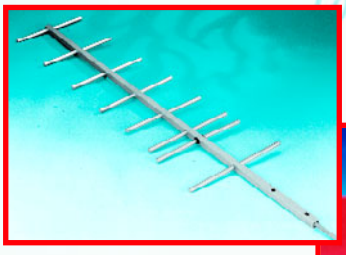
THE ELECTRONICS & COMPUTER MAGAZINE

APRIL 1999
£ 2·85

ELEKTOR ELECTRONICS

PC TOPICS:

video processing on the PC
DIY hard disk installation
how many drives?



Yagi aerial for GSM phones

servo tester

LightFinder

BASIC Stamp controls robot vehicle

multi-purpose power supply

April 1999 Number 276

Elektor Electronics
Volume 25
ISSN 0268/4519

CONTENTS

INFORMATIVE ARTICLES

CONSTRUCTION PROJECTS

MISCELLANEOUS

THIS MONTH IN PC TOPICS:

Modern computer programs offer an awful lot of features and extras, but these come at the price of large demands on hard disk space. Especially if your computer is already a few years old, it can be worthwhile to buy a new hard disk with more capacity than what you now have. In this article, we explain what you should look out for.

DIY hard disk installation

create lots of space — dirt cheap!

Photo: Western Digital

Operating systems and programs are becoming ever larger, and this means that they require more and more space on the hard disk. A full Windows installation can easily take up 300 to 500 MB, and let's not even talk about a full version of Microsoft Office! Fortunately, hard disk manufacturers also manage to continually increase the storage capacities of their products — you might think there is a deal between them and the software developers. On average, hard disk capacity at a given price has doubled every year for the past several years, and there seems to be no end in sight. Anyone who was proud to own a 2 MB hard disk three years ago is probably now struggling with insufficient disk capacity. Luckily, the prices of hard disks are presently quite low, so it is a good time think about buying a larger drive.

The main requirements

Let's go over the main things you should consider when buying a hard disk.

All modern hard disks are at least Ultra-DMA/33 types. The very latest models even support the Ultra-DMA/66 standard. This extended EIDE protocol provides fast data transfers between the hard disk and the motherboard. However, the motherboard must also support the protocol for it to be of any benefit. Even so, it doesn't do any harm to have it, since modern hard disks are backwards-compatible and thus will work faultlessly even with old-model IDE controllers.

Modern hard disks run at 5400 rpm or more. This is important with regard to the data transfer rate. There is an increasing number of hard disks gradually appearing on the market with significantly higher speeds, such as 7200 and even 10,000 rpm. Keep in mind that higher speeds usually mean higher noise levels. Also, high rotational speeds make little noticeable difference for normal applications.
Most hard disks have a cache (data buffer) of 256 or 512 kB. In principle, the bigger the cache the better.
The access time, which is the time required for the read/write head to move to a particular location, is an important quality factor. Commonly specified values range between 8 and 15 ms (average seek time). Be careful when comparing access time specifications between different manufacturers. One may specify 14 ms and the other 9 ms, but the latter may neglect to mention that this omits the 'latency time', which is the time required for the head assembly to settle to a stable position above the track. Latency adds around 5 ms to the seek time, and should actually be included in the average seek time specification.
In addition, as a rule the average seek time is less for 3.5" drives than for 5.25" drives. This is logical, since the heads must move farther in the larger drive.
Finally, consider how the disk will be fitted. You may need a set of mounting brackets to fit a 3.5" drive in a 5.25" mounting bay. There must also be a free power connector inside the computer housing (the large type; the small power connectors that were used several years ago are rarely found on modern hard disks). You will also need a 40-pin flat cable for connecting the disk to the motherboard. Make sure that this cable is no longer than 45 cm, since data corruption can otherwise occur, especially with Ultra-DMA drives (we're dealing with very high frequencies here).

Jumpers, jumpers, jumpers

An IDE drive always has a jumper block with (usually) 6, 8 or 10 pins, normally located next to the IDE connector. This is used to select the 'position' of the drive in the bus chain. Relatively modern motherboards always have two IDE connectors, and up to two drives can be connected to each IDE cable. The jumpers in the drive allow you to indicate whether the drive is the only one connected to the cable, or whether it is the first (master) or second (slave) drive if two

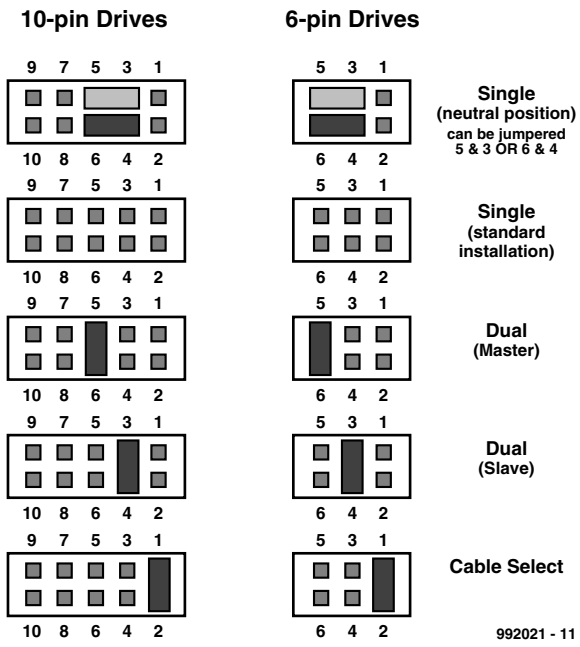


Figure 1. A hard disk can be set up as a solitary disk, a master or a slave by means of a few jumpers. This example shows the jumper settings for modern Western Digital drives.

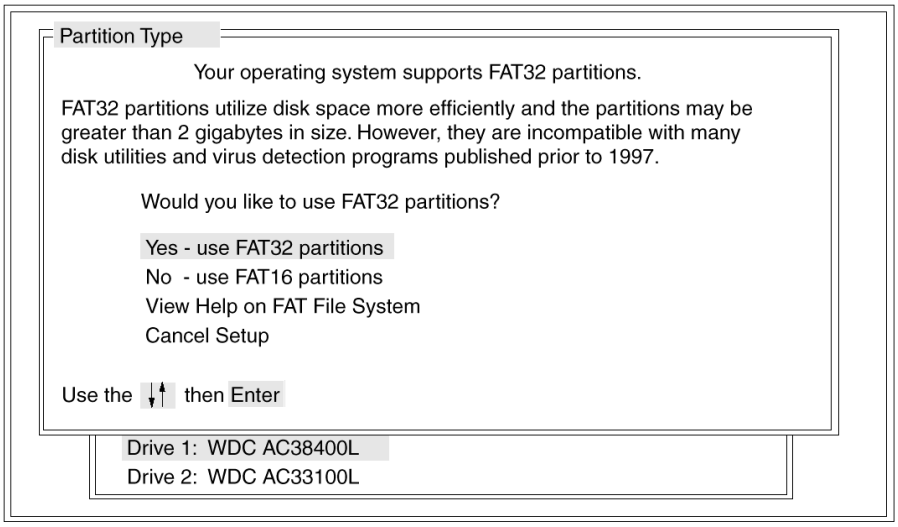


Figure 2. A disk manager, such as EZ-Drive, allows the full disk capacity to be used with an old version of the BIOS.

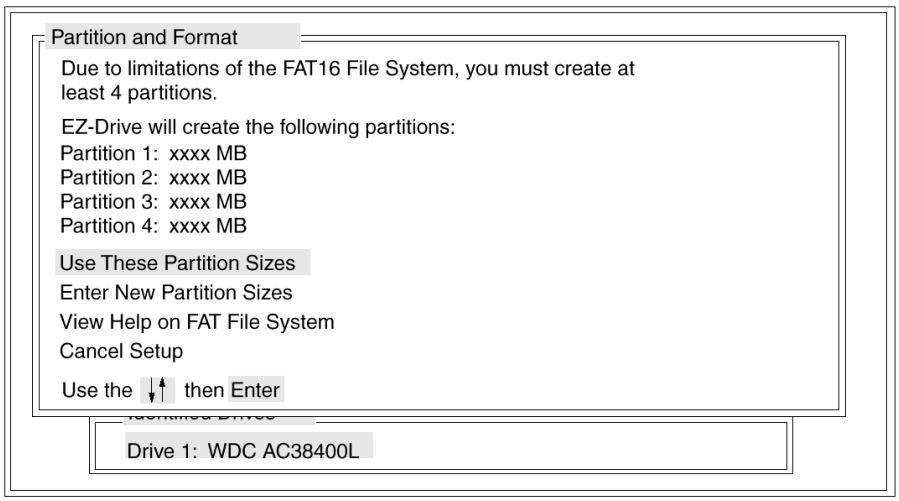


Figure 3. With the latest versions of Windows, you can select FAT32. After this you can divide the hard disk into primary and secondary partitions.

drives are connected.
The jumper settings are usually marked on the drive. If this is not the case, you should be able to quickly determine what they are from the vendor or the manufacturer's Internet site. Almost all manufacturers use three standard configurations: single (one drive only), master and slave (two drives). In addition, there is almost always a cable-select setting, but this is not needed for normal use. **Figure 1** shows an example of the jumper settings for modern Western Digital drives.
By the way, it is not always possible to use a fairly old drive (more than five years old) in combination with a new, modern drive. They may not cooperate with each other. You just have to try it and see whether it works.

BIOS settings

After you have fitted the new hard drive, you must enter the proper BIOS settings before it will be recognized by the rest of the hardware. Relatively modern motherboards have an intelligent BIOS that reads out the drive parameters for itself and then makes all the correct settings. In addition, there is often a BIOS menu item called 'IDE HDD auto detection'. If you select this, the computer searches through all devices connected to the IDE interface and requests their configuration data. It also checks for a number of mode options, such as PIO and LBA. You can select a particular configuration, which is then stored in the CMOS RAM on the motherboard. With an older-model BIOS, you will have to manually enter the necessary parameters in the CMOS setup (number of cylinders, heads and sectors). This information is normally printed on the hard disk drive. The 'precomp' setting is no longer used, so you can leave it at zero, and you should set the landing zone equal to the highest cylinder number or even more.
Don't forget to enable LBA (logical block addressing) in the BIOS if the hard disk is larger than 528 MB. Otherwise it's not possible to use the full capacity of the disk. However, this only applies to operating systems that work with FAT16 or FAT32 disk formats. In addition, none of this information about the BIOS settings applies to SCSI drives, which are installed via the BIOS of the SCSI controller.
There's one final trick that you should know about if you have used a particular disk in the past with a disk manager or boot manager, and now want to use it in a more recent computer. Frequently, this causes problems due to a conflict between the (hidden) utility program in the boot sector and the new BIOS or operating system. You can 'clean up' the boot sector of the hard disk by simply entering the command

```
FDISK/MBR
```

This will not cause any data loss on the hard disk.

Capacity limits

A hard disk is divided up into a number of cylinders, heads and sectors. This is a set of 'working parameters' that is passed on to the BIOS, but it usually does not represent the actual physical configuration of the hard disk. If we multiply the values of all three parameters together, we have the capacity of the hard disk. Before 1995, computers could only work with hard disks smaller than 528 MB (the maximum numbers were 1024 cylinders, 16 heads and 64 sectors). After that, the limits were extended to 2.1 GB (4096 cylinders). More recent motherboards can handle capacities up to 8.4 GB with no problems, and the very latest models have a theoretical limit of around 124 GB!
There is no general rule for determining the maximum disk capacity that a given motherboard can handle. Usually it is necessary to delve into the CMOS setup of the motherboard and see what the maximum values are that can be entered. It may be necessary to update the BIOS to provide support for large-capacity disks.
Another possibility for an older-model motherboard is to use a disk manager, which is a special program that works around the limitations of the motherboard. The disk manager stores a small program in the boot sector of the hard disk, and this program runs every time the computer is started up. It uses clever tricks to avoid the disk capacity limitations of the motherboard.
If you buy a 'retail' version of a hard disk,

you will probably find that a disk manager program (such as EZ-Drive or Disk Manager) is included (see **Figure 2**). If you need a disk manager, you can always download one for free from the Internet site of the disk manufacturer (among others, EZ-Drive is available from the Western Digital site and Disk manager from the Seagate site).

Partitioning

Although a brand new hard disk already has a basic storage structure, this is not enough to allow it to be used by the operating system. It is first necessary to divide the available capacity into one or more (active) partitions, and then to format each of the partitions to produce clusters of a particular size.

Partitioning is the process of splitting a hard disk into one or more 'logical' drives, each of which can be assigned its own drive letter. With DOS and Windows, a primary DOS partition is essential, and you may also have an extended DOS partition. An extended partition can be subsequently divided into a number of logical drives (stations).

For DOS and Windows, the program Fdisk is used to divide a hard disk into a number of partitions. It is a good idea to make a boot diskette containing the Fdisk and Format programs (among others), and preferably also the drivers needed to activate a CD-ROM drive, before you start to partition a hard disk. Make sure that you use the versions of these programs that came with the most recent version of Windows that you have.

If we take a modern Windows version as an example (Windows 95 OSR2 or Windows 98), the first thing that Fdisk asks after it starts is whether you want to enable support for large disks (see **Figure 3**). This means that you can choose here between a FAT16 system and a FAT32 system (FAT means 'file allocation table', which is a sort of library that keeps track of what programs are stored where on the hard disk). FAT32 is the system that supports large disks, and if you chose this then larger partitions and smaller clusters will be supported on the hard disk (we'll come back to this later).

It is best to select large-disk support, but you should be aware that FAT32 can only be used under Windows 95 OSR2 and Windows 98. A FAT16 partition is required for old-fashioned DOS programs that will not run in a Windows window.

If you use Windows 98, you can initially select FAT16 and later convert one or more partitions to FAT32 with the help of the program Drive Converter (DRV1.EXE), to be found on the Windows 98 CD-ROM (see **Figure 4**). If you do this, there's no way back — a FAT32 partition cannot be changed back to FAT16! However, a combination of FAT16/FAT32 is possible on a single hard disk by starting FDISK a few times with a different FAT selection.

One or two hard disks?

Normally, a DOS-based operating system assigns the drive letter A to the first physical floppy drive that it finds in the system. If there is a second floppy drive, it is assigned the letter B; otherwise B is not assigned.

The system always assigns drive letter C to the primary DOS partition of the first physical hard disk drive. After this, the system looks for a second hard disk. If it finds a second hard disk, it checks to see if it contains a primary DOS partition. If it does, then this partition receives the drive letter D.

If the system does not find any other

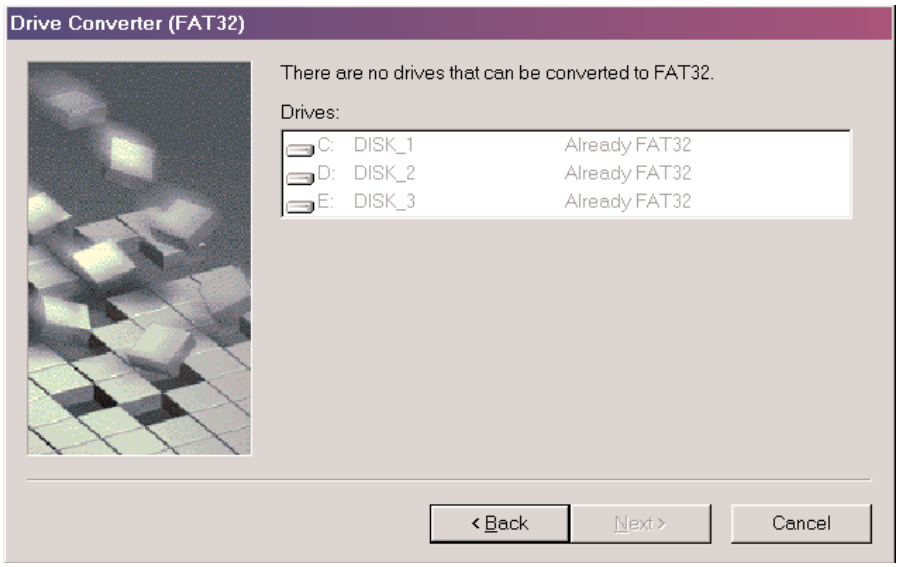


Figure 4. The Drive Converter utility that comes with Windows 98 can be used to convert a FAT16 partition to FAT32.

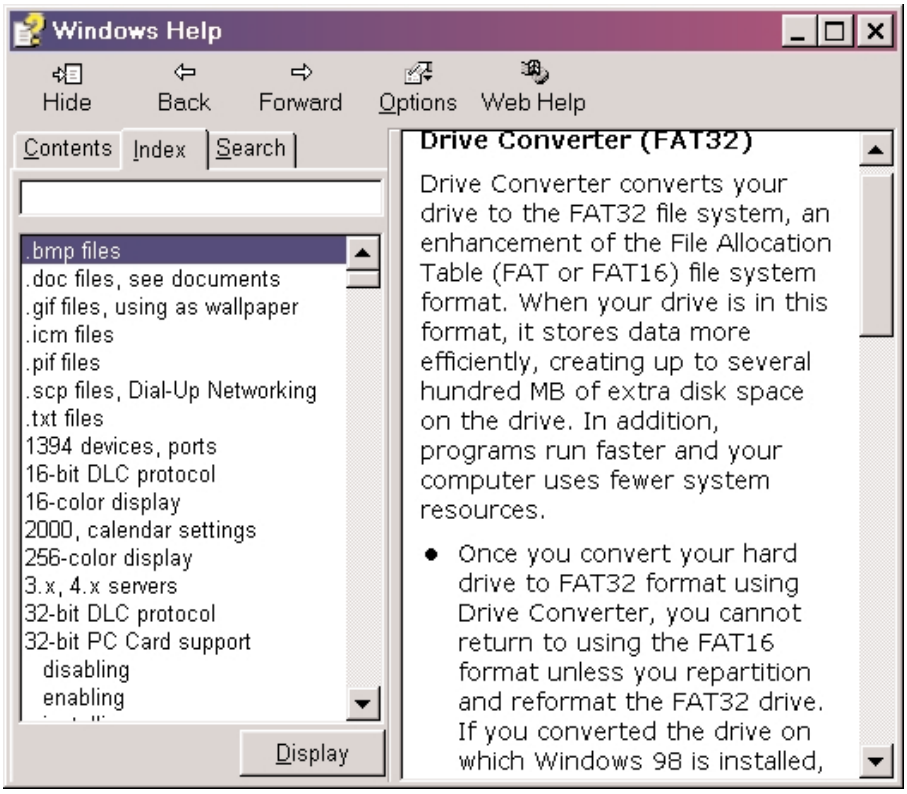


Table 1. Hard disk data transfer rates.

IDE-bus

IDE (ATA):	
single word DMA 0	2.1 MB/s
PIO mode 0	3.3 MB/s
single word DMA1, multi word DMA 0	4.2 MB/s
PIO mode 1	5.2 MB/s
PIO mode 2, single word DMA 2	8.3 MB/s
EIDI (ATA-2):	
PIO mode 3	11.1 MB/s
multi-word DMA1	13.3 MB/s
PIO mode 4, multi-word DMA 2	16.6 MB/s
Ultra-ATA (Ultra-DMA/33):	
multi-word DMA 3	33.3 MB/s

For comparison, a few transfer rates for SCSI hard disks:

SCSI-bus

	8-bit SCSI (50-lead cable)	16-bit SCSI (68-lead cable)
SCSI1	5 MB/s	-
Fast SCSI, SCSI II	10 MB/s	20 MB/s
Fast-20, Ultra SCSI	20 MB/s	40 MB/s
Fast-40, Ultra-2 SCSI	40 MB/s	80 MB/s

hard disks (a total of eight are allowed), it returns to the first hard disk and looks to see if it contains any logical drives (in an extended partition). It assigns sequential drive letters to any such logical drives. After this, it checks the second hard disk and assigns sequential drive letters to any logical drives it finds there.
In principle, all partitions have now been assigned drive letters. However, to maintain compatibility with old versions of DOS, the system next checks to see if there are any other primary DOS partitions on any of the drives. This was namely possible with some of the older versions of DOS. If any such partitions are present, the system assigns drive numbers to them as well. Virtual devices, such as CD-ROM drives and Zip drives, now get their turn. However, device drivers must usually be loaded for such devices before drive numbers can be assigned to them.
If there is only one hard disk in the system, then the situation is usually clear. In this case you create a primary DOS partition plus an extended DOS partition, and you divide the extended partition into several logical drives. These are assigned the drive letters C, D, E … in sequence.
With two hard disks, it's more complicated. You have to be especially careful if you want to use continue to use the old hard disk as a second drive and you want to partially copy its contents to the new hard disk.
There are two possibilities in this case. The first, and least confusing, is to completely repartition both drives if you plan to use the new drive as the first drive (which is to be recommended, since is will usually be faster than the old one). You then have something like the following:

Partitions:
Hard disk 1
drive C: (primary DOS partition)
drive D: (logical drive 1 in extended DOS partition)
drive E: (logical drive 2 in extended DOS partition)
Hard disk 2
drive F: (logical drive 1 in extended DOS partition)
drive G: (logical drive 2 in extended DOS partition)
drive H: (logical drive 3 in extended DOS partition)

As you can see, hard disk 1 has a primary DOS partition and an extended DOS partition, but the second hard disk has only an extended partition. This causes all partitions to be nicely assigned drive letters in sequence.
Suppose that instead of this, you want to use your old disk as the second disk as it is, without altering the way it is partitioned. You will in any case have to make a primary partition on the new hard disk, plus an extended partition if you wish, since otherwise the system cannot boot from the hard disk. If both the old disk and the new disk have three partitions, you end up with the following arrangement:

Hard disk 1
drive C: (primary DOS partition)
drive E: (logical drive 1 in extended DOS partition)
drive F: (logical drive 2 in extended DOS partition)
Hard disk 2
drive D: (primary DOS partition)
drive G: (logical drive 1 in extended DOS partition)
drive H: (logical drive 2 in extended DOS partition)

Naturally, you can learn to live with this, but you always have to take the 'non-sequential' assignment of drive letters into account. For instance, if you want to copy a file from the old drive C to the new disk, you have to do this by copying it from drive D to (for example) drive E.

Table 2. Standard cluster sizes for various partition sizes.

FAT16

Partition size	Cluster size
< 32 MB	512 KB
< 64 MB	1 KB
< 128 MB	2 KB
< 256 MB	4 KB
< 512 MB	8 KB
< 1 GB	16 KB
< 2 GB	32 KB

FAT 32

Partition size	Cluster size
< 512 MB	not possible (FAT16)
< 8 GB	4 KB
< 16 GB	8 KB
< 32 GB	16 KB
> 32 GB	32 KB

The limitations of FAT16

With Windows 95 and the first versions of Windows 98, the maximum addressable partition size was 2.1 GB (due to FAT16 and a maximum cluster size of 32 kB). This means that users of these systems must always divide hard disks with more than 2.1 GB capacity into partitions that are smaller than 2.1 GB. This restriction was removed by FAT32 in the second service release of Windows 95.

Formatting

After the hard disk has been divided into partitions, the operating system

can assign and recognize drive letters, but the disk is still not ready for use. First the partitions must be formatted, which amounts to organizing them into 'bite-sized' chunks that the FAT can work with. The familiar Format command, used for formatting floppies under DOS, is also used for formatting the hard disk. The command

```
format c: /s
```

causes drive C (or better, partition C) to be formatted, following which the system files of the operating system are immediately copied drive C so that it can subsequently be used as the start-up disk. The suffix /s is naturally not needed for formatting the other partitions.

Large and small clusters

When a drive is formatted, it is divided up into clusters. A cluster is not the same thing as a sector, which is the smallest physical data structure on the hard disk (512 bytes). A cluster consists of several sectors. Each cluster has an entry in the FAT, from which it can be assigned to a file. Given that the FAT has a maximum size, and considering that looking up clusters in the FAT becomes slow if the FAT is too large, the cluster size varies with the size of the partition. If only for this reason, it is advisable to divide a large hard disk into two or three partitions (such as three 2 GB partitions for a 6 GB hard disk). FAT16 is not only limited to a maximum drive capacity of 2.1 GB, it also supports a much smaller number of clusters than FAT32 (which is hinted at by the number '16'). Table 2 shows the cluster sizes for various partition sizes with FAT16 and FAT32. It is a good idea to choose a compromise between the cluster size and the total number of clusters. This will avoid wasting too much disk space on the one hand, and having a large FAT that degrades system performance on the other hand. As a simple example, if FAT16 is used with a partition size of 2 GB, the cluster size is 32 kB. If a 1 kB icon file is written to the disk, it thus takes up 32 kB. This can lead to a lot of wasted space, especially if a lot of very small files are stored, so that only 1 to 1.5 GB of the original 2 GB is effectively being used. Fortunately, the situation is better with FAT32, but you should be careful not to let the total number of clusters become so large that it slows down the computer. For example, with a partition size of 2 GB and a cluster size of 4 kB, we already have half a million cluster allocations!
It is advisable to manually increase the cluster size for partitions from 2 to 8 MB. This can be done with the undocumented format option /z:x, where x is the number of 512-byte sectors that should make up a cluster. For instance, the command

```
format /z:16
```

will divide the hard drive into clusters of 8 kB (16 ⇔ 512 bytes).

Conclusion

Keep in mind that 'low-level' operations on a hard disk are almost always destructive, which means that they erase all data on the disk. Before you start, be sure to make a backup of any data on the hard disk that you want to keep. If you want to alter partitions without losing the data already present, you can use a commercial program such as Partition Magic. Unfortunately, these programs are not inexpensive, and it's a shame to lay out so much money for something that you use only a few times.

(992021)

Although two drives are more expensive than a single drive with the same net capacity, two smaller drives are better than one large drive! With multiple drives you simply have more options. The most important of these relates to backups. Since it is unlikely that both drives will fail at the same time, copying the contents of one drive to the other one provides a quick way of making a backup. In addition, with a bit of construction, one can manage to solve the problem of how to run two separate software systems on a single PC.

By Dr W. Matthes

how many drives?

All about switchable two-disk configurations and tape cassette drive installation

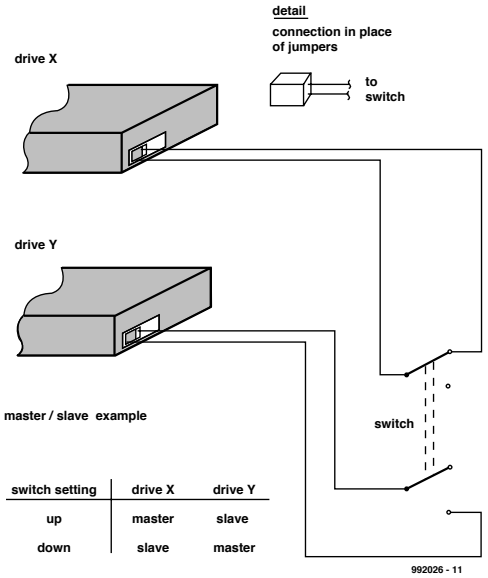


Figure 1. A simple drive switching arrangement.

A 'simple and stupid' drive-switching solution

Figure 1 shows how a simple double-pole changeover switch can be used to select either one of two drives as the master drive. All that is necessary is to switch the jumper connections which determine the disks' master/slave configuration (as identified in the documentation). One can then (for example) install a 'traditional' DOS/Windows 3.x system on one drive and (after switching over) install a newer version or a different system (such as Windows 95, OS/2 or Linux) on the second drive.

Since one of the drives is always the slave drive, all files of both systems are always accessible (assuming that both systems use a common file structure — such as the DOS FAT system — or that there is appropriate operating-system support, such as with the OS/2 hpfs).

Notes

1 *Practical construction (1):* use female connectors in place of the jumpers. These can be sawn from a double-row header and filed on the sides so that they fit between the jumper pins.

2 *Practical construction (2)*: one switch section is needed for each jumper position. At minimum a double-pole switch is required.

3 *Operation (1):* actual switchover often only happens after the machine has been switched off.

4 *Operation (2):* modifying the Setup parameters. If both drives are the same type then no modifications are necessary. Otherwise the following operational procedure is required: (1) switch off the system; (2) throw the changeover switch; (3) switch on the system; (4) call Setup (eg with [Del]) and (5) enter the master and slave drive parameters (if you have a 'handy' DOS version, start AUTO DETECT and confirm the returned parameter values). This may all sound a bit a complicated, but nonetheless you can in this manner quickly install a completely new operating system environment and work with it. If there are problems, it only takes about five minutes to restore the old system — and all of your files are still accessible.

Switching a drive off-line

IDE drives cannot normally be switched 'off line' via a jumper. You can work around this by breaking the interface Reset line (easily done since it is lead 1, at the outer edge of the cable). The drive which is to be placed out of service is held permanently reset by a simple switch. This arrangement can be added to the disk

changeover switch. **Figure 2** shows how it is done.

Tape cassette drive

This section briefly describes how to connect a backup drive to the floppy-disk interface.

Conventional tape cassette drives.
Traditionally the tape drive is attached in place of a floppy- disk drive. In PCs with one floppy-disk interface the tape drive thus replaces drive B:. The floppy- disk drive B: must then be 'deleted' in Setup. In principle the tape drive may be configured in place of any floppy- disk drive except for drive A:, since there is no BIOS support for a cold start from a tape drive.

Drive switching (1)
Two floppy-disk drives and a single tape drive can be used on a single interface by switching the Drive Select and Motor Enable signals (see **Figure 3**). Appropriate cables with built-in switches are commercially available, but you can also make your own. Separate the proper leads from the flat cable, cut them, and solder on extension leads as necessary. Don't forget to insulate the connections (preferably with heat-shrink tubing).

Drive switching (2)
With this approach the interface cable remains intact. Instead of modifying the cable, the drive select jumpers of the floppy-disk and tape drives are replaced by a changeover switch and associated cabling (see **Figure 4**).

Modern tape drives.
Modern drives are designed such that one tape drive and two floppy-disk drives can be used on a single (unmodified) interface. Note: usually the existing interface cable can remain in place; a special branched cable ('Y-cable') is used to connect the tape.

Practical tip
Configure the cassette tape drive as drive B:, delete floppy-disk drive B: in Setup and address the second floppy-disk drive via a different drive letter. While this does not save you from having to switch over the drives, it does mean that you do not have to restart the PC every time you do so. You can use the DRIVER.SYS command in the config.sys file to assign drive letters. For example, to configure a 5¼" floppy-disk drive enter

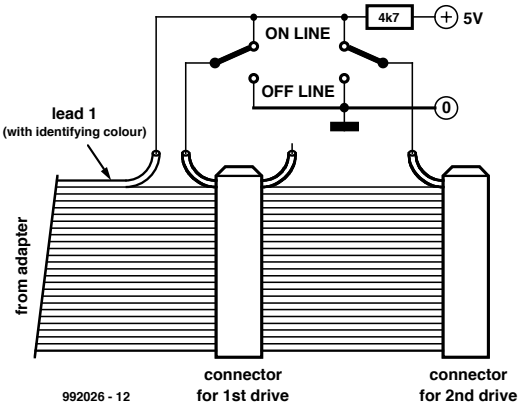


Figure 2. Drive on-line/off-line switch.

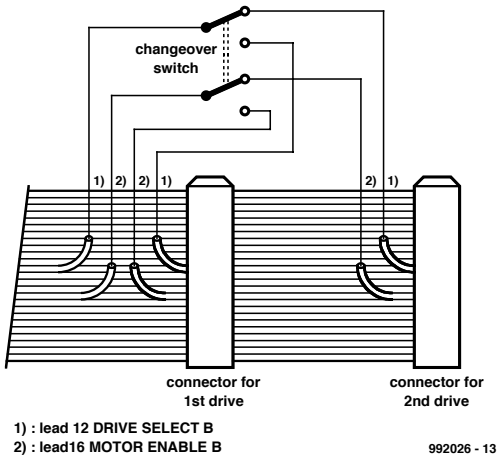


Figure 3. Drive switching via the interface cable.

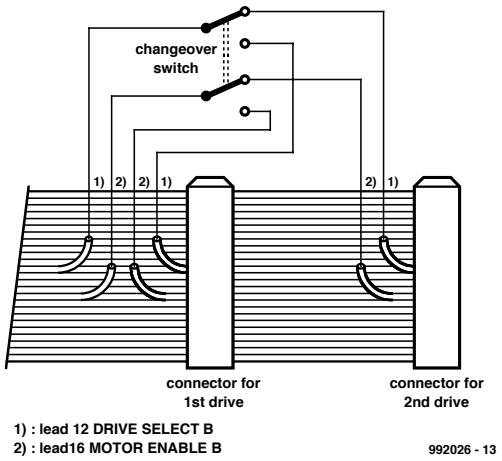


Figure 4. Drive switching via the Drive Select jumpers.

DEVICE=C:\DOS\DRIVER.SYS /D:1 /C /F:1

This assigns the next free drive letter after that of the last hard-disk drive

Notes
1 This technique can also be used to operate three floppy-disk drives on a single interface. You must of course switch the drives over (drive B: and drive F:, for example) according to which drive is to be used.
2 Tape drives normally do not need drive letters, since they are accessed via special device drivers instead of via the BIOS and DOS.

(992026-1)

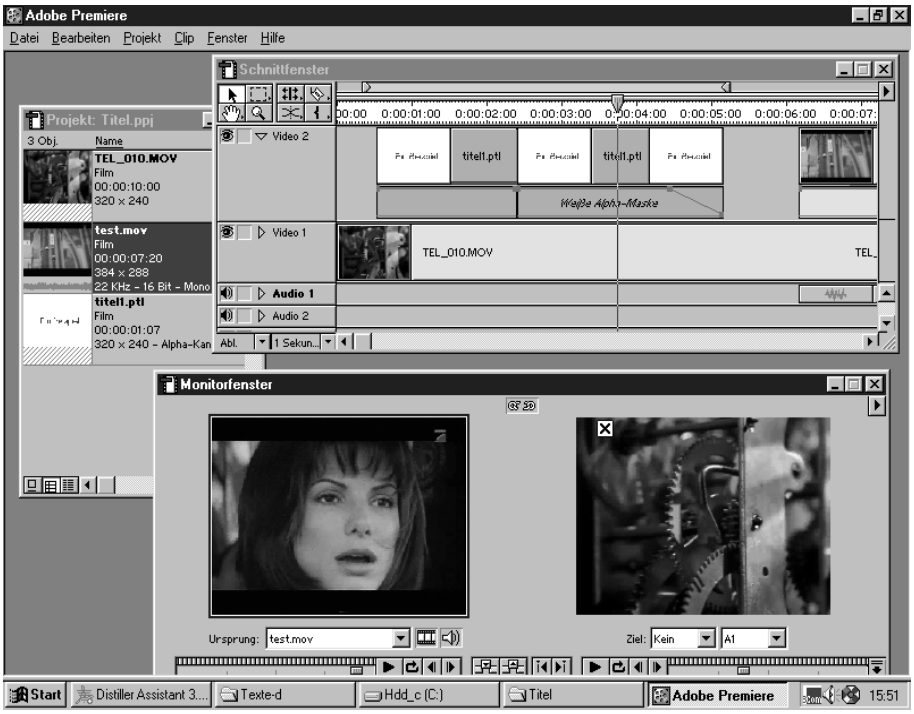
This article is an extract from the book *PC Service and Repair* by Dr W. Matthes, Elektor Electronics (Publishing), ISBN 0-905-705-41-6.

