (LSB = *Least Significant Byte*) de l'adresse de cylindre, en mode LBA se sont les bits LBA 8 à 15.

MSB Cylindre C00D_{HEX}

Il s'agit des 8 bits de poids fort (MSB = *Most Significant Byte*) de l'adresse de cylindre ou les bits LBA 16 à 23 en mode LBA.

Numéro de disque dur et de tête C00E_{HEX}

Les bits 0 à 3 représentent le numéro de tête codé en binaire. En mode normal il existe un maximum de 16 têtes. Le bit 4 sert à faire la distinction maître/esclave (*master/slave*) (1 = esclave, 0 = maître). On trouve, sous la forme des bits 5 à 7, le nombre d'octets par secteur (*byte/sector*). Le paramètre 512 octets/secteur y est fixé définitivement. En mode LBA on trouve à cet endroit les bits 24 à 27.

Registre d'état n°1 et d'instruction C00F_{HEX}

Ce registre reçoit les instructions ATA. Dans le cas d'une opération de lecture il s'agit du registre d'état (*status register*).

On peut dériver, des différents bits, l'état, à cet instant, du disque dur :

- Bit 0 (ERR)

Indique une erreur. Le type d'erreur peut être identifié par le biais des bits d'erreur à l'adresse 1F1_{HEX}.

- Bit 1 (IDX)

Est une impulsion d'index variant d'un fabricant à l'autre.

- Bit 2 (CORR)

Indique le transfert de données après correction ECC.

- Bit 3 (DRQ)

Le périphérique est prêt au transfert de données.

- Bit 4 (DSC)

L'opération de recherche est terminée.

Tableau 3. Brochage de l'interface ATA (KI).

Broche	Signal	Description
2,19,22,24,26,30,40	GND	Masse
1	RES	Signal de RAZ de l'ordinateur.
17	D0	Bus de données bidirectionnel entre le disque dur et l'ordinateur. Ce bus sert au transfert de données, d'instruction et d'autres informations d'état (status). Lorsque le lecteur est inactif, les lignes se trouvent à l'état de haute impédance.
15	D1	
13	D2	
11	D3	
9	D4	
7	D5	
5	D6	
3	D7	
4	D8	
6	D9	
8	D10	
10	D11	
12	D12	
14	D13	
16	D14	
18	D15	
20		Broche coupée pour protection contre une erreur de polarité.
21	/IOCHRDY	L'ordinateur devra attendre, lors d'accès au lecteur, que ce signal soit inactif.
23	/IOWR	Signal d'écriture pour les adresses de ports d'E/S.
25	/IOR	Signal de lecture pour les adresses de ports d'E/S.
27	/IOCHRDY	Idem que broche 21.
28	ALE	Adress Latch Enable
29	libre	Réservé aux extensions
31	IRQ14	Demande d'interruption destinée à l'ordinateur.
32	IO16	Indication de transfert de données sur 16 bits.
34	/PDIAG	Passed Diagnostics : Le lecteur-esclave signale au lecteur-maître la fin des diagnostics internes.
33	A1	Ligne d'adresse de l'ordinateur servant à la sélection des différents registres.
35	A0	
36	A2	
37	/CS0	Adresses 1F0 _{HEX} à 1F7 _{HEX}
38	/CS1	Adresses 3F0 _{HEX} à 3F7 _{HEX}
39	/ACT	Peut piloter une LED signalant les activités du lecteur.

- Bit 5 (DF)

Détection d'une erreur à l'écriture.

- Bit 6 (DRDY)

Le disque dur est paré, après mise sous tension par exemple.

- Bit 7 (BUSY)

Le disque dur est en cours d'exé-

cution d'une instruction.

Registre d'état n°2/Device Control C006_{HEX}

Une opération de lecture donne le même résultat que dans le cas d'un accès au registre d'état n°1. L'état de la ligne d'interruption n'est cependant pas modifié. Il est possible à ce

niveau, dans le cas d'un accès pour écriture, d'activer l'interruption en mettant le bit 1 à « 0 ». Le positionnement (mise à « 1 ») du bit 2 entraîne une réinitialisation (Reset) logicielle du disque dur. Les bits 3 à 7 restent inutilisés. Le bit 0 doit toujours être à « 0 ».

