K2603C TIMER.

TOETSENBEDIENING.

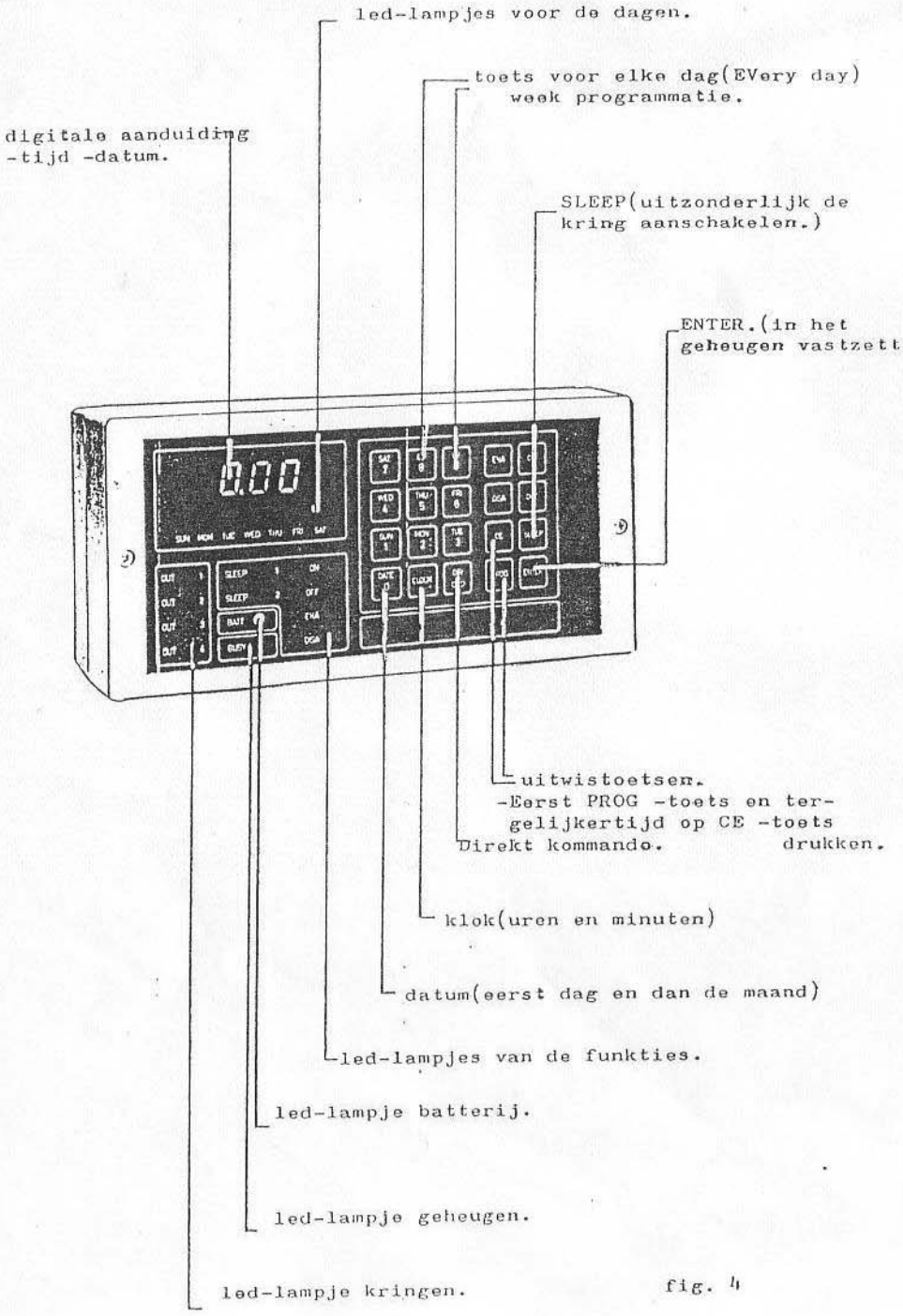


fig. 4

HET PROGRAMMEREN

1-De juiste tijd en datum instellen.

-Schakel de spanning in. Druk nu op de [CE] -toets.

-Programmeren of gelijkzetten van het UURWERK en de DATUM.

-Druk [PROG] en dan op [CLOCK], daarna de dag, .vb: [WED] en de datum [0][3][0][4] (eerst de dag en daarna de maand.), indien U fout bent druk verder.

Druk daarna [ENTER] (dit wordt opgenomen in het geheugen.).

Druk vervolgens de tijd in, [1][8][3][0] (eerst het uur en dan de minuten.), en [ENTER].

-Het BATTerij- ledlampje is nu uit.

Wanneer dit lampje brandt, staat de klok stil.

ter kontrole: U ziet nu het uur 1830, druk nu op [DATE] dan hebt U de datum 0304. Druk nu terug op [CLOCK].

SAMENGEVAT: vb:

[PROG] - [CLOCK] - [WED] - [0][3][0][4] - [ENTER] - [1][8][3][0] - [ENTER]

2-Vooraleer verder te programmeren, gaan we alle ongewenste stappen uitwissen.

-druk op [SLEEP], het ledlampje sleep 1 flikkert. Druk zoveel keer op [0] zodat er 3 nullen af te lezen zijn.(000). Druk nu op [ENTER] dan heeft men sleep 2 en druk op [0] zodat U ook 3 nullen afleest. Druk nogmaals op [ENTER] en de tijd verschijnt opnieuw.

-Overloopt nu de verschillende kringen 1-2-3 en 4 als volgt:

kring 1: -druk op 1 en dan op [ENTER].
Wanneer de ingestelde tijd blijft staan, is er niets geprogrammeerd voor kring 1, in het ander geval kan men telkens op [ENTER] drukken totdat men weer de ingestelde tijd kan aflezen.

Om het geprogrammeerde in de kring uit te wissen, druk op de toetsen [PROG] en blijf drukken terwijl U op de [CE] -toets drukt. Doe dit voor elke stap totdat kring 1 niets meer in het geheugen heeft.
Doe hetzelfde voor kring 2-3- en 4.

Nu is alles leeg en men kan beginnen met het programmeren.

Maak eerst een lijstje van het programma op papier, zo hebt U een goed overzicht bij de hand.

==========

HET PROGRAMMEREN

A-Direkt aan-en/of uitschakelen van de gastoestellen,aangsloten op de verschillende kringen.

-Druk op toets [DIR CMD] ,de ledlampjes van kring 1-2-3-en4 flikkeren.

Druk nu op [1] of [2] of [3] of [4] .(de gewenste kring) en vervolgens beginnen nu de lampjes van de funkties te flikkeren.
Druk nu naar keuze op [ON] of [OFF] .

SAMENGEVAT: [DIR CMD] - Nr kring,[1-2-3 of 4] - [ON] of [OFF] .
Dit kan ook met de [SLEEP] [1] of [2] . [ON] of [OFF]

B-Het programmeren van de kringen 1 - 2 - 3- of 4 .