Video capture cards have become so cheap that a PC can now be used for sophisticated video editing. Nearly all the cutting techniques and effects that we commonly see on TV can be achieved with a video capture card in combination with suitable software: blends and fades, digital effects, filters, animations, titling and so on. Of course, this all requires a lot of disk space, and calculating the various effects is very time-consuming. Practically speaking, this means that video clips prepared using an inexpensive editing system with 'normal' hard disks should be limited to less than 10 minutes of playing time.

By Dr M. Seiwert

Video processing on the PC

Part 1: basics



Video editing systems, consisting of a video capture card and matching video editing software, are available starting at around £200. The video capture card must be installed in a free PCI slot that has bus master capability, so that it can directly transfer data to and from the memory without the intervention of the processor. Video capture cards have VHS inputs and outputs with Cinch jacks, and most of them also have Super-VHS inputs and outputs with mini-DIN jacks. In use, the card is connected to a camcorder or video recorder, which acts initially as a playback source. It is preferable to use a S-VHS connection, since it transfers the chrominance and luminance signals separately. This reduces the crosstalk between the two signals. The sound signal is either processed in a separate part of the video capture card or connected directly to a sound card, depending on the particular video capture card used. The sound signal is connected via a Cinch cable. **Figure 1** shows the block diagram of a typical video capture card.

When the video capture card receives a television signal (from a video recorder, for example), it decodes the input signal, digitizes it and converts it into a sequence of bit-mapped images. Depending on the design of the video capture card, the images are displayed on the computer moni-

tor and/or a television set that serves as a control monitor. Each bit-mapped image occupies 1.4 MB at full resolution, which is roughly the capacity of a 3.5-inch diskette. The computer must of course handle up to 30 such images per second and store them in memory. Since even fast disks cannot handle data at this rate, the bit-mapped images are compressed before they are stored. Special ICs on the video capture card are used to perform this compression at the real-time image data rate, since the PC processor cannot normally handle this task. The compression ratio can be adjusted for the best compromise between the disk data transfer rate and image quality; this depends on the type and configuration of the PC used. If the compression ratio is too small, the disk will not be able to keep up and image data will be lost. On the other hand, if the compression ratio is unnecessarily high then the image quality will be degraded by compression artifacts.

The compressed image data are stored together with the sound data as *.avi files on the hard disk. These AVI files are different from the AVI files that you may be familiar with from multimedia applications in one important aspect: they can only be used on the system where they were generated, or on a system with the same model of video capture card. For playback, the image data must be decompressed at the required rate, which also takes place on the video capture card.

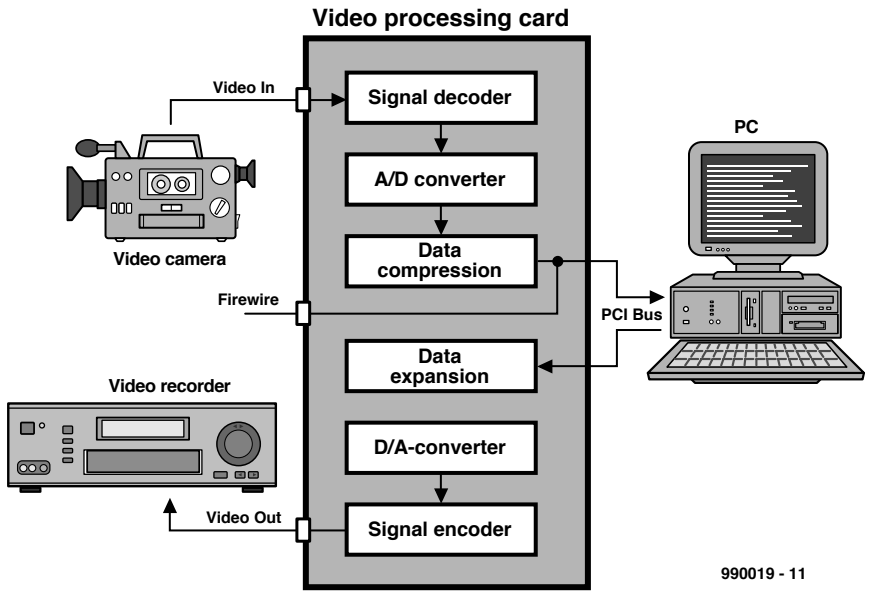


Figure 1. Block diagram of a typical video capture card.

Many video capture cards display played-back AVI files only on a television, and not on the computer monitor. The AVI files are the starting point for video editing, which is discussed in part 2 of this article. Once a video clip has been fully edited, it can be

Figure 2. An example of the comb effect. In this recording of a speech, the missing clappercard was replaced by a handclap. Left: no comb effect, right: comb effect

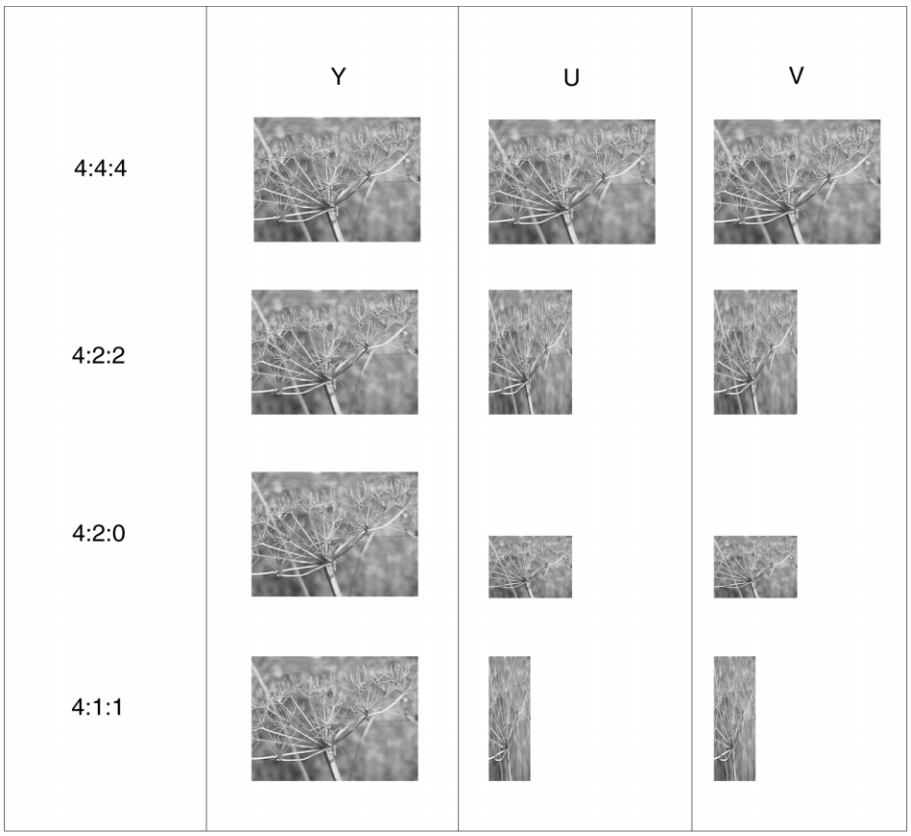


Figure 3. Summary of commonly used black/white and colour resolution relationships.

'played back' to the video recorder. The outputs of the video capture card are connected to the inputs of the video recorder for this. Playing back an AVI file involves the same processes as capturing a video signal, in the reverse order. The digital video and sound data are decompressed by the video capture card and converted into an analogue television signal that appears at the video output connectors. This signal can be recorded by the video recorder.

Alternatively, the finished clip can be further processed to produce AVI files that can be played back on any standard PC. This requires the data compression and decompression to be performed by a software algorithm, instead of by the codec hardware of the video capture card. The algorithm is independent of the video capture card and is executed by the PC's processor alone. Naturally enough, AVI files prepared in this manner are displayed on the computer monitor.

Table 1. The relationship between the compression ratio, data rate, file size and image quality.

Compression-ratio	Data rate, MB/s	Duration per GB, min:sec (PAL)	Image quality and applications
3:1	7	2:26	Digital Betacam – television, advertising
5:1	3,5	5:16	DV – news reporting, digital computer videos
7:1	3	5:41	S-VHS, Hi-8 – news reporting, analog computer videos
10:1	2,1	8:08	S-VHS, Hi-8 – computervideos
15:1	1,4	12:11	VHS – amateur projects

Digital camcorders and video capture cards

Digital camcorders have been commercially available for some time now. They are wonderfully compact, and their picture quality (relative to the demands of the consumer market) is excellent. In a digital camcorder, the video signal is digitized and compressed before it leaves the camcorder. The tape drive records the compressed digital signal. This means that the first three of the video signal processing functions that usually take place in a video capture card have already been performed in the camcorder.

There are 'digital' video capture cards that are designed to work with digital camcorders. These have connectors for the digital Firewire bus, in addition to the usual VHS and S-VHS connectors. (For an explanation of Firewire, see the PC Topics Supplement in the February 1998 issue of *Elektor Electronics*.) This allows the already digitized camcorder data to be transferred directly from the video cassette to computer files. The AVI file format is not used in this case; instead, the 'digital video' format is used to produce files with a *.dv extension. This file format directly corresponds to the data format on the video cassette. The compression ratio is not selectable in this case. Instead, it is fixed, and thus a sustained data transfer rate of 3.5 MB/s is required. If the hard disk drive cannot support this data rate, it's time for an upgrade. Of course, these cards can also capture and store standard video signals via their analogue inputs. Such signals are digitized, compressed and stored in the DV format, in exactly the same manner as a digital camcorder works when making a video recording.

Special aspects of the video signal

There are several different video signal standards. NTSC is commonly used in the US and Japan, while PAL and SECAM are used in Europe. This is not a problem, since the video capture cards can handle all these standards, and they can to a certain extent even automatically detect which standard is being used. Nevertheless, we as users must be aware of the major differences between the various standards, so that we can make the best selections for the capture parameters. NTSC uses a field rate of 60 Hz, which means that 60 fields (rasters) or 30 full frames ('pictures') are transmitted per second. In Europe, where the standard

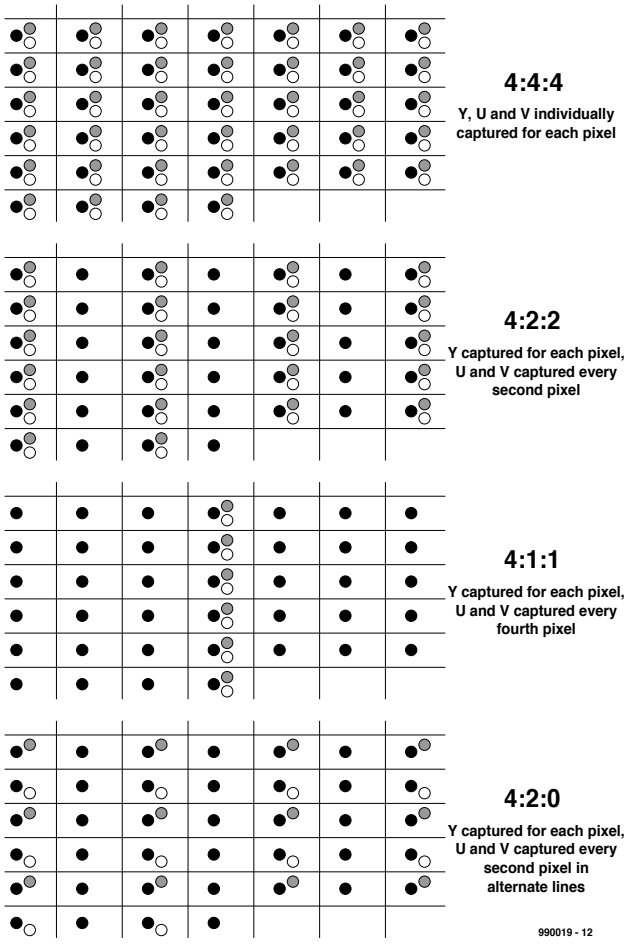
Sampling rates for digital video formats

The compleet video signal, consisting of a luminance component (Y) and two chrominance components, U and V, is stored as three monochrome (black and white) components. When all picture elements (pixels) of these three monochrome images are captured (sampled), the procedure is called 4:4:4 recording (top diagram). Because of the immense amount of data generated in this way, the system is rarely used in practice.

When 4:2:3 recording is applied, the U and V colour components are only sampled every other pixel. This results is a 1/3 reduction in the amount of data required for 4:4:4 recording.

With 4:1:1 recording, every fourth pixel is sampled in the colour channels. The data compression factor with respect to 4:4:4 recording is 50%.

As an alternative to 4:1:1 recording, a system called 4:2:0 is generally used in Europe. Only one colour component is sampled on every second pixel, the sampling being alternate on the U and Y components.



A comparison between different sampling rates. Solid dots represent brightness information (luminance, Y), the open dots and grey dots represent the colour components U and V respectively.

mains voltage frequency is 50 Hz, the rate is 50 fields per second or 25 frames per second. It is thus a good idea to capture NTSC video at 30 frames per second (fps) and PAL or SECAM video at 25 fps.
The picture dimensions are also different: NTSC video converts to bitmap images of 640 ⇔ 480 pixels at full resolution, while PAL and SECAM produce 768 ⇔ 576 pixels at full resolution. The vertical and/or horizontal resolution for data capture can be halved in order to reduce the size of the image data files. For example, if you want to produce computer videos with a resolution of 320 ⇔ 240 pixels, you can capture the video data at half of the full vertical and horizontal resolution without affecting the final image quality. The resulting data files will use only one fourth as much disk space as full-resolution files.

Half frames and the comb effect

In order to keep flickering to an acceptable level, television pictures at 25 or 30 frames per second are transmitted as alternating even and odd half frames at 50 or 60 fields per second. The first field (the even half frame) contains all even-numbered lines of the full frame, and the second field (the odd half frame) contains all odd-numbered lines (although this sequence may be reversed, since it is not standardized). This means that the picture that you see is refreshed 50 or 60 times per second, which significantly reduces the flickering effect. Video capture cards play along with this trick, and in fact capture one half frame every 50th of a second if they are configured for a 25 fps acquisition rate. In exactly the same way, the card generates 50 half frames per second, which are faithfully registered by the video recorder. If you use the system only for editing video clips, everything is in order and you need not be especially concerned about frames and half frames. The only thing that you might notice is that stationary ('halted') pictures from action scenes will flicker quite visibly on the control monitor. This is due to the fact that the video capture card displays both half frames in rapid sequence on the control monitor, even when playback is halted. If the two half frames are not identical, due to rapid movements, there will be visible flickering.
Problems can crop up if you want to capture video images for computer playback. PCs do not have enough processing power to update the monitor display 50 or 60 times per second. They therefore combine successive half frames into single composite frames. This violates the time continuity of the video imagery, since half frames that were recorded at two different times are displayed at the same time. This causes the image to be disrupted if there is rapid motion, This is called the comb effect because of the way it affects the picture (see **Figure 2**).
There is a simple way to avoid the comb effect: capture the video at one half of the full vertical resolution. In this case only every second raster line is captured, which is the same as capturing only every second half frame. The resulting 'reduced' resolution of 288 lines is adequate for computer videos, given the current level of PC technology. This number of lines also corresponds to the MPEG-1 standard.

Data reduction

We have no choice but to reduce the volume of the video data, so that the hard disk can handle the data stream and the file sizes remain manageable. As a first measure, we exploit a characteristic of human visual perception that has long been known from photography and television. We perceive detail at much higher resolution for the monochrome (black-and-white) component of an image than for the colour components. Since a composite colour television signal consists of a brightness component (containing the monochrome imagery) and two colour components (U and V), an obvious approach is to store the colour components at reduced resolution. Our eyes, which are 'half blind' in the colour realm, hardly notice the reduced resolution. **Figure 3** lists commonly-used resolution relationships for video data storage and shows sample images for each one.

For consumer applications, the relationships 4:2:0 (for PAL and SECAM) and 4:1:1 (for NTSC) are generally used. Here the colour components are stored at only one fourth the resolution of the black and white component. This is also considered acceptable in the professional realm, such as television news reporting. For costly advertising productions and other productions that require a certain quality margin, due to extensive digital manipulation of the imagery, a relationship if 4:2:2 is employed.

In addition to this reduction in the resolution of the colour components, the individual images are compressed. This is done by applying a discrete cosine transformation (DCT) followed by quantization. The popular JPEG image format, for example, is based on DCT processing. This technique is based on the fact that in a 'real' image the brightness and colour values vary only slightly in the neighbourhood of any given pixel, except at the edges of the image. This 'low-pass' character of natural images means that we can safely ignore the high-frequency components.

The compression ratio is adjustable with analogue video capture cards, so that it can be matched to the capability of the hard disk drive. The processing speed of the video capture card is also important, since the minimum level of compression that the card can handle determines the maximum quality that is possible for the captured video imagery. Table 1 clarifies the relationship between image quality and the compression ratio.

(990019-1)

WHEN ELECTRONICS WAS YOUNG (4)

The year 1831 was noteworthy for a number of reasons, none of them as yet connected with electronics per se:

1. *Sir David Brewster (1781–1868) publishes his 'Treatise on Optics';*

2. *Independently, Michael Faraday (1791–1867) and Joseph Henry (1797–1878) discover that electricity can be induced by changes in a magnetic field—a discovery leading to the first electric generators;*

3. *Joseph Henry describes a practical electric motor;*

4 *Sir Charles Wheatstone (1802–1875) and William Fothergill (1799–1868) create the first printing telegraph, a machine with an arrow that points to letters of the alphabet;*

5. *The British Association for the Advancement of Science is established*

6 *Charles Darwin (1809–1882) begins his epic five-year voyage on the Beagle*

7 *An otherwise obscure von Jacobi discovered that the Earth may be used as a conductor.*

Von Jacobi's discovery is, of course, of great importance in electrical engineering. Yet, neither his name nor his discovery is mentioned in most modern reference books.

In that year, Faraday argued that, since Oersted had shown that a current could produce a magnetic field, a magnetic field should produce a current. He found this to be so, discovering the important property of electromagnetic induction (earlier discovered by Joseph Henry). In this work, Faraday introduced the idea of lines and fields of force, an idea which was to prove highly productive. It enabled him to devise primitive motors, a transformer, and a dynamo. Faraday also examined capacitors and the properties of dielectrics.

William Faraday

Electromagnetic induction, the conversion of magnetism into electricity, had been discovered in 1830 by Joseph Henry, but Henry had not published his findings. In 1832, he discovered self-induction and this time he published immediately. Consequently, the unit of self-induction is named after him. A coil has a self-inductance of one henry (H) if a change of current through it of one ampere per second produces a back e.m.f. of one volt across it.

Sir Charles Wheatstone who popularized but did not invent the bridge named after him (it was invented by S Christie) developed, together with Cooke, a device with separate control and switching sections. However, the first patent for such a relay (as it came to be known) was taken out by Edward Davy in 1838, although Cooke and Wheatstone's patent was also accepted. Samuel Morse was granted a US patent in 1840 which is apparently similar to Davy's patent.

Samuel Finley Breese Morse (1797–1872) made use of relays to develop the binary (on-off) telegraph system, which he put into practice in 1844 after he obtained a government grant to connect Baltimore and Washington DC. Electrical telecommunications had been borne.

(Sources: Electronic Inventions and Discoveries; Chambers Dictionary of Scientists, Chronology of the Modern World; The Timetables of Science).

[995038]

PCB design and mains voltage

safety and reliability

Designing printed-circuit boards that may be safely connected to the mains supply need not give rise to unnecessary worries. A condition is, however, that the correct rules are adhered to.

1

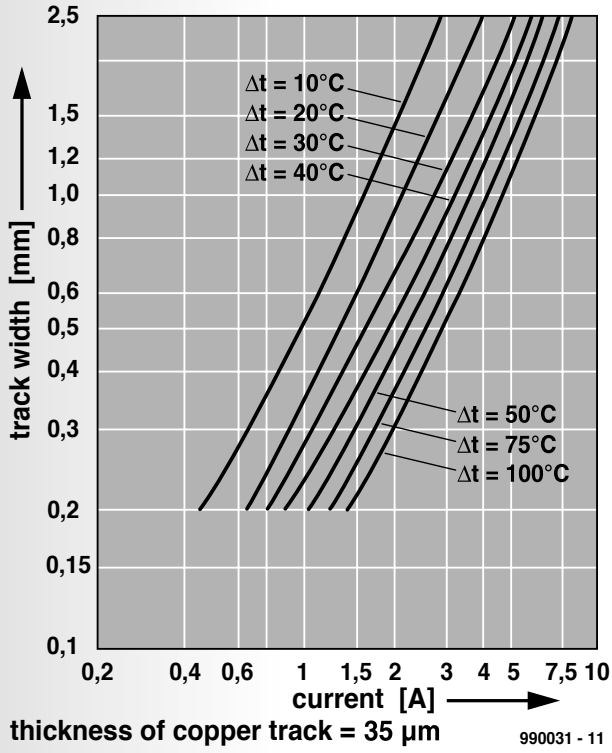


Figure 1. Rise in temperature of a 35 µm thick copper track as a function of the track width and the current level.

When a printed-circuit board is to be linked to the mains supply, so that certain copper tracks carry the mains voltage, a potentially dangerous situation arises if a wrong layout is chosen. Provided that the right rules are obeyed, anyone can design and make a PCB that may be safely connected to the mains supply.

There are two matters that need careful observation: the potential between two adjacent copper tracks, and the current flowing through a track.

Fortunately, the possible uses and applications of board material have been well recorded, so that all relevant data can be found rapidly from a couple of characteristic curves. **Figure 1** shows the rise in temperature of a copper track as a function of the track width and the level of the current flowing through the track. The curves are based on a printed-circuit board with copper tracks 35 μm thick, which is the standard thickness in the case of normal board material.

The temperature rise shown by the curves occurs in an ideal situation in which the board can readily radiate the heat. If ventilation is poor, perhaps because the board is used in an tightly enclosed space, the temperature may rise to a much greater extent. Although standard board material can withstand an appreciable rise in temperature (from ambient), it is sensible to keep it within reasonable limits: 30–40 °C.

WATCH OUT: HIGH VOLTAGE

The distance between adjacent tracks is an important parameter when high voltages will occur on the board. The characteristics in **Figure 2** show the minimum distance for a number of potential differences. The value derived from these curves is the minimum distance anywhere on the board between two adjacent tracks. See also **Figure 3**. At this distance, there is no risk of flashover. Since during the etching of the board tiny irregularities (jagged edges) almost inevitably arise, and later dust particles may settle between the tracks, a good safety margin must be provided.

The curves in Figure 2 show two situations of use of the board: indoor (dashed lines) and out of doors (solid lines). In both cases, it is assumed that the board is used at an altitude not

By K.S.M. Walraven

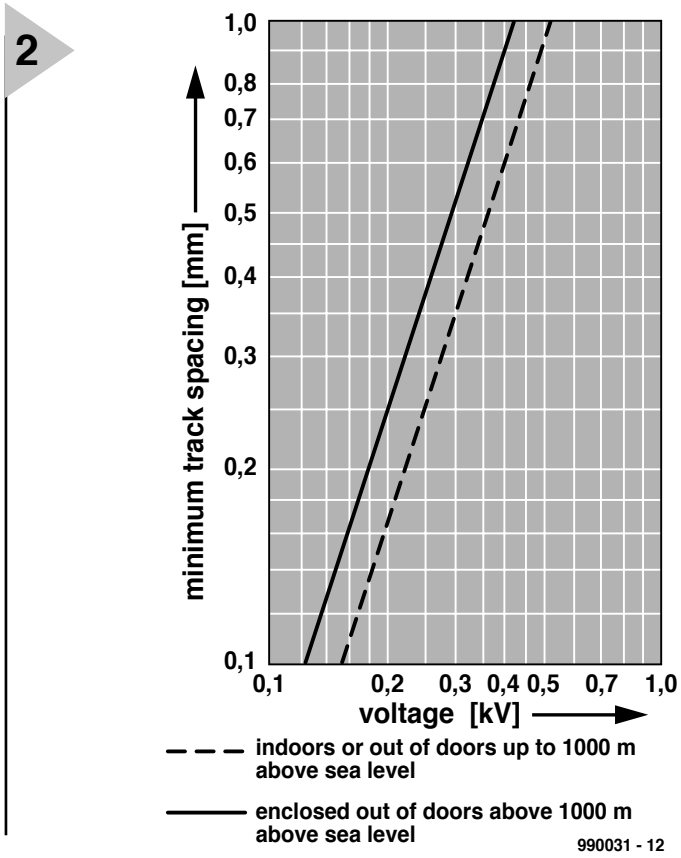


Figure 2. Minimum track spacing as a function of the applied voltage. Note the difference between indoor and outdoor use – see text.

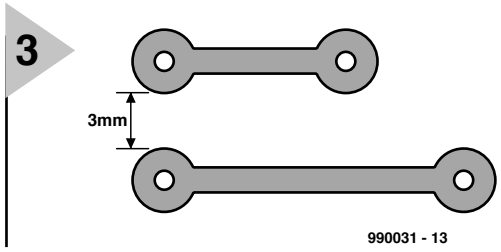


Figure 3. The safe track spacing is the smallest spacing between adjacent tracks anywhere on the board.

exceeding 1000 metres above sea level.

There are furthermore various legal requirements for boards that are to be connected to the mains supply. These include the stipulation that the minimum distance between mains-carrying copper tracks must be at least 3 mm, which is appreciable more than indicated by Figure 2. If the equipment in which the board is used is double-insulated (that is, it is not connected to the supply protective mains earth – which is not advisable in the case of home constructed equipment), the distance between mains-carrying tracks must be not less than 6 mm. This distance is also obligatory between the mains-carrying section and the low-voltage section of the board. Note, by the way, that if a Class II insulated transformer

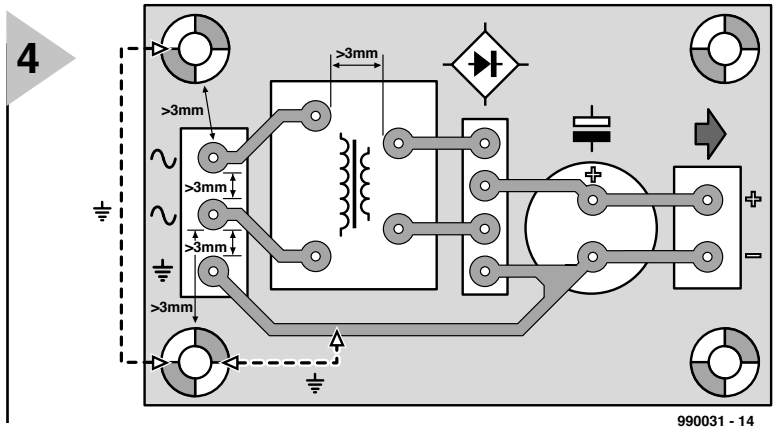


Figure 4. In apparatus with a supply protective earth (which should be the case with virtually all home built equipment) the minimum track spacing is (legally) 3 mm.

is used in a Class I equipment, this does not confer Class II status on the equipment. Also, if an electrically conductive enclosure is used to isolate and protect a hazardous supply voltage from user access, it must be protectively earthed, irrespective of whether the transformer is Class I or Class II.

Figures 4 and **5** give the main points which the designer of the board must observe and adhere to.

The material used does, of course, influence the properties of the board, but fibre-reinforced epoxy is a good practical starting point.

[990031]

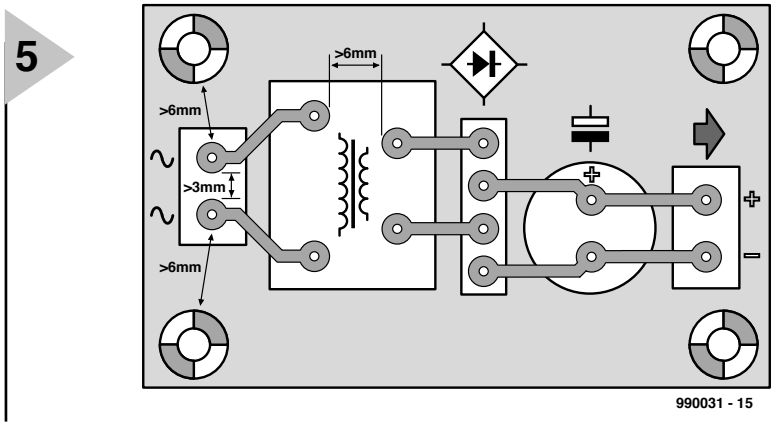
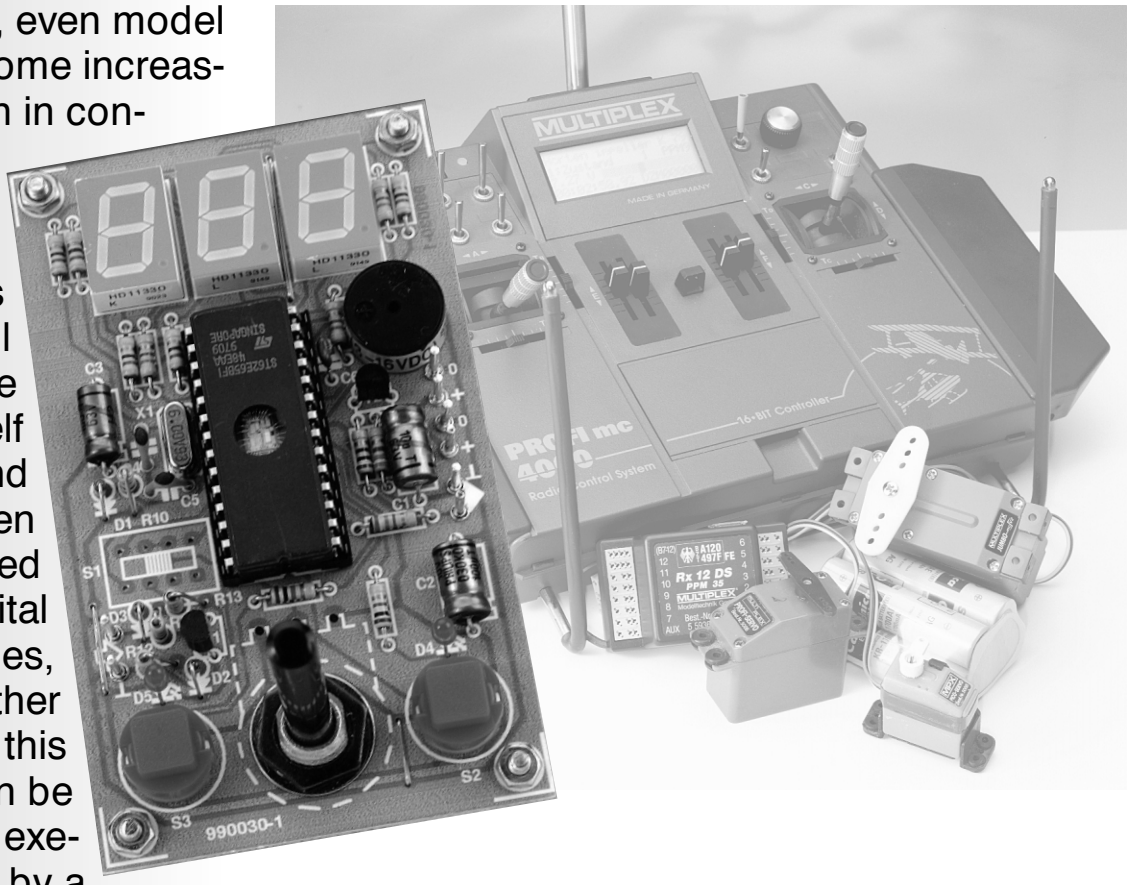


Figure 5. In apparatus without a supply protective earth (rare in electronic equipment – mainly applicable to power tools and some proprietary domestic electrical goods) the minimum track spacing must be (legally) 6 mm.

servo tester

with a built-in pulse generator

Nowadays, even model builders come increasingly often in contact with microcontrollers. Not only is the control of the model itself more and more often implemented using digital techniques, many other tasks in this hobby can be perfectly executed by a microcontroller. In this article, we present an intelligent tester for servo controllers. It generates servo pulses and can also measure pulse parameters.



Design by B. Stuurman

Servo pulses play an important role in controlling the operation of models. The motor speed, direction of travel and other aspects are all controlled by servo pulses. For the uninitiated, a brief explanation of servo pulses is in order. A servo pulse is a digital signal that is generated with a repetition interval of 20 milliseconds. The pulse width ranges from a minimum of 1 ms to a maximum of 2 ms. The 'neutral' width is around 1.5 ms. **Figure 1** shows an individual servo pulse.

The servo converts servo pulses into mechanical motion. If the pulse width is 1 ms, the servo position is at one extreme, while if the pulse width is 2 ms the servo position is at the other extreme.

The servo tester can measure both the width and the repetition interval (period) of a servo pulse. It can also generate pulses itself for testing servos. The tester is designed to produce a minimum pulse width of 0.7 ms and a maximum pulse width of 2.5 ms, so that all types of servos can be thoroughly tested.

THE DESIGN

Figure 2 shows the design of the servo tester. It is based on a ST62T65B IC, a powerful microcontroller made by SGS Thomson. This controller has two full 8-bit ports (PA and PB) and a 5-bit port (PC). The ports have slightly different characteristics. Port B can switch currents up to 20 mA to earth, while port

A can handle currents up to 5 mA in both directions. The hardware design takes these factors into account, so that the individual segments of the display are connected to port A while the cathodes (which carry the heavier currents) are connected to port B.

Pin PC1 of port C is used as the input. This pin is connected directly to a built-in timer. The timer runs if a high level is present on this pin, and it is halted if a low level is present. This hardware coupling guarantees high precision for the measurements. The input circuit, consisting of T1, D2, D3, R11, R12 and R13, acts as an emitter follower that buffers the input signal without changing its polarity. The diodes provide protection against excessive input voltages.

With the chosen clock frequency (6 MHz), the integrated timer counts in steps of 2 μs. This reduces the amount of computation that has to be performed by the controller.

The controller also has a timer with an adjustable interval. The output of this timer, pin PB7, is connected to an output amplifier consisting of T2, R14 and R15. The servo pulses that appear on this output can be used to directly control a servo. This signal is also made available to the controller via switch S1.