Adresse active C007_{HEX}

Il s'agit là d'un registre à lecture seule. Les bits 0 et 1 permettent de faire la distinction entre les lecteurs maître et esclave. Les bits 2 à 5 abritent le complément à 1 de la tête activée. Une activité d'écriture du lecteur se traduit par la mise à « 0 » du bit 6. Le bit 7 indique au PC que la disquette a été sortie du lecteur. Notre application n'utilise pas ce bit. L'auteur a écrit un petit programme en C (disponible sur le site Internet d'Elektor et sur disquette EPS000202-11) permettant de simplifier quelque peu la programmation; ce programme procède à une démonstration du pilotage du disque dur. **ATA.C** initialise l'interface sérieuse, vu que c'est par son intermédiaire qu'arrivent, ultérieurement, les messages. On a ensuite réinitialisation (reset) de l'interface. Une fois que le disque dur a indiqué qu'il était paré, le programme envoie une instruction d'identification de périphérique (IDENTIFY-DEVICE) au disque dur. Le disque dur se met alors à transférer les données internes dont il dispose. À l'examen il s'agit de données indiquant le modèle, la version, le numéro de série, le nombre de têtes et de cylindres, etc. Le transfert des données émises par le disque dur se fait par le biais de l'interface sérieuse. On a ensuite lecture et transmission du premier bloc de données du secteur de boot. Il est important à ce niveau que l'on ait, à chaque fois, lecture de la totalité des 512 octets vu que sinon on risque de gros problèmes lors d'opérations de lecture ultérieures. Le petit exemple proposé illustre éloquentement la facilité de communication avec un disque dur IDE. Il existe en outre un certain nombre de routines écrites en assembleur MCS51 que l'on pourra, le cas échéant, intégrer dans ses propres programmes.

Structure du disque dur

On a bien évidemment besoin, pour retrouver les données écrites sur le disque dur, d'un certain systématisme. Le premier niveau de la structure d'un disque dur est constitué par ce que l'on appelle le préformatage ou formatage de bas niveau (low level format). Au cours de cette opération la surface du disque dur est dotée de pistes comportant plusieurs secteurs chacune. La taille de chacun de ces secteurs est de 512 octets. Après ce préformatage le disque dur présente la structure d'une disquette « géante ».

Tableau 4. Cartographie de la mémoire de l'interface ATA.

Adresse	Lecture	Écriture
C000 _{HEX}	DB...D15	
C006 _{HEX}	alternate status	device control
C007 _{HEX}	drive address	
C008 _{HEX}	Data	
C009 _{HEX}	error	Feature
C00A _{HEX}	sector count	
C00B _{HEX}	sector number	
C00C _{HEX}	cylinder low	
C00D _{HEX}	cylinder high	
C00E _{HEX}	drive/head	
C00F _{HEX}	status	Command
Registre d'état		
D7	BUSY	device busy
D6	DRDY	device ready
D5	DF	device fault
D4	DSC	device seek complete
D3	DRQ	data request
D2	CORR	correction
D1	IDX	vendor specific
D0	ERR	error
Registre de commande de périphérique		
D7...D23		reserved
D2	SPST	soft reset
D1	NIEN	interrupt enable (active low)
D0	0	always 0

Pour certaines applications, cet état pourrait, en fait, être satisfaisant en l'état. Ainsi, il suffirait, si nous nous contentions d'écrire des données sur un long intervalle de temps, d'écrire de paquets de 512 octets dans une série de secteurs successifs. Cette approche séquentielle n'est cependant pas, dans la majorité des cas, suffisante.

Il n'est pas nécessaire, en tant qu'utilisateur de PC, de s'inquiéter du détail du fonctionnement du disque dur. Les pilotes de périphérique du système d'exploitation ou le BIOS se chargent de cette tâche de suivi. Il devient indispensable cependant, si l'on veut accéder à un disque dur par le biais d'un pro-

gramme que l'on aura écrit soi-même, d'en savoir un peu plus en ce qui concerne la structure interne du disque dur.