1)zonder datum.
-druk [PROG] de lampjes van de kringen 1-2-3 en 4 flikkeren.
druk nu de kring nummer. [1] of [2] of [3] of [4] , de lampjes van de mogelijke funkties knipperen.

Druk nu op de toets van de funktie:[-ON-OFF-ENA-DISA-SLEEP]

Nu flikkert de lampjes van de dagen.druk nu een bepaalde dag, of elke dag van de week(vb. [MON] of [EV 8] [EV 9])

Nu flikkeren de digitale cijfers,en druk nu de tijd in.
(uren en minuten -vb. [1][8][3][0])en dan wordt alles opgenomen in het geheugen door op [ENTER] te drukken.

SAMENGEVAT:
[PROG] -[2]-[OFF] -[MON] -[1][8][3][0] -[ENTER] .

Bij een vergissing,en men heeft [ENTER] nog niet gedrukt,druk op [CE] en herbegin.
-Als men reeds [ENTER] heeft gedrukt ,kan men dit uitwissen door de toetsen [PROG] en [CE] .

2)met datum.

[PROG] -[2]-[ON] -[DATE] -[datum] -[ENTER] -[tijd] -[ENTER]

Dus in plaats van een bepaalde dag of EV , druk men [DATE] .

==========

DE FUNKTIE " SLEEP ".

Een SLEEP wordt gebruikt om op een onregelmatig tijdstip de kring aan te schakelen.(tot maximum 99minuten 99 seconden.)

Instellen van de SLEEPtijden:
-Druk [SLEEP] -lampje bij sleep 1 begint te flikkeren,stel nu de gewenste tijd in.
Druk nu op [ENTER] .Sleeptijd 1 wordt nu in het geheugen opgenomen, en het lampje van sleep 2 begint nu te flikkeren.
Indien gewenst stel nu sleeptijd 2 in en druk opnieuw [ENTER] .

Direkte sleep: [DIR CMD] - [nr kring] - [SLEEP] [1] of [2]

Na de verlopen sleeptijd is dit verdwenen uit de timer,en is de kring uitgeschakeld.

-SLEEP met dag of datum:

[PROG] -[nr kring] -[SLEEP] [1] of [2] [dag] of [datum] -[ENTER](bij DATE) -

druk nu de -[tijd]- en [ENTER] .

Nadien moet die geprogrammeerde stap met sleep uitgewist worden door de toetsen [PROG] - [CE] .

DE FUNKTIES DISAble(loskoppelen) en ENAble(aankoppelen) met het geheugen.

Bij een DISAble MOET altijd een ENAble geprogrammeerd worden.

DISAble.(LOSkoppelen)

druk: [PROG] → [nr kring] → [DISA] → [DATE]-of-[DAG] → [ENTER] → [tijd] → [ENTER]
[tijd] → [ENTER]

ENAble.(AANkoppelen)

[PROG] → [nr kring] → [ENA] → [DATE]-of-[dag] → [ENTER] → [tijd] → [ENTER]
[tijd] → [ENTER]

Voorbeeld:gebruik voor een verlofperiode.

==========

MICRO-PROCESSOR CONTROLLED TIMER WITH CMOS MEMORY

CONSTRUCTION DESCRIPTION

In general

The whole consists of 2 printed circuit boards; the microprocessor board and the display board. The microprocessor board contains the brain of the timer, namely the single chip microprocessor, the memory and the real time clock chip, together with the relays and the power supply. The display board contains the displays, the leds and their control logic.

As you can see, a lot of electronic circuitry has to be packed into a relative small housing. Therefore, a well-kept assembly is very important : you should mount the components close to the board, upright if necessary, solder them well and quickly, and after having mounted every component check you didn't cause any short-circuits.

Microprocessor board

-Fit the jumpers J1 (next to C1), J4 (next to IC1), and J6 (underneath R24).

-Fit resistors:

R1 and R2, 1K (brown, black, red, gold)
R3 through R11, 10K (brown, black, orange, gold)
R13, R16 and R67, also 10K (brown, black, orange, gold)
R17, 18K (brown, grey, orange, gold)
R18, 4M7 (yellow, violet, green, gold or silver)
R19, 1K8 (brown, grey, red, gold)
R20, 330K (orange, orange, yellow, gold)
R21, 2K2 (red, red, red, gold)
R22 through R24, 4K7 (yellow, violet, red, gold)
R25, 15K (brown, green, orange, gold)
R63 through R66, 47 ohm 1/2W (yellow, violet, black, gold)

-Fit D1 through D4, all diodes from the 1N4000-series. Pay attention to the polarity !

-Fit D5 through D18, 1N914 or 1N4148 type small signal diodes. Pay attention to the polarity ! The 1N4148 type may have a colour code (large yellow ring, brown, yellow, grey). In this case the large yellow ring must correspond to the strip on the board. If the diode is identified by its number, the black ring has to correspond to the strip on the board.

-Fit ZD1, 6V8 zenerdiode, pay attention to the polarity !

-Fit ZD2, 3V3 zenerdiode, pay attention to the polarity !

-Fit IC-sockets for IC1 through IC5

-Fit a 24 pin socket for IC6, the memory chip. Attention : the socket may be mounted on either the component side (position IC6A) or solder side (position IC6B). When you mount the socket on the solder side, then, later on, to put a new memory chip (EEPROM or UV-EPROM) into the socket, you'll only have to remove the front panel. (see further for the memory options)

-Fit T1 through T11, T13 and T15, BC547, 548 or 549 type transistors.

-Fit T23, BC557, 558 or 559 type transistor.

-Fit the resistor array RA1A or RA1B, according as the memory chip is mounted in position A or B. In both cases the array has to be mounted on the component side. Such an array consists of nine resistors which are connected to each other on one side, and packed together into one single housing. The common pole is indicated by a dot or a notch, and must be inserted in the hole, marked 'C'.

-Fit C3 through C6, 100nF Sibatit capacitors.

-Fit C7 and C8, 10nF Sibatit capacitors.

-Fit C11, 330nF MKM capacitor.

-Fit C12 and C13, 22pF ceramic capacitors.

-Fit CV1, 39pF trimmer capacitor. Turn the trimmer in the centre position, that is so that the plates overlap each other for about 50 percent and that the overlapping decreases as you turn to the right.

-Fit VR1, a 7805 type voltage regulator or equivalent, as follows: put VR1 in the appropriate holes, with its metal backside towards the hole for the heatsink. Put the heatsink in its place and pass a 10 mm long bolt from the underside through the board and the heatsink. Bend the regulator backwards to put it into flat position, posed with its metal back on the heatsink. Fix all three parts together with a nut. Tighten the nut well in order to obtain a good thermal contact between regulator and heatsink. Check that the leads of the regulator do not touch anything and then solder them.

-Fit X1, 4MHz crystal.

