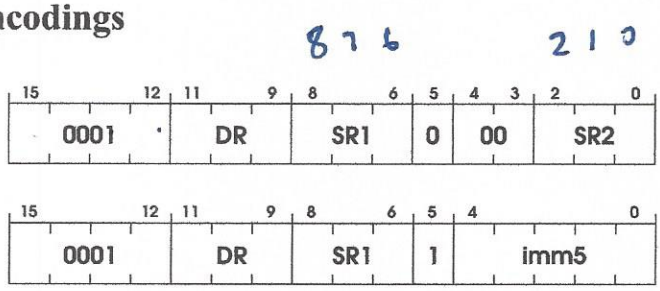
ADD

Addition

Assembler Formats

```
ADD   DR, SR1, SR2
ADD   DR, SR1, imm5
```

Encodings



Operation

```
if (bit[5] == 0)
  DR = SR1 + SR2;
else
  DR = SR1 + SEXT(imm5);
setcc();
```

Description

If bit [5] is 0, the second source operand is obtained from SR2. If bit [5] is 1, the second source operand is obtained by sign-extending the imm5 field to 16 bits. In both cases, the second source operand is added to the contents of SR1, and the result stored in DR. The condition codes are set, based on whether the result is negative, zero, or positive.

Examples

```
ADD   R2, R3, R4     ; R2 ← R3 + R4
ADD   R2, R3, #7     ; R2 ← R3 + 7
```

ADD

ALU: $S_0 = H$
$S_1 = L$
$S_2 = L$
$S_3 = H$
$M = L$

RF A ← SR2

RF R ON

ALU R_2W ON ← SR2 Loaded

ALU R_2W OFF

RF A ← SR1

ALU CE ON ← Result obtained

ALU CE OFF

RF R OFF

~~ALU R_2W OFF~~

RF A ← DR

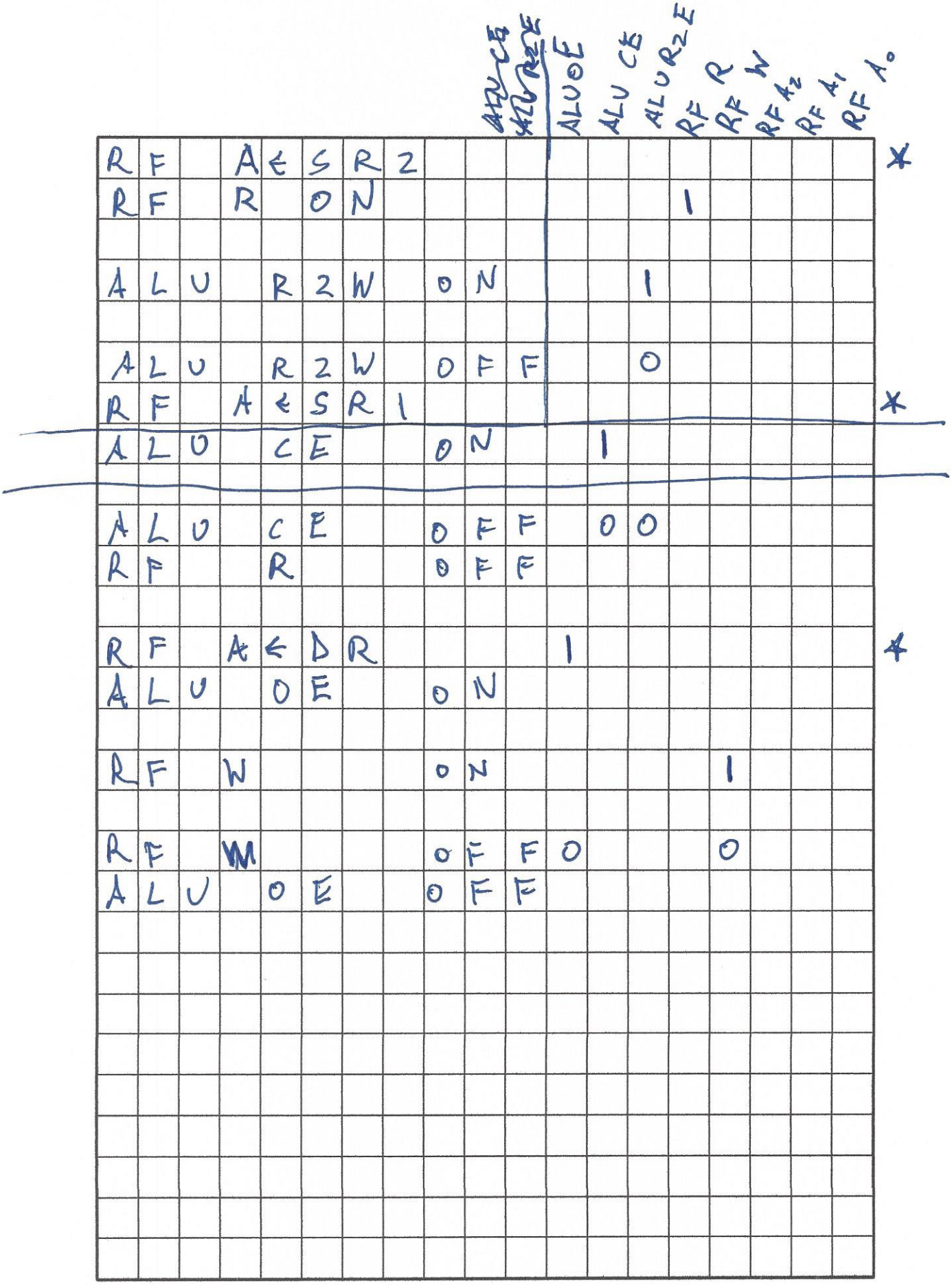
~~RF W ON~~

ALU OE ON

RF W ON ← DR Written

RF W OFF

ALU OE OFF


ALU CE
ALU R2E
ALU OE
RF R
RF W
RF A2
RF A1
RF A0
RF A ← SR2
RF R ON
ALU R2W ON
ALU R2W OFF
RF A ← SR1
ALU CE ON
ALU CE OFF
RF R OFF
RF A ← DR
ALU OE ON
RF W ON
RF W OFF
ALU OE OFF