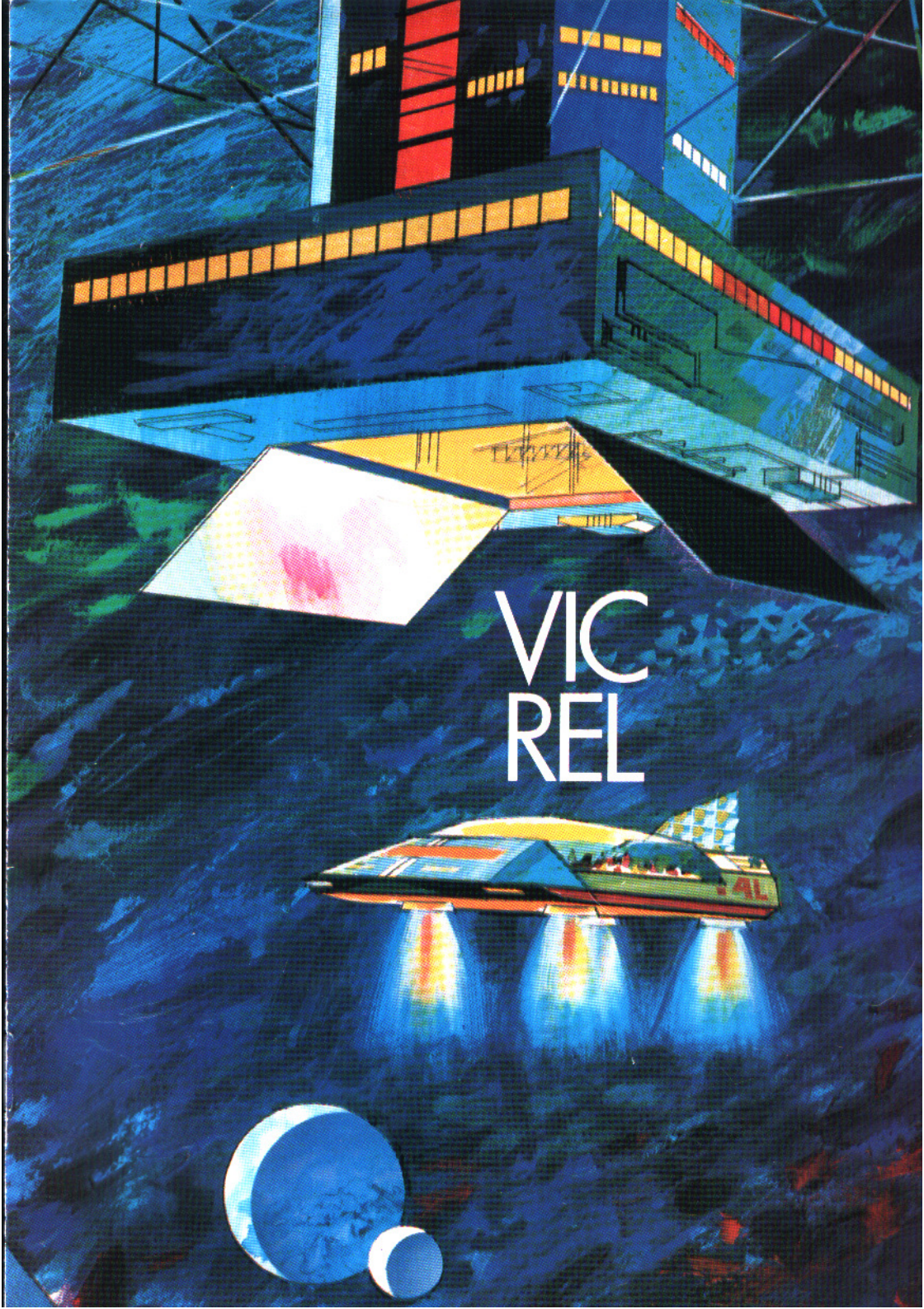

VIC
REL

VIC-RELAY-CASSETTE

In general

The purpose of this cassette is to simplify control of, for example, burglar alarms, garage doors, doorlocks, heating elements, lamps, transmitters, remote controllers, valves, pumps, telephones, accumulators, irrigation systems, electrical tools, stop watches, ventilators, humidifiers, etc, etc. The cassette contains 6 relay outputs, and 2 inputs of type optocoupler.