-Fit X2, 32.768KHz crystal (small metal tube).

-Fit C1, 2200 µF electrolytic capacitor. Pay attention to the polarity!

-Fit C2, 10 µF electrolytic capacitor. pay attention to the polarity!

-Fit E1, the thin 1,2V NiCd accu.

-Fit E2, the bigger 2,4V NiCd accu.

-Fit the 4 relays RY1 through RY4, type SRU-6V or equivalent.

ATTENTION: the following components have to be fitted on the solder side of the board, the soldering of their connections consequently is done on the component side !

-The connectors J7 and J8. Cut a STRIP of 13, and one of 10 female contacts from the supplied tape. Don't detach the contacts from the supporting strips! First solder the middle and the two outmost contacts of each row. Check the contacts stand upright, and the distance between them and the board is about half a millimetre (so that the edges cannot make any short-circuit with the leads between the contacts). Correct if necessary. Then solder the rest. Next break off the supporting strip bending it softly up and down.

-The 10 pole keyboard connector J9. If the connector is not equipped with contacts on both sides in the groove, then you must take care that the contacts come on the printedge side.

-The 2 pole screw connector for "220VAC".

-The screw connectors for the outputs.

-The fuseholder "F1"

-The (isolated) wires for the connection of the power supply.

1)when using an EXTERNAL 8VAC or 12VDC power supply. Connect the points 'TR' to the points '8VAC'. The polarity is unimportant. Scratch off the text '220' from the PCB overprint in order to avoid later misunderstandings. Insert a 1 A fuse into the holder, and press the cap onto the holder.

2)when using a built-in transformer (VELLEMAN TR2603). Solder four about 15 cm long wires to the points '220VAC' and '8VAC'. The wires have to be inserted from the solder side, and be soldered on the component side ! Insert a 250 mA fuse in the holder, and press the cap onto the holder.

-Insert the IC's into their sockets:

IC1, type VLK2603, with its notch towards C5.
IC2, type SAF3019, with its notch towards IC3.
IC3, type 555, with its notch towards the relays.
IC4, type 74LS148, with its notch towards IC5.
IC5, type 74LS373, with its notch towards the resistor array.
IC6, type 6116 or 5517, with its notch towards the fuse.

Construction of the display board.

-Fit the 12 jumpers "J".

-Fit the resistors:

R12, R14 and R15, 10K (brown, black, orange, gold)
R26 and R27, 180 ohm (brown grey, brown, gold)
R28, 220 ohm (red, red, brown, gold)
R29 through R35, 180 ohm (brown, grey, brown, gold)
R36 through R41, 100 ohm (brown, black, brown, gold)
R42 through R55, 47K (yellow, violet, orange, gold)
R56 through R62, 22K (red, red, orange, gold)

-Fit IC sockets for IC7 and IC8.

-Fit a 40 pin IC socket for the displays.

-Fit C9 and C10, 68 nF Sibatit capacitors.

-Fit the transistors. Fit them close enough to the board (total heigth max. 8 mm), otherwise the board will not fit into the housing.

T12, T14, and T16 through T22, type BC547, 548 or 549.
T24, type BC557, 558 or 559.
T25 through T31, type BC516.

-Fit the buzzer BUZ1. Pay attention to the polarity !

-Insert DY1 through DY4, D350PA or TIL701 type displays or equivalent, in their sockets. Take care that the decimal point is below, in other words towards the leds.

-Fit LD13 through LD19, red 3 mm diameter leds. Pay attention to the polarity ! (shortest connection = cathode = flat side on the PCB overprint). First only solder one connection and position the leds so that their tops are equally high as the displays. Correct if necessary and then solder the second connection.

-Fit LD1 through LD12, red 5 mm diameter leds. Pay attention to the polarity (flat side) ! The widening at the connections must touch the board, so that the leds and the displays are equally high.

-Fit J7 and J8: Cut a STRIP of 13, and one of 10 male contacts from the supplied tape. The plastic has to be on the component side, and the long end of the pins passes through the board (see figure). First only solder 2 pins of each connector. Check the plastic completely touches the component side of the board and the contacts stand upright. Then solder the remaining contacts.

-Insert the IC's into their sockets:

IC7, type 74LS374, with its notch towards T14 and R15.
IC8, type XR2203 or ULN2003, with its notch towards C9 and C10.

Mounting into the housing

Insert J7 and J8 of the display board carefully into the connectors of the microprocessor board. Take care that each pin of J7 and J8 is correctly inserted in the corresponding female contact and that there exists no short-circuit between the different con-

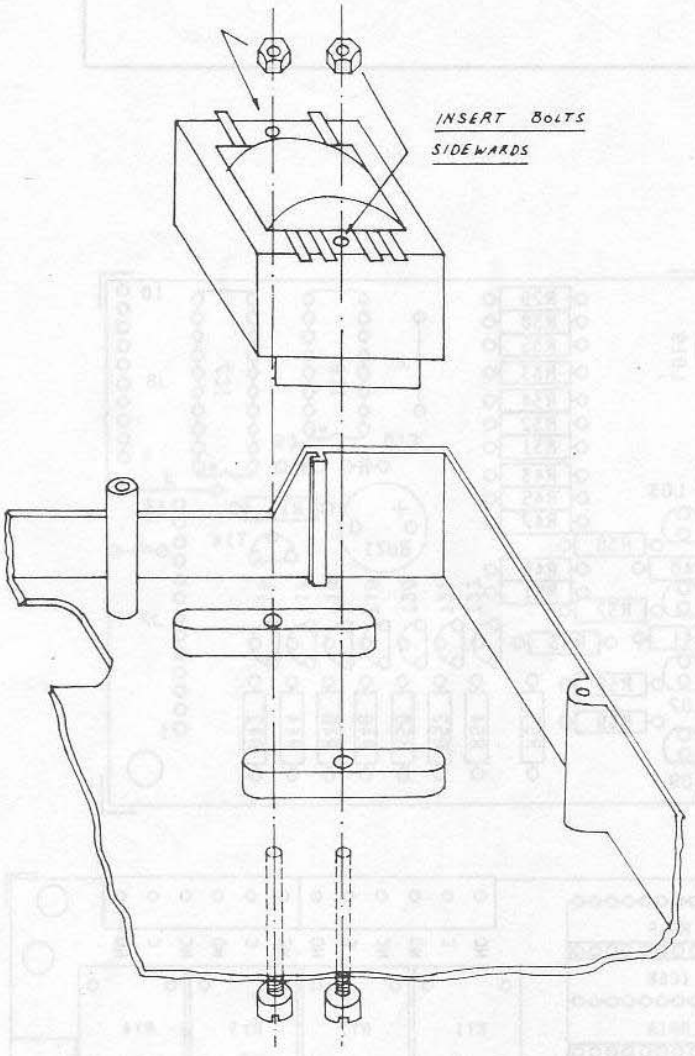
tacts. Install two spacers between the boards: the first one in the left corner upside the display and the other one in the right corner unterneath connector J7. Pass a 15 mm long bolt through each spacer and fit them with nuts. Tighten the nuts sufficiently to make the boards constitue a firm unit.
Now fix the microprocessor board into the bottom of the housing by means of 3 (short) parker screws. Then install a third spacer between the boards, in the left corner unterneath the leds. Use the somewhat longer parker screw to fix the board onto the bottom at that place.
Connect the power supply:
1)when using an EXTERNAL 8 VAC or 12 VDC power supply. The connection wires have to be connected to the screw connector "VAC" (!). The polarity is unimportant, even when using direct current. The fuse has to be a 1 A type.
2)when using a built-in transformer (Velleman type TR2603 was developped specially for this purpose). Proceed as follows (see also figure):
Take the bolts out of the transformer core. Pass the two 40 mm long bolts through the bottom, shove the transformer over the bolts and fix the whole with the nuts. Connect the transformer: the primary winding to "TR", and the secondary winding to "8VAC". DON'T solder the connection wires to the connection pins of the transformer (these are provided in case the transformer has to be mounted directly onto a PCB), but at the place where the windings are soldered. The pins have to be cut off up to the plastic. Check you didn't confound the primary and secondary windings ! The mains has to be connected to the "220VAC" screw connector. The fuse has to be a 250 mA type.

