computer corner

Z-80 *A look at the nine addressing modes included in the Z-80 instruction set.*

WILLIAM BARDEN, JR.

LAST MONTH, WE DISCUSSED THE Z-80 INstruction set. This month, we will take a look at the various addressing modes available to the Z-80 programmer.

There are nine types of addressing included in the Z-80 instruction set, seven of these are included in the 8080 instruction set. The 8080 addressing types include the following:

1. Register Addressing
2. Implied Addressing
3. Register Indirect Addressing
4. Extended Addressing
5. Immediate Addressing
6. Immediate Extended Addressing
7. Modified Page Zero Addressing

The additional types of Z-80 addressing are:

8. Relative Addressing
9. Indexed Addressing

A register addressing type instruction specifies which CPU registers will be used in the instruction. The INC R instruction increments the contents of the specified register by one, for example. In this example, the C register is specified by the programmer.

Mnemonic	Instruction Configuration
INC C	00001100
	001 specifies register C
	000 register B, etc.

Implied addressing is used on any instruction in which the use of one or more CPU registers is implied. An example of this would be the one-byte instruction ADD B which adds the contents of the B register to the A register and places the result in the A register. Use of the A register is implied.

Mnemonic	Instruction Configuration
ADD B	10000000
	000 specifies register B
	001 register C, etc.

Register indirect addressing type instructions generally use the H,L register pair as a pointer to memory, although pairs B,C and D,E are also employed. The Z-80 block instructions for moves, searches and I/O are of this type, in addition to the more standard 8080 instructions. Here the instruction is a one-byte instruction and the H,L register pair holds the address of the operand in memory. If the H,L register pair held 2000_{10}, for example, the following instruction would add the contents of location 2000_{10} and the A register, and put the results in the A register.

Mnemonic	Instruction Configuration
ADD (HL)	10000110

Extended addressing is also used in the 8080. Here the instruction is three bytes long with the first byte specifying the operation code and the next two bytes specifying a 16-bit memory address of an operand *or* a jump location. The 8080 JMP instruction becomes JP in the Z-80.

Mnemonic	Instruction Configuration
JP 200H	11000011 00000000
	00000010
	byte 1 byte 2 byte 3, 200_{16}

Immediate addressing and immediate extended addressing are 8080-type addressing modes. In the first, the second byte of the two-byte instruction specifies an 8-bit operand to be used in the instruction. In the second type, the immediate operand to be used is 16-bits, or two bytes.

Mnemonic	Instruction Configuration
AND 7	11000110 00000111
	byte 1 byte 2
	The contents of the A register is logically AND'ed with 7 and the result placed in the A register.
LD BC,200H	00000001 00000000
	00000010
	byte 1 byte 2 byte 3, 200_{16}
	200_{16} is loaded into register pair B,C.

Modified page zero addressing in the Z-80, as in the 8080, is a special CALL instruction of one byte. The instruction is designated a *restart,* and causes a transfer to a page 0 location. It is used for commonly used subroutines or (usually) for interrupt processing for multi-interrupt capability. Page 0 is defined as locations 0 through FF_{16}.

Mnemonic	Instruction Configuration
RST 6	11110111
	110=6
	The contents of the program counter is pushed into the stack and the CPU jumps to page 0 location 6 × 8 or 48_{10}.

Relative addressing is not found in the 8080. The Z-80 uses this addressing mode only in jump type instructions. Here, the second byte of the two-byte instruction specifies an 8-bit signed *displacement* that is added to the current contents of the program counter to produce the jump address. Since the displacement may be +127 to −128, a relative jump can conditionally or unconditionally jump back −126 or forward +129 from the jump instruction (program counter points to next instruction after the jump). Since many jumps in a program are within this range, this instruction can be used to save one byte over an extended-type JP, and of course, to execute in a shorter time than a JP.

Mnemonic	Instruction Configuration
JR NC,10FOH	00110000 11101110
	byte 1 byte 2
	JR instruction is at location 1000_{16}. A jump is executed if carry flag = 1. The jump address is 1002_{16} (program counter contents) + EE_2 or $10FO_{16}$.

In indexed addressing, one of the two index registers IX or IY is used. The contents of the specified index register is added to an 8-bit displacement in the third byte of the three-byte instruction. The result is the memory address to be used. This is a very powerful type of addressing not found in the 8080.

Mnemonic	Instruction Configuration
ADD (IX + 20H)	11011101 10000110
	00100000
	byte 1 byte 2, byte3
	op code displacement
	If register IX held 2000H, the contents of location 2020H would be added to the contents of register A and the result would be placed in register B.

Many of the instruction types already discussed use several types of addressing modes. In general, the addressing types to be used for a given instruction should be the one that produces the shortest instruction to optimize both the number of bytes required to store the instruction and the time to execute it. **R-E**