Un disque dur comporte une, voire plusieurs, partition(s). Ces partitions, connues sous la dénomination de VOLUMES, sont en fait différents domaines parfaitement délimités du disque dur. Le but de la manœuvre est la préparation des différents domaines du disque dur pouvant être utilisés, le cas échéant, par des systèmes d'exploitation différents. Après le test de la mémoire présente, la vérification des fonctions de base de la carte-mère et l'initialisation de la carte graphique, le BIOS charge le premier secteur d'un disque dur (tête 0, cylindre 0, secteur 1), le secteur de partition. Lorsque le BIOS détecte, dans les 2 derniers octets du bloc de 512 octets, la séquence de code 55AA_{HEX}, il en déduit qu'il se trouve en présence de code exécutable et démarre l'exécution du programme en utilisant le premier octet du secteur à l'adresse 0000:7C00.

Le code stocké à cet endroit a pour fonction de reconnaître la partition active et d'exécuter le secteur de boot qu'elle comporte. Le contenu de la table de partition (normalement à partir de l'offset 01BE_{HEX} dans le secteur de partition) donne à la routine les informations requises sur la position de ce secteur. Cette table comporte, pour chaque partition, une information longue de 16 octets. La routine a vite fait de découvrir laquelle de ces partitions est la partition de boot active puisque les premiers codes sont 80_{HEX} dans le cas d'une partition active et 00_{HEX} s'il s'agit d'une partition inactive. La routine interrompt l'exécution si elle ne trouve pas de partition active ou si elle en rencontre plusieurs simultanément.

Le **tableau 2** récapitule la signification d'une information présente dans la table de partition. Ces informations ne sont pas impératives, mais existent avec pratiquement n'importe quel système d'exploitation. Le système d'exploitation ne peut pas faire grand chose avec la partition. Il faut commencer par formater le domaine qu'elle délimite. Pour ce faire, les secteurs physiques sont subdivisés en blocs logiques (*clus-*

ter), que le système d'exploitation est ensuite en mesure d'utiliser après les avoir concaténés (mis à la queue leu leu).

Le traitement à faire subir à ces clusters est fonction du système d'exploitation. Sous DOS, on utilise la table d'allocation de fichier, la fameuse FAT (*File Allocation Table*). Chacune des données présentes dans cette structure désigne l'un des clusters.

Il y est indiqué à quel fichier spécifique appartient un cluster donné et quel est le cluster suivant du dit fichier. On recherchera dans la littérature accompagnant le système d'exploitation les informations nécessaires et suffisantes quant au mode d'emploi de cette FAT en lecture et écriture de fichiers, sachant qu'entrer dans ces détails nous ferait sortir du cadre de cet article.

Le contenu de l'octet 04_{HEX} décrit le système d'exploitation :

Octet	Système d'exploitation
04 _{HEX}	
00 _{HEX}	Partition non utilisée
01 _{HEX}	DOS FAT à 12 bits
04 _{HEX}	DOS FAT à 16 bits
05 _{HEX}	Partition étendue
08 _{HEX}	OS/2
0B _{HEX}	FAT-32
41 _{HEX}	PowerPC Boot
51 _{HEX}	Novell
64 à 69 _{HEX}	Novell
81 _{HEX}	Linux
83 _{HEX}	Linux swap
A5 _{HEX}	FreeBSD
A9 _{HEX}	NetBSD

Se voir connecter un disque dur est loin de constituer la seule capacité de cette interface. Étant donné que l'interface CD-ROM ATAPI utilise le même modèle de registres, on pourra également utiliser la présente réalisation pour une application de ce type. Il existe, pour ces périphériques ATAPI, une série d'instructions spécifiques par le biais desquelles il est relativement facile d'accéder aux fonctions de lecture classiques de ces appareils. Ne serait-il pas temps de penser à rendre vie à ce vieux lecteur de CD-ROM Double-Speed abandonné au grenier ?

On trouvera des adresses Internet où découvrir des informations sur le thème abordé ici sur le site Internet de l'institut de spécification du standard ATA, à savoir le comité T13 :

<http://www.t13.org>