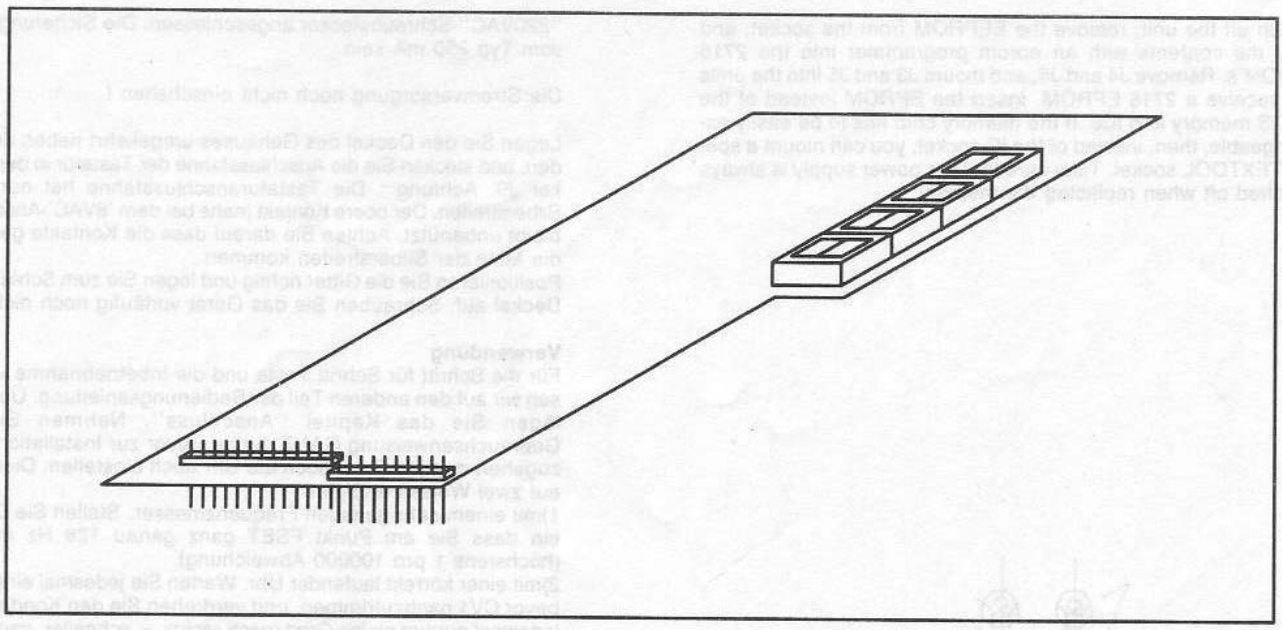
Don't switch on the power supply yet.

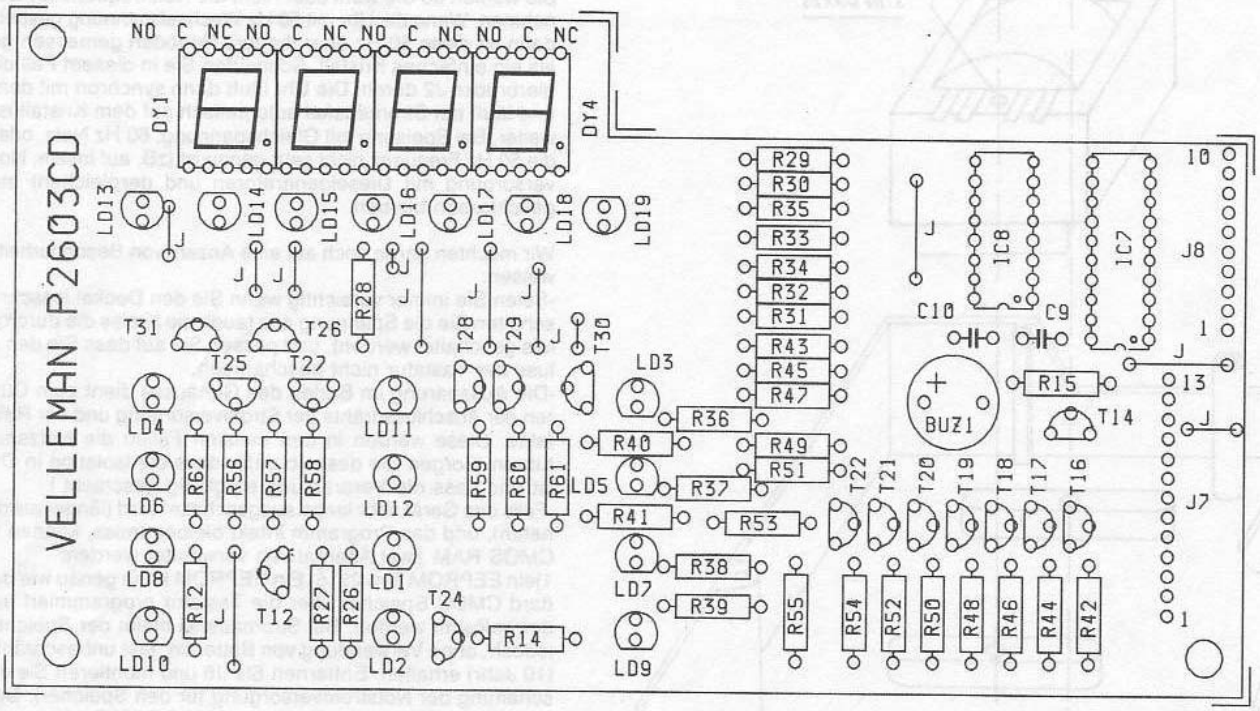
Put the cover of the housing upside down next to the bottom, and insert the keyboard plug into connector J9. Attention : the keyboard plug has only 9 silverstrips. The upper contact (close to the '8VAC' connection) is not used. Take care that the contacts are exactly in the middle of the silverstrips.
Put the grids on place and finally the cover on top. Don't fit it with screws yet.