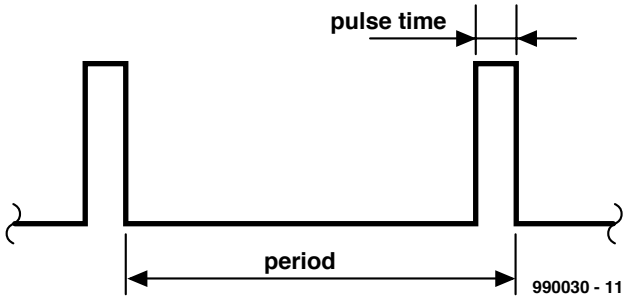
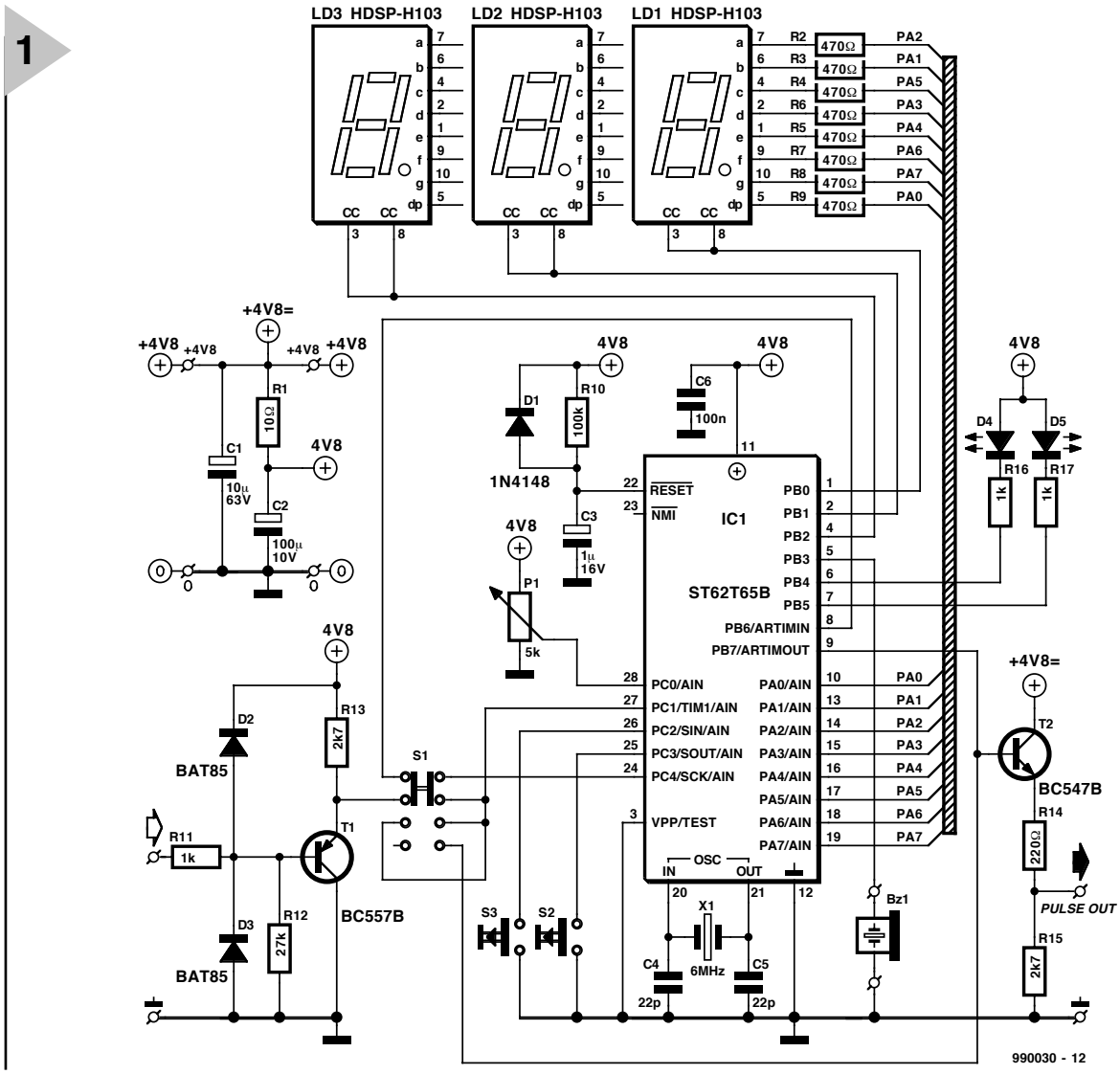


Figure 1. The servo signal is a pulse-width modulated signal with a repetition interval of 20 ms. The pulse width ranges from 1 ms to 2 ms.

The rest of the hardware is formed by switches S2 and S3, LEDs D4 and D5, seven-segment displays LD1 through LD3 and a buzzer. Finally, R10, C3 and D1 form a reset circuit. This ensures that the controller is properly initialized when the power is switched on.

Switch S1 has three positions. Position 1 selects the period measurement function. In this mode, the input of the circuit is connected to pin PB6, while pin PC4 is connected to pin PC1. The software provides the PC4 input with a 2:1 divider, which is advantageous for making period measurements. With this

Figure 2. This circuit can be used for testing servo signals as well as servos.



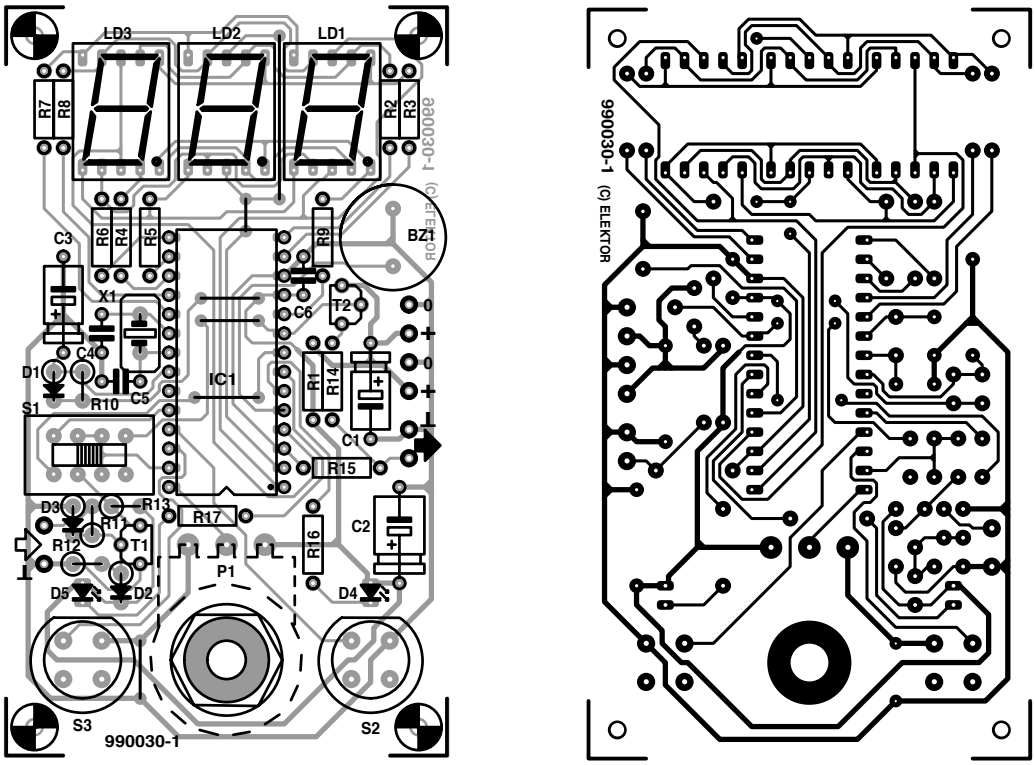


Figure 3. The complete circuit can be built on this ready-made printed circuit board.

COMPONENTS LIST

Resistors:
R1 = 10Ω
R2-R9 = 470Ω
R10 = 100kΩ
R11,R16,R17 = 1kΩ
R12 = 27kΩ
R13,R15 = 2kΩ7
R14 = 220Ω
P1 = 5kΩ potentiometer, linear

Capacitors:
C1 = 10µF 63V
C2 = 100µF 10V
C3 = 1µF 16V
C4,C5 = 22pF ceramic
C6 = 100nF ceramic

Semiconductors:
D1 = 1N4148
D2,D3 = BAT85
D4,D5 = LED, red, high efficiency
T1 = BC557B
T2 = BC547B
IC1 = ST62T65B (order code 996507-1)

Miscellaneous:
X1 = 6 MHz
LD1,LD2,LD3 = HDSP-H103, high efficiency, common cathode, Siemens
S1 = slide switch, 3 positions, 2 poles (Knitter switch type MFP-230)
S2,S3 = pushbutton D6-R-RD (ITC)
Bz1 = passive piezo buzzer type SEP 2242
PCB: order code 990030-1 (see Readers Services pages)
3.5" disk with source code file: order code 996008-1 (see Readers Services pages)

divider, the period can be measured using the same routine that is used for measuring the pulse width.

In position 2, the circuit input is connected to pin PC1. This mode is used for measuring the width of the servo pulse.

Position 3 is also used for pulse width measurement, but in this case pin PB7 is connected to pin PC1 and the controller measures the width of the pulse that it generates itself.

Note that switch S1 is a special type. It has two rows with four contacts each, instead of the usual two rows with three contacts each. In each position, two adjacent contacts are connected to each other. Make sure that you purchase the correct type!

Potentiometer P1 can be used for various adjustments, such as manually setting the pulse width and setting the minimum and maximum pulse widths. The two pushbutton switches S2 and S3 are used to enter settings for the minimum and maximum pulse width, respectively. LEDs D4 and D5 indicate the entry of these values.

The 4.8 V supply voltage for the circuit is simply borrowed from the receiver supply.

PRACTICAL ASPECTS

Regardless of how nice the tester may look on paper, what counts is how it works in practice. To test this, you must first build the circuit. This is made a lot easier by a ready-made printed circuit board, available through the Publishers' Readers Services. **Figure 3** shows the copper track layout and the component layout.

Building the tester should not present any particular difficulties, since the component placements are clearly indicated. Don't forget the three wire links that have to be inserted under the controller socket. It's all too easy to overlook them.

The best way to connect a servo is to use a servo extension cable with one end cut off. You can quickly make a good electrical connection with such a cable. **Figure 4** shows the connection schemes for a number of different types of servos. This should help you to properly connect your own servos to the tester.

AND NOW TO WORK!

Using the tester is remarkably easy. Once you have decided what sort of measurement to make, the actual measurement process requires no more than selecting the correct mode, connecting the signal or the servo to be measured or tested, and reading the test result from the bright LED display.

To measure the period of a servo pulse, put switch S1 in position 1. The measured period will be shown on the LED display in units of milliseconds. The repetition rate of the servo signal can be calculated by dividing 1000 by the measured period in milliseconds.

If the pulse width or period is longer than the maximum value that the tester can measure, 'HHH' is shown on the display. If the input level is low for the duration of the entire measurement, all the bottom segments

of the display are illuminated. If by contrast the input level is high for the duration of the measurement, all the topmost segments are illuminated. Note however that these low and high level indications are only possible for pulse width measurements, due to the use of a 2:1 divider for period measurements.

To measure the width of a servo pulse, put switch S1 in position 2. Connect the input of the tester to the servo output of a receiver. The pulse width will be shown on the display, with a resolution of two digits after the decimal point.

When switch S1 is in position 3, you can read the width of the test servo pulse on the display. This value will change if P1 is adjusted, and the width of the test servo pulse will change accordingly. If a servo is connected to the tester output, it will follow the rotation of potentiometer P1.

Automatic testing of a servo is also possible. To do this, press S2 (Set Low) and S3 (Set High) simultaneously. The servo will now move back and forth between its minimum and maximum positions. You can change the speed at which this occurs by adjusting P1. Briefly press either S2 or S3 to restore normal manual operation.

The minimum and maximum widths of the test pulse can be easily programmed. If you press and hold the Set Low pushbutton (S2), the associated LED will be illuminated. You can read the current setting of the minimum pulse width on the display. Adjust potentiometer P1 until the desired width is displayed, and then release S2. Similarly, the maximum pulse width can be read and if necessary adjusted by means of the Set High pushbutton (S3). The minimum and maximum pulse-width settings are stored in the EEPROM of the controller.

If the selected value for the minimum pulse width is greater than that for the maximum pulse width, the circuit will give an error indication: the buzzer will beep and both LEDs will be illuminated. In this way, you are prevented from selecting an erroneous set of values.

With all of the features that this tester provides, you can use it for quickly checking and testing receivers as well as servos. It is a compact and handy piece of test equipment, as will you will quickly find once you have used it a few times!

(990030-1)

4

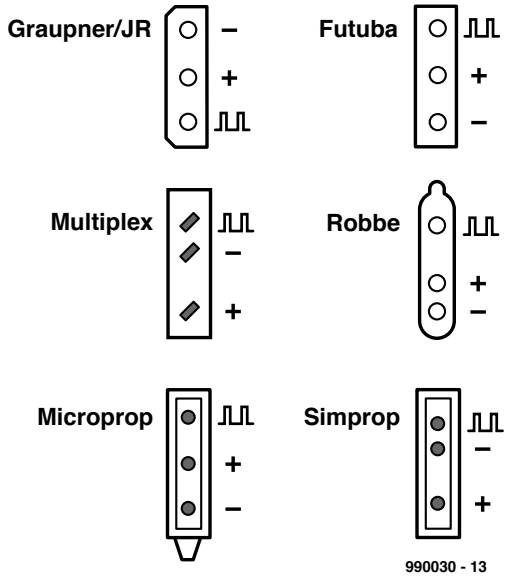


Figure 4. There are several manufacturers of servos, and each of them uses a different type of connector. This summary should help to clarify the situation.

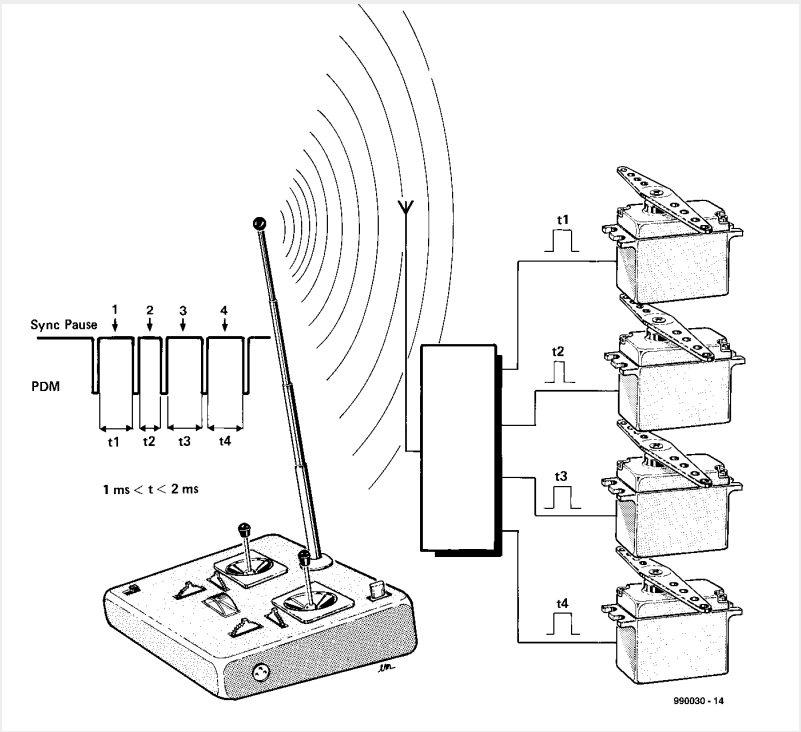
Model control with servos

Every model builder uses servos. They are connected directly to the servo output of a receiver in the model to be controlled, be it a car, a boat or an aeroplane. Several servos are commonly used in a model to control the direction of travel (left/right and up/down) and the speed. The diagram shows how four servos can be remotely controlled via a four-channel transmitter and four-channel receiver.

The actual control takes place via repetitive voltage pulses. These are labelled t1 through t4 in the drawing. The repetition interval for any given pulse is 20 ms.

The width of each pulse is variable, and in practice it ranges from 1 to 2 ms. With proportional control, all intermediate values are possible. Non-proportional (binary) controllers, which are used with inexpensive models, switch between one extreme and the other — which, by the way, does not actually require the use of a servo.

The servo itself contains a miniature electric motor that drives the item to be controlled. In addition, the rotation of the motor changes the position of an internal potentiometer. This potentiometer, which has a resistance of around 5 kΩ, controls a monostable multivibrator that is also housed in the servo. Built-in electronics serve to keep the width of the internally-generated pulse equal to that of the input pulse. Since there is a fixed relationship between the angle of rotation of the potentiometer and the pulse width, the amount of rotation of the servo motor can be directly controlled by the width of the applied pulse.

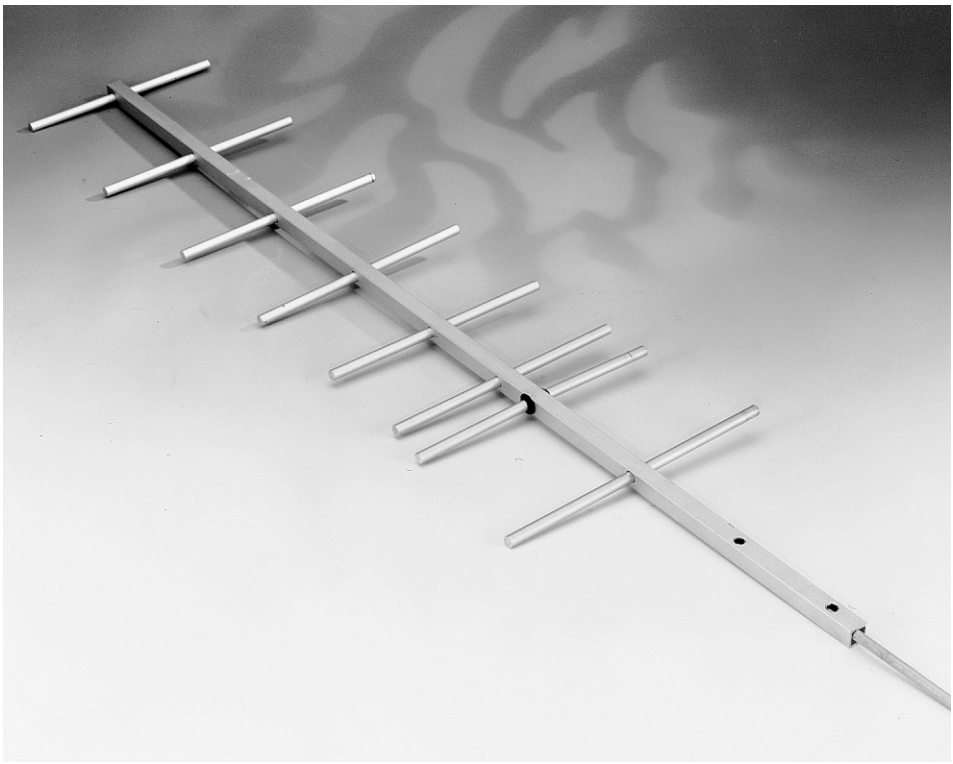


A conventional 'digital-proportional' PWM system. The servo positions are controlled by pulses whose widths can be varied between 1 and 2 ms.

A Yagi aerial for 900-MHz GSM

a fixed antenna for better connections

Although the adverts would have you think that you can use your mobile phone absolutely everywhere, probably every owner of a GSM phone has experienced situations in which the signal quality was hardly worth the name, or the phone simply could not find a transponder. The fixed external antenna presented here might be a help in such situations, since it extends the range of a GSM phone by many kilometres.



You can find them in the Alps, in border regions and in the middle of the countryside — those notorious regions in which even a passable mobile connection is still a pipe dream, and will probably remain so. However, if you mainly use your GSM phone in a fixed location, you can remedy such problems with an external fixed antenna. Such an antenna is the subject of this article. It extends the range of the GSM phone and improves the quality of the data transfer between the phone and your preferred transponder.

A Yagi antenna for UHF

It's no particular problem to select a suitable type of antenna from the many different sorts that are available. First off, it should be compact, robust, inexpensive and easy to build. Next, it should be directional and have a good gain figure, and it only has to work over a narrow frequency band. Given these requirements, and considering that GSM mobile phones use transmission frequencies in the microwave region at around 900 MHz, there is only one choice: a Yagi antenna. Thanks to the relatively short wavelength (λ = 33.33 cm), we can easily construct a 'long' Yagi that promises to have respectable gain. (A Yagi antenna that is longer than around one wavelength is generally said to be 'long'; otherwise it is 'short'.)

A Yagi antenna always consists of three different types of elements. The first of these is the driven element, which is a half wave dipole. Put simply, its job is to convert the electric field between its far ends into a current at its centre (or the other way around, since almost any antenna can be used for both reception and transmission). At the centre of the dipole is the feed point, where the antenna cable is attached. The driven element can be built in various ways. This simplest sort is a straight or 'open' dipole. This has the considerable advantage that the impedance at its feed point is nearly 75 Ω, which corresponds roughly to the antenna impedance of the GSM set (50 Ω). Of course, a Yagi antenna is balanced (symmetrical), and if an (unbalanced) 75-Ω coaxial cable is connected directly to the antenna then half of the antenna is effectively earthed, which reduces its gain. A balanced antenna feeder cable, however, has a typical impedance of 240-300 Ω. To be sure, a folded dipole (consisting of two closely spaced, parallel half-wave sections

Design by R. Gerstendorf

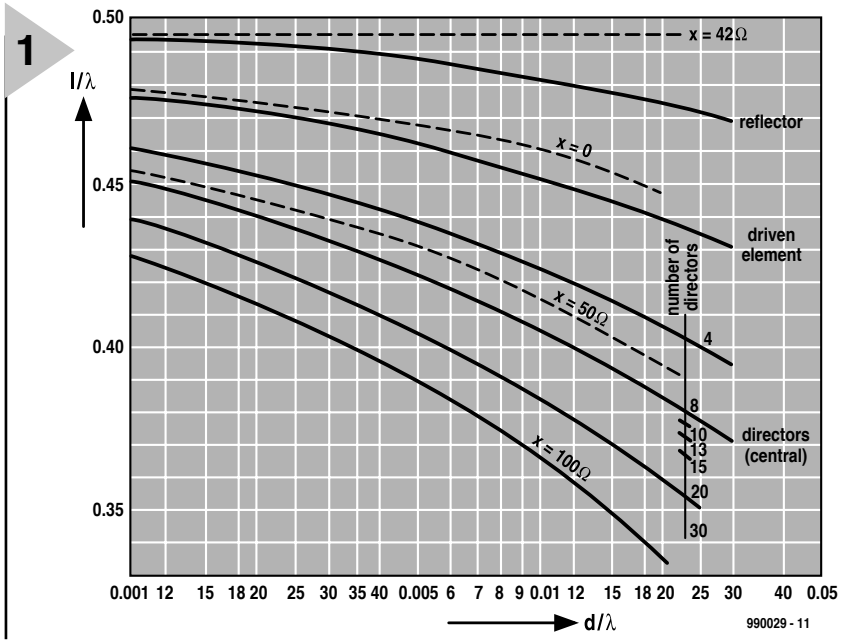


Figure 1. Optimum element lengths as a function of their diameter.

joined at the far ends) could be used for the driven element, and it has a typical feed point impedance of 240 Ω. However, this impedance would still have to be converted to the required value of 75 Ω.

In the first case, all that is needed is a balanced-to-unbalanced converter (a balun), while in the second case both a balun and an impedance converter are necessary. There are many ways to construct a combined balun and impedance converter, but none of them meet one of the most important requirements for the antenna, which that it be mechanically robust. A compact balun, as shown in **Figure 4**, appears by contrast to be ideal for this application: it can placed inside the boom that supports the antenna elements, and it can be easily constructed by winding a 10 to 15 cm length of UHF flat cable (60 Ω impedance; the small difference with 75Ω is not important) into a coil. The way that this balun works is easy to explain. It acts as a choke for unbalanced currents, while allowing balanced currents to pass unhindered. With this balun, an open dipole can be used as the driven element in the GSM Yagi antenna. Note that the balun is not absolutely necessary, but it allows the full gain of the antenna to be realized.

The second type of element is the director. A Yagi antenna has several director elements arranged in a row, in front of and parallel to the driven element. The distance from the reflector (behind the driven element) to the last director element is the length of the antenna, which determines its gain. The distance between the individual director elements is less important, but it should be less than 0.4 λ.

So far, what we have is an antenna consisting of a number of equal-sized elements with constant spacing. This is called a plain Yagi. Numerous experiments have however proven that the gain of a Yagi antenna can be increased if the lengths of the elements and the distances between the elements are not constant. Such an antenna is called a tapered Yagi. The gain of a tapered Yagi lies 1 to 2 dB higher than that of an equivalent plain Yagi, so that nowadays only tapered Yagis are used. It is very difficult to compute the optimum values for the element lengths and spacings, so most such antennas are designed from charts and tables.

The third and last type of antenna element is the reflector. This is placed behind the driven element (opposite the director elements) to suppress the radiation or reception of signals in the backward direction. It reflects the signals to the front and thereby increases the gain.

TAKING STOCK

Before we can determine the optimum values for the element lengths and locations, we must decide on the length of the antenna. The antenna described here has six director elements and thus a length of 1.54 λ (51.3 cm), according to **Table 1**. The theoretical maximum gain of such an antenna is around 11.5 dB. A longer antenna would of course be possible, but the gain increases only relatively slowly with increasing length, reaching roughly 16 dB at 7 λ (Table 1 simply stops at this point). The middle column of Table 1 shows the spacing of the successive elements. You can see that the element spacing increases with the distance away from the driven element, but it never exceeds 0.4 λ. Table 2 lists the full specifications of the 900 MHz GSM antenna, including the lengths and spacings of all the elements. The optimum element spacings, taken from Table 1, are shown in the second column.

To determine the optimum lengths of the elements, we have to use the chart shown in **Figure 1**. The cross-sectional area is a factor here, since the larger this is the shorter the elements must be. The elements become progressively shorter from the reflector to the final director. The chart shown in **Figure 2** indicates the proper lengths of the director elements according to their positions with respect to the driven element. For the prototype antenna, I used solid aluminium round stock with a diameter of 6 mm (0.018 λ). The optimum element lengths for this diameter are listed in the third column of **Table 2**.

Up to now I haven't said anything about the antenna boom, which supports the elements of the antenna. It has a marginal influence on the dimensions of the elements. Its cross-sectional shape is not important. The elements can be attached to the boom in various ways, either insulated from the boom or in electrical contact with the boom. If the elements are mounted such that they are separated from the boom by more than half of the diameter (or width) of the boom, the influence of the boom is negligible, regardless of whether the mounting is insulated or conductive. However, if the elements pass through the boom and are electrically attached to the boom, as is the case with our antenna, then they must be made somewhat longer. The correc-

Table 1. Optimum element spacings

Element	*Spacing in λ*	*Antenna length in λ*
Reflector	***0.240***	
Dipole		
1. Director	*0.075*	
2. Director	*0.180*	
3. Director	*0.215*	
4. Director	*0.250*	
5. Director	*0.280*	
6. Director	*0.300*	*1.54*
7. Director	*0.315*	*1.88*
8. Director	*0.330*	*2.19*
9. Director	*0.345*	*2.53*
10. Director	*0.360*	*2.89*
11. Director	*0.375*	*3.27*
12. Director	*0.385*	*3.65*
13. Director	*0.390*	*4.05*
14. Director	*0.395*	*4.44*
15. Director	*0.400*	*4.84*
16. Director	*0.400*	*5.24*
17. Director	*0.400*	*5.64*
18. Director	*0.400*	*6.04*
19. Director	*0.400*	*6.44*
20. Director	*0.400*	*6.84*

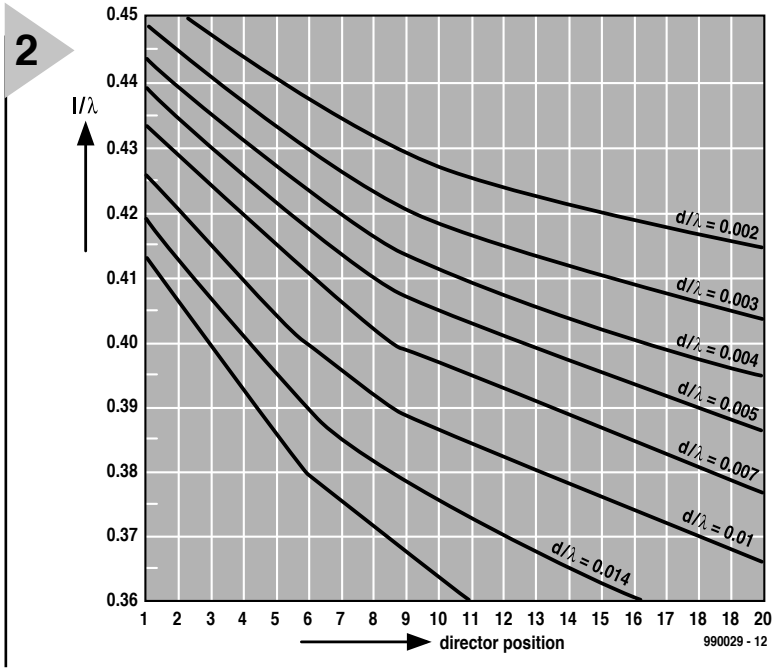


Figure 2. The lengths of the director elements depend on how they are mounted.

tion factor is shown in **Table 3** as a function of the width of the boom. It is easy to understand why this correction is necessary, since the conductive boom reduces the effective electrical length of the elements. This means however that it is important to ensure reliable and durable electrical contact between the boom and the elements!

To construct the antenna boom, I used a length of aluminium square box-section stock with an edge dimension (width) of 10 mm (0.03 (). This and the round stock for the elements are available in every DIY home improvement shop. By the way, it doesn't particularly matter whether you use anodized or plain aluminium. This is because a bare aluminium surface quickly oxidizes when it is exposed to the elements, forming an oxide layer that protects the underlying metal from further weathering.

In any case, the fourth column in Table 2 lists the final lengths of the elements after the addition of a correction factor of 5.3 mm. Finally, **Figure 3** shows a dimensioned drawing of the full 900 MHz GSM antenna, together with a number of construction details.

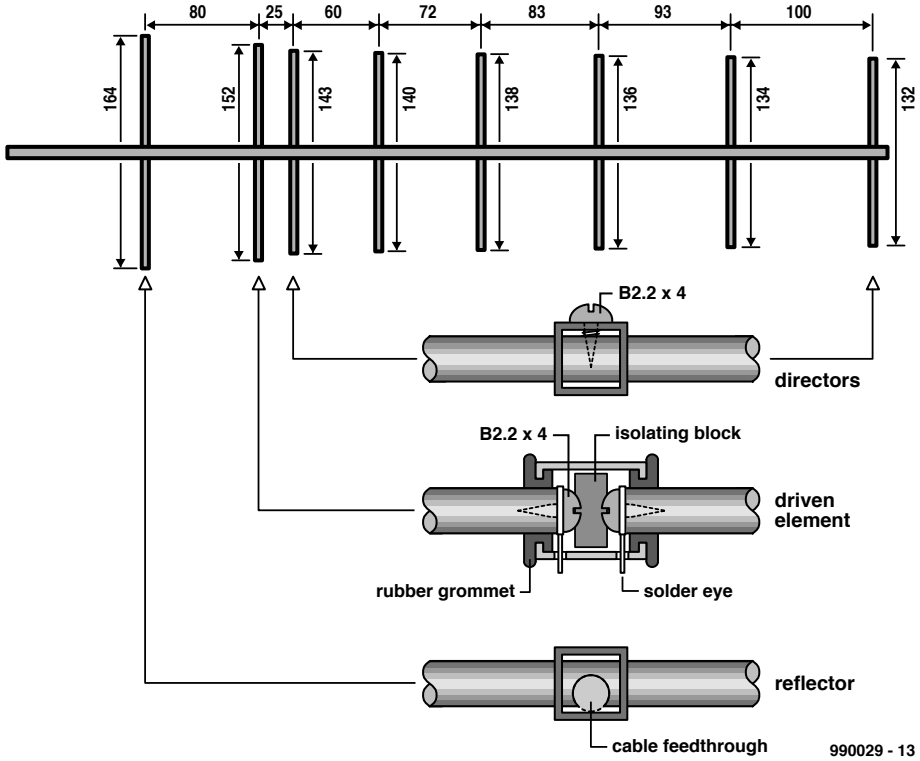
Figure 3. The dimensions of the 900 MHz GSM Yagi antenna, with construction details.

CONSTRUCTION

Now for some construction hints. The boom should be mounted by attaching it to a mast or bracket behind the reflector element. Thanks to the light weight and low wind load of the antenna, this can be easily done. Other mounting arrangements might possibly disrupt the operation of the antenna. For this reason, you should be sure to make the boom long enough to allow for mounting the antenna.

To attach the director elements, first carefully measure and mark their positions along the boom, centre punch the marked locations and then drill 6-mm diameter holes through the boom at each location. In addition, drill a small hole in the lateral face of the boom (parallel to the element axis)at each location. This hole should be just large enough to pass a B2.2 × 4 sheet-metal screw (see the top detail in **Figure 3**). Later on, you can file each of these holes into an oval shape along the element axis, to allow the elements to be precisely centred.

Now insert each of the director elements into the boom and precisely mark the middle point of each element, using the point of a sheet metal screw pressed into the previously described lateral holes. Remove the director elements and drill 2-mm diam-



eter holes through them at the marked middle points. These will be used later for the fastening screws.

A 6 mm hole must also be drilled in the boom for the reflector element. However, instead of drilling the middle of the reflector element for a mounting screw, you must file a round notch in it that is just big enough to pass the antenna cable (see the bottom detail in **Figure 3**). Before doing this, check that the reflector element passes readily through its mounting hole in the boom. There is not a lot of material left after the notch is filed, and the reflector element can easily break if too much force is applied to it. If the fit with the mounting hole is too loose, you can screw a small sheet-metal screw into the gap between the element and the boom to secure it and provide a good electrical contact. Do this only after the antenna is fully assembled.

Fitting the driven element requires a certain amount of manual dexterity. This element requires 8-mm diameter mounting holes in the boom. Cut the element exactly through the middle, file the cut ends flat and drill 2-mm diameter holes into the ends. Make these holes deep enough to receive fixing screws for a set of solder lugs (see the middle detail in **Figure 3**).

The driven element is insulated from the boom by a pair of rubber grommets having an inner diameter 6 mm and an outer diameter 8 mm. If necessary, trim the grommets with a knife to obtain a snug fit between the driven element and the boom. It should not be necessary to force the element sections into the boom.

Now it's time to prepare the balun and the coaxial antenna cable (RG58U). First, try to work the end of the coaxial cable past the reflector element and up to the mounting hole for the driven element. Once you have managed this, pull the end of the cable through one of the mounting openings and strip off about 3 cm of the outer insulation. Then remove about 1 cm of insulation from the inner conductor. Twist each of the conductors into a pigtail, and then cover and seal the end of the outer insulation with a length of heat-shrink tubing. Solder the leads of the cable to the leads of one end of the tightly coiled balun cable, and insulate these joints with heat-shrink tubing as well. Separate the leads of the balun cable at the other end for a few centimetres, strip the ends and solder them to the solder lugs. Cover the joints with heat-shrink tubing, so that only the rings of the solder lugs remain exposed.