Connection

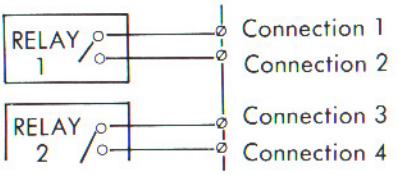
This cassette is connected to VIC-USER-PORT (on the BACK to the LEFT). Turn the cassette so that the text and the lights face upwards. The green socket is connected to the VIC. The Red light indicates that the inlets 1—6 are activated (closed). The Green light indicates that the inlets 1—2 are activated (5—12v DC connected).

In and outputs

Outputs: 1—6.

Connections 1—12 are connected in pairs to a relay in the VIC RELAY.

FIG-1.



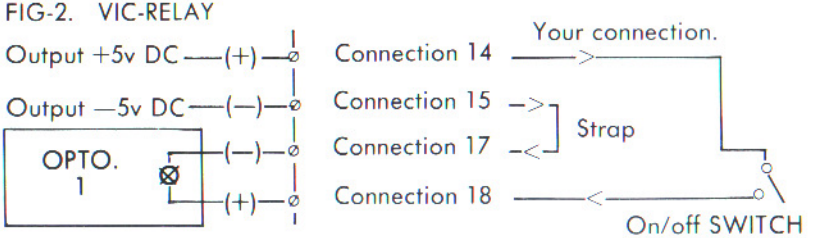
These OUTputs are galvanically separated from the VIC.
If the Red light (1) is lit this indicates that the socket on relay (1) is connected, that is to say 1—2 are contacted with each other. As a result of this if the same light is not lit, there is no connection between 1—2.

A relay is simply a switch which can be programmed by the VIC to switch on and off.

INPUTS 1—2

Or connections 17—18 19—20.
These constist of two optocouplers, indicating that when there is 5—12v DC connected by you to some other machine the Green light is lit. The same happens with the optocoupler. The light is caught by a light sensitive transistor which then informs the VIC that one of the inputs has been connected to 5—12v DC. This is to ensure that all your connections to the VIC-userport are isolated.

FIG-2. VIC-RELAY



PROGRAMMING

Initiation, i.e. the programming of the different connections on VIC's user-port which should be inlets and outlets respectively, is done by using a simple command.

poke 37138,63

This puts pins 1—6 as outputs and pins 7—8 as inputs. (se VIC programming manual).

This command should always be first in your program.

Now everything is ready for a short test program. Write the following on your VIC.

```
10 reg=37138  : dat =37136  : rem data
20 poke reg,63 : gosub 100  : rem initiating the i/o
30 poke dat ,1  : gosub 100  : rem switch on lamp (1)
40 poke dat ,2  : gosub 100  : rem switch on lamp (2)
50 poke dat ,4  : gosub 100  : rem switch on lamp (3)
60 poke dat ,8  : gosub 100  : rem switch on lamp (4)
70 poke dat ,16 : gosub 100  : rem switch on lamp (5)
80 poke dat ,32 : gosub 100  : rem switch on lamp (6)
90 end
100 for t = 1 to 700 : next :return : rem wait (loop)
```

If everything is all right, all the red lights should be lit and all outputs closed.

As you will see in the example above, every relay has its own number.

FIG-3.

RELAY	CONNECTION	POKE-DATA
1	1—2	1
2	3—4	2
3	5—6	4
4	7—8	8
5	9—10	16
6	11—12	32

Study these figures very carefully.

FIG-4.