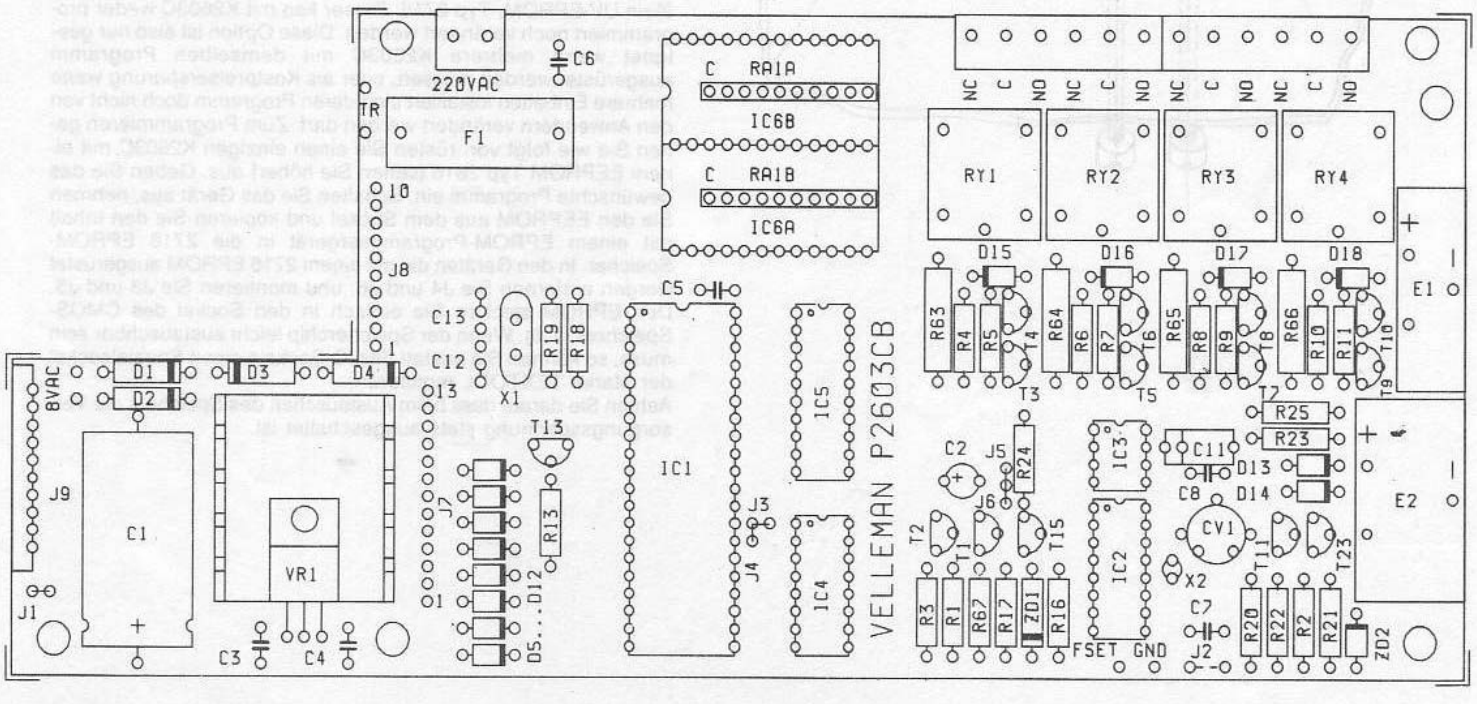
Usage
For the step by step testing and commissioning, we refer to another part of te manual. Omit the chapter "connection". Go through the COMPLETE directions for use.
Before starting installation the clock has to be adjusted. This can be done in two different ways:
1)with a very precise frequency meter. Adjust with CV1 until you measure very precisely 128 Hz at point FSET (maximum deviation: 1 per 100000).
2)with a known precise clock. Wait each time one day before adjusting CV1, and adjust the trimmer only a few degrees (to the right = faster, to the left = slower). Otherwise you turn beyond the ideal setpoint. Once the clock runs perfectly at the crystal frequency, you may choose whether or not you wish to use the mains frequency as the time base. If the timer is supplied with 50 Hz alternate current, thi 50 Hz is, measured over longer periods, more exact than a ordinary crystal. In that case cut through the jumper J2. The clock now runs synchronously with the mains frequency, and continues, in case of a power failure, automatically with the crystal oscillator. In case of direct current supply, 6O Hz mains, or when the 50 Hz is not very precise (for instance on isles, emergency power supply with Diesel generators and similar things) J2 should remain closed.

We would still like to point out a number of particularities:
-Be always careful when you open the unit: switch off the supply (also of the circuits switched by the relays) and take care not to damage the keyboard connection.
-The opening in the bottom of the housing serves as a passage for the connection wires of the power supply and the relay contacts. In most cases these wires will carry the mains voltage. Therefore take care that the isolation is O.K. and the wiring is done carefully !
-If the unit is switched off for a long period of time (more than three months), and the program in memory may not get lost, then two alternatives can be used for the CMOS RAM:
1)an EEPROM type 2816. An EEPROM can be programmed and erased through the keyboard just like the standard CMOS memory. In case of a power failure, the memory contents remain intact for a practically unlimited time (10 years), without the use of batteries. Remove J6 and mount J5 (switching-off the battery backup for the memory). Now insert the EEPROM 2816 instead of the CMOS RAM into IC6.
2)an UV-EPROM, type 2716. This one cannot be programmed or changed by means of the K2603C. Therefore this option is only suitable when several K2603C's have to be equipped with the same program, or as a cost saving when several units are installed with a program that may not be changed by the users. For the programming you proceed as follows: equip one K2603C with a 2816 type EEPROM (see above). Enter the desired program. Switch off the unit, remove the EEPROM from the socket, and copy the contents with an eprom programmer into the 2716 EPROM's. Remove J4 and J6, and mount J3 and J5 into the units that receive a 2716 EPROM. Insert the EPROM instead of the CMOS memory into IC6. If the memory chip has to be easily exchangeable, then, instead of the IC-socket, you can mount a special TEXTOOL socket. Take care that the power supply is always switched off when replacing the memory.