Now carefully push the balun into the boom, and pass one of its leads through the opposite opening. Insert the previously prepared grommets into the openings, and fasten the solder lugs to the ends of the driven element sections. Finally, insert the two halves of the driven element into the boom — but before doing this, you must prepare an insulator that sits between the two ends of the driven element. You should use a 2 mm thick Teflon disk; this can best be made using a piece of the inner insulation from a 'thick' HF transmitter cable like RG213.

Once the driven element is mounted in the boom, measure its overall length and then reduce it to the correct length (which can be somewhat tedious). After you have mounted and centred the director elements and attached a connector to the far end of the antenna cable, you can attach the antenna to its mount. The GSM Yagi antenna is now ready for use. It works very well with a GSM phone as a base station for a private or mixed DECT telephone installation.

(990029-1)

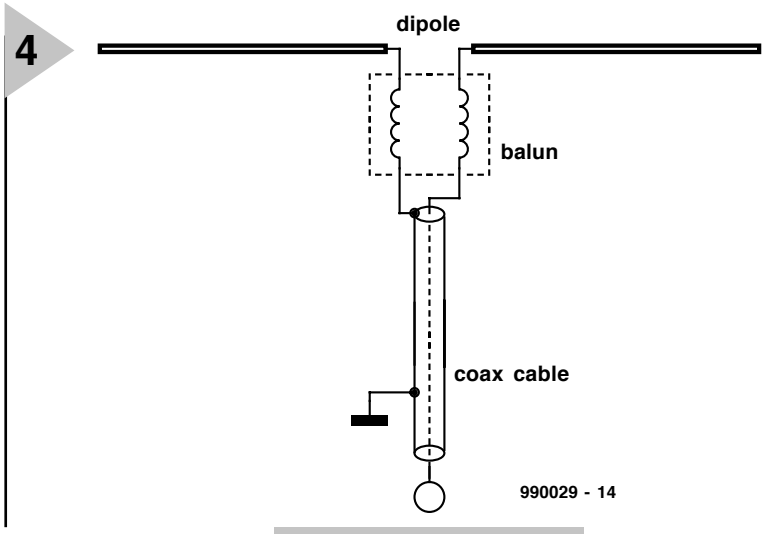


Figure 4. Balun to match the cable impedance.

Reference:

Karl Rothammel Y21BK, *Antennenbuch*, Telekosmos-Verlag, Frankh'sche Verlagshandlung
Stuttgart 1984, Germany

Table 2. Calculation of Yagi antenna parts

Basics		
Frequency	*900 MHz*	
Wavelength λ	*33.33 cm*	
Antenna length	*1.54 λ*	
	51.3 cm	
Element diameter	*0.018 λ*	
	6 mm	
Boom side length	*0.03 λ*	
	10 mm	

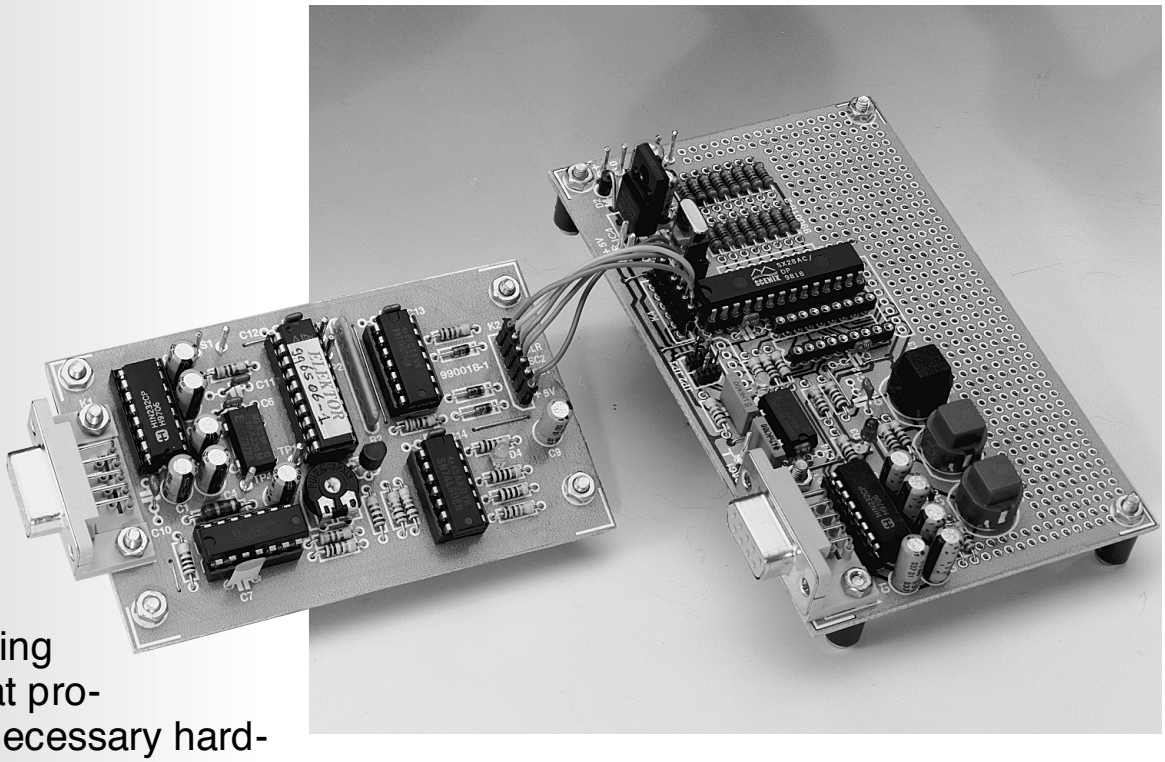
Element	***Distance (Table 1)***		***Length (Figs. 1/2)***		***Length (Fig. 3)***
Reflector	*0.240 λ*	*80 mm*	*0.476 λ*	*159 mm*	*164 mm*
Dipole			*0.441 λ*	*147 mm*	*152 mm*
1. Director	*0.075 λ*	*25 mm*	*0.412 λ*	*137 mm*	*143 mm*
2. Director	*0.180 λ*	*60 mm*	*0.405 λ*	*135 mm*	*140 mm*
3. Director	*0.215 λ*	*72 mm*	*0.398 λ*	*133 mm*	*138 mm*
4. Director	*0.250 λ*	*83 mm*	*0.392 λ*	*131 mm*	*136 mm*
5. Director	*0.280 λ*	*93 mm*	*0.386 λ*	*129 mm*	*134 mm*
6. Director	*0.300 λ*	*100 mm*	*0.380 λ*	*127 mm*	*132 mm*

Table 3. Element length correction due to insertion in boom

D in λ	***Correction Δl in λ***
0.010	*+ 0.003*
0.015	*+ 0.005*
0.020	*+ 0.008*
0.025	*+ 0.010*
0.030	*+ 0.016*
0.040	*+ 0.026*
0.050	*+ 0.038*

SX-microcontroller evaluation system (3)

part 3: the SX prototyping board



This month we present the SX prototyping board that provides the necessary hardware and a prototyping area for your own experiments. Add the PICKALOCK programming adaptor and you are able to use a PC to load programs into the SX microcontroller. The prototyping board is suitable for 18 as well as 28-pin versions of the Scenix SX microcontroller.

Design by M. Ohsmann

The prototyping board was designed to help you learn about practical SX programming and system development. The circuit diagram is shown in **Figure 1**. Originally, separate boards were designed for the two versions of the SX controller (SX18 and SX28). However, as the design was gradually finished, it was decided to make the system even more versatile by combining the two boards.

OSCILLATOR HARDWARE

The prototyping board offers three options for the oscillator configuration. Normally, jumper **JP1** is fitted, enabling the internal oscillator and its satellite parts X1, R7, C8 and C9 to act as a 16-MHz clock generator. If, however, jumper **JP2** is fitted, the SX chip is clocked at 48 MHz by integrated clock oscillator IC3. These two frequencies were selected for the demonstration programs. For your own experiments, it is, of course, possible to use other quartz crystals or external oscillator modules. If jumper **JP3** is fitted, the R-C network R6-C7 is connected to the OSC1 oscillator pin. In this configuration, the RC oscillator option of the SX microcontroller is employed.

ISP ADAPTOR

With jumper **JP1** installed, the SX micro

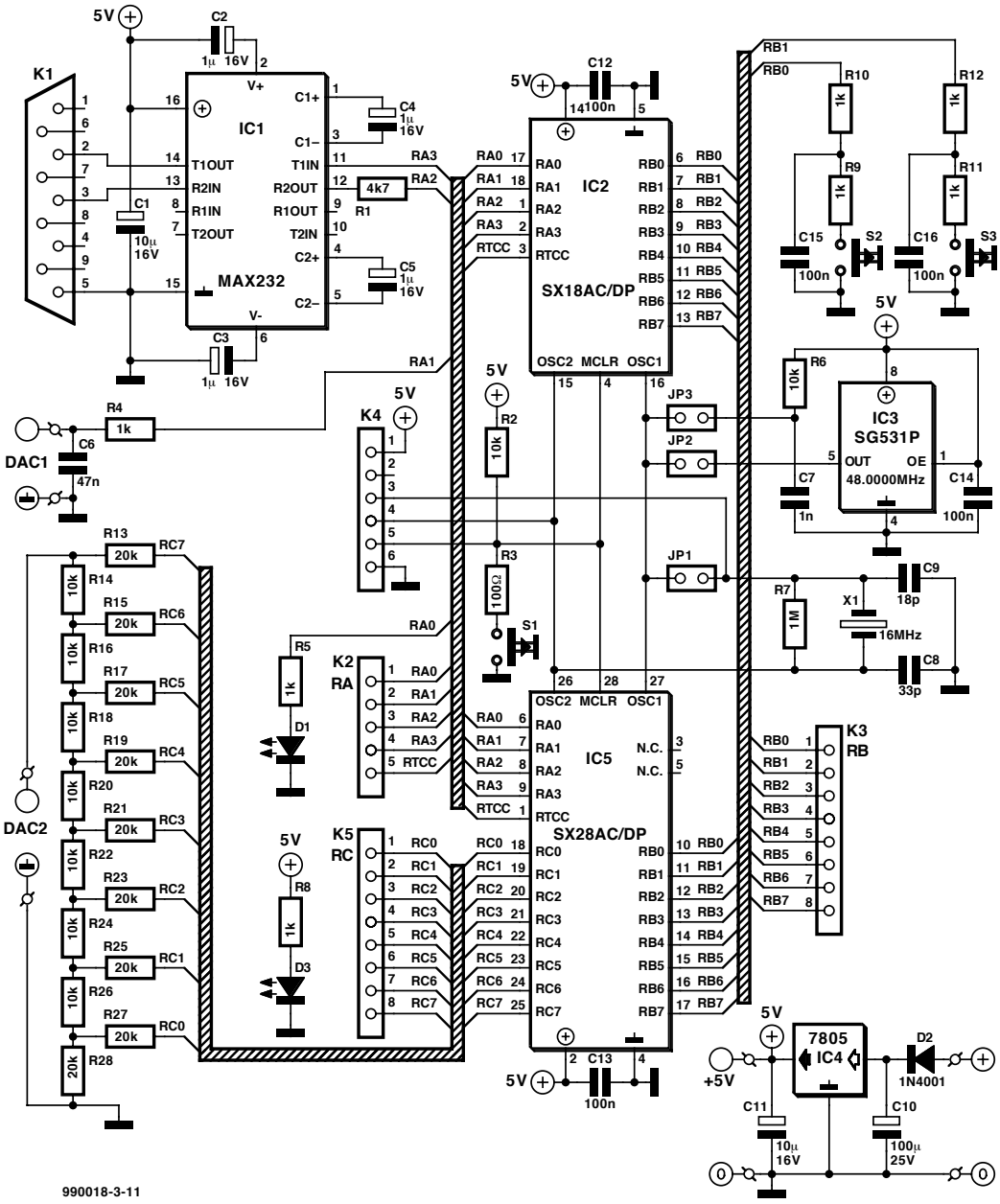


Figure 1. Circuit diagram of the SX prototyping board.

may be connected to the SX-PICKALOCK programmer via the ISP connector, K4. In this configuration it is possible to re-program the SX chip 'in-circuit'. The connecting cable should be not be too long to ensure that the 16-MHz oscillator works reliably with the programming adaptor connected-up.

SIMPLE I/O

Port RA provides a simple input/output interface on the prototyping board. An LED is connected to port line RA.0, and an RC network to RA.1. The latter enables analogue voltages to be generated through pulsewidth modulation (PWM). Port lines RA.2 and RA.3 are linked to the MAX232 RS232 interface circuit (IC1), enabling your programs to communicate with a PC (or a dumb terminal) via a serial cable connected to K1. A simple D-A converter is connected to port RC of the 28-pin version of the SX micro. The DAC enables fast analogue signals to be generated under software control. Push-buttons S2 and S3 may be used as rudimentary input devices. They are connected to port RB to enable you to experiment with the possibilities of 'wake up on interrupt'. Because all processor I/O pins are accessible via pinheaders, they are easily connected to extension circuits built on the prototyping area of the board. A reset switch is also available: S1.

The board is powered by a 9-volt supply. The supply voltage connected to the prototyping board is available at a pin on K4, enabling a programming adaptor to be powered also.

CONSTRUCTION

The prototyping board is a single-sided type whose copper track layout and component mounting plan are given in **Figure 2**. This board is available ready-made through the Publishers' Readers Services as well as through kit suppliers supporting Elektor projects. Although populating the board is not particularly difficult, the component placement and soldering should be carried out with great care and precision. Also, be sure to fit all wire links as indicated on the component overlay. Note that K1 is a 9-way sub-D **socket** (i.e., female connector). We recommend using IC sockets with turned pins for the SX microcontrollers (positions IC2 and IC5) because these will typically be fitted and removed quite a few times. The two SX micros **should never be on the board at the same time!**

Once the hardware is ready for use (see **Figure 3**), you may start thinking about the system software. However, before 'assembling' your first program

2

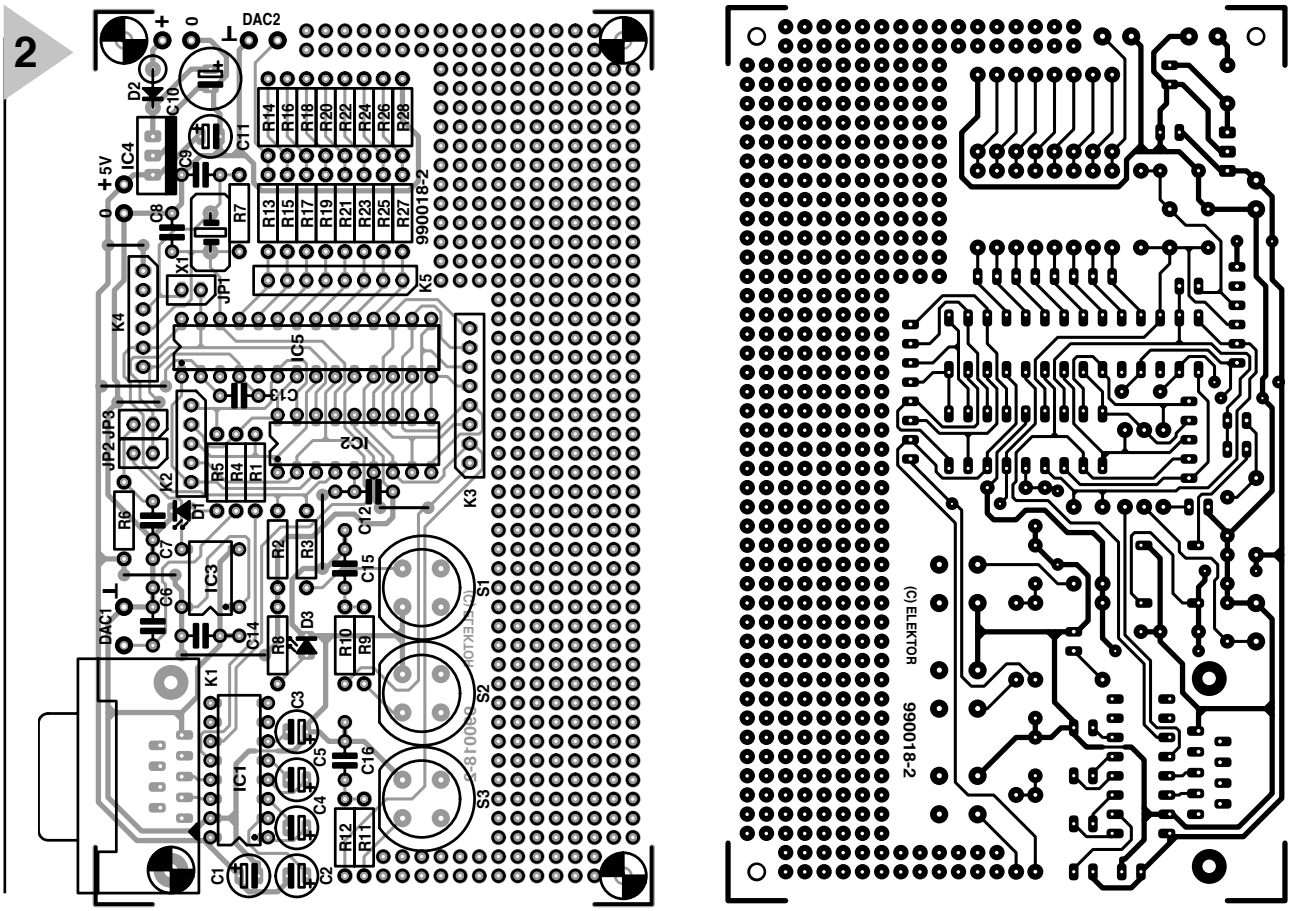
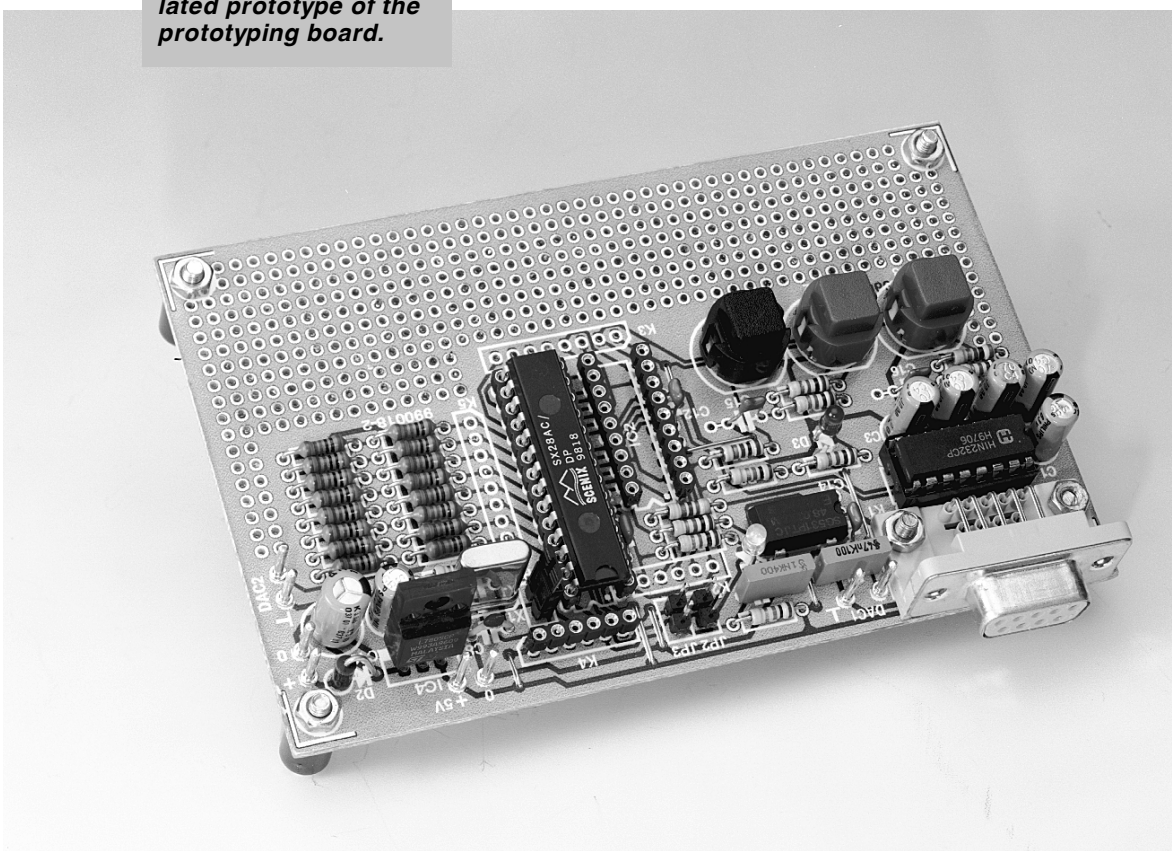


Figure 2. PCB copper track layout and component overlay (board available ready-made).

Figure 3. Ready populated prototype of the prototyping board.

3



COMPONENTS LIST

Resistors:
R1 = 4kΩ7
R2,R6 = 10kΩ
R3 = 100Ω
R4,R5,R8-R12 = 1kΩ
R7 = 1MΩ
R14,R16,R18,R20,R22,R24,R26 = 10kΩ 1%
R13R13,R15,R17,R19,R21,R23,R25,R 27,R28 = 20kΩ 1%

Capacitors:
C1,C11 = 10μF 16V radial
C2-C5 = 1μF 16V radial
C6 = 47nF MKT
C7 = 1nF MKT
C8 = 33pF ceramic
C9 = 18pF ceramic
C10 = 100μF 25V radial
C12-C16 = 100nF ceramic

Semiconductors:
D1,D3 = LED, high efficiency
D2 = 1N4001
IC1 = MAX232
IC2 = SX18AC/DP*
IC3 = SG531P 48.0000MHz (Epson) (Eurodis, Bolton)
IC4 = 7805
IC5 = SX28AC/DP*

Miscellaneous:
JP1,JP2,JP3 = 2-way pinheader w. jumper
K1 = 9-way sub-D socket (female), angled pins, PCB mount
K2 = 5-way SIL pinheader
S1,S2,S3 = pushbutton D6R-RD (ITC) (Eurodis, Bolton)
K3,K5 = 9-way SIL pinheader
X1 = 16MHz quartz crystal
K4 = 6-way SIL pinheader
1 off 18-way IC socket with turned pins (for IC2)
1 off 28-way IC socket with turned pins, 0.3-in pin distance (for IC5)
PCB, order code 990018-2 (see Readers Services page in March 1999 issue)

on the PC, and actually program the first SX micro, it is useful to discuss the processor and the control software in some more detail.

SXASM ASSEMBLER AND SX INSTRUCTION SET

The SXASM assembler was developed to enable you to write your own programs for the SX micro. It is supplied on a 3.5-inch disk supplied for this project through our Readers Services. Because the disk also contains the assembler source code file, it should be possible to write your own extensions for the assembler. Admittedly, the assembler is not as powerful as the one supplied by Parallax, but that need not be a problem for everyday use. For example, SXASM does not support local macros or labels. However, the standard macro instructions of the SX micro are available so that problems will rarely occur. A description of the assembler's features may also be found on the project diskette (see SXASM.DOC).

The instruction set is largely similar to that of the Microchip PIC microcontroller. Like the PIC, the SX controller requires you to take a hard look at the system of 'memory banks' in the RAM range, and the 'page' structure with jumps and subroutines. Because the SX is a RISC processor, many operations require the use of two instructions. Although similar operations use just one instruction on an 8051, the much higher clock rate of the SX ensures a very high processing speed.

YOUR FIRST PROGRAM

An example of an SX program as created with the aid of the SXASM assembler is shown in **Figure 4**. As you can see, the approach is traditional. Lines consist of an (optional) label, followed by an opcode, an operand and, (optional) comment. Opcodes and operands are case-insensitive, i.e., they may be written in upper or lower case. Our very first program does nothing sensational except perhaps flash LED D1 on the prototyping board! The first program is always the most difficult because you will typically encounter lots of pitfalls. To keep 'clerical' errors like typos and syntax errors to a minimum, the diskette contains a number of tested example programs.

STEP BY STEP

Figure 5 illustrates the connections between the modules that make up the SX evaluation system. Start by reading the sequence described below — practical points will be discussed further on.

The source code text of the program (LED1.SRC) is assembled using the SX assembler (SXASM). This is done by typing

SXASM LED1 <return>

Figure 4. Try this program to get the feel: a LED flasher for the SX micro.

4

```
****** LISTING of SXASM (LED1) ******
LVL LINE LOC  OBJ              SOURCE
(0)    1 0000                  device turbo,pins28
(0)    2 0000          divider equ    200  ; repetition of outer loop
(0)    3 0000          ra      equ    5    ; define rort A
(0)    4 0000
(0)    5 0000                  org    8    ; start of variable space in RAM
(0)    6 0008          cnt0    ds     1    ; define 3 counters as bytes
(0)    7 0009          cnt1    ds     1
(0)    8 000A          cnt2    ds     1
(0)    9 000B
(0)   10 000B                  org    0     ; program memory starts at 0
(0)   11 0000                  reset  start ; reset vector jumps to start
(0)   12 0000
(0)   13 0000 C00 005  start   mov    !ra,#0; set port A to outputs
(0)   14 0002 907      loop    call   time  ; call delay subroutine
(0)   15 0003 505              setb   ra.0  ; turn on LED
(0)   16 0004 907              call   time  ; call delay again
(0)   17 0005 405              clrb   ra.0  ; turn off LED
(0)   18 0006 A02              jmp    loop  ; start all over
(0)   19 0007
(0)   20 0007 CC8 02A  time    mov    cnt2,#divider ; how often outer loop
(0)   21 0009 C1E 029  lp3     mov    cnt1,#30  ; next level 30 times
(0)   22 000B 068      lp2     clr    cnt0      ; inner level 256 times
(0)   23 000C 2E8 A0C  lp1     djnz   cnt0,lp1  ; inner loop
(0)   24 000E 2E9 A0B          djnz   cnt1,lp2  ; next level loop
(0)   25 0010 2EA A09          djnz   cnt2,lp3  ; outer loop:count=divider
(0)   26 0012 00C              ret
(0)   27 0013                  end
FUSE  = 07FA
FUSEX = 0F7E
RESET = 0000
********** SYMBOLTABLE (11 symbols) **********
   divider :00C8        ra :0005      cnt0 :0008      cnt1 :0009
      cnt2 :000A     start :0000      loop :0002      time :0007
       lp3 :0009       lp2 :000B       lp1 :000C
 (27 lines, 0 errors, 19 bytes)
```

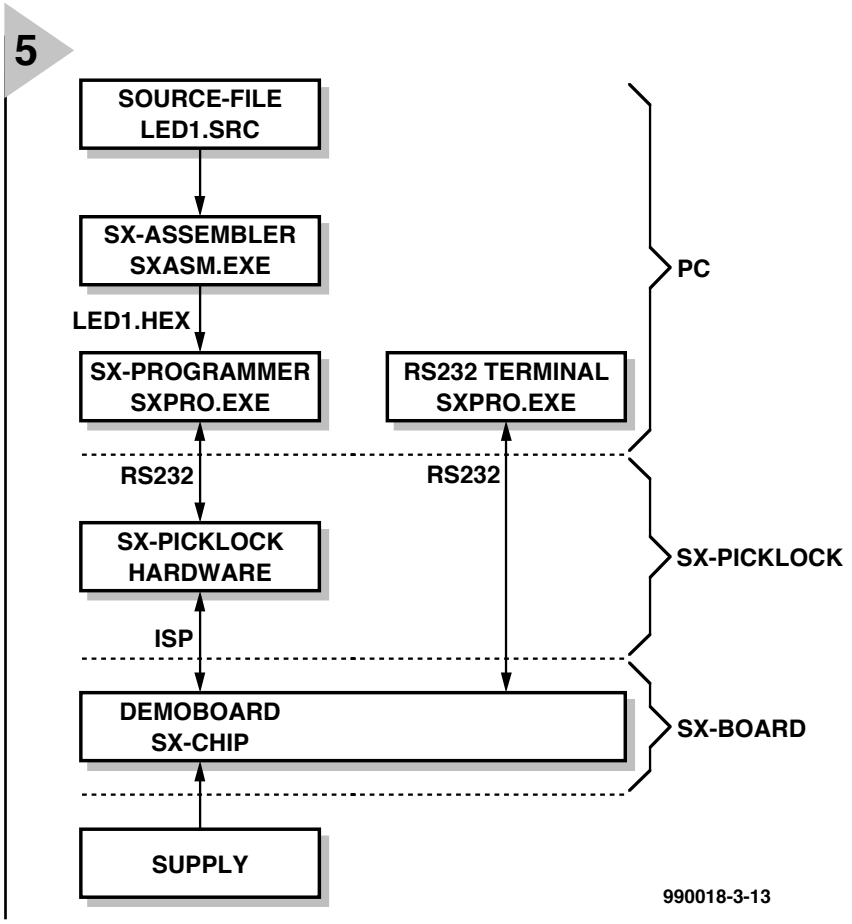


Figure 5. Block diagram of the modules that make up the SX Evaluation System.

This creates the hexadecimal file 'LED1.HEX' and the list file 'LED1.LST'. Next, activate the PC part, SXPRO.EXE, of your programming adaptor by typing

SXPRO LED1 < return>

This causes the SXPRO program to load the indicated HEX file and start communicating with the SX PICKALOCK programming adaptor via the RS232 link on the PC. The SX PICKALOCK hardware is connected to the prototyping board by way of the ISP (in-system programming) interface implemented on connector K1. Having connected all this hardware, you may switch on or plug in the 9-V mains adaptor. The prototyping board is then powered, and the SX microcontroller will start to operate. At the same time, however, the SX PICKALOCK is activated because it is also powered (via K4). At this point, SXPRO is allowed to send the SX program to the prototyping board by way of the ISP. This only requires typing

P < return>

in SXPRO (P for Program). Doing so causes the file 'LED1.HEX' to be copied into the SX microcontroller. This program is executed after a reset, i.e., after you press S1.

So much for a general overview of the programming sequence, the next thing to do actually performing these steps. However, we should first take a look at the listing as shown in Figure 4.

OPTIONS AND INITIALISATION

The Scenix SX microcontroller has a number of options that have to be included when programming the FUSE and FUSEX registers. The configuration of these bits is implemented by the 'device' statement in line 1 of the program. In this example, the SX chip is switched to 'turbo' mode, which causes each instruction to be executed with a minimum of clock cycles. We also 'declare' that the 28-pin version is used. As to the oscillator type selection, no specification is required because a fast quartz crystal is employed. The program is stored on disk under the name LED1.SRC. It is first assembled by typing

SXASM LED1 < return>

Only then are you allowed to connect the PICKALOCK programming adaptor to the PC and the prototyping board, and apply power to the board. Next, launch the SXPRO program on the PC by typing

SXPRO LED1 < return>

The option 'P' in this program then allows you to actually load the SX chip with the designated program (attention: jumper JP1 has to be installed). Next, press the reset pushbutton on the board, S1, whereupon the LED should start to flash! If this works, the hardware may safely be assumed to work as specified, and you can start programming! Just in case the LED does not flash, inspect the hardware to locate any faults.

MACRO INSTRUCTIONS AND CLOCK CYCLES

Each simple instruction the SX micro is capable of executing has the size of a 'word' (12 bits). However, a close look at the listing in **Figure 4** reveals that some instructions generate two words (for example, the one on line 13). These are macro instructions. Introduced by Parallax, macro instructions provide a kind of shorthand notation similar to that used with PIC processors. Be careful, because many macro commands 'invisibly' modify the W register or the flags. Some of these were even modified by the CARRYX option. If you want to know all the details, study the SX documentation in great detail, there is no other way! In addition to the documentation supplied by Parallax and Scenix, the project diskette (order code **996007-1**) also provides useful information. It should also be noted that the macro instructions require more than one cycle. The same goes for jumps and subroutines, however all this should be familiar from other RISC processors. Consequently, a speed comparison between the SX processor and other processors is not that simple. In the example program discussed above, three 8-bit variables are declared (set up) in RAM in lines 5 to 9, next, port RA (Register RA) is programmed as an output in line 13.

Next comes a loop in which port pin RA.0 is set and cleared again. As you probably know, this port line controls LED 1. A subroutine named 'time' is called in between the LED on and LED off instructions. It consists of three nested loops that provide the necessary delay.

NEXT MONTH

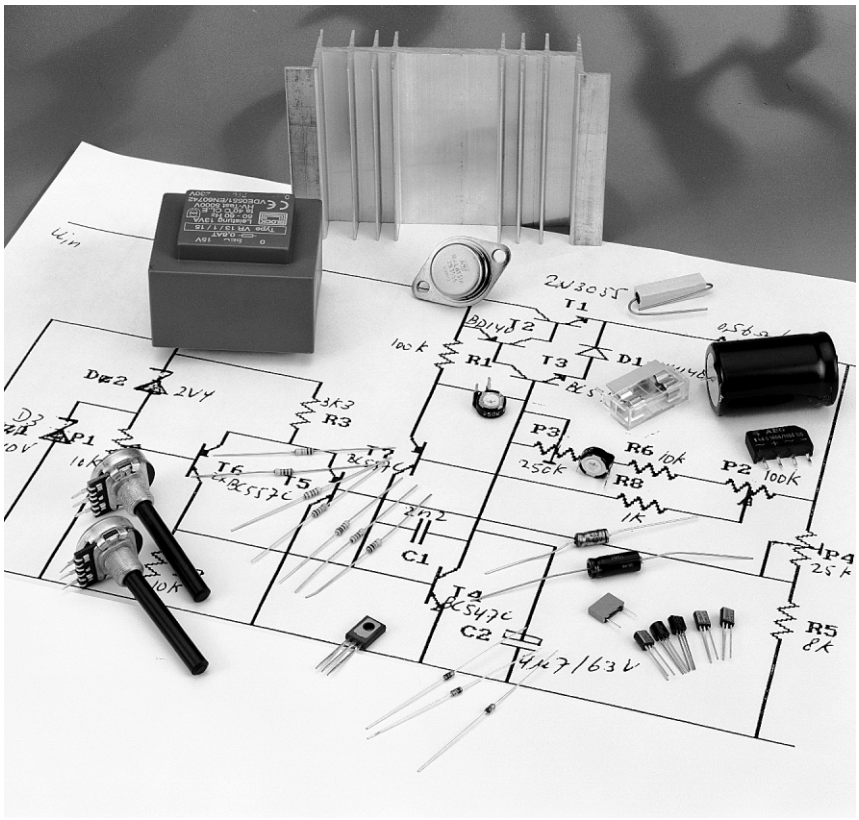
In next month's fourth and penultimate instalment of the series we will discuss further programs and applications for the SX microcontroller.

