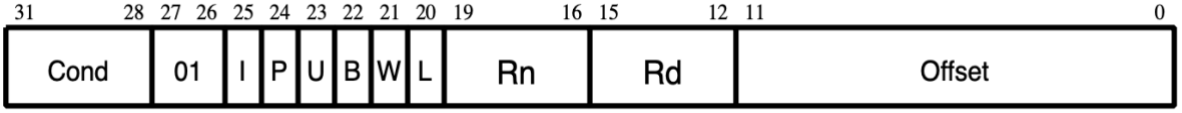
# Format of Half word and Signed Data Transfer Instructions in ARM

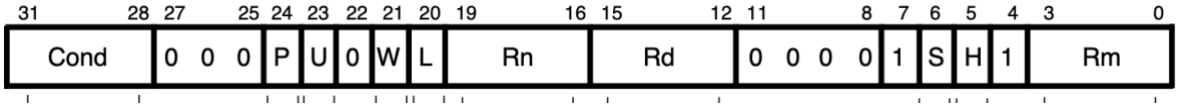
Half word and Signed DT Instructions in ARM have a different format than the full word and byte transfer instructions. For convenience, I am reproducing the relevant information from the document that had been posted on moodle earlier - "arm-instructionset.pdf".

First, let us compare the overall structure of the two formats.

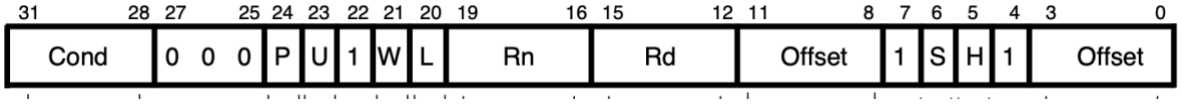
(a)Full word and byte DT (ldr, str, ldrb, strb)




(b) Half word and signed DT with offset in register




(c) Half word and signed DT with immediate offset




Strangely, F field in (b) and (c) is "00" and I-bit (bit 25) is '0' which are same as that in DP instructions with operand2 in register! However, it does not clash with DP instructions which have a '0' either in bit 4 position (constant shift amount) or in bit 7 position (register specified shift amount), whereas these instructions have both these bits as '1'.

P, U, W and L bits have same meaning as in (a). Interpretation of S and H bits is as follows.

| S | H | |
|---|---|---|
| 0 | 0 | swap (swp) - - not to be implemented |
| 0 | 1 | unsigned half word load or store (ldrh or strh) see Lbit |
| 1 | 0 | signed byte transfer load (ldrsb) |
| 1 | 1 | signed half word load (ldrsh) |

Note that signed/unsigned distinction is only for loads, not stores. This is so because a byte or a half word has to be extended to make a full word while loading into register file, whereas there is no such extension in case of store.

Also note that in (a), there is a provision for shift/rotate of offset when the offset is given by a register, but there is no such provision in (b).

Figures on the next page elaborate the formats further. Full details about the instructions are available in "arm-instructionset.pdf".