VELLEMAN P2603CD
NO C NC NO C NC NO C NC NO C NC
DY1
DY4
LD13
LD14
LD15
LD16
LD17
LD18
LD19
R28
T31
T26
T25
T27
T28
T29
T30
LD3
LD4
LD11
LD5
LD6
LD12
LD8
LD1
LD7
LD10
LD2
LD9
R62
R56
R57
R58
R59
R60
R61
R40
R36
R49
R51
R37
R41
R53
R38
R39
R12
R27
R26
R14
R55
R54
R52
R50
R48
R46
R44
R42
R29
R30
R35
R33
R34
R32
R31
R43
R45
R47
T22
T21
T20
T19
T18
T17
T16
T12
T24
IC8
IC7
J8
10
1
C10
C9
BUZ1
R15
T14
13
J7


VELLEMAN P2603CB
220VAC
C6
TR
F1
J8
10
1
C RA1A
IC6B
C RA1B
IC6A
NC C NO NC C NO NC C NO NC C NO
RY1
RY2
RY3
RY4
D15
D16
D17
D18
R63
R4
R5
T4
R64
R6
R7
T6
R65
R8
R9
T8
R66
R10
R11
T10
T3
T5
T7
T9
E1
C5
C13
C12
X1
R19
R18
T13
R13
D5...D12
J7
13
1
IC1
IC5
IC4
J3
J4
C2
J5
R24
J6
T2
T1
T15
IC3
IC2
C11
C8
CV1
D13
D14
R25
R23
T11
T23
X2
C7
J2
E2
R3
R1
R67
R17
ZD1
R16
FSET
GND
R20
R22
R2
R21
ZD2
D1
D3
D4
D2
8VAC
J9
C1
VR1
J1
C3
C4

VELLEMAN K2603C MICROPROCESSOR CONTROLLED TIMER WITH CMOS MEMORY INSTRUCTIONS FOR USE

General:

Most mechanical or electronic programmable timers available up to now had, according to their price, a number of serious and less serious shortcomings, such as the total loss of the program memory and/or the stopping of the clock in the case of a power failure, a very limited number of program steps, no possibility of programming with the date, in addition to day or week programs, only one fixed sleep time or perhaps none at all, etc.

To avoid all these disadvantages we used the most modern technologies: a special single-chip microprocessor developed by us, a memory chip and a clock chip, with standard emergency power supply by NiCd batteries, membrane keyboard, and so on...

The keyboard, the location and functioning of the display and the indicator LEDs, and the step-by-step progamming, controlled by the applance itself, ensure great ease of operation.

The program steps are specially powerful: not only is it most unusual to be able, via the program, to block an output in the on or off position (with the disable function, so that it does not follow the rest of the program until it receives an enable order; this is of great interest, for example for interrupting a day program during the weekend (and only two program steps are needed for this!)), but in addition to this, it is even possible to repeat any instruction whatever, according to choice, each minute, each hour, each day, each week, each month or each year !!!

Even combinations of these possibilities are feasable: for instance it is possible, with only one (!) instruction, to activate an output each first day of the month between 8 and 9 o'clock for 10 seconds every minute!

There are two sleeptimes, quite freely programmable: you can choose their length between 1 second and 99 minutes 99 seconds.

To prevent the loss of the memory content, there are several precautionary measures:

- on power failure, through the use of NiCd batteries, the contents of the CMOS memory remain intact, even over several months if necessary!
- the memory can be protected against changes made by unauthorized persons, by cutting through a wire bridge (interruption of the write signal).
- via the keyboard, programs can be cleared only step by step, and this can be done only by simultaneously pressing the PROG and CE keys. This means that involuntary clearing of a step is completely impossible.

Application: for household use, in weekend dwellings (heating, lighting, water heaters, ventilation, etc.), schools and businesses (time signals, stoking boilers, pumps, pressure groups, lighting, security of access doors outside working hours, furnaces, etc.).

Technical data:

Built up around a single-chip microprocessor.

Equipped with NiCd batteries (standard) as a back-up for the memory and as an emergency power supply for the clock in case of a power failure.

- Automatic re-start after power failure.
- User-friendly and very powerful programming.
- Two completely independent and programmable sleep times, from 1 second to 99 minutes and 99 seconds.
- Erasing or modification of the memory contents (by accident or not) by unauthorized persons can be avoided by simply removing a jumper.
- Standard program memory of 336 steps with date or 560 steps without date.
- Standard four relay outputs (220V/3A, all 4 relays included).
- Membrane keyboard; debouncing is ensured by the microprocessor.
- completely independent unit with housing, easy to fix to the wall.
- Power supply 8 VAC/1A or 12 VDC or TR2603 transformer.
- Dimensions: 242 x 116 x 59 mm.

Connection

Loosen the two parker screws on the front. Lift up the cover and put it next to the bottom (as cover and bottom were connected to each other by a hinge on the right-hand side). Connect the supply voltage to the two pole screw connector. Check that the voltage corresponds to that for wich the unit was built (primary voltage of the transformer equals the mains voltage, or 8VAC or 12VDC if an external power supply has to be used). Put the cover back on the housing, but don't fasten it yet. First go through the complete directions for use, so that you can adapt the configuration if necessary (by changing the jumpers), before proceeding to installation.

Operation:

In this description we assume that the appliance is being used for the first time.

We shall explain all the possibilities and their applications step by step. We recommend that even those who are well acquainted with appliances of this type should read through these instructions carefully, so that they can learn to make the best possible use of it.

Starting up a new appliance

Switch on the current.

The BUSY lamp lights up for about fifteen seconds, then the displays and the day-of-the-week lamps (afterwards referred to by the abbreviation DOW - day of the week) begin to flash and the BATT LED lights up. BUSY indicates that the processor is busy writing something in the memory. On starting up with a new RAM, the processor will start by initializing it: on all free places a ZERO order is written (a program step that will never be executed) and in the beginning of the memory comes the "system data" which allows the processor to test the RAM on a number of parameters (further details of this below).

BATT comes on as soon as the NiCd accu voltage falls too low to guarantee the proper working of the clock chip.

If BATT comes on during normal working, then the timer stops just like a conventional clock. The outputs remain the state they were in when BATT lit up. If, however, the timer is started up in a BATT condition, e.g. after a power failure with a too low Ni Cd accu voltage, then it will ask you to set the clock (by making the DOW and the displays flicker).
Let us for the moment assume that it is 18h00 on Wednesday 6 December. On the keyboard you wil see that on the same key as the 4, there is "WED" or Wednesday. Press WED. The displays continue to flash, but of the DOW only WED flashes to indicate that you have input the day of the week, and that it is now the turn of the date. Type in 0 6 1 2. Note that the flashing figures disappear as you input the "right" figures. Always input numbers completely, including the zeros before the relevant numbers, In the date, the day always comes first, then the month. Press ENTER to tell the processor that you have typed the date in correctly.
The displays start flashing again, but this time DOW lights up continuously. The processor is indicating that you should now input the time. Type in 1 8 0 0. The hour is input in the same way as the date, but in this case first the hour and then the minutes. Here you must also type the zero before the relevant figures. Press ENTER. The processor accepts the time and starts the clock, BATT goes out.