(990018-3)

general purpose variable power supply

easily adapted to individual requirements

A variable power supply is described that in spite of its simple design has two clear benefits. In the first place, it is built from discrete, readily available components, and in the second place, it can be readily adapted to individual requirements. Rules of thumb to do this are included in the article.



Design by G Baars

INTRODUCTION

The power supply is designed along fairly traditional lines, resulting in a unit whose output voltage as well as its current limiting is variable. In principle, both can be varied from nought, but in this design it was decided to make the peak values of voltage and current variable. These peak values can be varied from 10 V to 40 V and from 500 mA to 2.5 A. This makes the supply suitable for use in a variety of applications.

CIRCUIT DESCRIPTION

Although the supply does not contain a single integrated circuit, it is still very compact.

The circuit diagram in **Figure 1** deals merely with the regulator section, since the ratings of transformers, bridge rectifier and smoothing capacitors depend on the required output voltage and current.

The regulator proper is transistor T_1 in series with input and output terminals. In conjunction with transistors T_2 and T_3, it forms an emitter follower that has good current amplification and a low base-emitter potential.

The differential amplifier formed by T_5 and T_6 compares the voltage at the wiper of P_1 with a part of the output voltage derived from potential divider R_5-P_4. The amplifier tends to equalize these voltages by varying the drive to the emitter follower via T_4. When the output voltage exceeds the value set

with P_1, T_4 will conduct harder and pull the base of T_3 to earth. Transistors T_1–T_3 then conduct less hard, whereupon the output voltage drops. When the output voltage is lower than set with P_1, the opposite happens.

Transistor T_7 provides the requisite current limiting, for which R_4 functions as the current sensor. When the current through, and therefore the voltage across, this resistor exceeds a certain level, the potential across network P_3-R_6-P_2 increases to a value that causes T_7 to conduct harder (owing to the potential at the base of T_2). The resulting higher current into the base of T_4 causes this transistor to reduce the base voltage of T_3. The emitter follower then conducts less hard, whereupon the output voltage drops. Since the current through T_3 and D_1 is small, there is hardly any heat dissipation and the current limiting is virtually immune to temperature drift.

Summarizing the action of the potentiometers: P_1 enables the output voltage to be set between 0 V and maximum.

Potentiometer P_4 sets the peak output voltages according to

$$U_{max} = 10(1 + P_4/R_5) \quad [V]$$

where the values of P_4 and R_5 are in kΩ. When $P_4 = 0$ (wiper at U_{out}), the peak output voltage is 10 V, and when $P_4 = 25$ kΩ, $U_{max} = 40$ V.

Potentiometer P_2 enables the output current to be set between 0 A and a peak value determined by the setting of P_3. With $P_3 = 0$ (wiper at R_6), the peak output current is 2.5 A and with $P_3 = 250$ kΩ, the peak current is about 500 mA.

A PRACTICAL CASE

The rating of the transformer, bridge rectifier and electrolytic smoothing capacitor to precede the regulator is determined on the basis of the circuit in **Figure 2**.

Assume that an output voltage variable between 0 V and 18 V, and the current limiting variable between 0 A and 1 A, is required. This requires a mains transformer rated at not less than 18 V a.c. and an output current of not less than $\sqrt{2} \times 1 = 1.4$ A. Furthermore, losses in the bridge rectifier, series transistor (T_1) and resistor R_4 must be taken into account. This is why it is better to use the rule of thumb that the transformer

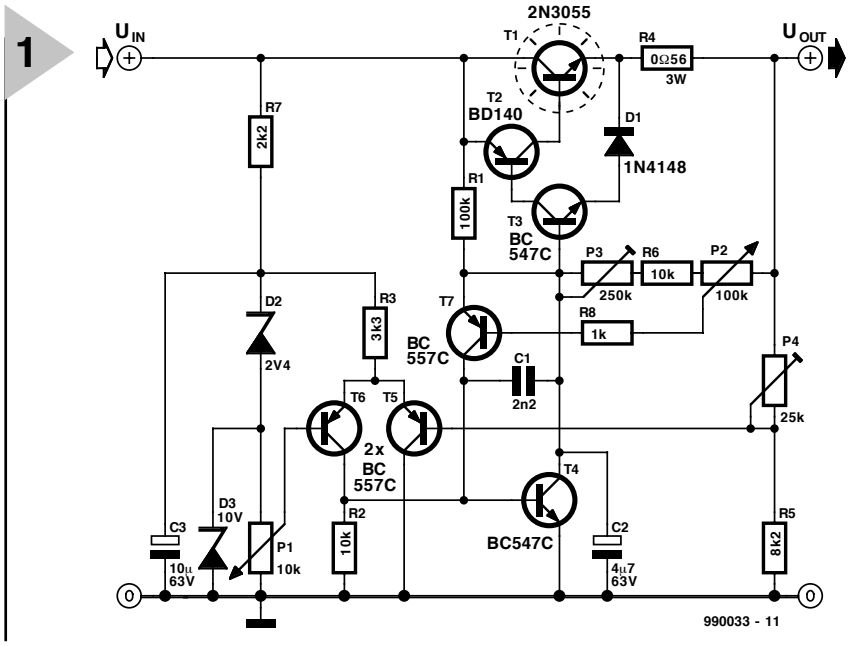


Figure 1. Circuit diagram of the regulator section of the power supply unit.

rating should be about 50 per cent higher than the theoretically needed power: in this case, a secondary alternating voltage of 18 V and a rating of 27 VA.

By rule of thumb, the bridge rectifier should also be rated about 50 per cent higher than theoretically required. This means that the requisite one should be a 35 V/1.5 A type.

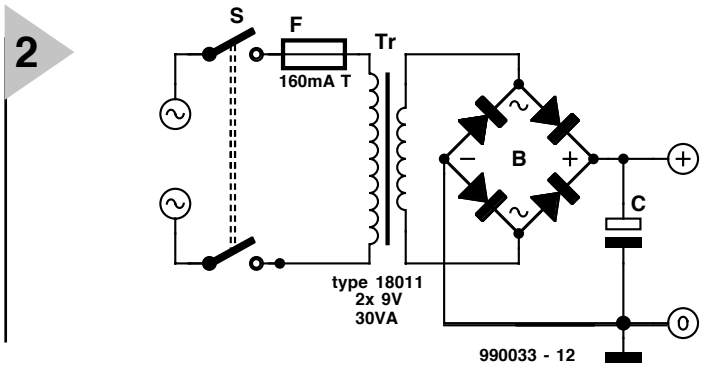
The value of smoothing capacitor *C* determines the minimum input voltage to the regulator. Since the wanted output voltage is 18 V, and the potential drop across R_4 with an output current of 1 A is 0.56 V, and the drop across T_1 is about 3 V, the minimum input voltage is 21.6 V.

The transformer provides, after rectification and smoothing, a voltage of $\sqrt{2} \times 18 = 25.4$ V. This should be reduced by about 1.4 V to account for losses in the bridge rectifier, but in practice these losses are compensated by the fact that most transformers provide a higher voltage than nominal.

The maximum permitted ripple voltage is the difference between the available direct voltage and the requisite input voltage, that is, 3.8 V.

The value of *C* in microfarad is calculated with the formula $C = IT/U_r$, where *I* is the peak output current in A, *T* is the period after full-wave rectification (0.01 s), and U_r is the ripple voltage in V. Substituting the values found earlier

Figure 2. Circuit diagram of the power section: the values and ratings of the components depend on individual requirement.



gives:

$C = 1 \times 0.01/3.8 = 2632$ [μF]

Bearing in mind the drop across the bridge rectifier, this value should be rounded upward to, say, 3300 μF. The operating voltage should 35 V There may be situations where a value of 3300 μF is not easily obtained, and a 4700 μF type should then be used.

It is advisable to precede the mains transformer by a mains fuse. This should be rated at $1.25I_{max}$, that is in this case:

$1.25 \times 27/230 = 147$ mA,

which is rounded upward to 150 mAT (T is slow-blow).

HEAT SINK

Finally, the rating and size of a suitable heat sink must be determined. Again, this may be done with the aid of rule of thumb.

The maximum dissipation of the series transistor is the product of the input voltage and the maximum output current. In this case, this is

$26 \times 1 = 26$ W.

It will be assumed that the case temperature of the transistor should not rise above 70 °C. The inner junction will, of course, get hotter (in the case of an 2N3055 about 105 °C), but this does not matter, because the transistor is capable of withstanding 120–150 °C.

The heat sink temperature of 70 °C is about 50 °C higher than the average room temperature. This means that there is 50 K (K=°C) available to dissipate 26 W. This means that the thermal resistance of the heat sink should be $50/26 = 1.9$ K W^{-1}.

When a heat sink of this rating is used, the power supply can provide an output voltage of 1 V and an output current of 1 A for hours without any problems whatsoever. If t is assumed that this extreme situation will never arise and that T_1 may on occasion get hotter than 70 °C, a heat sink with a thermal resistance of 3 K W^{-1} may be used. Bear in mind that isolation washers between transistor and heat sink increase the thermal resistance by 0.2–0.9 K W^{-1}, depending on the material of the washer.

It should be noted that a heat sink at 70 °C can singe your skin badly and it should therefore never be touched with bare hands.

ADDITIONAL NOTES

Although the regulator circuit can be used for many requirements, the input voltage to it should not drop below 15 V to ensure that zener diodes D_2 and D_3 continue to function correctly.

Also, it is advisable to insert a fuse between the smoothing capacitor and the regulator. Its rating should be equal to, or slightly higher than, the peak output current.

To ensure the requisite stability, an electrolytic capacitor with a value of about 1/10 of that of the smoothing capacitor should be added across the output terminals.

SETTING UP

When the unit has been completed and checked thoroughly, connect a voltmeter (50 V d.c. range) to the output terminals, turn P_1 fully clockwise (maximum), and adjust P_4 until the voltmeter shows the wanted output voltage (for instance, as in the foregoing example, to 18 V).

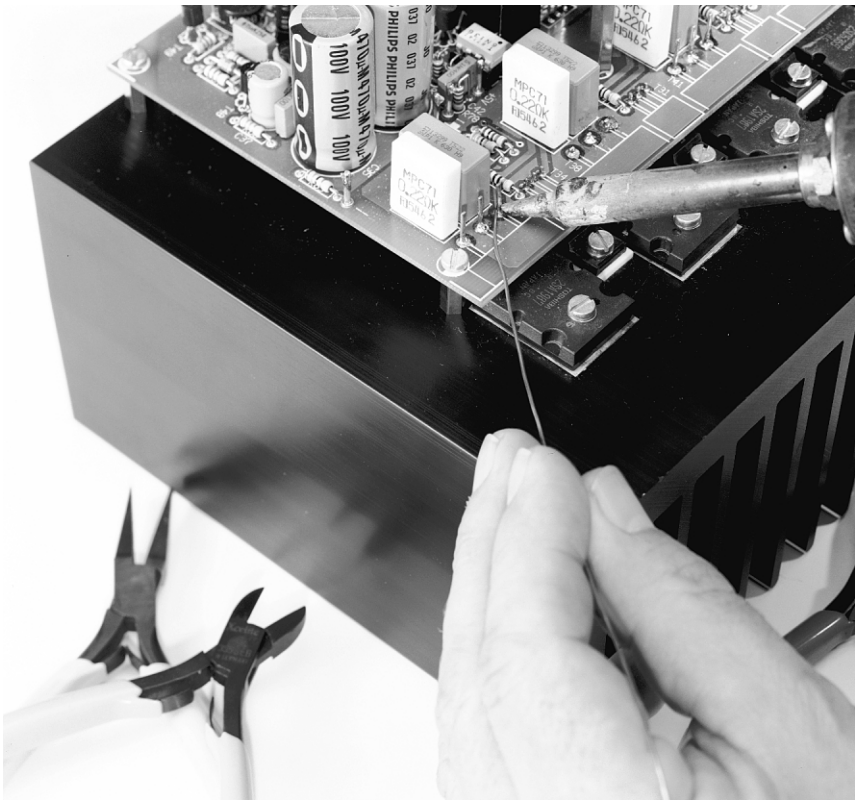
Terminate the output by a 24 V, 50 W load (a halogen light bulb is ideal for this) in series with an ammeter (set to 10 A). Turn P_3 fully clockwise (lowest current), adjust P_2 for maximum current, and set P_3 for the wanted maximum current.

[990033]

Titan 2000

Part 3: construction and setting up

This third of four parts deals primarily with the construction of the amplifier and ends with a brief resume of its performance and specifications. Let the constructor beware, however: the Titan 2000 is not an easy project and certainly not recommended for beginners in electronic construction.



Design by T. Giesberts

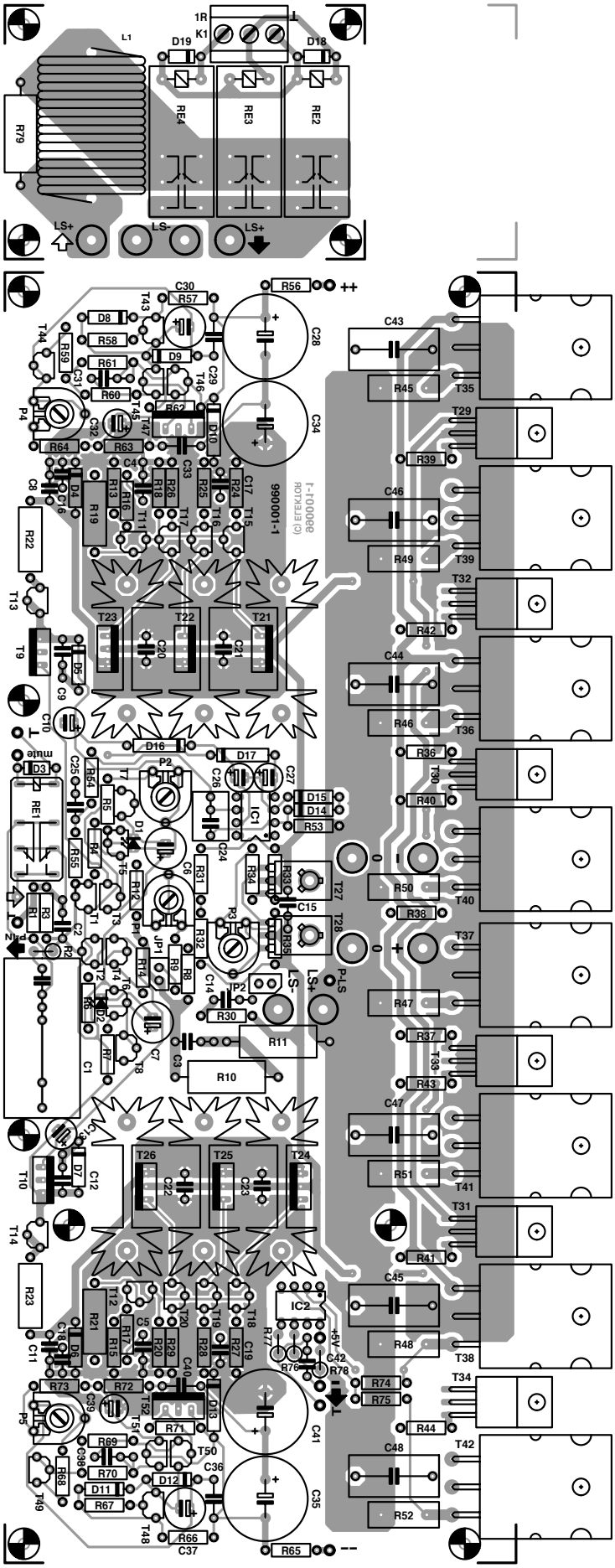
INTRODUCTION

It is clear from the first two parts of this article that the Titan 2000 is a complex unit that needs to be constructed and wired up with with great care to ensure the specified performance. For that reason, the construction notes will be more detailed than is usual with projects in this magazine. It is assumed that the protection network and auxiliary power supply have already been built and tested.

MOTHER BOARD

It must be borne in mind that in the case of a fast power amplifier like the Titan 2000, with a gain/bandwidth product of about 0.5 GHz, the board must be an integral part of the circuit. The mother board is therefore designed together with the remainder of the circuit. The length of the tracks, the area of the copper pads, the positions of the decoupling capacitors, and other factors, are vital for the proper and stable operation of the unit. Constructors who make their own boards are therefore advised to adhere strictly to the published layout.

Owing to the power requirements, the various stages are parallel configurations. When these are mounted on the heat sinks, a fairly large parasitic capacitances to earth ensue. This is because for reasons of stability all seven heat sinks must be strapped to earth. It



Parts lists

It is regretted that, owing to circumstances beyond our control, component codings in the various sections have been duplicated. Consequently, the mother board, protection network board, and auxiliary power supply board contain many components with the same identification (R_1-R_{36}, C_1-C_{26}, D_1-D_{12}, T_1-T_6, IC_1-IC_2, JP_1, K_1).

Amplifier

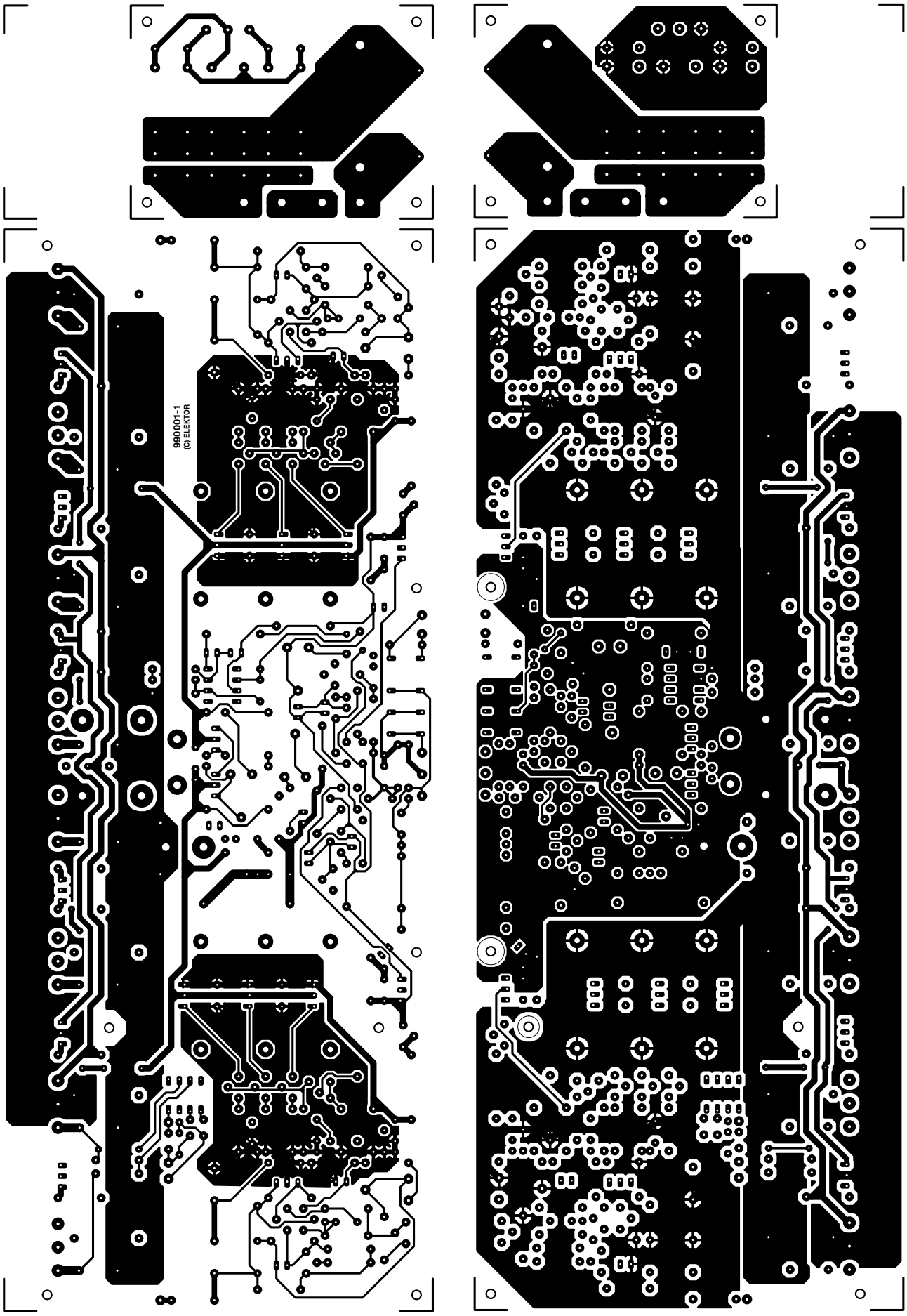
Resistors:

R_1, R_{53} = 1 MΩ
R_2 = 562 Ω
R_3 = 47 kΩ
R_4, R_6, R_{12}, R_{14}, R_{60}, R_{61}, R_{69}, R_{70} = 22 Ω
R_5, R_{62}, R_{71} = 330 Ω
R_7, R_{34} = 470 Ω
R_8 = 22.1 Ω
R_9 = 390 Ω
R_{10}, R_{11} = 470 Ω, 5 W
R_{13}, R_{15} = 1.00 kΩ
R_{16}, R_{17}, R_{38} = 150 Ω
R_{18}, R_{20}, R_{58}, R_{67} = 270 Ω
R_{19}, R_{21} = 10 kΩ, 1 W
R_{22}, R_{23} = 3.3 kΩ, 1 W
R_{24}–R_{29} = 68 Ω
R_{30} = see text
R_{31}, R_{32} = 22 kΩ
R_{33}, R_{35} = 220 Ω
R_{36}, R_{37} = 560 Ω
R_{39}–R_{44} = 10 Ω
R_{45}–R_{52} = 0.22 Ω, inductance-free
R_{54}, R_{55} = 4.7 MΩ
R_{56}, R_{65} = 15 Ω
R_{57}, R_{63}, R_{66}, R_{72} = 15 kΩ
R_{59}, R_{68} = 5.6 kΩ
R_{64}, R_{73} = 12 kΩ
R_{74}, R_{76}, R_{77} = 100 Ω
R_{75} = 33 Ω
R_{78} = 2.2 kΩ
R_{79} = 2.2 Ω, 5 W
P_1, P_4, P_5 = 4.7 kΩ (5 kΩ) preset
P_2 = 250 Ω, preset
P_3 = 500 Ω, preset

Capacitors:

C_1 = 2.2 µF, metallized polyester (MKP)
C_2, C_3, C_{42} = 0.001 µF
C_4, C_5 = 0.0022 µF
C_6, C_7 = 220 µF, 25 V, radial
C_8, C_9, C_{11}, C_{12}, C_{15} = 0.1 µF
C_{10}, C_{13} = 100 µF, 25 V, radial
C_{14} = see text
C_{16}–C_{23} = 100 pF, 100 V
C_{24} = 1 µF, metallized polypropylene (MKT)
C_{25} = 0.68 µF
C_{26}, C_{27}, C_{32}, C_{39} = 2.2 µF, 63 V, radial

Figure 12. The double-sided printed-circuit board is intended to be combined with the heat sink into a single entity. Before that can be done, however, the section for the output relay and the inductor must be cut off the main section.


990001-1
(C) ELEKTOR

C_{28}, C_{34}, C_{35}, C_{41} = 470 µF, 100 V, radial
C_{29}, C_{33}, C_{36}, C_{40} = 0.22 µF, 100 V
C_{30}, C_{37} = 47 µF, 63 V, radial
C_{31}, C_{38} = 0.015 µF
C_{43}–C_{48} = 0.1 µF, 630 V

Inductors:
L_1 = see text

Semiconductors:
D_1, D_2 = LED, red, flat
D_3, D_{18}, D_{19} = 1N4148
D_4, D_6 = zener, 5.6 V, 500 mW
D_5, D_7 = zener, 15 V, 1.3 W
D_8, D_{11} = zener, 30 V, 1.3 W
D_9, D_{12} = zener, 39 V, 1.3 W
D_{10}, D_{13}, D_{16}, D_{17} = 1N4004
D_{14}, D_{15} = zener, 12 V, 500 mW
T_1, T_4, T_5, T_{15}–T_{17} = BC560C
T_2, T_3, T_6, T_{18}–T_{20} = BC550C
T_7, T_8, T_{43}, T_{48} = BF245A
T_9 = BF871
T_{10} = BF872
T_{11}, T_{50}, T_{51} = BC640
T_{12}, T_{45}, T_{46} = BC639
T_{13}, T_{14} = BF256C
T_{21}–T_{23} = MJE350
T_{24}–T_{26} = MJE340
T_{27} = BD139
T_{28} = BD140
T_{29}–T_{31} = 2SC5171 (Toshiba)
T_{32}–T_{34} = 2SA1930 (Toshiba)
T_{35}–T_{38} = 2SC5359 (Toshiba)
T_{39}–T_{42} = 2SA1987 (Toshiba)
T_{44}, T_{49} = BF256A
T_{47} = BD712
T_{52} = BD711

Integrated circuits:
IC_1 = OP90G
IC_2 = 6N136

Miscellaneous:
JP_1, JP_2 = 2.54 mm, 2-way pinstrip and pin jumper
K_1 = 3-way terminal block, pitch 5 mm
Re_1 = relay, 12 V, 600 Ω
Re_2–Re_4 = relay, 12 V, 16 A, 270 Ω
Heat sink for T_{21}–T_{26} = 38.1 mm, 11 K W^{-1} (Fischer Type SK104-STC; TO220)
Heat sink for drivers/output transistors, 150 mm, 0.25 K W^{-1}, Fischer Type SK157
Ceramic isolation washers for T_{21}–T_{34}: Fischer Type AOS220
Mica isolating washers for T_{35}–T_{42}
PCB Order no 990001-1 (see Readers Services towards end of this magazine)

is, of course, of paramount importance that these capacitances are as small as feasible. For this reason, it is vital that in the thermal coupling of T_{21}–T_{34} 1.5 mm thick ceramic—not mica—isolating washers are used. Mica washers may, however, be used with the output transistors since parasitic capacitances there are of no significance.

The component and track layouts of the mother board are shown in **Figure 12**. It will be seen that the board consists of two sections: the mother board proper and the output-relay board. The latter must be cut off before any other work is done. Later, when it is built up, it is mounted on the mother board with the aid of four 50 mm long metal spacers in such a way that the LS– and LS+ terminals on the two boards are above each other. The spacers also provide the electrical link between the boards.

The completed relay board is shown in **Figure 13**. Inductor L_1 is made from a doubled-up length of 1.5 mm enamelled copper wire wound in two layers of eight turns each around a 16 mm former (such as a piece of PVC pipe). After the coil has been wound, the PVC pipe is removed and the four windings connected in parallel. See **Figure 14**.

Ignoring the drivers and output transistors for the moment, the construction of the mother board is traditional. As always, great care must be taken during the soldering and placing of components. Do not forget the thermal coupling of T_1-T_3, T_2-T_4, D_1-T_5, D_2-T_6, T_{45}-T_{46}, and T_{50}-T_{51}, as already pointed out in Part 1. Also, T_{21}–T_{23} and T_{24}–T_{26} must be mounted on a heat sink, and isolated from it by means of a ceramic washer. When this is done, fit the composite heat sinks on the board, and link them to earth.

The input signal and the ± 85 V supply lines are linked to the board via standard solder pins.

For connecting the ± 70 V supply lines and the relay board, 3 mm screw holes are provided. Metal spacers are to be fixed to these and cable connectors to the top of the spacers.

MAIN HEAT SINK

When the mother board has been completed, and carefully checked, as far as described, it and the drivers and output transistors, T_{27}–T_{42}, must be mounted on the main heat sink. This is a 150 mm high Type SK157 from Fischer with a thermal resistance of 0.25 K W^{-1}. This is admittedly a very tedious job. It is vital that all requisite fixing holes are drilled accurately in the heat sink and preferably tapped with 3 mm thread. The template delivered with the ready-made board is almost indispensable for this work.

When the holes have been drilled (and, possibly, tapped) transistors T_{27} and T_{28} should be fitted first (this is important because they become inaccessible after the board has been fitted). They must be located as close as possible to the output transistors and not in the position indicated on the board. Again, the template makes all this clear. Their terminals must then be extended with the aid of short lengths of equipment wire, which are later fed through the relevant holes on the board and soldered to the board via, for instance, a three-way pin header.

The terminals of the drivers and output transistors must be bent at right angles: those of the former at the point where they become thinner and those of the latter about 5 mm from the body of the device. When this is done, screw all transistors loosely to the heat sink, not forgetting the isolating washers. If it is intended to use fan cooling, the requisite temperature sensor—that is, a Type BD140 transistor— should also be attached to the heat sink at this stage. The template does not show a location for the sensor, but it seems sensible to fit it at the centre close to T_{37} or T_{40}.

The next step is to fit all ten spacers to the heat sink: these should all be 10 mm long. In the prototype, spacers with a 3 mm screwthread at one end were used. Two of the spacers merely provide additional support for the relay board and another two form the electrical link between the negative supply line and the heat sink.

When all this work is done, the board should look more or less like that in **Figure 15**. Note that because of tests later on, there are, as yet, no ceramic isolating washers fitted on the prototype.

The next, and most tedious, step is to combine the board and heat sink. It is, of course, vital that all spacers are exactly opposite the relevant fixing holes and—even more tedious—that the terminals of all transistors are inserted into the correct mounting holes. Bear in mind that the metal

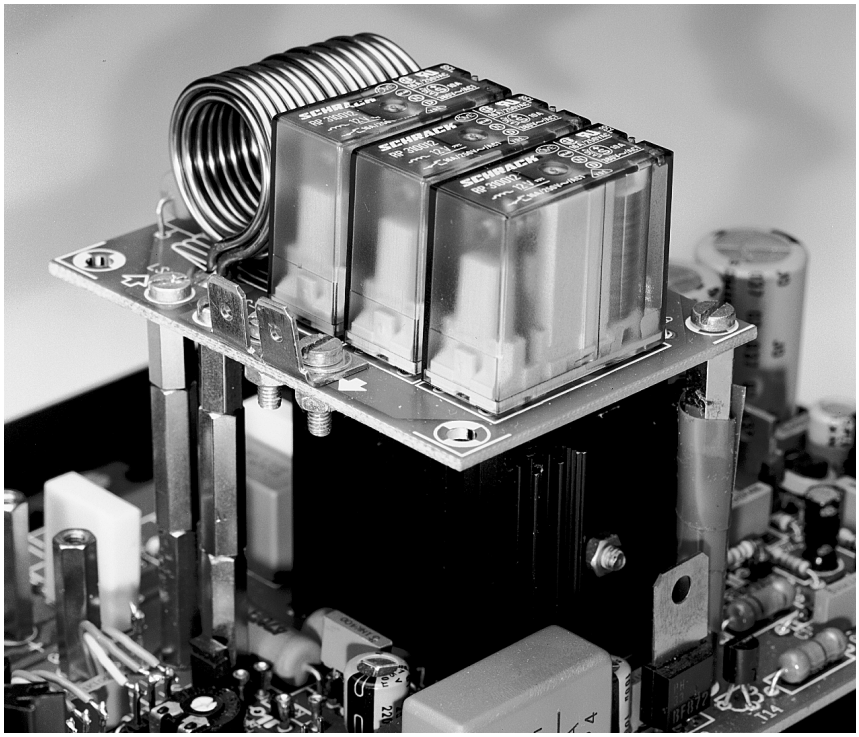


Figure 13. Illustrating how the relay board is mounted on the mother board with the aid of spacers.

spacers for linking –, + , LS+ , and LS–, are already on the board. As the terminals of the output transistors are slightly longer than those of the drivers, it may be possible to do this work in two stages: output transistors first and drivers second. It may prove necessary to turn one or more of the transistors slightly, which is the reason that the fixing screws have not yet been tightened. When all terminals are correctly inserted, these screws must, of course, be tightened firmly.

The final step is to fix the relay board on the spacers that form the link for the LS– and LS+ terminals.

SETTING UP

Before the amplifier module can be taken into use, presets P_2–P_5 must be set as required. Preset P_1 is intended only for possibly adjusting the balance in case of a bridge configuration.

Start by turning P_3 (the quiescent-current control) fully anticlockwise and P_2, P_4, and P_5, to their centre position. Check the outputs of the power supply and auxiliary power supply and, if these are correct, link the + 70 V line to pins '+ ' and '0', the –70 V line to '–' and '0', the + 85 V line to '+ + ' and the -85 V line to '--'. For absolute safety, link the ± 70 V lines temporarily via a 10 Ω, 5 W resistor.

Next, set P_4 and P_5 for voltages of + 78 V and –78 V respectively at the cases of transistors T_{47} and T_{52} respec-

Figure 14. Air-cored inductor L_1 is formed by laying two windings each of eight turns of doubled-up each on top of one another. The former is a length of 16 mm diameter PVC pipe as used by plumbers. The resulting four windings are simply connected in parallel.

tively (the cases of these transistors are linked to the output of the relevant regulator). It is important that the negative and positive voltages are numerically identical.

Since the parameters of the n-p-n and p-n-p transistors in the input stage are never exactly identical, there may be a slight imbalance. This may be corrected by adjusting the output of current source T_5 with the aid of preset P_2 to give a potential of exactly 0 V at the output (pin 6) of IC_1 (when 'cold').

Finally, insert an ammeter (set to 500 mA or 1 A range) in the + 70 V or –70 V line, and adjust P_3 carefully for a quiescent current of 200 mA (cold condition—that is, immediately after switch-on). With a large drive signal, the quiescent current may increase to some 600 mA, but at nominal temperatures, its level will stabilize at 200–400 mA. Note that these fluctuations have no noticeable effect on the performance of the amplifier.

CHECK AND TEST

When the amplifier has been switched on for about half an hour, the voltages shown in Figure 2 (Part 1) may be verified. Note that voltage levels depending on the setting of current sources habitually show a substantial spread: 30 per cent is quite common. All measurements should be carried out with a good digital voltmeter or multimeter with a high-impedance input.

Other than the test voltages in the circuit diagram, there are some others that may be checked. For instance, the proper functioning of the output transistors may be ascertained by measuring the voltage across R_{45}–R_{52}. Hold one test probe against the loudspeaker terminal and with the other measure the potential at the emitters of all output transistors. The average value should be about 20 mV, but deviations of up to 50 per cent occur.

The voltage amplifier operation may be checked by measuring its current drain: if this is within specification, the voltage across R_{56} and R_{65} must be within 0.8–1.1 V (after the amplifier has been on for at least half an hour).

Finally, the potential drops across the emitter resistors of differential amplifiers T_{45}-T_{46} and T_{50}-T_{51} must not differ by more than a factor 2. Too large a factor is detrimental to the stable operation of the amplifiers. A too large difference may be corrected by changing the value of R_{62} or R_{71}, as the case may be. If this is unsuccessful, the relevant transistor pair will have to be replaced.

When all is well, the resistors in series with the ± 70 V lines should be removed. Note that a rectified voltage of 70 V, let alone one of 140 V, is lethal. It is therefore absolutely essential to switch off the power supply and verify that the residual voltages have dropped to a safe value before doing any work on the amplifier.