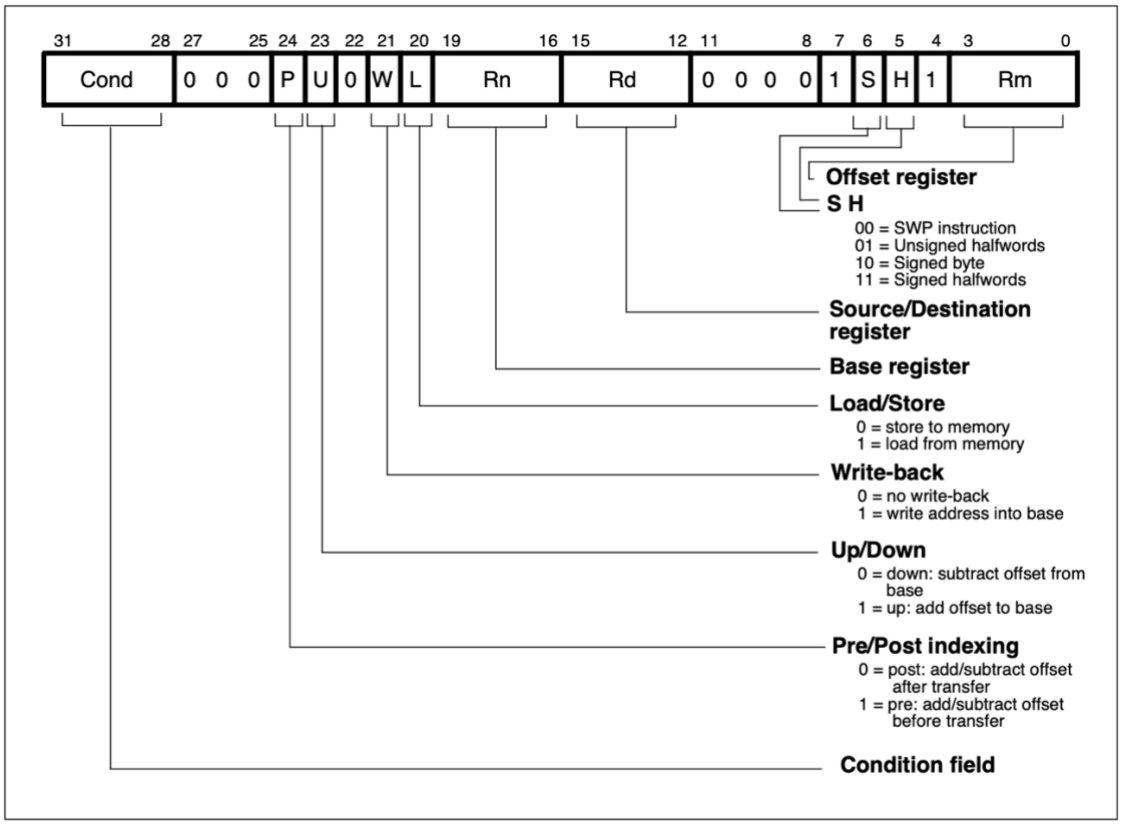



***Figure 4-16: Halfword and signed data transfer with register offset***

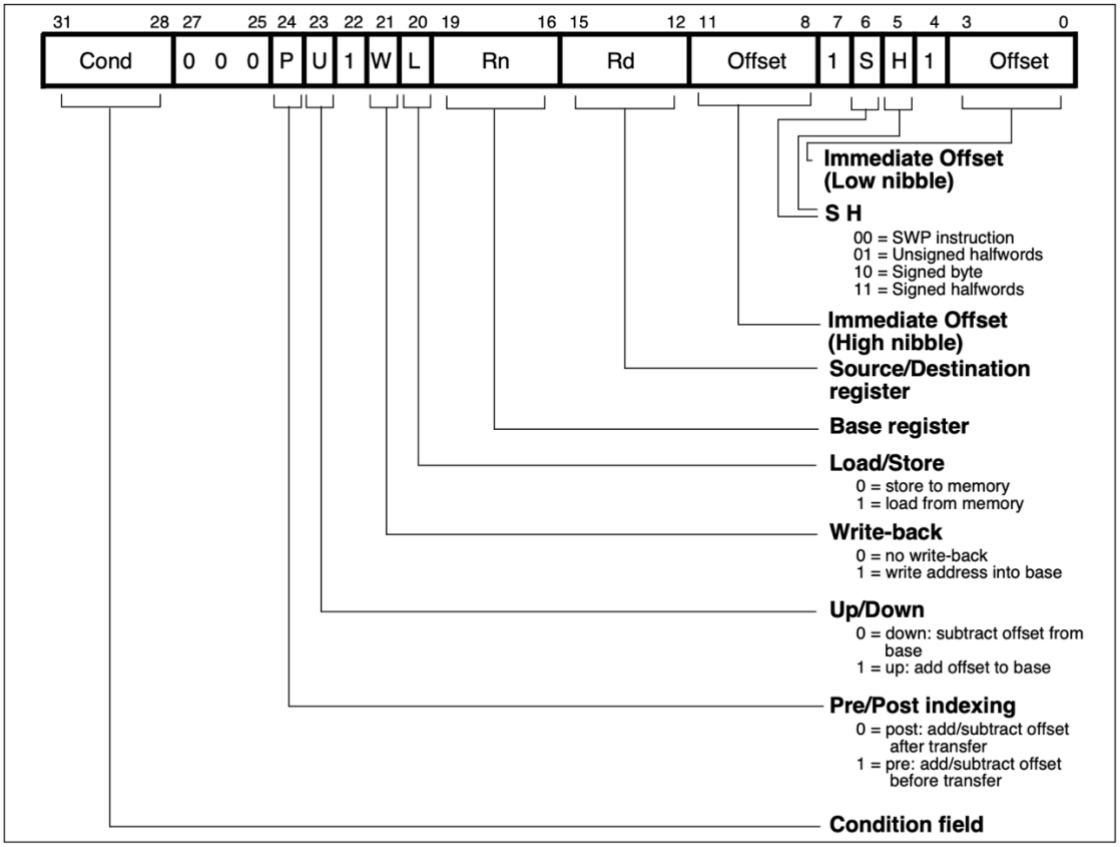



***Figure 4-17: Halfword and signed data transfer with immediate offset***