To display the date again, press DATE (this is the same key as the 0). You can display the time by pressing CLOCK (the key next to it).

The sleep times
Let us now adjust the sleep times: press SLEEP (on the far right column of the keyboard).
Since up to now we have never input a sleep time, the present value of the sleep time appears on the displays "flashing 8888". SLEEP 1 flashes to indicate that you must type in the first sleep time. Firsts give the number of minutes, then the seconds, e.g. 0 0 0 5, and ENTER, BUSY goes on, and a flashing SLEEP 2 instructs you to input the second sleep time, e.g. 1 minute and 1 second. Type 0 1 1 0 ... oh! that was wrong. Not to worry; you can just input further figures till you get the desired value on the display. Thus type 1 0 1 and ENTER. The display now gives you the time.
If you wish to verify or change the sleep times, always proceed as follows. Assuming that we wish to change sleep time 2 to 1 minute and 11 seconds. Press SLEEP. Sleep time 1 appears. Press ENTER, since we do not wish to change sleep time 1. Now sleep time 2 appears. Type 1 1. The displays now show the desired time. Press ENTER to finish the operation.

The CE key
When you have started to input something and you have second thoughts or have made a mistake, then it is always possible to interrupt the instruction with the CE key. The time then returns to its normal tasks, and the display just shows the time. If information appears on the display that is unexpected or incomprehensible for you, or when you are busy with a task but have lost the thread, then press CE and start again.

The direct commands
All of this is very fine, but the timer was really intended for switching... and that is what we are now going to test with the "Direct Commands". Press DIR CMD (in the middle of the bottom row). All the OUT lamps flash. Give the number of the output you want to switch, e.g. 6. OUT 1 to 4 continue flashing, since output six does not exist! Input an existing number, e.g. 1 (ENTER is not necessary). The MODE leds now flash and OUT 1 lights up to indicate that you are giving a command for output 1. You can choose between five keys in the two last columns of the keyboard: ENA, DISA, ON, OFF and SLEEP. Press ON. Output 1 is activated.

Function of ENA, DISA and SLEEP
Although the purpose of the ON and OFF keys is obvious, yet ENA, DISA and even SLEEP call for a word of explanation. By putting an output into the DISAble mode, the working of the relay is, as it were, disconnected from the program control. However, the timer stores the state the output would have been in if it were not in the disable mode. As soon as the ENAble command comes, then the output is reconnected to the normal program, and the output switches to its correct state.
Try this out:
press successively DIR CMD, 1 and DISA. Output 1, which we had just switched on, remains unchanged at on. Now press DIR CMD, 1 and OFF, and the output will not react, because with the privious command you put it into disable. Internally, however, the OFF instruction is "noted". Press DIR CMD, 1, ENA. The output switches off, in response to the last command given during the disable period (OFF in this case).

Naturally, this facility will mostly be used to bypass day programs during the weekend in holidays periods, and the like. But it is also handy as a direct command, for example to leave the heating on later exceptionally for one evening or to switch it off for a few days during your absence. Further on in the text you will find a number of examples of applications.
One tip: if an output does not appear to work as you had expected, don't panic. First of all give the command ENA. It is quite possible that the timer has been put info disable as the result of an instruction some time ago that you had forgotten, or that somebody else has put the timer into disable without your knowledge.

The working of the sleep times also requires some explanation. When you give a sleep command, the time is put into a counter and the output switched on. For each minute the clock advances, the minute part of this counter is decremented. Every second, the processor checks wether the minute counter is already at 0. If so, it reduces the second register. If both are at 0 and the ouput is not activated by an ON instruction, then the output will be switched off. Try this out with DIR CMD, 1, SLEEP. You will see that SLEEP 1 and SLEEP 2 flash, asking you to give the number of the sleep time (the processor accepts only 1 or 2).
Here, for example, give 1. Output 1 switches on for about 5 seconds and then it returns out. Now try SLEEP 2: press DIR CMD, 1, SLEEP 2. The output now stays on till 11 seconds after the following minute-change.
This immediately illustrates two limitations on the use of the sleep times: first of all, that the length of the first "second" of the sleep time is not exactly determined, but could better be described as being maximum 1 second in length; secondly that in using sleep times via a direct command, the first "minute" lasts only until the following minut-change. The latter does not apply when you make use of sleep times in the program.
There now remains only one key which we have not discussed, and it is the most important - PROG.

Programming the outputs
The programming is always done in a fixed sequence, led by

the processor.
We give all the examples for output 1, but naturally the method is identical for all the other outputs.
If you come to a point where you are not certain whether you have input everything correctly, or if you make a mistake, you can always interrupt the instruction with the CE key and start again.

To order the timer to switch on OUT 1 every minute for five seconds, proceed as follows: PROG, 1, SLEEP 1. So far, the programming was similar to the inputting of direct commands. The DOW now flash. You have three possibilities:
- either the instruction should by executed only on 1 specified day of the week, and you press one of the figures 1 to 7 (SUN ... SAT).
- or the instruction every day of the week, and you press 8 or 9 (EVery).
- or the instruction must be executed only on certain dates.

Press EV, thus key 8 or 9. All DOW indicator LEDs light up (the instruction wil be executed all days of the week) and the displays ask you to input the time at which the instruction is to be executed, first hours then minutes. Here too, you have the possibility of giving an EVery indication: this applies for both the hours and the minutes if you type in a number greater than 80, which means that the processor considers every hour of the day (or respectively every minute of the hour) as being in conformity with the real time, and therefore executes the instruction. Type 9 9 9 9 and ENTER. BUSY lights up, and the displays show the real time again. The instruction is accepted and stored in the memory. Just a little patience until the following minute-change, then you will see output 1 activated for 5 seconds. This will now happen each minute, and will go on for ever, unless it is altered. Now input the following program steps:
PROG, 1, ENA, EV, 8 0 0 0, ENTER
PROG, 1, DISA, EV, 8 0 0 5, ENTER.
What you have now done is to block the output 5 minutes after every hour, and unblock it at the hour-change.
For the moment, do not worry about the reaction of the output, but control the program for output 1.

Verifying the programs
Just press key 1. On the displays and the indicator LEDs appears the first program step we have input: we are dealing with output 1, the "time" is 99.99, DOW is EVery, since all the DOW lamps are lit up, and finally it was a sleep time 1 instruction.
Press ENTER. The following instruction for OUT 1 appears. Once more ENTER for the third instruction, and the last time ENTER to end the "check".
For the other outputs the procedure is identical, but so far there is nothing in the memory for them, so the processor will just continue to give the time.