Next month's instalment will deal with the wiring up of the amplifier and its performance, including specifications.

[990001-3]

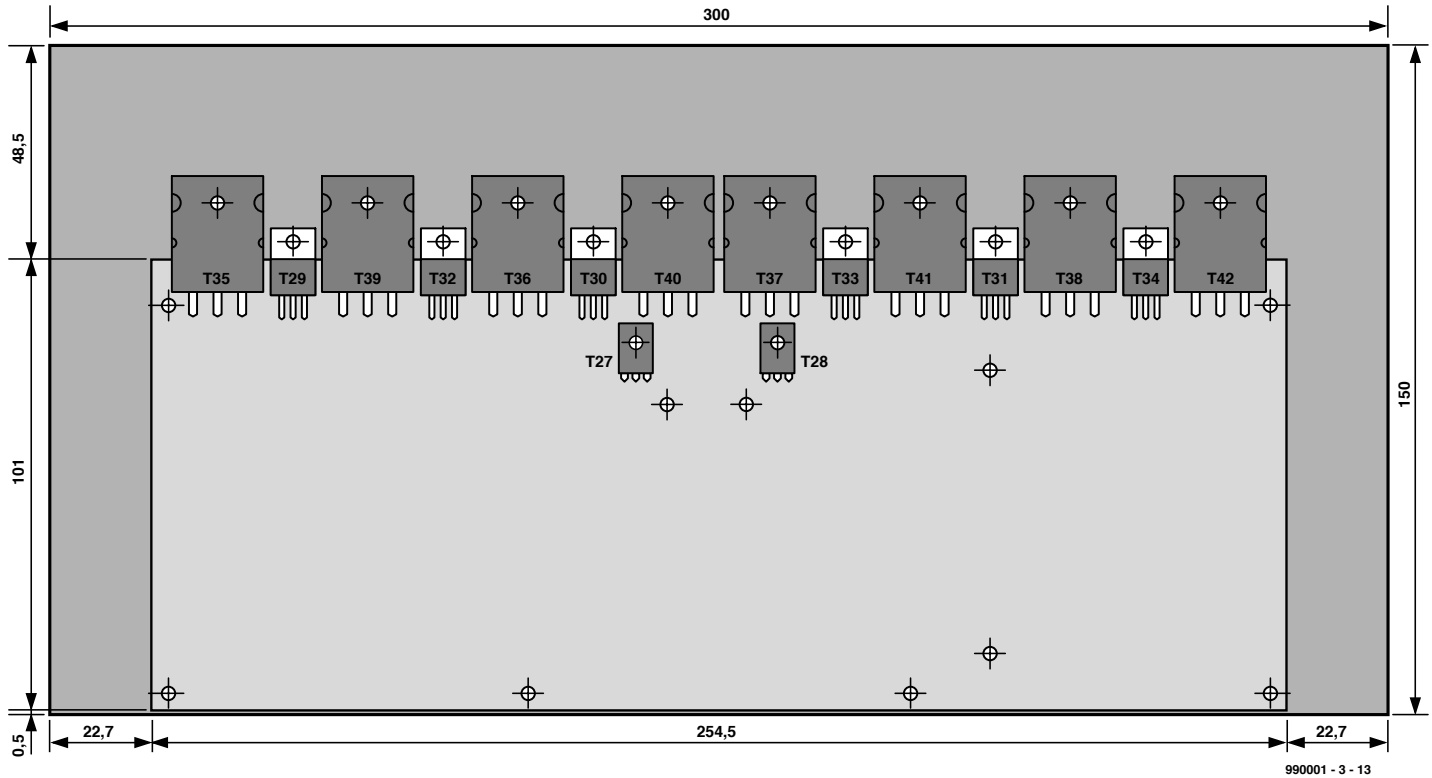


Figure 15. The PCB is delivered with a template to ensure that the transistors are fitted at the correct location on the heat sink.

evaluation system for SAB80C166 (2)

part 2: construction and use

Despite its apparent complexity, the practical circuit of the SAB80C116 evaluation system is only marginally different from typical 8-bit systems as published in Elektor Electronics.

Design by K.-H. Domnick

Just like almost any other microcontroller system, the 80C166 16-bit board is marked by buses for data, address and control signals. Because the data-bus of the 80C166 has a width of 16 lines, many peripheral ICs like latches and memory elements have to be duplicated. The 80C166 offers a total of six ports (P0-P5), of which P5 is employed by the 8-channel A-D converter. The corresponding lines are protected against voltage surges by zener diode-resistor networks R10-R19, R21/22/D3-D13.

Port P0 is multiplexed as the combined address and data bus. The actual address bus starts behind two latch components, IC9-IC10. The bus has an addressing capacity of 64 kBytes. The address bus is routed to memory devices IC11-IC18, GAL IC4, Real Time Clock IC5 and decoder IC6. Moreover, it is made available on extension connector K2 for use by external circuits.

The digital inputs and outputs are taken to connector K1 by way of buffer devices IC19-IC22. The level conversion to RS232 signals is carried out in IC1 and IC2.

Resistors R1-R4 are optional, and intended for special applications of the SIOs (serial input/output devices). Circuit IC7 is also optional for unusual applications employing the controller's Ready signal and a lengthened output signal on port line P1.13.

Because the chip select signals are decoded in a GAL, it is possible to modify the memory configuration.

Two field-effect transistors (FETs) type BS170 (T1 and T2) are included to ensure that the current through the pull-up resistors disappears when the board is switched off with the RAMs still powered by the backup battery.

After any write operation to the EEPROM, a pause of 2-4 ms should be observed until the next access. With some EEPROMs, Read/Busy information is available on pin 1. If such a device is employed, then their status may be interrogated by a program reading the ERDY1/ERDY2 (EEPROM-Ready) lines.

Decoder IC6 supplies the chip-select signals for the clock device, IC5, the watchdog in the MAX690 (IC3) and the extension bus.

Table 1. Minimum configuration of C166 board

R7	resistor, 1 kΩ	Pull-up for $\overline{S0}$
R9	resistor, 10 kΩ	Pull-up for $\overline{WR}$
R23	resistor, 4.7 kΩ	Pull-up for $\overline{RSTINT}$
R24	resistor, 10 kΩ	Pull-up for $\overline{NMI}$
C1,C2	capacitor, 10 µF, tantalum	Decoupling
C9,C10,C16,C17	capacitor, 100 nF	Decoupling
C22-C25	capacitor (tantalum), 10 µF	SIO2
T1	transistor, BS170	For S0 signal (Chip Select 0 for RAM)
IC1	MAX232	SIO2
IC4	GAL 16V8, 15-12 ns	Programmed
IC8	SAB 80C166M	Microcontroller
IC9/IC10	74HC573	Address latch
IC11/IC12	62256	Static RAM
IC13/IC14	27C256	EPROM (user program or system monitor)
Q1	40 MHz	Oscillator or quartz crystal w. C18/C19
JP1,JP2,JP5		

Table 2. Modified contents of configuration file

```
;*****************************************************************
;*     configuration file for EVA165/166/167 monitor MON16X      *
;*                        (c) ertec GmbH                         *
;*****************************************************************
;
bios=0                    ; use direct port i/o
; baud=9600               ; baudrate 9600
baud=38400                ; baudrate 38400
com=2                     ; use COM2
; boot=boot.166           ; loader for monitor MON166
; monitor=mon.166         ; monitor MON166
;*****************************************************************
;
```

SOLDER A CENTIPEDE

Most of you will agree that soldering the 100-pin microcontroller on to the board is a daunting task. Alas, there is no alternative because the 80C166 is only supplied in a 100-pin MRFP case.

Apart from fine-gauge solder (max.

1 mm dia.) for fitting the 'regular' parts you will need suitable solder cream and solder flux. The solder iron should have a fine tip, and be temperature controlled at about 320 °C. The microcontroller is the first part to be soldered on to the board. Before you start soldering, make sure that all 100 pins are properly aligned to the pads on the PCB. If necessary, carefully adjust their position. Next, mix five or six parts of solder flux with one part of solder cream, and dilute this with a drop of methylated spirit until the mixture starts to become liquid. The mixing is conveniently done in a beer bottle cap, using a toothpick as a stirring device. The mixture is then carefully applied to the edges of the solder pads, or to the ends of the microcontroller pins. This is best done using a very small brush. In general, the less mixture you apply, the better.

Next, the controller may be positioned on the pads again, and fixed in position by soldering two diagonally located pins. Now once again check the correct alignment of each and every pin over its own solder pad. If no obvious problems are detected, you can start soldering. Using one finger to push the controller firmly on to the board and at the same time check its temperature, carefully and slowly pull the tip of the solder iron past the pin extremes (i.e., not across the pins). Provided the pad and the pins are uniformly heated, the solder cream will also contract and solidify in a uniform manner through the solder flux. When you feel that the chip gets too hot, take a short break, and continue where you left off.

Use a magnifying glass to inspect all solder joints, and correct any one that needs to be brought in line. If an unwanted joint has been made, press desoldering braid and some flux on to the link, heat the joint and remove the excess solder by a gradual pulling motion. Flux and solder cream residue may be removed with a cotton swab dipped in methylated spirit.

With the 80C166 safely ensconced on the board, construction may proceed, starting the lower-profile parts (resistors), and, finally, the taller parts like the electrolytic capacitors. The parts list, incidentally, indicates a minimum configuration (M) for the board, see also **Table 1**.

As a replacement for IC3 (MAX690), connect a 10-μF tantalum electrolytic capacitor to IC3 pin 7 (+ wire) and pin 3 (– wire). This capacitor will act as

1

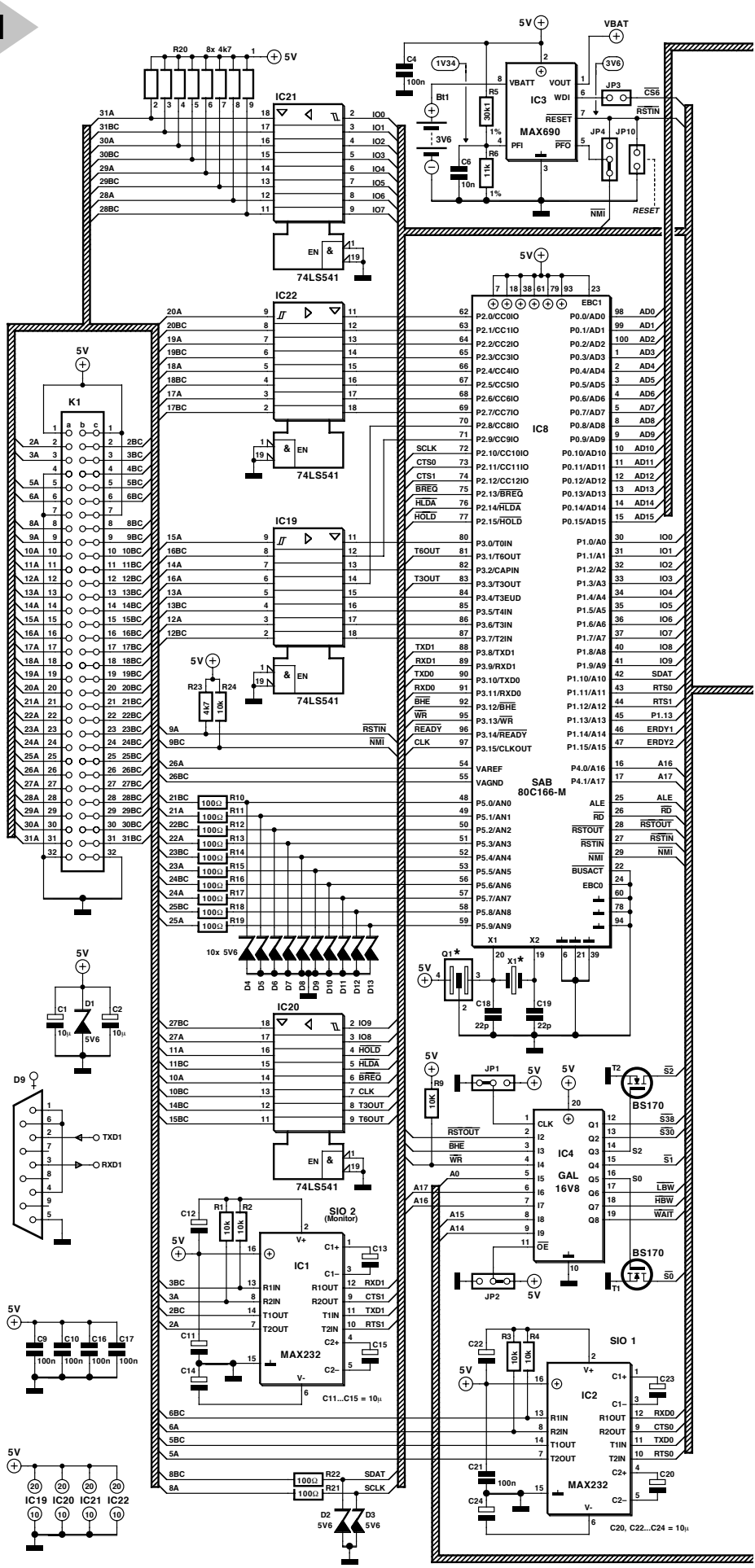
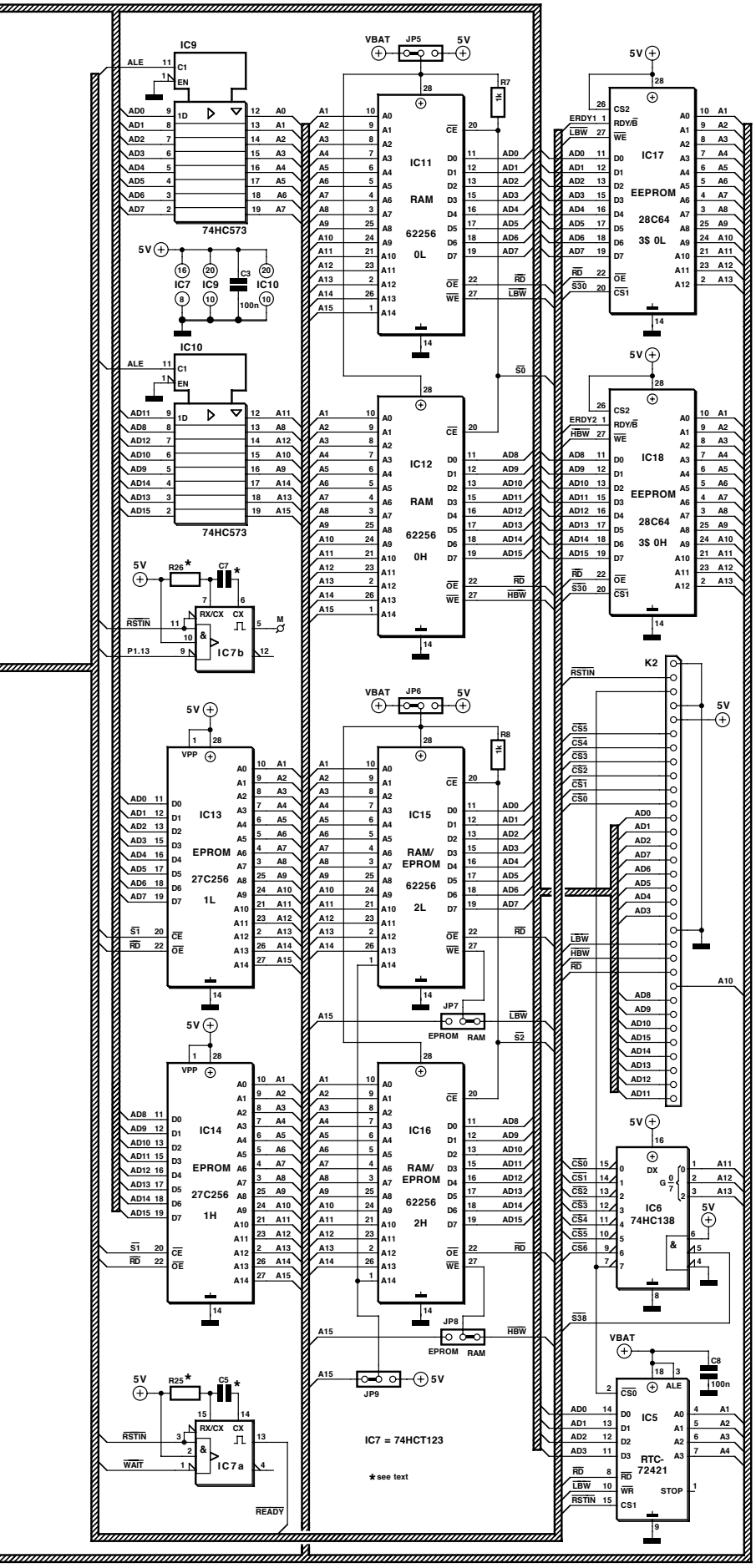


Figure 1. The quite extensive circuit diagram of the 80C166 16-bit microcontroller evaluation system.



the power-on reset. When the MAX690 is employed, jumper JP3 should be pulled. If not, the MAX690 will faithfully produce a reset signal at 1-second intervals.

To power the RAM chips, install JP5 in the 5-V position, and connect GAL pins 1 and 11 to ground using JP1 and JP2.

When you are satisfied with your construction efforts, it is time to give the board a last and thorough visual inspection. If no faults are discovered, the ICs may be inserted into their sockets.

POWERING-UP

Connect the 80C166 board to a suitable 5-volt dc power supply whose current limit is set to 0.3 A. Switch on, and check that the current consumption settles at about 0.2 A.

ENTER YOUR PC

To enable the 80C166 board to communicate with a PC, it has to be fitted with two EPROMs (IC13 for the low byte and IC14 for the high byte) containing the monitor program. On the PC, the program MON16X.EXE or MON16X2.EXE has to be installed with the files MON16X.OVL, MON16X.CFG or MON16X.XSR. Connect the serial cable between SIO2 and a free communication port (RS232) on your PC, and then launch the program MON16X.EXE. Two parameters should be included:

```
[Drive:\path\]MON16X
–c [Drive:\path\]MON16X.CFG
–s [Drive:\path\]MON16X.SCR
```

where
–c indicates the drive/path/name of te configuration file;
–s indicates the drive/pathname of the script file.

To keep things as simple as possible, the program should be launched with the aid of batch file.

The contents of the original MON16X.CFG configuration file has to be modified as shown in **Table 1**. The lines in **bold** print are new, while those that are no longer required have a semicolon (;) in front of them.

The script file, MON16X.SCR, could have the following contents:

```
key 1 "Help" A
key 4 "S" A
key 5 "PROC" A
key 9 "EXIT" A
stat
```

The PC program first transmits a dummy byte to enable the monitor program on the board to detect the baud rate. The board returns 0AAh to

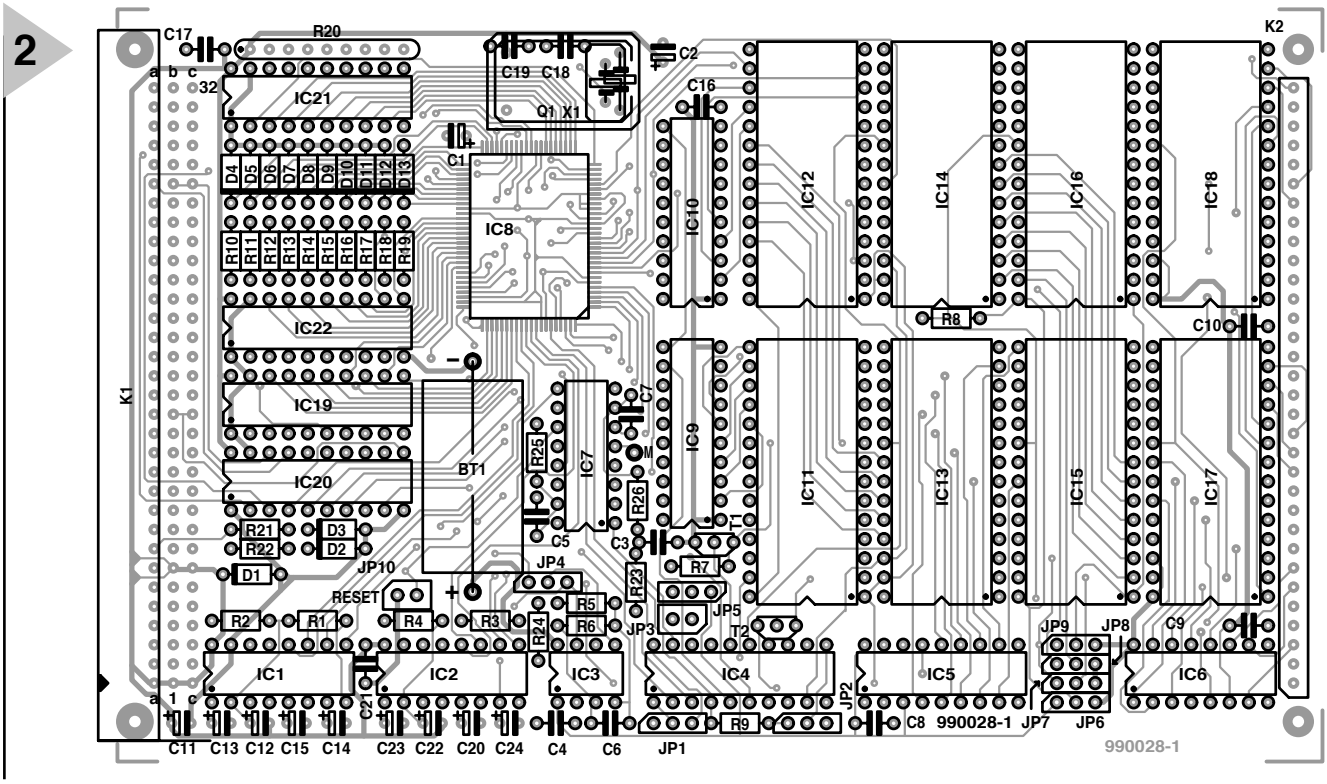


Figure 2. Densely packed: the double-sided, through-plated, Eurocard-sized PCB for the evaluation system.

COMPONENTS LIST

Resistors:
R1-R4 = 10kΩ (SIO1)
R5 = 30kΩ 1% (I/O)
R6 = 11kΩ 1% (I/O)
R7 = 1kΩ (M)
R8 = 1kΩ (RAM/ROM)
R9,R24 = 10kΩ (M)
R10...R19,R21,R22 = 100Ω (I/O)
R23 = 4kΩ7 (M)
R25,R26 = see text (RDY)

Capacitors:
C1-C4,C8,C9,C10,C16,C17,C21 = 100nF ceramic, raster 5mm (M)
C5,C7 = see text
C6 = 10nF raster 5mm (I/O)
C11-C15 = 10µF 16V radial (M)
C18,C19 = 22pF (only with 100MHz-crystal)
C20,C22,C23,C24 = 10µF 16V radial (SIO1)

Semiconductors:
D1 = ZD 5V6/1W3 (M)
D2-D13 = zener diode 5.6V 1.3W (I/O)
T1 = BS170 (M)
T2 = BS170 (RAM/ROM)
IC1 = MAX 232 (M)
IC2 = MAX 232 (SIO1)
IC3 = MAX 690 (I/O)
IC4 = GAL 16V8-15 (programmed, 996508-1) (M)
IC5 = RTC 72421 (RTC)
IC6 = 74HCT138 (BATT, RTC)
IC7 = 74HCT123 (RDY)
IC8 = SAB 80C166 M (M)
IC9,IC10 = 74HCT573 (M)
IC11,IC12 = RAM 62256-100 (M)
IC13= EPROM 27C256-120 (Low Byte, programmed, 996509-1) (M)
IC14 = EPROM 27C256-120 (High Byte, programmed, 996509-2) (M)
IC15,IC16 = RAM 62256-100 or EPROM 27C256-120 (RAM/ROM)
IC17,IC18 = EEPROM 28C64-250 (EEPROM)
IC19-IC22 = 74LS541/540 * (I/O)

Miscellaneous:
Q1 = oscillator module 40MHz (M)
IC sockets, 8/16/20/28-pin (M)
IC socket, 18-pin (RTC)
64-way pinheader or DIN-connector 41612 (M)

Items available through the Publishers' Readers Services:

996509-1, -2:	programmed EPROMs
996508-1:	programmed GAL
996009-1:	disk, C166 monitor (compiled and source code), and GAL listing
996009-2	disk, high-level debugger from Hitex
990028-1:	Printed circuit board

(M)	Minimum configuration
(I/O)	Input/Outputs
(BATT)	MAX690, battery
(SIO1)	SIO1 extension
(RTC)	Real-time clock extension
(RAM/ROM)	RAM or EPROM extension
(EEPROM)	EEPROM extension
(RDY)	READY extension

indicate that it is ready. The connection with the PC is ready for use after the version texts and the final prompt **MON166>** have appeared.

Once the monitor program is 'on line', programs may be uploaded to the 80C166 board. This is done with the command 'L' (for Load), followed by the filename, a 80C166 program in hexadecimal format, or as an 'absolute 'file:

```
L
"[Drive:\path\]filename.hex"
or
L "[Drive:\path\]file[.abs]"
```

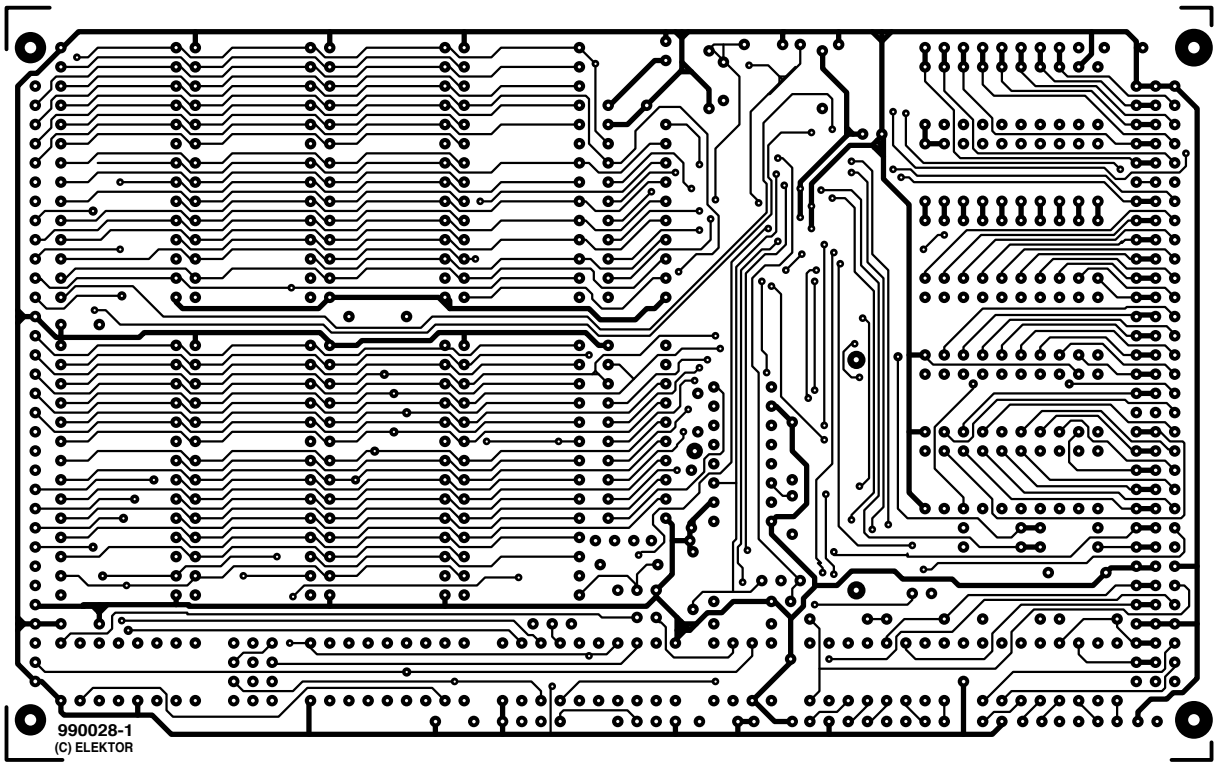
Using the command G (Go) and the relevant start address, the previously loaded program is executed.

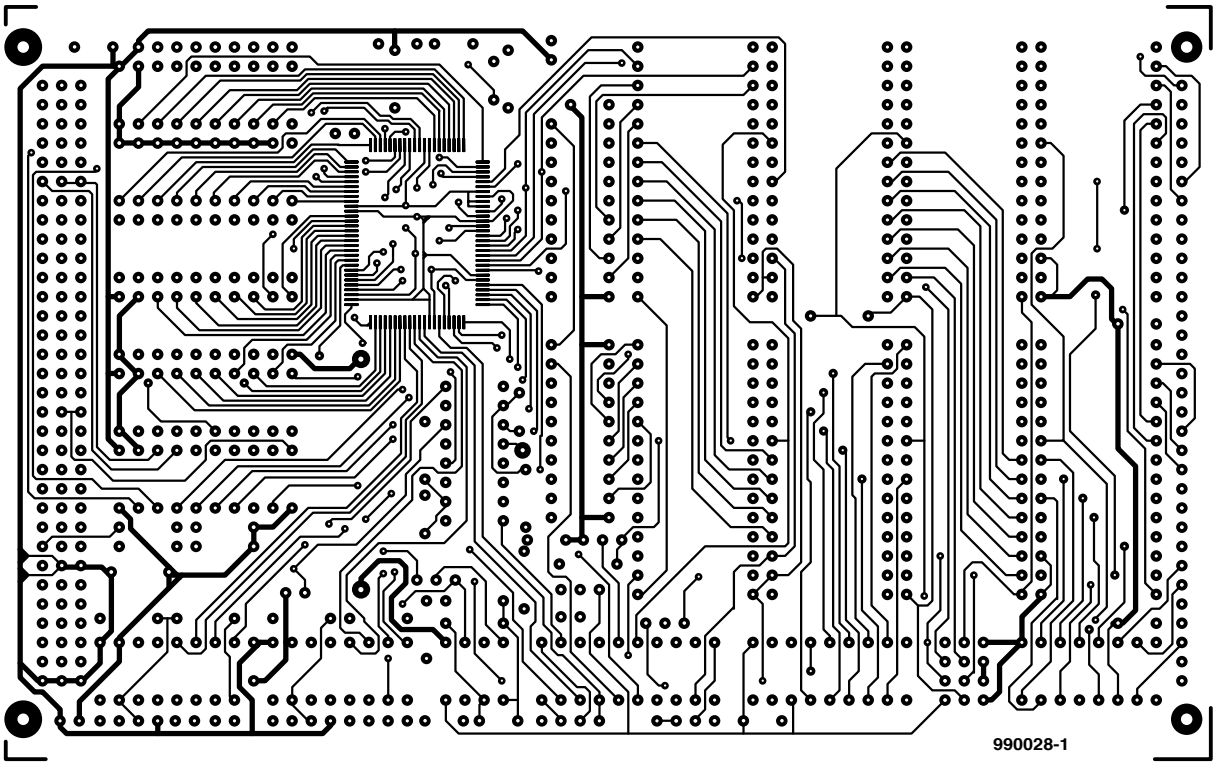
The command H is available to enable all monitor commands to be listed. The monitor program also allows further hardware tests and memory checks to be run.

If, after some time, you see one of the following error reports on the PC display: *serial send error*, *communication error*..., *timeout*, *fatal error ... abort*, or similar, you have to get ready for a faultfinding session.

HARDWARE DEBUGGING

To begin with, inspect all solder joints, particularly those on the microcontroller. Also check that all parts have been fitted in the right PCB positions. A correctly operating serial link is also essential.
Here are a few hardware-related problems you may come across:





- Wrong serial port, or wrong entry in the PC configuration file MON16X.CFG.
- RxD and TxD lines to PC swapped.
- No interconnections between handshaking lines CTS and RTS, DTR, DSR and CD, on the PC interface. These links are needed to inform the PC that a DTE (data terminal equipment, i.e., the 80C166 board) is connected to the serial port.
- Missing (monitor) program in EPROMs, or 'low' and 'high' EPROMs swapped.
- GAL (IC4) not fitted, not programmed or incorrectly programmed.
- JP1 and/or JP2 (pins 1 and 11 of the GAL) incorrectly installed.
- Missing SIO (IC1) or electrolytic capacitors required for correct functioning.
- Quartz crystal or oscillator module does not oscillate (measure at Q1 pin 3).
- Reset input remains low. When the MAX690 (IC3) is used without the battery, reset may remain low (measure at RESET pin 7 IC3 or K1 pin 9a).
- Output $\overline{\text{RSTOUT}}$ remains low. Dur-

Table 4. Contents of file 80C166.EQN

```
; GAL for 80C166 Controller Board
chip 80C166 GAL16V8
; Pin  1     2     3     4     5     6     7     8     9     10
       j1    RST   BHE   WR    A0    A17   A16   A15   A14   GND
       j2    S38   S30   S2    S1    S0    LBW   HBW   WAIT  VCC
; Pin  11    12    13    14    15    16    17    18    19    20
equations
S0          = /A17 * /A16 *  RST                ; RAM              0$0000h-0$FFFFh
              + /A17 * /A16 * /RST *  j2        ; (with /RST)      0$0000h-0$FFFFh
/S1         = /A17 * A16                        ; EPROM            1$0000h-1$FFFFh
              + /A17 * /A16 * /RST * /j2 * /j1  ; (with /RST)      0$0000h-0$FFFFh
S2          =  A17 * /A16                       ; RAM/EPROM        2$0000h-2$FFFFh
              + /A17 * /A16 * /RST * /j2 * j1   ; (with /RST)      0$0000h-0$FFFFh
/S30        =  A17 *  A16 * /A15                ; EEPROM           3$0000h-3$8000h
/S38        =  A17 *  A16 *  A15 * /A14         ; Decoder          3$8000h-3$BFFFh
/LBW        = /WR  * /A0                        ; Low Byte Write
/HBW        = /WR  * /BHE                       ; High Byte Write
; Optional
/WAIT       = A17 * A16 * /A15                  ; EEPROM           3$0000h-3$8000h
              + A17 * A16 *  A15 * /A14         ; Decoder          3$8000h-3$BFFFh
```

ing a reset, $\overline{\text{RSTOUT}}$ should drop low, and return to high as soon as the reset condition is ended. If that does not happen, the (monitor) program is simply not executed, and the controller will never reach the EINIT (End Init) instruction. EINIT returns the $\overline{\text{RSTOUT}}$ output to high (measure at IC4 pin 2).

- Address Latch Enable (ALE) signal not available. No matter if the controller contains a fully debugged award-winning program, or just fetches nonsense, memory access will occur every 150 to 850 ns, with the ALE signal being pulled high for about 20 ns (measure at IC9/IC10 pin 11).

- Missing or wrong address bits. Address bits 0-15 are latched on every ALE pulse. The latch components IC9 and IC10 have to be fast enough to recognize the short ALE signal, and faithfully copy the address bits which are available for about 40 ns (measure at IC9/IC10 pins 12-19).