RELAY	ON	OFF
1	OR 1	AND(63—1)
2	OR 2	AND(63—2)
3	OR 4	AND(63—4)
4	OR 8	AND(63—8)
5	OR 16	AND(63—16)
6	OR 32	AND(63—32)

If relay-2 has to be closed (1) You have to write:
poke(37136),peek(37136) or 2
If relay-2 has to be opened (0) You have to write:
poke(37136),peek(37136) and (63-2)
What happens next is the following, VIC reads the register as it exists at PEEK (37136) then we perform a logical operation 'OR' with '2' (data to

relay two is added to existing data). You can get further explanation about the OR function in your VIC-manual. As you will be reading about the OR function, why not take the opportity to read about the AND function also.

If you want to reset all the relays, just write: POKE 37136,0

But if you only want to reset a particular relay and leave the others in their actual state (see fig-4 and) write:
POKE 37163,PEEK(37163) AND 8
In this case only relay no. 4 will be reset. (poke data for relay no. 4 = 8. (see fig-3).

All the numbers which you write on the screen are in decimal. These are then interpreted by the VIC as hexadecimal or binary numbers.
Vic's user-port is a register containing 8 bits (1 byte).
OK! With this in mind we'll have a look at the convertion table:

FIG-5.

DECIMAL	HEXADECIMAL	BINARY
1	01	0000 0001
2	02	0000 0010
3	03	0000 0011
4	04	0000 0100
5	05	0000 0101
6	06	0000 0110
7	07	0000 0111
8	08	0000 1000
9	09	0000 1001
10	0A	0000 1010
11	0B	0000 1011
12	0C	0000 1100
13	0D	0000 1101
14	0E	0000 1110
15	0F	0000 1111
16	10	0001 0000
17	11	0001 0001 Etc.

Let us return to the first example. You switch on relay (1) with the code 1 or 0000 0001 (as the VIC sees it). This data is placed in the register which correspondens to the VIC's user-port. A close up: NOTE reversed data!!

FIG-6.

1	0	0	0	0	0	0	0	<= VIC USER-PORT WITH DATA.
R	R	R	R	R	R	I	I	
E	E	E	E	E	E	N	N	
L	L	L	L	L	L	P	P	
A	A	A	A	A	A	U	U	
Y	Y	Y	Y	Y	Y	T	T	
1	2	3	4	5	6	1	2	

1000 0000 = POKE DATA 1 (poke dat,1).

HOW TO READ THE INPUTS

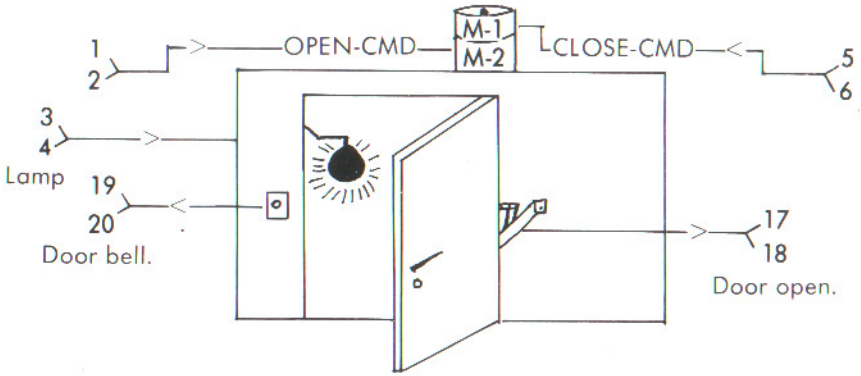
To read input no. 1
PRINT PEEK(37136) AND 64
If the answer = 0 then input one is active. (17—18 connected to 5—12v DC)

All other answers mean that input one is not active (Without 5—12v DC).
The same goes for input two except for the 'AND' data.
PRINT PEEK(37136) AND 128
NOTE! 128 for input two. See following example:

When input-1 has 5—12v (1), then "peek(37136) and 64" = 0
When input-1 loses 5—12v (0), then "peek(37136) and 64" = 64
When input-2 has 5—12v (1), then "peek(37136) and 128" = 0
When input-2 loses 5—12v (0), then "peek(37136) and 128" = 128

Let's say that the button on your front door (door bell) is connected to connections 19 and 20 (input-2). When someone presses it, the motor to the automatic door-opener starts up (connected via a power relay to output-1). When the door is fully open, it presses a button which is connected to input-1 that switches off the motor and waits a few seconds before switching on the light in the hall (connected via a power relay to output-2). And lastly, the motor for closing the door, motor two which is connected to output-3, starts up.

