# 7.2.1 FLP Multiplier

## Hardware

* Use 4-bit fractions and 4-bit exponents.
* Negative numbers are represented in ***2's complement***.

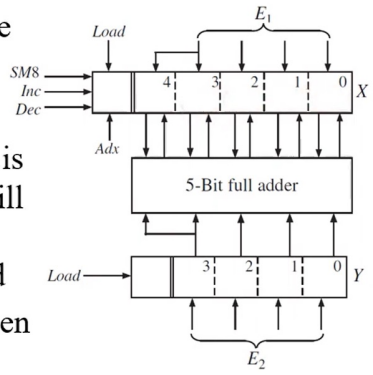
- Major components of the FP multiplier:
  - **Exponent adder**: uses a 5-bit full adder for the 2's complement exponents
  - **Fraction multiplier**: implements a ***shift*** and ***add*** multiplier algorithm for the 2's complement fractions (Refer to the faster multiplier in §4-10)
  - **Major control unit**: provides the signals to perform the appropriate ops of ***right shifting***, ***left shifting***, ***exponent incrementing/decrementing***, and so forth.

datapath

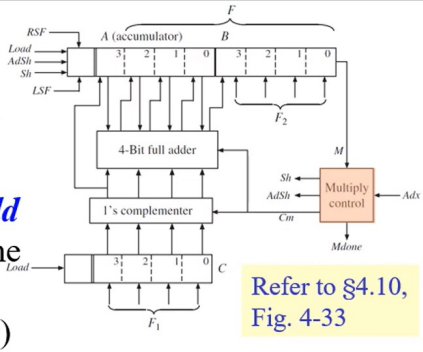


## Exponent Adder

- Exponent adder:
  - uses a 5-bit full adder for the 2's complement exponents
  - ***Load***, ***Adx***
  - ***Inc***, ***Dec***: when the fraction is normalized, the exponent will have to be correspondingly incremented or decremented
  - ***SM8***: in the special case when product is 0, the register should be set to the value **1000**



# Fraction Multiplier




Refer to §4.10, Fig. 4-33

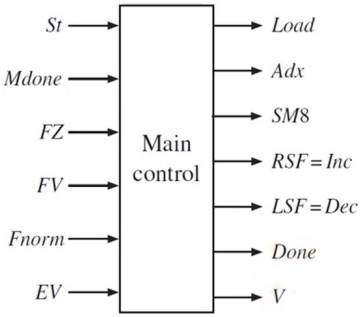
- Fraction multiplier:
  - implements a ***shift*** and ***add*** multiplier algorithm for the 2's complement fractions (faster multiplier in §4-10)
  - Multiply 3 bits plus sign by 3 bits plus sign.
    ⇒ The result will be 6 bits plus sign, i.e., the lower 3 bits of A concatenated w/ B.
  - Multiply control: generates appropriate shift and add signals depending on the multiplier bits
    - The controller is *linked* into the main controller: ***Adx***, ***Mdone***

linked state machine
(similar to procedure call)



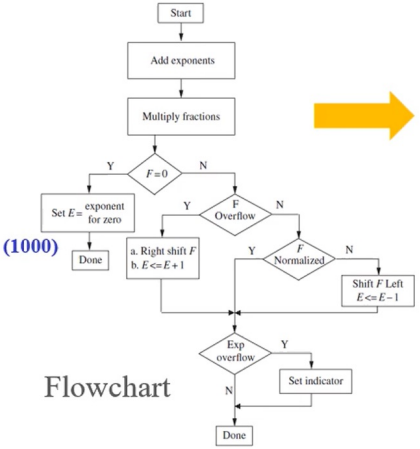
# Major Control Unit

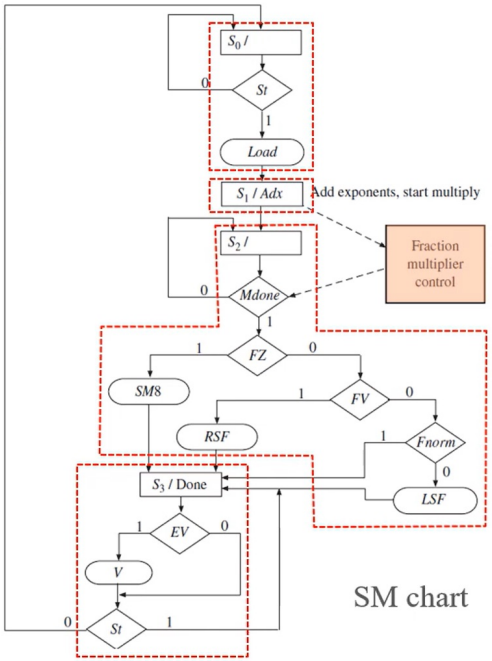
- Main control unit:
  - provides the signals to perform the appropriate operations of ***right shifting***, ***left shifting***, ***exponent +1/−1***, ….



# SM Chart of *Main Controller*

- SM chart for floati
  point multiplicatio




- ***St:*** Start the floating-point multiplication.
- ***Mdone:*** Fraction multiply is done.

**Input signals**

- ***FZ:*** Fraction is zero.
- ***FV:*** Fraction overflow (fraction is too big).
- ***Fnorm:*** $F$ is normalized.
- ***EV:*** Exponent overflow.

---

- ***Load:*** Load $F_1$, $E_1$, $F_2$, $E_2$ into the appropriate registers; also clear $A$.

**Control signals**

- ***Adx:*** Add exponents; also starts the fraction multiplier.
- ***SM8***: Set exponent to − 8 (1000, to handle special case of 0).
- ***RSF:*** Shift fraction right; also increment $E$.
- ***LSF:*** Shift fraction left; also decrement $E$.

---

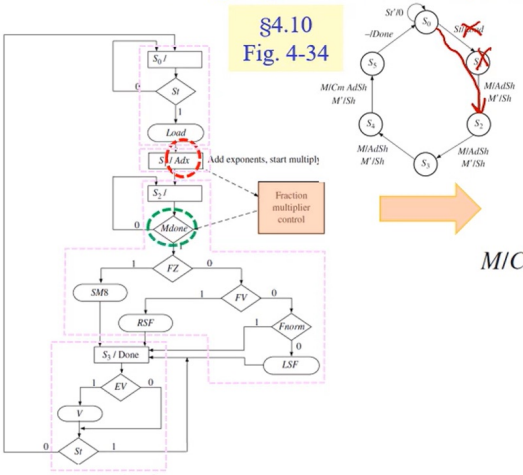
**Output signals**

- ***V:*** Overflow indicator.
- ***Done:*** Floating-point multiplication is complete.

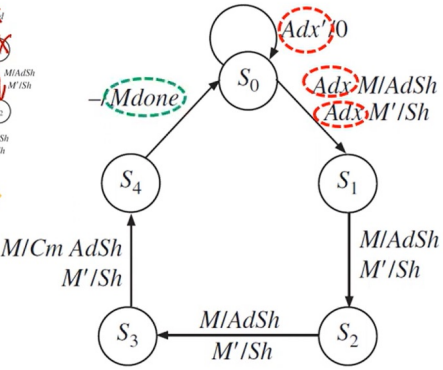
# State Graph of *Multiplier Control*

- State graph for multiplier control:

§4.10
Fig. 4-34




SM chart




State graph

*** "*Load*" has been done in main controller!**