- Missing chip-select signal on EPROMs (measure at IC13/IC14 pin 20). The chip select signals for the memory devices are generated by the system GAL (IC4). The bit combinations shown in **Table 3** are required to for combining into the 'chip select-1' signal for the EPROMs (pin 15 on IC4). The bits should be measured on GAL IC4. During a reset, all lines are at high impedance (i.e., at 'input').

Table 3. CS1 on GAL

After reset	After EINIT
$\overline{\text{RSTOUT}}$ (Pin 2) = 0	$\overline{\text{RSTOUT}}$ (Pin 2) = 1
A17 (Pin 6) = 0	A17 (Pin 6) = 0
A16 (Pin 7) = 0	A16 (Pin 7) = 1
A15 (Pin 8) = 0	A15 (Pin 8) = 0
A14 (Pin 9) = 0	A14 (Pin 9) = 0
$\overline{\text{S1}}$ (Pin 15) = 0	$\overline{\text{S1}}$ (Pin 15) = 0

- No serial connection. Each time MON16X is started, you should be able to measure serial transmission signals on the RxD line (IC1 pin 13/12). The monitor program replies via the TxD line (IC1 pins 11/14). Measure at IC1 pins 13, 12, 11 and 14).

- Simply doesn't work...? Your construction may contain other errors or

Pin functions, extension

Pin	Name	Function
1	GND	Supply ground
2	$\overline{\text{RSTIN}}$	Restart input (Reset)
3		
4	GND	Supply ground
5	+5	+5 V supply
6	$\overline{\text{CS5}}$	Chip select 5: 3A800-3AFFFh
7	$\overline{\text{CS4}}$	Chip select 4: 3A000-3A7FFh
8	$\overline{\text{CS3}}$	Chip select 3: 39800-39FFFh
9	$\overline{\text{CS2}}$	Chip select 2: 39000-397FFh
10	$\overline{\text{CS1}}$	Chip select 1: 38800-38FFFh
11	$\overline{\text{CS0}}$	Chip select 0: 38000-387FFh
12	AD0 / P0.0	Address/data bit 0
13	AD1 / P0.1	Address/data bit 1
14	AD2 / P0.2	Address/data bit 2
15	AD7 / P0.7	Address/data bit 7
16	AD6 / P0.6	Address/data bit 6
17	AD5 / P0.5	Address/data bit 5
18	AD4 / P0.4	Address/data bit 4
19	AD3 / P0.3	Address/data bit 3
20	GND	Supply ground
21	$\overline{\text{LBW}}$	Low Byte Write
22	$\overline{\text{HBW}}$	High Byte Write
23	$\overline{\text{RD}}$	Read
24	A10	Address bit 10
25	AD8 / P0.8	Address/data bit 8
26	AD9 / P0.9	Address/data bit 9
27	AD10 / P0.10	Address/data bit 10
28	AD15 / P0.15	Address/data bit 15
29	AD14 / P0.14	Address/data bit 14
30	AD13 / P0.13	Address/data bit 13
31	AD12 / P0.12	Address/data bit 12
32	AD11 / P0.11	Address/data bit 11

a defective part you have not noticed so far. Further investigation is required.

FREEWARE AND MORE

The hardware of the 80C166 evaluation system should be complemented by a range of software products. The aim of the discussion below is to prevent confusion arising about this particular subject.

Basically, a distinction has to be made between software on the board and software running on the PC.

On the board, the monitor program is permanently stored in two EPROMs. This monitor program was specially developed by the author for the 80C166 board. It is faster, better and less bug-ridden than comparable, non-adapted products. Its only disadvantage is that the EPROM pair reduces the memory range by 64 kBytes. Unfortunately, the 80C166 does not support bootstrap loading, this is only possible on the 80C165 and 80C167. The monitor program is supplied in two ready-programmed EPROMs (Publishers' order codes **996509-1** and **996509-2**). Alternatively, burn your own EPROMs using the hex file found on the project diskette (Publishers' order code **996091-1**).

The GAL chip may also be obtained ready-programmed (order code **996581-1**). However, if you have a programmer for the GAL16V8, you may want to use the GAL equations file listed in **Table 4**.

The MON16X program for use on the PC was not developed by the author. It is supplied as shareware or test software on CD-ROM or through the Internet, by a number of manufacturers of microprocessor-related products, including our advertiser Hitex. See, for example, these web sites:

www.hitex.com
www.keil.com
www.tasking.com
www.ertec.com

With nearly all suppliers, C or C+ is the 'dominant' programming language.

The author uses the Keil assembler/linker/locator (which is not free of charge), in combination with collected software from ertec GmbH, packed in the file **EVAAL.ZIP** (found in the 'download' area on ertec's web site). Actually, this software is intended for ertec's type EVA166 development board, which is compatible with the evaluation system described in this article series. The zipped file contains, among others, these programs which are of special interest to us:

MON16X2.EXE	Windows 95 program
MON16X.EXE	MS-DOS program
MON16X.OVL	overlay for MON16X
MON16X.SCR	script file for things to be automatically done by MON16X or MON16X2
MON16X.CFG	configuration file for launching MON16X or MON16X2
ASS16X.EXE	C166 assembler for DOS
ASS16X2.DLL	C166 assembler for Windows 95

The bootstrap loader has to be disabled in the configuration file by putting semicolons in front of the relevant command lines. The EPROM-resident version of the monitor as developed by the author achieves a baud rate of 38,400 bits/s.

Finally, a vast number of application examples, FAQs and links to other information on the 'C166 micro may be found at

www.hitex.demon.co.uk/c166/miscdocs.html

(990028-2)

Pin functions, controller board

Function		Name	Pin		Pin		Name	Function	
+ 5V	Supply	**+ 5V**	1bc	1	2	1a	**+ 5V**	+ 5V	Supply
Output	SIO2 TxD Mon	P3.8 / TXD1	2bc	3	4	2a	P1.12	Output	SIO2 RTS
Input	SIO2 RxD Mon	P3.9 / RXD1	3bc	5	6	3a	P2.12/CC12IO	Input	SIO2 CTS
GND	SIO2	**GND**	4bc	7	8	4a	GND	GND	SIO1
Output	SIO1 TxD Prog	P3.10 / TXD0	5bc	9	10	5a	P1.11	Output	SIO1 RTS
Input	SIO1 RxD Prog	P3.11 / RXD0	6bc	11	12	6a	P2.11/CC11IO	Input	SIO1 CTS
+ 5V	Serial	+ 5V	7bc	13	14	7a	GND	GND	Serial
*Input/Output Serial Data**		P1.10	8bc	15	16	8a	P2.10 /CC10IO	*Input/Output Serial Clock**	
Input	Non Maskable Int	NMI	9bc	17	18	9a	**RSTIN**	Input	Reset
Output	(20 MHz Clock)	P3.15 / CLK	10bc	19	20	10a	P2.13 / BREQ	Output	(Bus Request)
Output	(hold acknowledge)	P2.14 / HLDA	11bc	21	22	11a	P2.15 / HOLD	Output	(Hold)
Input	(Timer 2 Input)	P3.7 / T2IN	12bc	23	24	12a	P3.6 / T3IN	Input	(Timer 3 Input)
Input	(Timer 4 Input)	P3.5 / T4IN	13bc	25	26	13a	P3.4 / T3EUD	Input	(Timer 3 Up/Dn)
Output	(Timer 3 Output)	P3.3 / T3OUT	14bc	27	28	14a	P3.2 / CAPIN	Input	(Capture Input)
Output	(Timer 6 Output)	P3.1 / T6OUT	15bc	29	30	15a	P3.0 / T0IN	Input	(Timer 0 Input)
Input	(Capt/Comp) 10	P2.9 / CC9IO	16bc	31	32	16a	P2.8 / CC8IO	Input	(Capt/Comp) 9
Input	(Capt/Comp) 8	P2.7 / CC7IO	17bc	33	34	17a	P2.6 / CC6IO	Input	(Capt/Comp) 7
Input	(Capt/Comp) 6	P2.5 / CC5IO	18bc	35	36	18a	P2.4 / CC4IO	Input	(Capt/Comp) 5
Input	(Capt/Comp) 4	P2.3 / CC3IO	19bc	37	38	19a	P2.2 / CC2IO	Input	(Capt/Comp) 3
Input	(Capt/Comp) 2	P2.1 / CC1IO	20bc	39	40	20a	P2.0 / CC0IO	Input	(Capt/Comp) 1
Input.	*(Analogue) 1 **	P5.0 / AN0	21bc	41	42	21a	P5.1 / AN1	Input	*(Analogue) 2 **
Input.	*(Analogue) 3 **	P5.2 / AN2	22bc	43	44	22a	P5.3 / AN3	Input	*(Analogue) 4 **
Input	*(Analogue) 5 **	P5.4 / AN4	23bc	45	46	23a	P5.5 / AN5	Input	*(Analogue) 6 **
Input.	*(Analogue) 7 **	P5.6 / AN6	24bc	47	48	24a	P5.7 / AN7	Input	*(Analogue) 8 **
Input	*(Analogue) 9 **	P5.8 / AN8	25bc	49	50	25a	P5.9 / AN9	Input	*(Analogue) 10 **
Analogue Ground (GND)		VAGND	26bc	51	52	26a	VAREF	Analogue Reference (+ 5V)	
Output	10	P1.9	27bc	53	54	27a	P1.8	Output	9
Output	8	P1.7	28bc	55	56	28a	P1.6	Output	7
Output	6	P1.5	29bc	57	58	29a	P1.4	Output	5
Output	4	P1.3	30bc	59	60	30a	P1.2	Output	3
Output	2	P1.1	31bc	61	62	31a	P1.0	Output	1
GND	Supply	**GND**	32bc	63	64	32a	GND	**GND**	Supply

Evaluation System for 80C166 (parts 1 and 2)

March & April 1999, 990028

Errors in Components List

IC9 and IC10 should be type 74HC573, not HCT573.

C1 and C2 should be 10µF 16 tantalum bead.

R20 should be an 8-way SIL array, value 4kΩ7.

The crystal frequency mentioned with C18, C19 should be 40MHz, not 100MHz.

IC3, MAX690 should be listed as MAX690 (BATT), not MAX690 (I/O)

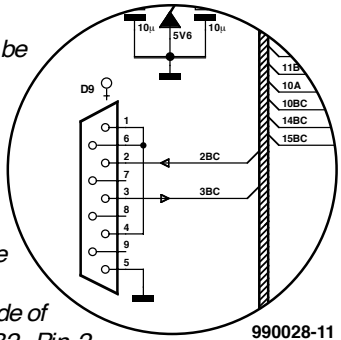
Miscellaneous

Pull-up resistors R1-R4 need not be mounted with the serial channels.

When a 40-MHz crystal is used for X1, it may oscillate at the fundamental frequency (13.333 MHz) instead of the third overtone. If this happens the system baudrate will not be correct and the serial communication will fail to work as described. Suggested solutions are (1) to use a 40-MHz crystal oscillator module, (2) use a different 40-MHz crystal or (3) adapt the value of C18 and C19 until oscillation occurs at the third overtone.

The battery may be a 3.6-V NiCd type, but note that this is very slowly charged by the MAX690. A better solution is to use a Lithium battery.

The circuit diagram does not make it clear how the serial connection is made via D9. Pins 2 and 3 of this connector should not be connected TxD1 and RxD1 (i.e. the TTL side of the MAX232), but to the other (RS232) side of the MAX232. Pin 2 of D9 is therefore connected to pin 2BC via bus K1. Likewise pin 3 of D9 is connected to pin 3BC via bus K1. The correct connections are shown in the illustration.



Development System for 68HC11F1

June 1999, 990042

JP1 is a simple jumper. In the text and parts list, it is erroneously referred to as K3.

Part K3 is a 4-way SIL pinheader. In the text and parts list, it is erroneously referred to as JP1

R4 is erroneously listed as a SIL array with a value of 10kΩ. The PCB however only accommodates eight discrete resistors. These are numbered R4 and R6-R12. Resistor R13 in the circuit diagram equals R4 on the PCB.

PC-Controlled Model Railway: EEDTS Pro

June 1999, 990082-2

On page 60, the text references t S3, D1 and S4 should read S2, D2 and S1 respectively.

EVALUATION SYSTEM FOR 80C166 (PARTS 1 AND 2)
March & April 1999, 990028

Errors in Components List

IC9 and IC10 should be type 74HC573, not HCT573.

C1 and C2 should be 10µF 16 tantalum bead.

R20 should be an 8-way SIL array, value 4kΩ7.

The crystal frequency mentioned with C18, C19 should be 40MHz, not 100MHz.

IC3, MAX690 should be listed as MAX690 (BATT), not MAX690 (I/O)

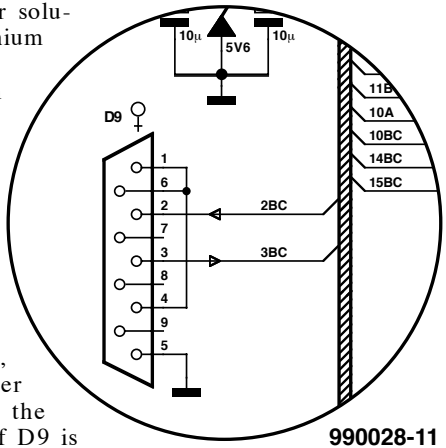
Miscellaneous

Pull-up resistors R1-R4 need not be mounted with the serial channels.

When a 40-MHz crystal is used for X1, it may oscillate at the fundamental frequency (13.333 MHz) instead of the third overtone. If this happens the system baudrate will not be correct and the serial communication will fail to work as described. Suggested solutions are (1) to use a 40-MHz crystal oscillator module, (2) use a different 40-MHz crystal or (3) adapt the value of C18 and C19 until oscillation occurs at the third overtone.

The battery may be a 3.6-V NiCd type, but note that this is very slowly charged by the MAX690. A better solution is to use a Lithium battery.

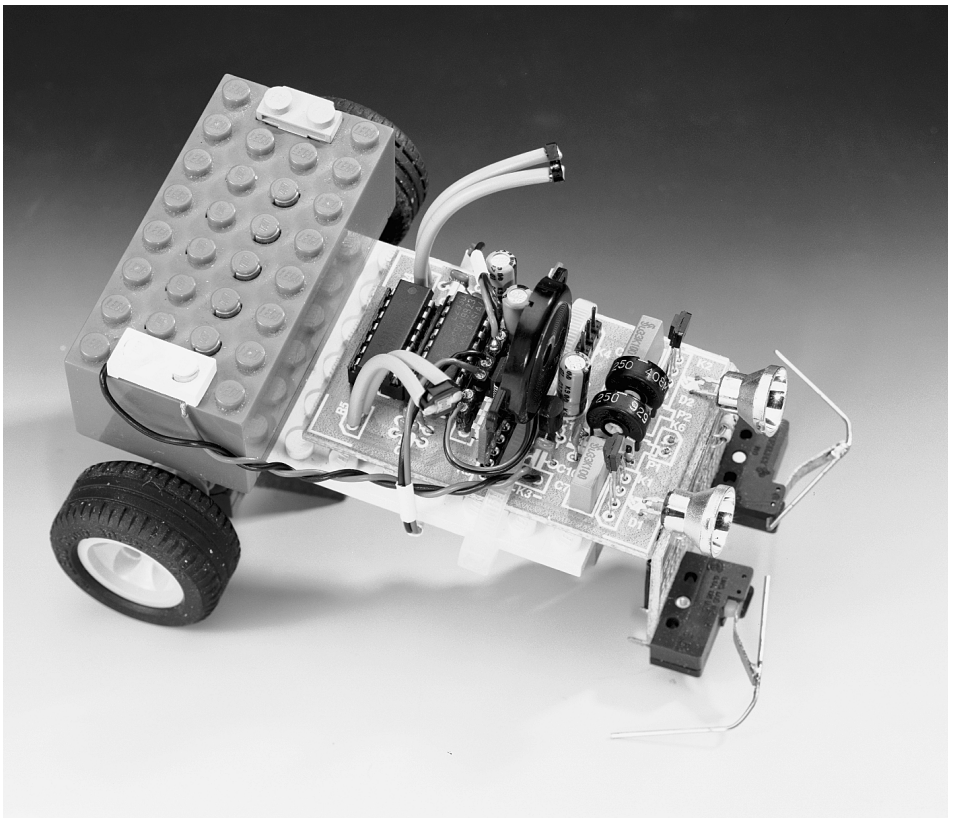
The circuit diagram does not make it clear how the serial connection is made via D9. Pins 2 and 3 of this connector should not be connected TxD1 and RxD1 (i.e. the TTL side of the MAX232), but to the other (RS232) side of the MAX232. Pin 2 of D9 is therefore connected to pin 2BC via bus K1. Likewise pin 3 of D9 is connected to pin 3BC via bus K1. The correct connections are shown in the illustration.



LightFinder

a Stamp-powered Lego vehicle

A compact and programmable little robot vehicle may be built from ingredients like the famous BASIC Stamp, assorted electronic parts, and Lego bricks and motors. The LightFinder described in this article proves that intelligence is the keyword in elementary cybernetics.



Specification

- 9-V battery powered
- Two motors
- Two front/side contact switches
- Two light sensors
- Two infrared light sensors
- Programmable in BASIC
- 256 bytes program code
- Program written on PC

Cybernetic models (electronically powered systems simulating living organisms) were published on several occasions in *Elektor Electronics*. However, in retrospect these systems were complex, slow and bulky. The availability of advanced microcontroller has made it much easier to build an small vehicle doted with a basic amount of artificial intelligence (AI). The LightFinder described in this article is a fine educational project, with a V.1 BASIC Stamp from Parallax acting as the 'brains'. The project also comprises a compact printed circuit board containing all I/O components, two Lego motors with built-in reduction gear, and some Lego building blocks. Throw it all together and you are looking at a miniature robot vehicle capable of finding its way to a light source, or (by reprogramming), avoid a light source, despite obstacles in its path.

Thanks to the use of the BASIC Stamp as the heart of the system, the behaviour of the LightFinder is readily changed to meet your own requirements. No knowledge of source code programming is required, the Stamp module is programmed in BASIC using your PC.

Design by G. Nöcker

STRUCTURE

The block diagram shown in **Figure 1** shows the elementary structure of the LightFinder cybernetic model. The vehicle is powered by two compact dc motors with internal reduction gear. The sensors are a pair of LDRs (light-dependent resistors) for light/dark detection, a pair of infrared sensors for distance measurement, and a pair of switches to detect obstacles in the way.

The LightFinder's intelligence is vested in a ready-made BASIC Stamp Version 1 module from Parallax, Inc. The LightFinder communicates with you via beeping sounds from its on-board piezo-buzzer. The Lego motors are driven by an integrated motor driver IC, which is also shown in the block diagram.

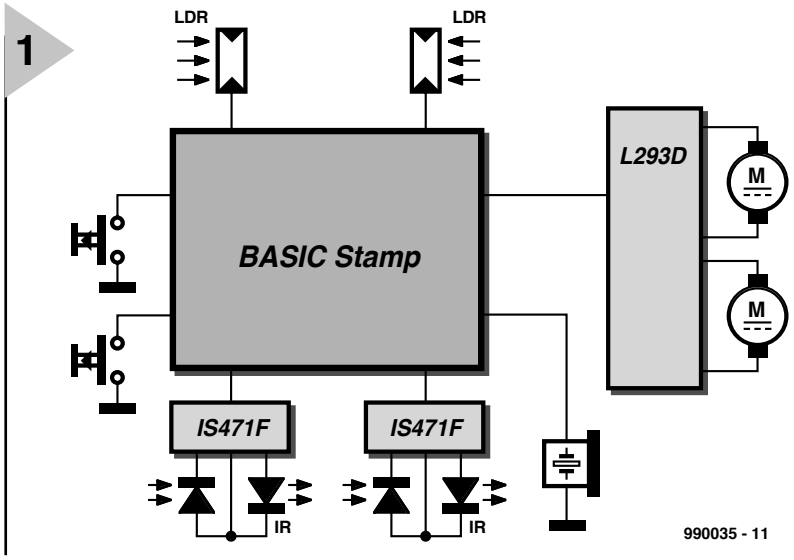


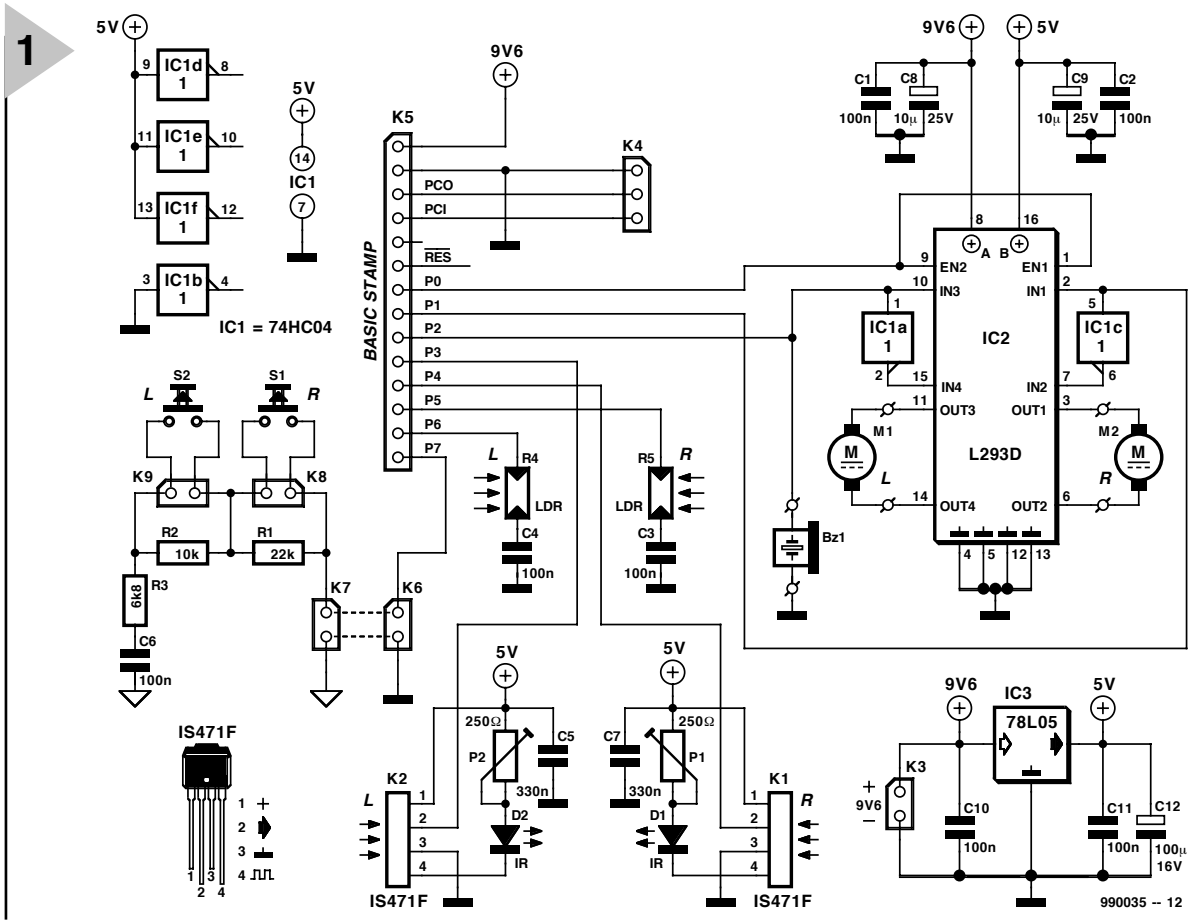
Figure 1. Block diagram of the LightFinder cybernetic model.

The two infra-red sensitive 'eyes' of the LightFinder are pretty advanced electronic circuits capable of detecting objects by illuminating them with a modulated infra-red light source. Depending on the sensitivity set by the user, these 'eyes' are capable of detecting large obstacles at a distance of about 20 cm.

PRACTICAL CIRCUIT

The circuit diagram of the LightFinder's control system is shown in **Figure 2**. The heart of the circuit is formed by the PIC-based BASIC Stamp connected to K5. Because the BASIC Stamp is used as a ready-made module in this project, its operation is not discussed in this article. All electronic circuits that link the processor to the real world is connected to the pins of K5. Because of the processor's rather limited I/O capacity, switches S1 and S2 are connected to an analogue input, PC4, via a voltage divider. The processor is capable of measuring the value of a resistor, potentiometer or LDR by connecting the relevant resistance in series with a capacitor, and then establish the RC time produced by the combination. Because the RC time measured on PC4 will change when S1 is closed (R = 16.8kΩ) or S2 is closed (R = 28.8 kΩ), the system software is capable of detecting whether the left-hand or right-hand bumper switch was closed by

Figure 2. The circuit diagram of the LightFinder's electronics is no more than a collection of sensors. All computing power resides in the BASIC Stamp module.



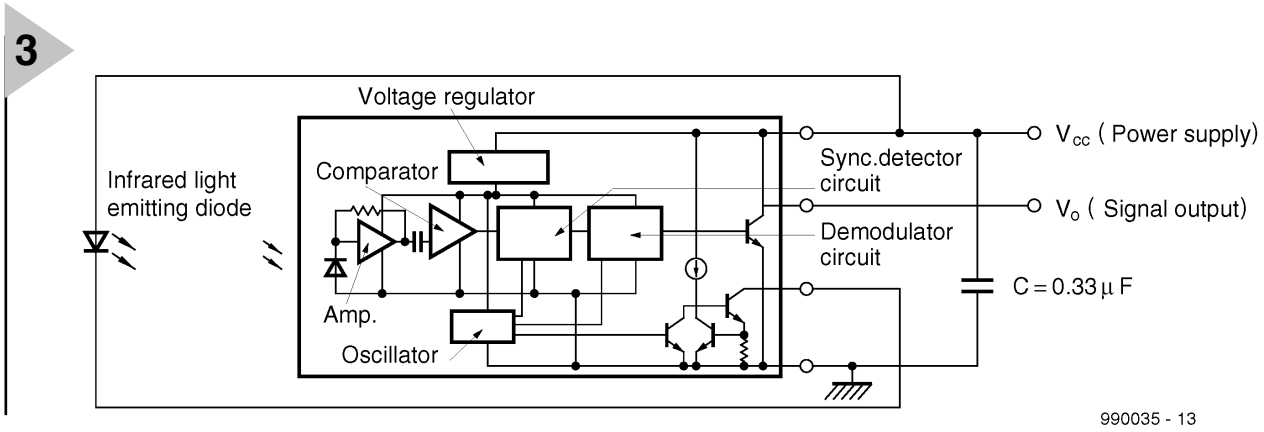


Figure 3. Internal structure of the IS471F from Sharp. Two of these sensors act as the LightFinder's 'eyes'.

the LightFinder running into an obstacle. When both switches are open, the Stamp measures a resistance of 38.8 kΩ. The two LDRs are 'read' in the same way by connecting them in series with a 100-nF capacitor.

INFRARED EYES

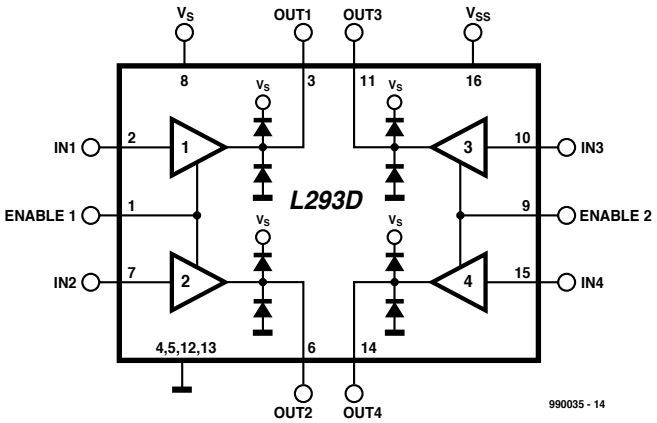
The LightFinder uses an ingenious method to determine its distance from a large obstacle such as a wall. Infrared light is employed for this function, in particular, an IS471F integrated circuit from Sharp. The block diagram of this IC is shown in **Figure 3**. More technical details, in particular, on the opto-electrical characteristics, may be found on this month's *Datasheets*. The IS471F contains a complete modulator, demodulator, voltage regulator and an oscillator. The latter is used to drive an external infrared emitting diode (IRED) via a buffer. The light emitted by the IRED is detected again by the on-chip receiver diode. The resulting signal is first amplified and then converted into a digital pulse-shaped signal by means of a comparator. The integrated synchronisation detector and demodulator circuit then compare the received signal with the pulse signal applied to the sender diode. If the two signals are equal, the digital output of the IC (pin 2) is enabled. By adapting the resistance in series with the IRED, the 'transmit power' may be increased or decreased. In this way, the optical sensitivity may be controlled within certain limits. The configuration used in the circuit allows a large object at a distance of about 20 cm to be detected without problems.

The circuit may be powered by voltages between 4.5 V and 16 V, so that battery voltages of 9 V or 12 V are good options. The pulse repeat time is typically 130 (s (approx. 8 kHz), and the pulse on-time is 8 (s, so that the battery is used economically. The LightFinder has two infra-red eyes so that objects at either side of the vehicle may be detected.

The buzzer is a simple piezo-electric beeper. It is controlled via port line P2 of the BASIC Stamp. The tone frequency is programmable over a wide range: approx. 95 Hz to over 10 kHz.

The integrated motor driver, a Type L293 from SGS-Thomson (and second sources) allows the hardware to be reduced considerably. As shown by the internal diagram in **Figure 4**, the IC contains four power drivers. These are capable of supplying a continuous direct current of 600 mA or 1.2 A peak. On-chip flyback diodes protect the IC outputs against voltage surges caused by the switching of inductive loads (such as the motor coils). Because the inputs of the L293 are TTL compatible, they can be directly linked to the microcontroller outputs. An on-chip thermal protection circuit disables the buffer if an overload condition arises.

Figure 4. A special IC, the L293D, is used to drive the motors.



The two motors of the LightFinder, M1 and M2, are totally controlled by just three processor control lines. Port lines P0, P1 and P2 form the link between the controller and the motor driver. P0 carries the common enable signal for the Lego motors. If the enable line is inactive (low), the power drivers in the L293 are switched off, or, more precisely, their outputs are switched to high-impedance. In this state the current consumption is minimized.

Port lines P1 and P2 convey the motor direction information. Because the buffers are set up in a bridge configuration, an inverter is included between the two drivers assigned to one motor. This is done to make sure that the two motor terminals are always polarized or, of course, at 0 V.

The power supply has been kept as simple as possible, with all parts capable of working at 9 V connected directly to the battery. The motor driver and the BASIC Stamp can only work at 5 V, and they receive their supply voltage from regulator IC3. Attentive readers may have noticed that the circuit diagram indicates a supply voltage of 9.6 V. The reason is that this voltage is supplied by some rechargeable batteries that can be used as substitutes for 9-V blocks. Rechargeable batteries were successfully used during the entire development period of the LightFinder, with excellent results, and to stay keep our governors happy as far as budgets are concerned! In use, the circuit draws more than 100 mA when the vehicle is on the move.

ELECTRONIC CONSTRUCTION

The copper track layout and component overlay of the PCB designed for

COMPONENTS LIST

Resistors:
R1 = 22kΩ
R2 = 10kΩ
R3 = 6kΩ8
R4,R5 = LDR
P1,P2 = 250Ω preset, vertical

Capacitors:
C1-C4,C6,C10,C11 = 100nF, raster 5mm
C5,C7 = 330nF
C8,C9 = 10μF 25V radial
C12 = 100μF 16V radial

Semiconductors:
D1,D2 = IRED, e.g. LD271
IC1 = 74HC04
IC2 = L293D
IC3 = 78L05

Miscellaneous:
Bz1 = piezo buzzer (passive)
K1,K2 = IS471F (Sharp)
K3,K6-K9 = 2-way SIL socket
K4 = 3-way SIL header
K5 = 14-way SIL socket
M1,M2 = motor, see inset on Lego parts
S1,S2 = microswitch, 1 make contact
BASIC Stamp, V. 1 (Milford Instruments)
PCB, order code 990035-1, see Readers Services page

5

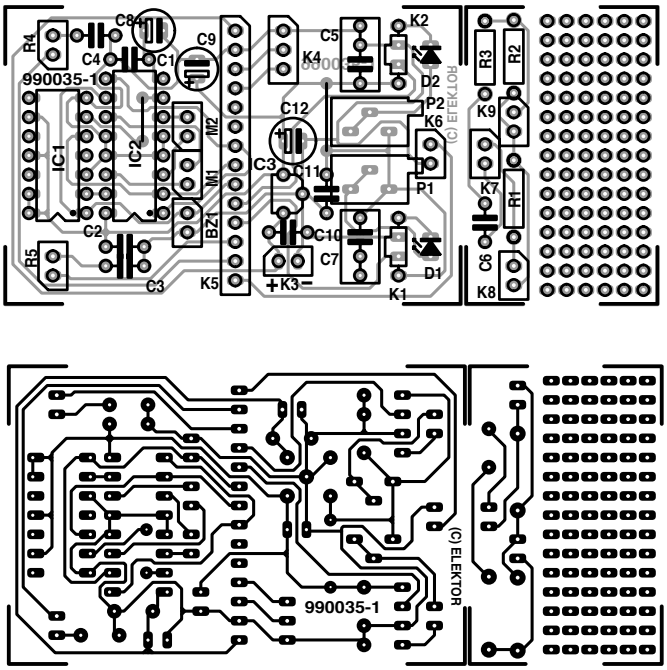


Figure 5. Copper track layout and component mounting plan of the LightFinder PCB (board available ready-made).

the LightFinder may be found in **Figure 5**. This board is available ready-made through the Publishers' Readers Services. It is plain to see that his board has to be cut in two, and we recommend doing this first. The two bumper switches S1 and S2 are fitted on the board with holes only. The photograph of the prototype shows the practical realization. Two pieces of metal wire are used to make the actual 'bumpers' that serve to extend the contact range of the switches. The wires are cut and bent such that they extend a little beyond the wheels. In this way, the bumpers allow obstacles beside the vehicle to be detected also.

Using two pieces of massive wire, the board holding the switches is fitted at right angles on to the main board.

The BASIC Stamp is inserted in SIL socket K5 — the component side points in the direction of IC2. The Stamp computer may be programmed later using the Parallax programming cable hooked up to the PC and 3-way SIL header K4.

Pay attention when mounting the sensors on to K1 and K2. As indicated by the component overlay, the protruding part of the case should point to the rear of the vehicle. The infrared light produced by the associated IRED (D1 or D2) then reaches the sensor face via the reflecting object. The behaviour of the IR sensors may be fine tuned as follows: with the motors disconnected, set the IRED power presets (P1, P2) to maximum resistance. Then measure the voltage at the sensor output (pin 3). Next, place a fairly large obstacle at the desired distance from the sensor, and turn the relevant preset (P1, P2), until the output goes low. Turn the sensors sideways a little — they should not look straight ahead, and horizontally they should remain in line with the associated IRED.

