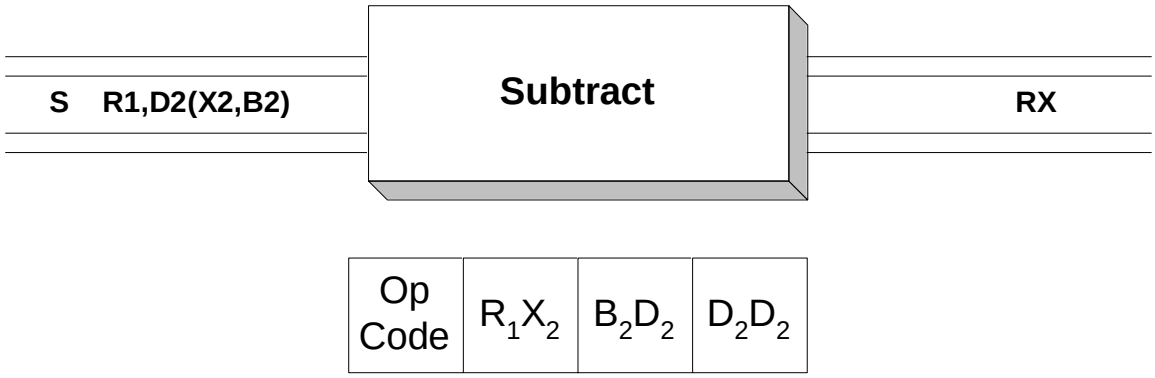
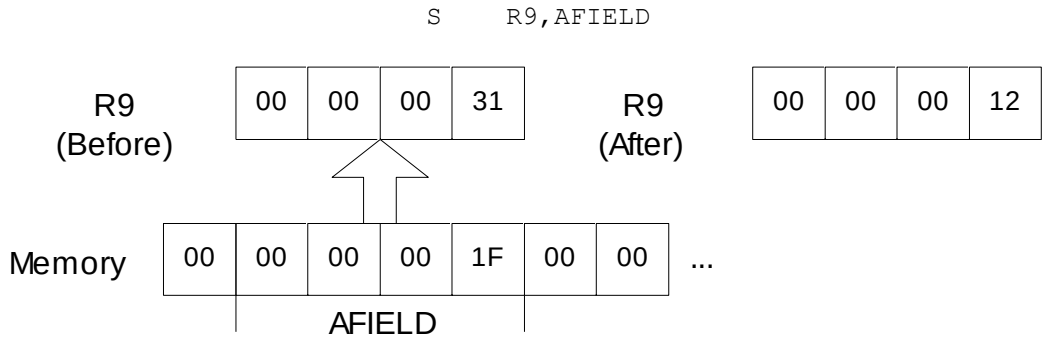


The Subtract instruction performs 2's complement binary subtraction. Operand 1 is a register containing a fullword integer. Operand 2 specifies a fullword in memory. The fullword in memory is subtracted from the fullword in the register and the result remains in the register. The fullword in memory is not changed. Consider the following example,



The contents of the fullword "AFIELD", x'0000001F' = 31, are subtracted from register 9 which contains x'00000031' = 49. The difference is 18 = x'00000012' destroying the previous value in R9. The fullword in memory is unchanged by this operation.

Since **S** is an RX instruction, an index register may be coded as part of operand 2 (see Explicit Addressing.

Examples

Some Unrelated Adds

```
R4 =  X'FFFFFFD5'  -43 IN 2'S COMPLEMENT
R5 =  X'00000028'  +40 IN 2'S COMPLEMENT
R6 =  X'00000004'   +4 IN 2'S COMPLEMENT

DOG   DC   F'35'
CAT   DC   F'4'

S     R4,=F'20'     R4 = X'FFFFFFC1' = -63
S     R5,=F'-20'    R5 = X'0000003C' = +60
S     R6,=F'20'     R6 = X'FFFFFFF0' = -16
S     R6,=F'-5'     R6 = X'00000009' = +9
S     R6,CAT        R6 = X'00000000' =  0
```

```
S     R5,DOG        R5 = X'00000005' = +5
S     R6,DOG(R6)    R6 = X'00000000'  INDEXING IS ALLOWED
```