Executing the programs
When the timer is switched on, it always starts with all outputs off and in the enable state. Afterwards, it obeys the instructions that correspond with the actual time.
Instructions are specified for the same time are never executed by the processor at exactly the same time, but shortly after one another, and IN THE SEQUENCE IN WHICH THEY APPEAR IN THE MEMORY, and as they appear when you check the programs for an output. The time-difference, with a completely loaded memory, can be somewhat less than a second.
Our program now runs as follows: at Xh05 the output switches on, and immediately afterwards it is blocked in the ON state until the following hour-change. At that moment, sleep time 1 starts (but you do not see this, because the output is still on!) and it is unblocked. Five seconds later it cuts out. It still gives four pulses of five seconds before starting the whole process over again af the fifth minute.
This is a typical example of a complicated pattern which, by judicious choice of instructions and their sequence can be reduced to only a few steps. Without a doubt, in your own applications you will be able, with a bit of thinking, to reduce the number of instructions drastically. One good piece of advice straight away: note programs first before you type them in. They will be simpler, shorter and more efficient, and the chance of errors is reduced. Furthermore, it will be easier for you to use "tricks" as in the example above, by changing the sequence of "simultaneous" instructions.

Setting the clock right
To "program" the clock, press PROG, but as output number press CLOCK. You come into the same routine as when you first adjusted the clock, so first DOW, then the date, and then the time. Set the clock to 18h03 on Wednesday 6 December: PROG, CLOCK, WED, 0 6 1 2 ENTER, 1 8 0 3 ENTER.
This enables you after a short time to test the working of the above program.
If the clock has now reached 18h06, set it again, this time at Wednesday 6 December at 18h58.
You can then control the working of the second part of the above program.

Clearing instructions from memory
Clear the first instruction from memory as follows:
press key 1 to start displaying the program. The first program step appears on the displays. Press PROG and simultaneously CE. Do not release the PROG key while you are pressing CE, otherwise all you will clear will be the display of the program. BUSY now lights up, and the following instruction is now shown. ENTER until the cycle is completed.
Each time you want to clear an instruction (but not now since we still need the other program steps!) you must follow this procedure: control the programs for the output concerned, press ENTER until the instruction you want to clear is displayed, then clear it by pressing PROG and CE. Then if you also wish to clear following instructions, you can also do this with the PROG and CE keys.

Programming with date
We are now once more going to let output 1 pulse every minute, but only "today", i.e. on 6 December: PROG, 1, SLEEP, 1. The DOWS now flash. Press DATE. The displays and DOW both flash, to show that you must input the desired date, first the day and then the month. Here the EVery possibilities are the same as those for the time: a number larger than 80 lets the instructions be repeated on every day of the given month, or on the given day of every month. Type, for instance, 0 6 8 0 and ENTER. DOW now goes out to indicate that an instruction with date has been input, and the displays now "ask" for the hour. Give 8 0 8 0 and ENTER.
Now once more control the program for OUT 1: the new instruction appears in the first place, where the orignal (cleared) instruction used to be. DOW stays out to indicate that a date has also been programmed. Press DATE, and the programmed date appears. Press CLOCK to make the time appear again. Press ENTER to control the following steps and finish the cycle.
Now the switching pattern will be carried out only on the sixth day of each month.

Option
As the timer is now adjusted, you can input 336 program steps with date. If you try to input more steps, they will simply not be stored; the BUSY lamp will not light up.

If you are never going to use a program step with date, then you can open J1 and thus increase the number of steps to 560. But reflect well before doing this: in the "system data" described in the beginning of the text, the type of program involved is stored: with possibility of date or not. When starting up, J1 is tested, and THE WHOLE MEMORY IS CLEARED if the state of J1 does not correspond to the system data (which corresponds to the state of J1 on the previous start-up!) The difference consists in the number of

bytes (places in the memory) the processor needs for one instruction: five if you want the possibility of programming with dates, and only three if you never have dates in your programs. The consequence, however, is that the processor would get completely lost in a program that is stored in the memory in one form, while the processor itself "thinks" that it is in the other form.
When J1 is opened, you can no longer input program steps with date. If you press DATE for DOW, the timer will not accept this, but will ask for a specific day or EV.
Nor will you get any reaction if you press DATE in verifying the memory.

Protecting the memory against overwriting
If the memory contains long, complex programs, then you can protect yourself against jokers who, for instance, clear an instruction or alter the sleep times, and give you a lot of searchwork. Open J4 (or cut the wire bridge through) after you have carefully verified and tested your programs. Then there is no way in which the processor can write in the memory. It is impossible to clear program steps or to add them, to alter the sleep times, or to change from the mode with date to the mode without date or vice versa.
Attention: if you deliberatly want to change from one mode to another then RESTORE the wire bridge or SHUT J4, otherwise you will find the craziest "programmes" in the memory and the control of the outputs will be completely haywire!

Time base:
Yous may choose whether or not you wish to use the mains frequency as the time base. The 50 Hz mains frequency is, when measured over longer periods, more exact than an ordinary crystal. When the timer is supplied with 50 Hz AC, you can remove jumper J2. The clock then runs synchronously with the mains frequency, and continues, in case of a power failure, automatically with the crystal oscillator. In case of direct current supply, 60 Hz mains, or when the 50 Hz is not very precise (for instance on isles, emergency power supply with Diesel generators and similar things) J2 should remain closed.

Installation
Lift the cover off the unit and carefully pull the keyboard plug out of the connector. Remove the grids. In the bottom of the housing there are two holes for wall mounting. The connection wires enter the housing through the rectangular opening. Drill two 6 mm holes in the wall and insert a plug into each hole.
Pass the connection wires through the rectangular opening in the bottom and fix the bottom onto the wall by means of 2 (wood) screws.
The relay outputs are single pole single throw switches, with "C" the common contact, 'NO' the normally open contact (closed when the relay is activated) and "NC" the normally closed contact. Each relay can switch a resistive load of max. 3A at max. 240VAC. If the operation of the timer gets disturbed by the switching of highly inductive loads (even when the switched power is not too high), then this is caused by the sparks produced by the switching of the relay. In most cases this can be remedied by putting a VDR (for example SIEMENS S14K275) over the contacts. If necessary, a series connection of a 100 ohm 1/2 watt resistor with a 47 or 100 nF/400V capacitor installed in parallel with the VDR will further reduce the sparks (see fig.). If this remedy doesn't satisfy, a DIRECT CURRENT relay can be mounted close to the load to be switched. This DC relay has to be supplied from a separate direct current source, and a diode should be mounted parallel (in REVERSE) to the coil (see fig.). The sparks are then kept at a distance from the timer, making the electromagnetic radiation harmless.
After you have finished all connections, the grids should be put back in their places. Reconnect the keyboard. Attention: the keyboard plug has only 9 silverstrips. The upper contact (close to the '8VAC' connection) is not used. Take care that the contacts are exactly in the middle of the silverstrips. Finally fasten the cover by means of the two long parker screws.

ATTENTION: always switch off the mains before opening the unit. If ever the fuse of the timer should blow, then replace it by the original type: 0,25 A/slow when using 220/240VAC, or 1 A when using an external supply (8VAC or 12VDC).