FIG-7.




```
10 POKE 37138,63 :REG=37136      Initiation
20 A = PEEK (REG) AND 128        Read inlet two (door bell button)
30 IF A <> 0 THEN 20             If not pressed read again
40 POKE REG,PEEK(REG) OR 1       Start door motor R1=1
50 B = PEEK (REG) AND 64         Read inlet one
60 IF B <> 0 THEN 50             Door not fully open yet
70 POKE (REG),PEEK(REG)AND       Open! switch off motor R1=0
   (63—1)
80 FORT = 1 TO 800 : NEXT T      Wait a few seconds.
90 POKE (REG),PEEK(REG) OR 2     Switch on the light in the hall. R2=1
100 POKE (REG), PEEK(REG) OR 4   Close the door, leave the light on.
110 FORT1=1 TO 1000 : NEXT T1    Wait a few seconds
120 POKE REG,PEEK(REG)AND        Closed! Stop motor. R3=0
   (63—4)
200 GOTO 20                      Be ready for the next guest.
```

NOTE! NOTE! NOTE! NOTE! NOTE! NOTE! NOTE! NOTE!!

Please Note! That the relays which are found in the VIC-I/O can only take a max. of 24 volts 10 watts and that the circuit in the VIC-I/O is not designed for the intended load (current) which may be needed for the motors and lamps. That is why it's recommended that one or more POWER RELAYS be installed by an electrician. To make VIC see input 1 and 2 you must activate them by applying 5—12v DC. This you may take from connection 14 which is (+5v) and connection 15 which is (—5v). Please note! MAXIMUM LOAD is 50 mA. This power is connected in series with some kind of switch, which will activate the inputs. NOTE! Make connections (—) minus to minus, and (+) plus to plus.

Look at bottom of first page for correct wireing.

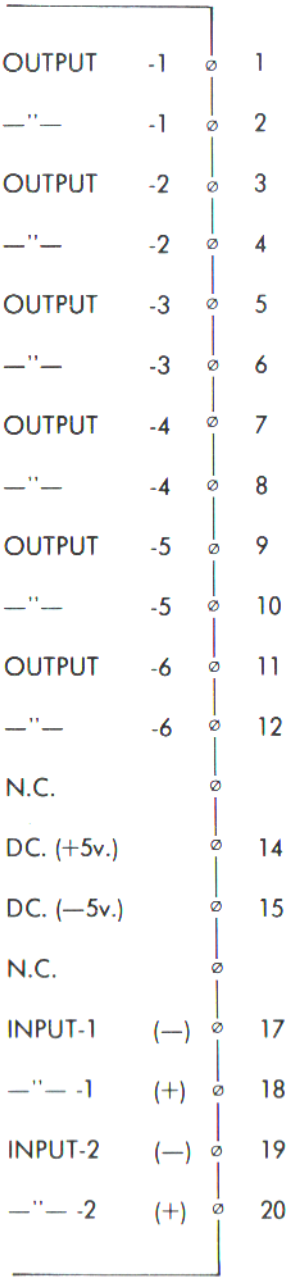
Next page contains the beginning of future experiments, here you might draw your own diagrams to VIC-REL. GOOD LUCK!.

VIC REL and CBM 64

Address 37136 is changed to 565 77
Address 37138 is changed to 565 79

VIC-RELAY-CONNECTIONS

OBJECT:
DATE:




WILLY-E
02 04
commodore
COMPUTER