MECHANICAL CONSTRUCTION

The prototype of the LightFinder was built mainly from Lego bricks and other parts. The vehicle is powered by two compact 9-volt dc. motors with internal gear reduction boxes. The 'floor plate', nose wheel and battery

Lego bricks & other parts

The version of the LightFinder as shown on this month's front cover was built from standard Lego bricks and parts. Because some of the special parts may be difficult to find individually, an overview is given of the relevant order codes. Further information may be found on Lego's wonderful website at www.lego.com.

Micro motor:	*5119*
Small low-profile tyres:	*5268*
Nose wheel:	*5050*
Hinge:	*5388*
Cross axles:	*5226*
Plates with wires:	*5311*
9V battery box:	*5391*

Items 5050, 5388 and 5226 are part of the Plane Accessories Set.
It should be noted that two motors and two cable sets have to be ordered. In addition to these components you will also require a base plate and a few standard bricks. Anyone who has ever worked with Lego should have these available.
For a copy of the Lego Service brochure, or information on Mail Order Service for Lego sets, contact

Consumer Service
LEGO UK Limited
Ruthin Road
Wrexham LL13 7TQ
Main Switchboard Number: (01978) 290900

box are also from the Lego 'Technic' series. This month's cover photograph shows the result of one evening of 'vehicle building for the young'. If you want to copy our construction, the relevant parts descriptions and Lego order numbers may be found in the inset.

Users of Meccano, Knex or Fischer Technic parts may easily produce similar constructions of the vehicle, the general structure being uncritical. If the vehicle is made much larger or from different parts (metal), it will probably become so heavy that you have to resort to six (rechargeable) AA-size batteries instead of the 9-V PP3 (6F22) block. Non-rechargeable AA batteries have a much larger capacity than a 9-V PP3 (up to 1500 mAh), and they will typically last longer.

Other motors may be used, as long as their current consumption does not exceed the 600-mA limit imposed by the power drivers in the L293.

Special connecting cables are available for the Lego motors. If these are cut, you have two connecting cables, which may be soldered in the designated sockets (M1 and M2).

The wires of the LDRs will be long enough to give these devices a far view when positioned high on the vehicle. Be sure to isolate the wires, though, and bend them carefully only if you need to adjust the viewing direction. The choice of the LDRs is not particularly critical. If necessary, the 'Alt-P' option in the control software (Stamp.exe) may be used to adapt the sensitivity to the LDR characteristics (see the Stamp manual for further details).

SOFTWARE

The BASIC control program developed for our prototype of the LightFinder is listed in **Figure 6**. This listing is intended to get you started. Because the program is not too extensive, it is easily typed into the Editor utility of the Stamp.exe program.

The BASIC Stamp kit comes with programming tools, a programming cable, a manual and a carrier board. This kit may be ordered directly from Parallax, or through authorized distributors. Readers in the UK should contact Milford Instruments, South Milford, Leeds, tel. (01977) 683665, fax (01977) 681465, email *info@milinst.demon.co.uk*, web site at *www.milinst.demon.co.uk*.

There is a cheaper alternative to the complete kit, however: The Parallax web site at *www.parallaxinc.com* allows you to download the complete Stamp manual as a 'pdf' (Adobe Acrobat Reader) file, as well as the development software. If you feel you are able to make your own programming cable, then all you have to order is one BASIC

6

Figure 6. Listing of the BASIC program that makes the LightFinder come alive.

```
'-----------------------------------------------------------
' Program to control a LEGO-robot (type NANOBOT)
'
' NANO_BOT.BAS 08.03.98
' Author Gerhard Nöcker
' 1998 Copyright Elektor, Beek (L), The Netherlands
'-----------------------------------------------------------

'----- Port Pin Assignment ---------------------------------
'Port                          Function              Type
'Pin 0 Motor current           0/1 == on/off         DigOut
'Pin 1 Right motor             0/1 == back/fw        DigOut
'Pin 2 Left motor (buzzer)     0/1 == back/fw        DigOut (sound)
'Pin 7 Switch right/left                             AnIn
'Pin 4 Right IR obstacle det.  0/1 == obstacle/free  DigIn
'Pin 5 Right LDR                                     AnIn
'Pin 6 Left LDR                                      AnIn
'Pin 3 Left IR obstacle det.   0/1 == obstacle/free  DigIn
'----- Remark ----------------------------------------------
'Pin 2 is also used to produce sounds. Both motors will be halted.
'The direction bit of the left motor controls the buzzer.
'There are four different sounds:
'Obstacle on the right side:   high/low
'Obstacle on the left side:    low/high
'Too much light:               high
'Too dark:                     low

'----- Declaration of constants ----------------------------
symbol sw_pin = 7       'switch input at port pin 7
symbol sw_scale = 33    'switch scale factor
symbol ri_ldr = 5
symbol ri_ldr_sc = 82
symbol le_ldr = 6
symbol le_ldr_sc = 82
symbol l_thres = 15
symbol halt = 0
symbol fwd = 7
symbol backw = 1
symbol Right = 5
symbol Left = 3

'----- Declaration of variables ----------------------------
symbol r_sw = bit0              'storage for right switch (on/off)
symbol l_sw = bit1              'storage for left switch (on/off)
symbol f_flag = bit2    'NOT USED
symbol b_flag = bit3    '
symbol l_flag = bit4
symbol sw_val = b4       'storage for analog switch value
symbol lw_re = b5
symbol lw_li = b6
symbol duration = w4
symbol Rnd = w5

'----- Program start ---------------------------------------
        dirs=%00000111     'port bit 7..3 input, bit 2..0 output
        r_sw = 0
        l_sw = 0

'
' If a switch is opened/closed at power up the robot will
' search/avoid light.
'
         pot sw_pin, sw_scale, sw_val           'read switches
         b_flag = 1
         If sw_val < 240 Then lbl1
         goto loop
lbl1:    b_flag = 0
         l_flag = 0

'----- Program main loop -----------------------------------
Loop:    pot ri_ldr, ri_ldr_sc, lw_re          'read right LDR
         pot le_ldr, le_ldr_sc, lw_li          'read left LDR
```

```
        lw_re=lw_re min l_thres
        lw_li=lw_li min l_thres
        b8 = lw_li - l_thres
        b9 = lw_re - l_thres
        If lw_re < 200 Then clearfl
cont0:
        If lw_re < b8 And b_flag = 0 Then turn_l 'compare light
        If lw_li < b9 And b_flag = 0 Then turn_r 'and
        If lw_re < b8 And b_flag = 1 Then turn_r 'behaviour
        If lw_li < b9 And b_flag = 1 Then turn_l
        If lw_re = l_thres And lw_li = l_thres Then turn 'too much light
        If lw_re = 255 And lw_li = 255 And l_flag = 0 Then wait 'too dark

forw:   pins = fwd
        GoTo switch

turn_r: pins = Right
        goto loop

turn_l: pins = Left
        goto loop

switch: 'switches and IR obstacle detection
        pot sw_pin, sw_scale, sw_val    'read switches
        If sw_val > 180 And sw_val < 240 Or pin3 = 0 Then sl
        If sw_val < 180 Or pin4 = 0 Then sr
        If sw_val > 240 Then Loop

sl:     r_sw = 0
        l_sw = 1
        GoTo cont1
sr:     r_sw = 1
        l_sw = 0

cont1:
        pins = halt
        pause 1000
        pins = backw
        pause 1500
        pins = halt
        If l_sw = 1 Then cont2
        sound 2,(123,40,110,40)
        pins = Left
        GoTo cont3
cont2:
        sound 2,(110,40,123,40)
        pins = Right

cont3:
        random Rnd
        duration = 1000 + b10
        pause duration
        pins = halt
        pause 1000
        goto loop

turn:
        pins = halt
        sound 2,(125,30)
        pins = Right
        duration = 2800
        pause duration
        pins = halt
        pins = fwd
        pause 3000
        GoTo switch

wait:
        pins = halt
        sound 2,(80,30)
        pause 10000
        l_flag = 1
        GoTo switch

clearfl:
        l_flag = 0
        GoTo cont0
```

Stamp module Version 1.

The BASIC Stamp is ready for programming once the control program has been typed in using Stamp.exe, the programming cable is connected at both ends (LightFinder and PC), and the LightFinder is switched on. Note that Stamp.exe is a DOS program whose Upload function will only work properly in 'real DOS' mode (i.e., not in a Windows DOS box).

At this point give your cybernetic model a test spin — see if it can find a light source! If you are not satisfied with its behaviour, the control program may be modified to make the model respond differently to the various signals it is capable of detecting. If you have ideas for improvements, let us know!

(990035-1)

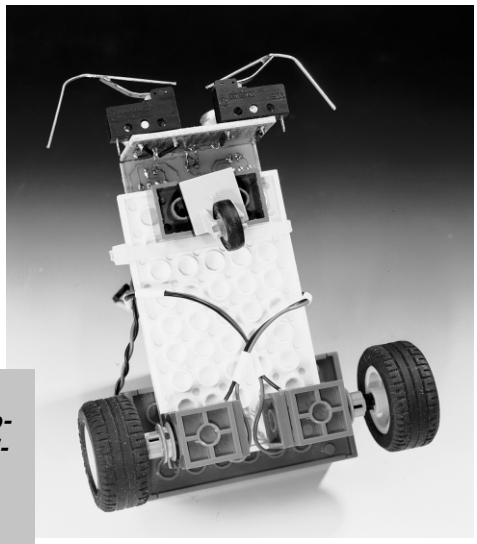
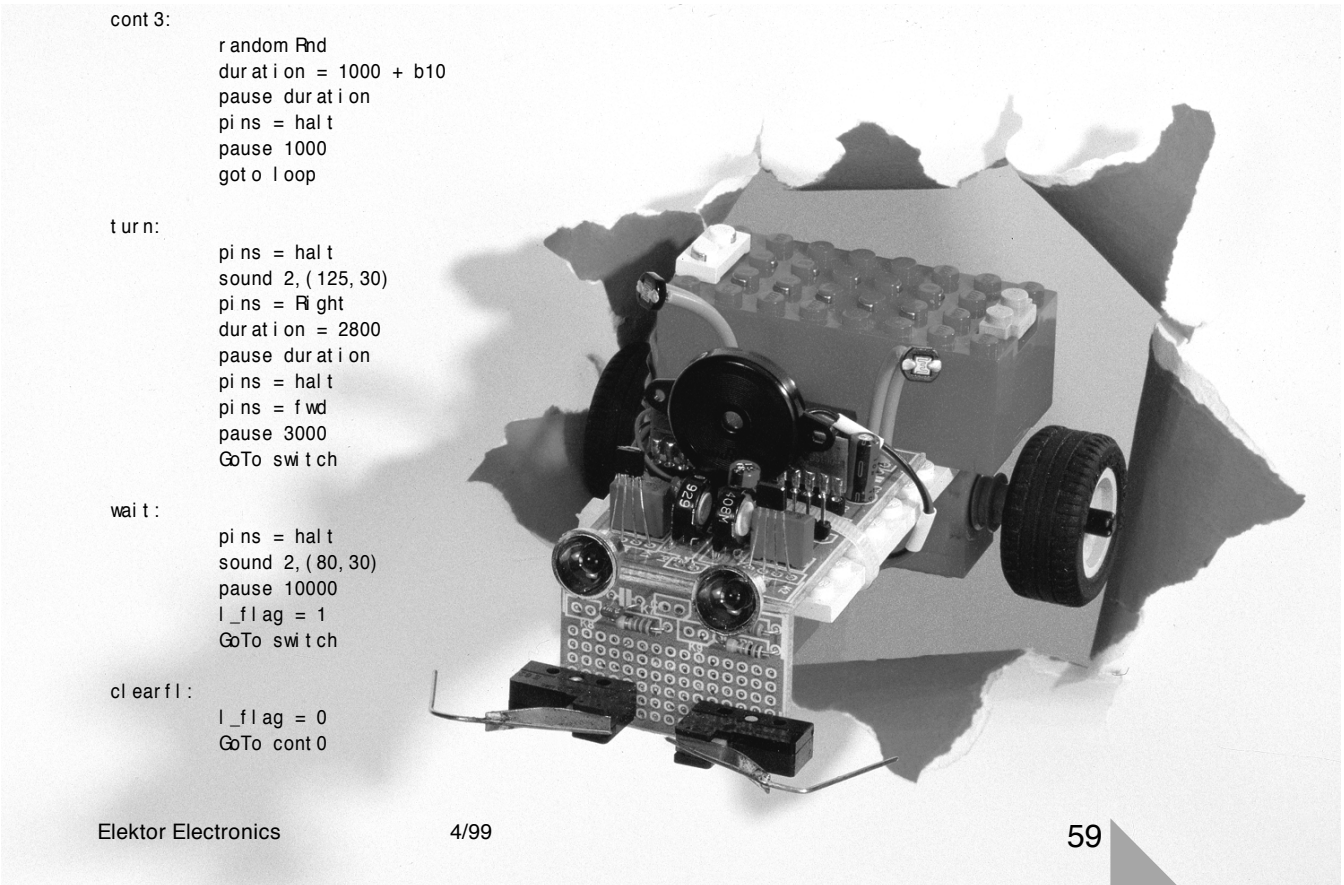


Figure 7. Photograph of our prototype. The mechanical construction consists of Lego bricks and parts.



IS471F

ELEKTOR ELECTRONICS

Sensors

DATASHEET 4/99

IS471F
OPIC™ Light Detector with Built-in Signal Processing Circuit for Light Modulation System

Manufacturer
Sharp Corp.

SHARP

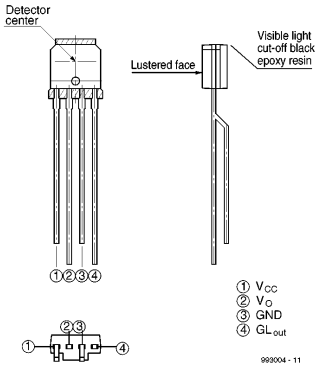
Features
- Impervious to external disturbing lights due to light modulation system
- Built-in pulse driver circuit and sync. detector circuit on the emitter side
- A wide range of operating supply voltage (V_{cc}: 4.5 to 16V)

Applications
- Optoelectronic switches
- Copiers, printers
- Facsimile equipment

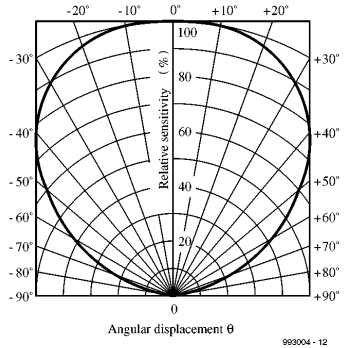
Application Example
LightFinder, *Elektor Electronics* April 1999

Absolute Maximum Ratings (T_a = 25°C)

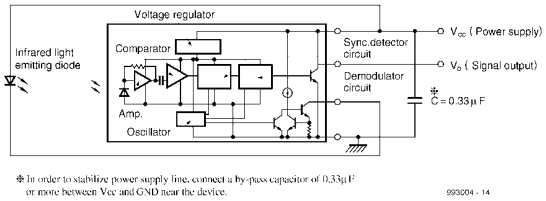
Parameter		Symbol	Rating	Unit
Supply voltage		V_{CC}	–0.5 to 16	V
Output	Output voltage	V_O	16	V
	Output current	I_O	50	mA
GL output	Output voltage	V_{GL}	16	V
Power dissipation		P	250	mW
Operating temperature		T_{OPR}	–25 to + 60	°C
Storage temperature		T_{STG}	–40 to + 100	°C
Soldering temperature		T_{SOL}	260	°C



OPIC (Optical IC) is a trademark of the Sharp Corporation. An OPIC consists of a light-detecting element and signal-processing circuit integrated onto a single chip.



Sensitivity diagram (T_a = 25°C)



Basic circuit

MAX192

ELEKTOR ELECTRONICS

Integrated Circuits
A-D Converters

DATASHEET 4/99

MAX192
Low-power, 8-channel, Serial 10-bit ADC

Manufacturer
Maxim Integrated Products (UK) Ltd., Unit 3, Theale Technology Centre, Station Road, Theale, Berks RG74XX. Internet: www.maxim-ic.com

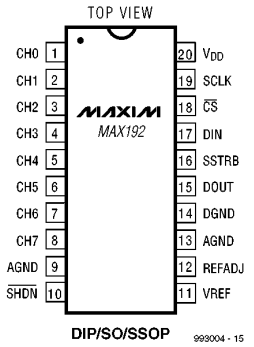
MAXIM

Applications
- Automotive
- Pen-entry Systems
- Consumer Electronics
- Portable Data Logging
- Robotics
- Battery-Powered Instruments, Battery Management
- Medical Instruments

Features
- 8-Channel Single-Ended or 4-Channel Differential Inputs
- Single + 5V Operation
- Low Power: 1.5mA (operating) 2μA (power-down)
- Internal Track/Hold, 133kHz Sampling Rate
- Internal 4.096V Reference
- 4-Wire Serial Interface is Compatible with SPI, QSPI, Microwire, and TMS320
- 20-Pin DIP, SO, SSOP Packages
- Pin-Compatible 12-Bit Upgrade (MAX186/MAX188)

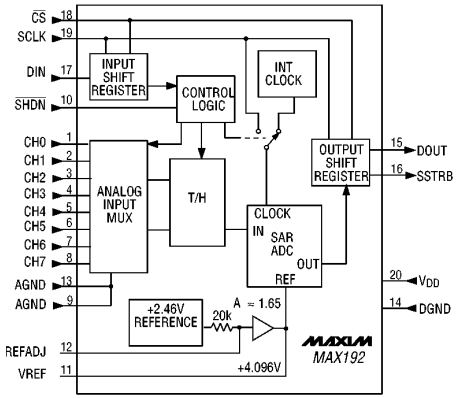
Application Example
Video DVM, *Elektor Electronics* May 1999.

General Description
The MAX192 is a low-cost, 10-bit data-acquisition system that combines an 8-channel multiplexer, high-bandwidth track/hold, and serial interface with high conversion speed and ultra-low power consumption. The device operates with a single + 5V supply. The analog inputs are software configurable for single-ended and differential (unipolar/bipolar) operation. The 4-wire serial interface connects directly to SPI™, QSPI™, and Microwire™ devices, without using external logic. A serial strobe output allows direct connection to TMS320 family digital signal processors.
The MAX192 uses either the internal clock or an external serial-interface clock to perform successive approximation A/D conversions. The serial interface can operate beyond 4MHz when the internal clock is used. The MAX192 has an internal 4.096V reference with a drift of ± 30ppm typical.
A reference-buffer amplifier simplifies gain trim and two sub-LSBs reduce quantization errors. The MAX192 provides a hardwired $\overline{SHDN}$ pin and two software-selectable power-down modes. Accessing the serial interface automatically powers up the device, and the quick turn-on time allows the MAX192 to be shut down between conversions. By powering down between conversions, supply current can be cut to under 10μA at reduced sampling rates.
The MAX192 is available in 20-pin DIP and SO packages, and in a shrink-small-outline package (SSOP) that occupies 30% less area than an 8-pin DIP. The data format provides hardware and software compatibility with the MAX186/MAX188. For anti-aliasing filters, consult the data sheets for the MAX291–MAX297.

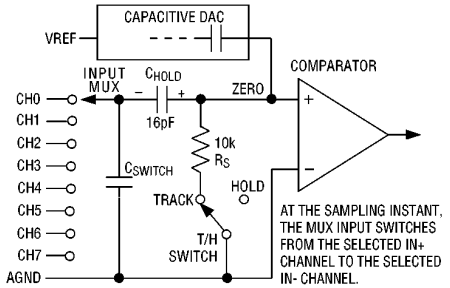


Pin Description

Pin	Name	Function
1–8	CH0–CH7	Sampling Analog Inputs
9, 13	AGND	Analog Ground. Also IN– Input for single-enabled conversions. Connect both AGND pins to analog ground.
10	$\overline{SHDN}$	Three-Level Shutdown Input. Pulling $\overline{SHDN}$ low shuts the MAX192 down to 10µA (max) supply current, otherwise the MAX192 is fully operational. Pulling SHDN high puts the reference-buffer amplifier in internal compensation mode. Letting $\overline{SHDN}$ float puts the reference-buffer amplifier in external compensation mode.
11	VREF	Reference Voltage for analog-to-digital conversion. Also, Output of the Reference Buffer Amplifier. Add a 4.7µF capacitor to ground when using external compensation mode. Also functions as an input when used with a precision external reference.
12	REFADJ	Reference-Buffer Amplifier Input. To disable the reference-buffer amplifier, tie REFADJ to VDD.
14	DGND	Digital Ground
15	DOUT	Serial Data Output. Data is clocked out at the falling edge of SCLK. High impedance when $\overline{CS}$ is high.
16	SSTRB	Serial Strobe Output. In internal clock mode, SSTRB goes low when the MAX192 begins the A/D conversion and goes high when the conversion is done. In external clock mode, SSTRB pulses high for one clock period before the MSB decision. SSTRB is high impedance when $\overline{CS}$ is high (external mode).
17	DIN	Serial Data Input. Data is clocked in at the rising edge of SCLK.
18	$\overline{CS}$	Active-Low Chip Select. Data will not be clocked into DIN unless $\overline{CS}$ is low. When $\overline{CS}$ is high, DOUT is high impedance.
19	SCLK	Serial Clock Input. Clocks data in and out of serial interface. In external clock mode, SCLK also sets the conversion speed. (Duty cycle must be 45% to 55%)
20	VDD	Positive Supply Voltage, + 5V ± 5%



Block Diagram.

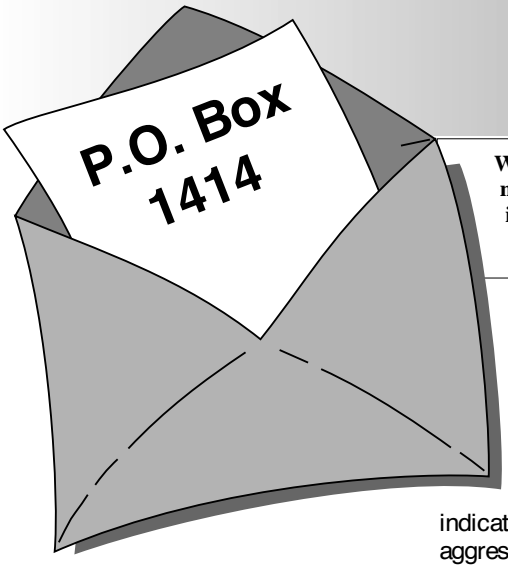


Equivalent Input Circuit.

Electro-optical characteristics (V_{CC} = 5V, T_a = 25°C)

Parameter		Symbol	Conditions	Min.	Typ.	Max.	Unit
Operating supply voltage		V_{CC}	-	4.5	-	16	V
Supply current		I_{CC}	V_O OL terminals shall be opened	-	3.5	7.0	mA
OL output	Output	Low level output voltage	V_{OL}	I_{OL} = 16mA, E_{VP} = 500lx, E_{VD} = 0	-	0.15	0.35	V
		High level output voltage	V_{OH}	E_{VD} = E_{VP} = 0	4.97	-	-	V
		Output short circuit current	I_{OS}	E_{VP} = E_{VD} = 0	0.25	0.5	1.0	mA
	Low level output current		I_{OL}	V_{OL} = 1.2V	40	55	70	mA
	Pulse cycle		t_p	-	70	130	220	µs
	Pulse width		t_w	-	4.4	8	13.7	µs
Low-to-high threshold irradiance			$E_{eP}LH$	E_{eD} = 0, light emitting diode (λ_p = 940nm)	-	0.4	2.66	µWmm^2
High-to-low threshold irradiance			$E_{eP}HL$		-	0.7	2.8	µWmm^2
Hysteresis			$E_{eP}LH/E_{eP}HL$		0.45	0.65	0.95	-
Response time	High-to-low propagation delay time		t_{PHL}		-	400	670	ms
	Low-to-high propagation delay time		t_{PLH}		-	400	670	ms
External disturbing light illuminance			E_{VDX}	E_{eP} = 7.5µWmm^2, λ_p = 940 nm	2000	7500	-	lx

E_{eP} represents illuminance of signal light in sync with the low level timing of output at OLout terminal.
E_{eD} represents illuminance of DC light. Light source: LED (λ_p = 940 nm).
E_{VP} represents illuminance of signal light in sync with the low level timing of output at OLout terminal.
E_{VD} represents illuminance of DC light. Light source: OE standard light source A.



We can only answer questions or remarks of general interest to our readers, concerning projects not older than two years and published in *Elektor Electronics*. In view of the amount of post received, it is not possible to answer all letters, and we are unable to respond to individual wishes and requests for modifications to, or additional information about, *Elektor Electronics* projects.

Clipping and the CD

Dear Editor—As someone who makes a living creating CD masters for record labels, and an equipment designer also, I have to respond to the article by Mr Giesberts ('Clipping Indicator' – Elektor Electronics, October 1998 – which also appeared in Audio Electronics – January 1999).

The timing could not have been better since I was working on an album that contains tracks mastered both here and at perhaps the best-known mastering facility on the West Coast. I was in the process of examining peak levels. This topic also relates to a page called 'TechNews' that I'm about to open on my website, www.drtmastering.com

For major and most independent record label releases, it is mastering facilities, not recording studios, that are responsible for setting the levels that you find on a CD. This is a sector of the industry that is not well known outside the business itself, and there are some common practices worth mentioning.

It is very common to have clipping on CDs, in all music styles, but especially in Pop. This takes two forms:

1. Actual clipping the waveform at the maximum positive or negative value for two samples or more.

A little history. The Sony PCM1610 and 1630 processors, writing a video signal to a 3/42 Umatic tape deck, produced the vast majority of master tapes for CD production through the 1980s and early 1990s. These had a clipping or 'Over' indicator that would light up at one of three presettable levels: 4, 8 or 16 consecutive samples at the positive or negative rail. The default setting that most professionals used was 8 samples. This means that you could clip the signal for 7 samples and no 'Over' would be indicated. Folks into more aggressively hot music could set the limit at 15 samples. Depending on the program type, and what compression was used just below the 'Over' threshold, Sony and most others determined that clipping on this scale was not a problem for most consumers. So, like it or not, there has been clipping on CDs from the start. If it was always restricted to below 8 samples, or even 16, that would be the good news …

(I have to comment here on the editor's parenthetical remark. It is precisely because of commercial acumen that label executives and producers demand that their material be recorded as hot as possible onto a CD. There is an unfortunate tendency in the industry to want your CD louder than the rest, because it gets people's attention, at least for a short while. Any mastering facility or studio that refuses to record a hot level for a client who demands it, and (trust me) many do, will lose that client.)

And now, the bad news …

2. Clipping below maximum possible level.

Mastering engineers who are asked to cut a hot level (and who want to keep their jobs) do the sensible thing: they insert a limiter ahead of the CD burner that will restrict the maximum level to perhaps just a sample or two below the rails. Since there are roughly 65,536 discrete levels that can be represented by a 16-bit number, throwing away two or four is not a big deal. Dynamic range is not affected noticeably, the CD replication plant will accept the master as having no 'Overs', and will cut the disc without further signal processing. (The fact that these plants often do processing is yet another can of worms.)

Since the disc is technically within specification, the clients are free to insist on even hotter levels for their next album. This means that more of the disc will be clipping, but now at a 'safe' level just below the rails. It is not uncommon to see MANY dozens, even HUNDREDs, of consecutive samples clipping in this way, depending on the music style.

The only silver lining is that if you apply the proper type and amount of compression and limiting, so that the transition into (moderate) clipping is smooth, many people actually enjoy the sounds of the artefacts that result. Harmonics are generated, the sound gets momentarily brighter in character, and people think they are hearing clarity, rather than distortion. (Consequently, getting this transition zone right – the upper few dBs of dynamic range – is where I continue to spend a large amount of time when designing mastering gear. In EE and AE tradition, it is possible to build equipment that outperforms anything you can buy.)

This is big business for hardware manufacturers. They sell hard limiters, soft limiters, tube and solid state, compressors with hard and soft knees (transition zones). One manufacturer of A/D converters used on tens of thousands of CDs builds a 'soft clipper' right into the converter, with the express purpose of making the music sound louder.

You can also buy multiband digital processors or software for your PC which will anticipate when the signal is about to clip, and reduce the level before the clipping occurs. Radical er… waveshaping (don't call it distortion – a bad thing) may be required to do this, but the result is that you can get crushing average levels onto the CD.

Does this sound good? ummmm… wellllll… it's loud. What to do about all this? Demand more conservative levels? Don't buy CDs with clipping on them? Change the label's practices? Perhaps all of these.

Monitoring the signal with an oscilloscope is very informative. If readers are interested, I can suggest a very simple circuit ($US5.00 in Radio Shack parts) that you can insert ahead of the scope, doubling the vertical resolution and make it much easier to read. I use it as one of my visual monitors when working on CD masters.

David Torrey
DRT Mastering
001 603 924 2277

Thanks for opening our (and our readers') eyes to yet another aspect of commercial practice that is not given general publicity.
Any comments to this letter should be addressed to the Editor, Elektor Electronics, PO Box 1414, Dorchester, England DT2 8YH.

CTO: the world's first message switch

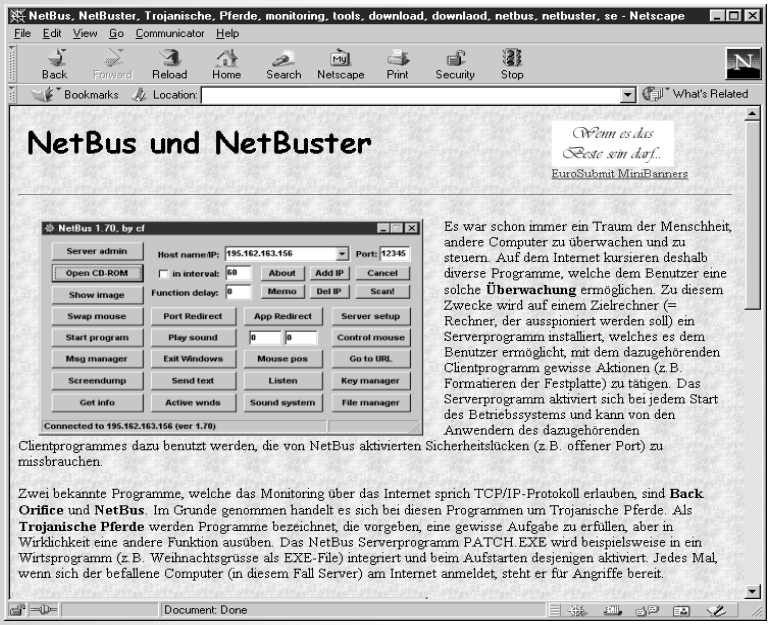
Dear Editor—The otherwise excellent article by Gregg Grant (Elektor Electronics, February 1999) is marred by his discussion on loading coils (page 14). Twisted pair losses are at a minimum when LG= RC. This equation is not satisfied on underground telephone cables. Thus, loading coil (series inductance) were added in an attempt to make the equation balance. Typically, 22 mH was added every 2000 yards, and such loaded cables were in universal use in the audio trunk and junction network for at least 50 years.

However, lumped loading causes the twisted pair to act as a low-pass filter with a cut-off frequency around 3900 Hz. So, rather than combat high-frequency losses as suggested by the author, they had just the opposite effect!

Chris Wood, Oxted, Surrey

Mr Grant replies: "Mr Wood is entirely correct: my thanks to him for pointing out my oversight. Also, apologies to readers I have unwittingly misled.".

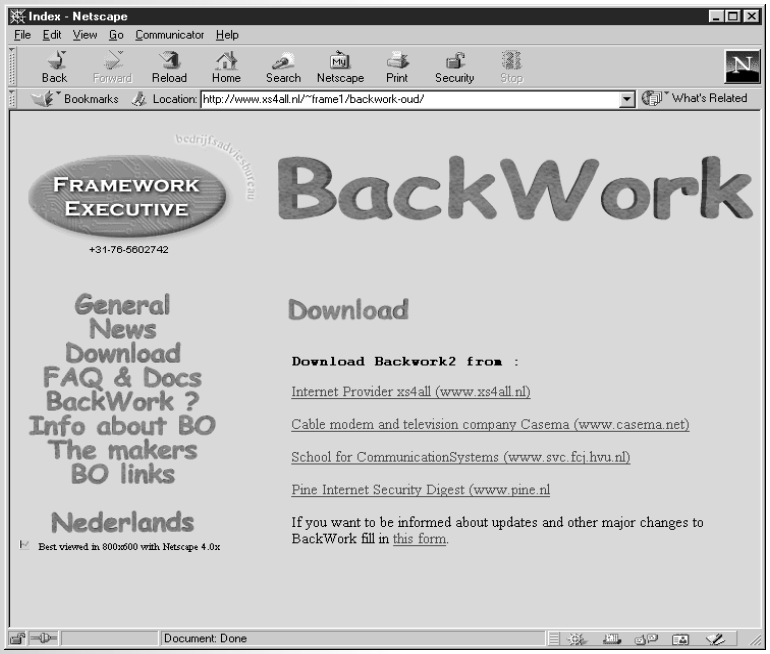
Trojan Horses strike again!



A new Trojan War broke out recently, this time the battle-field is not 'the wind-swept plain' but the Internet. This time he victims are the PCs of innocent users. Fortunately, a good defence against the nefarious invader may also be found on the same Internet.

Have you ever heard of Netbus or Back Orifice? If not, then you have been spared the disasters brought asbout by two new viruses disseminated via the Internet. These so-called *Trojan Horses* are worse than almost any other virus known so far, because the sender gets control of all 'infected' computers. In practice, 'control' includes remotely operating your keyboard or mouse, activating the CD-ROM drive, or placing a banner on your screen. Effectively, the PC gets a kind of invisible remote control which is operated via the Internet.

As soon as an infected computer is connected to the Internet, the sender of the Trojan Horse viruses may control all the PC functions that are within his power. Of course, this is no longer possible as soon as the Internet connection is terminated.



The sender of these viruses also gets full control over all files stored on the infected PC, which is left extremely vulnerable to electronic vandalism of the worst sort: gone are your personal notes and other confidential information like credit card numbers — all of it may be thrown on the street for all sorts of unauthorized use. It will be your worst nightmare!

The typical behaviour of these Trojan Horses bears a great resemblance to the trick used by the ancient Greeks to invade and conquer the city of Troy after years of unsuccessful besieging. The virus is usually spread by means of a nice little program, or a dressed-up email message which, when opened on the receiving PC, infects the system by installing a small server program. Next, this server utility makes itself invisible and immediately starts its destructive work as soon as the PC is linked to the Internet.

The good news is that the virus infection may be neutralized by means of a disinfectant program, **Framework**, which may be downloaded free of charge at *www.framework.nl*. On detecting one of the 45 known Trojan Horses, Framework stalls it, and prompts the user to eliminate it. If the answer is affirmative (what else can we answer, we wonder?), then the horse is killed in action.

(995035-1)