
CRU ACCESS INSTRUCTION

Format CALL IO(type,bits,cru-base,variable,variable
 [,...])

The IO types 2 and 3 can be used to control devices. IO always must be the CRU address divided by 2 as any value above 8192 will be out of range. The cru-base must be divided by 2 as the 9901 chip ignores the least significant bits of the base register it uses. See Editor Assembler Manual page 61. The CRU data to be written should be right justified in the byte or word. The least significant bit will output to or input from the CRU address specified by the CRU base address. Subsequent bits will come from or go to sequentially higher CRU addresses. If the CRU input reads less than 8 bits, the unused bits in the byte are reset to zero. If the CRU input reads less than 16 bits but more than 8 bits, the unused bits in the full two 8 bit bytes will be reset to zero.

Programs

Line 100 display what it does for you.	>100 DISPLAY AT(1,1)ERASE ALL : "THIS PROGRAM CHECKS FOR UNUSUAL KEYS BEING PRESSED , EVEN IF MORE THEN FOUR KEY ARE BEING PRESSED AT ONCE"
Line 110 scans CRU at >0006 and reports keys pressed.	>110 CALL IO(2,16,3,A,B):: IF A=18 AND B=255 THEN 110 ELSE CALL HPUT(24,3,RPT\$(" ",30 ,24,24,STR\$(A)&" "&STR\$(B))
Line 120 more reports.	>120 IF A=146 THEN CALL HPUT(24,3,"FUNCTION KEY")ELSE IF B=191 THEN CALL HPUT(24,3,"C ONTROL KEY")ELSE IF B=223 TH EN CALL HPUT(24,3,"SHIFT KEY ")
Line 130 still more reports.	>130 IF B=251 THEN CALL HPUT(24,3,"ENTER KEY")ELSE IF B=2 53 THEN CALL HPUT(24,3,"SPAC E BAR")ELSE IF B=254 THEN CA LL HPUT(24,3,"PLUS/EQUAL KEY ")
Line start over scan of keys.	>140 GOTO 110