

**LOGIC SOFT  
RELOCATER V1.6  
Z-80 RELOCATER PROGRAM FOR MACHINE CODE.  
ON  
NASCOM MICROCOMPUTERS**

The program is supplied on a self – generating tape cassette recorded on one side At 300 BAUD and the other at 1200 BAUD.

Once loaded to its normal locations (1000-15DO) it can then be relocated itself to Any free memory block 05DOH long. (Note: it will be necessary to change array Pointers in the main program if loaded into ROM).

On execution the program will ask nine questions to which it will require a user Input:

- 1) **ORIGIN** .                    The start address of the program to be relocated. The address to which the program was meant to start from but not necessarily its actual position.
  
- 2) **NEW ORIGIN**                The start address to where the program is to be relocated to.
  
- 3) **LOWER LIMIT**                The address below which code in the program to be relocated is not altered.  
E.G. if a limit of 1000H was set and during execution the Relocator finds a CALL to an address below 1000H then It will not relocate that code.
  
- 4) **UPPER LIMITS**                The opposite of the lower limit. Addressing above the limit is not altered.
  
- 5) **END ADDRESS**                The end address of the program that is to be relocated  
Note: this is the physical end address of the program Where it is actually located in RAM.
  
- 6) **1 BYTE TEST**                Answer this question either “Yes” or “NO” failure to do so will result in an assumed No. “Yes” will set up a mode such that if the relocater finds a ‘LD’ instruction for a single byte operand (eg : 3E,0e,06 etc.) then the relocater will stop and the prompt ‘c?’ will be shown . This is asking if you wish to change the code. Typing ‘n’ will cause the relocater to resume relocating where as ‘Y’ will position the cursor over the operand and allow it to be changed using the keyboard. Once satisfied press ENTER to continue relocation.

**7) ACTUAL ORIGIN** The actual first address of the program, to be relocated in RAM. Note: If the program is in its correct position relative to the origin then make ACTUAL ORIGIN = ORIGIN.

**8) OPTIONS** - (C D L M O Q W )

C: Copy relocated code to new origin.

D: Delay (about .75s @4Mhz )

L: List relocated code to a centronics printer (uses ports 4, 5,6, 7 )

M: Manual mode – waits for a key to be pressed before Proceeding . (Note: don't use ENTER key ).

O: Over copy – copies relocated code over the original Program.

Q: Query – if code is within the upper and lower limits Then it will halt the program and the prompt ' (YNC)?' Will be displayed.

'C' allows the code to be altered as described in the BYTE TEST mode.

'Y' allows the relocater to relocate the code.

'N' tells the computer not to relocate the code.

Enter one of these letters

W: Write – sends the VDU contents to the UART.

**9) DATA AREAS?** This allows the operator to define areas within the program which are to be ignored by the relocater but still to be copied in under the C and O commands. To exit from the mode simply press ENTER without inputting any address As many defined areas may be inputted as the operator Wishes as there are no restrictions on the array length. The First input (LOW) points to the first byte to be treated as Data and the second input (HI) to the byte to be treated as An instruction .(Note: make sure that a space between Inputs).

On inputting options up to 10 letters may be entered at any one time. This means that you may input as many delays or lists as you wish (up to 10 of course). ie for a delay of 1.5s input 'DD' .

It is important that the order of inputs is noted as the relocater operates on them as they have been inputted.

## RELATIVE RELOCATION

In the version (V1.6) the relative relocation facility has been fully supported. Eg if for example you have a program which starts at 0D00 and you want to relocate it to 0F00 but the program can only be loaded into 0C00 then the relocator can relocate

This to 0F00 if you input the following parameters:  
ORIGIN = 0D00H  
NEW ORIGIN = 0F00H  
ACTUAL ORIGIN = 0C00H

Then on running the program will relocate to code to 0F00 though it usually starts at 0D00 and is situated at 0C00H .

## PROGRAM STRUCTURE

To enable the relocator to be located in ROM it will be necessary to point array locations to address outside the relocator in RAM.

Two arrays are needed:

- 1) WORD: 10 bytes long: org 10B5H
- 2) DATA:XX bytes long: org 15D0H

The above address point to their current location.XX means that the length is dependent on the number of inputs.

$$\text{LENGTH} = \text{NUMBER OF INPUTS} * 4 + 2 \text{ bytes}$$

To rewrite the pointers alter location:

11D9: 1390:1332: for WORD array and  
14B4: 152D: for DATA array

Logic soft can not be held responsible for the corruption of any program or data arising from the misuse of this program. Logic soft also reserve the right to amend, update or otherwise alter this program, etc

Copyright © LOGIC-SOFT , SEPTEMBER 1981.  
Copyright © Hellenic Systems Ltd, 2006