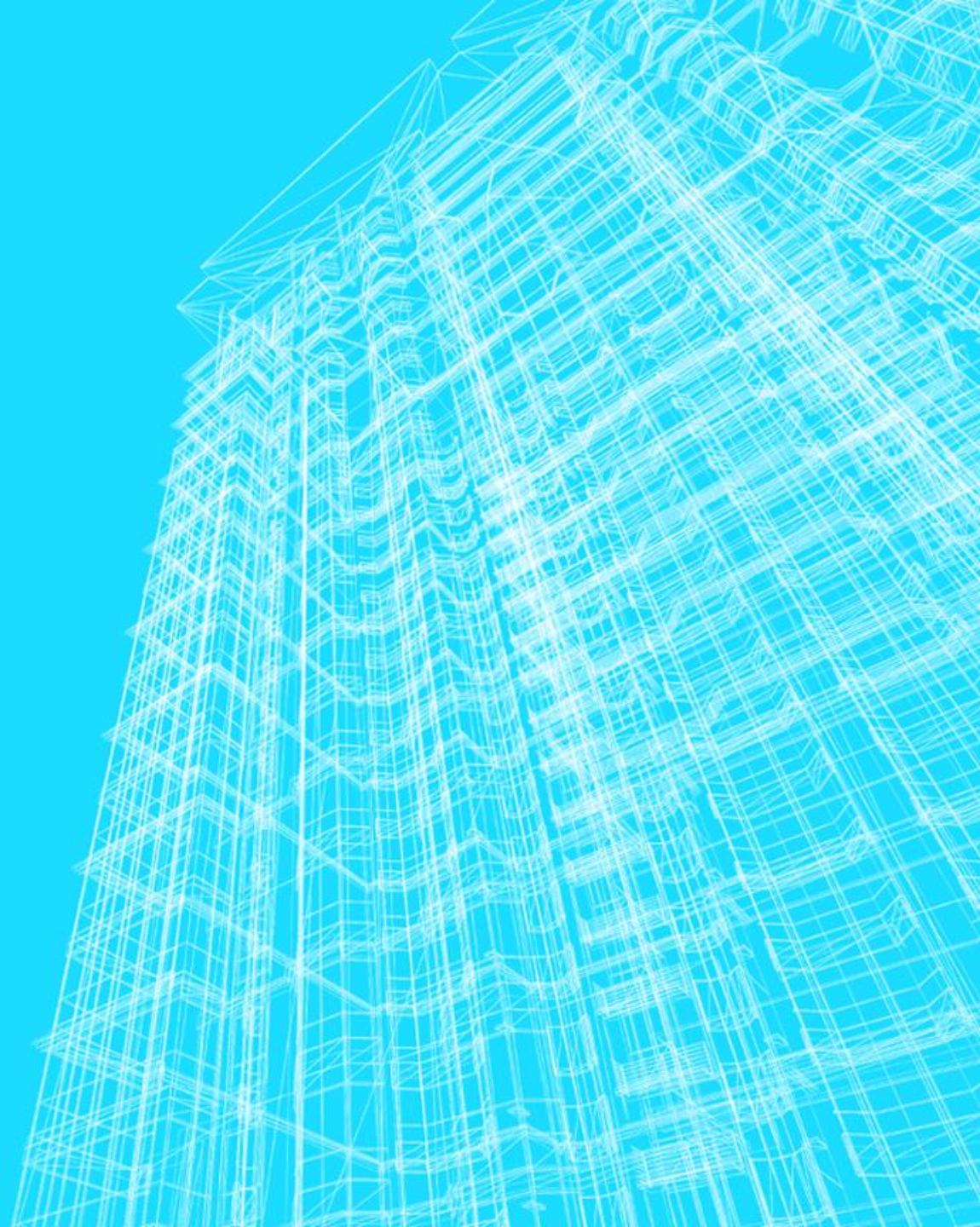
[64 16] FX CALCULATOR

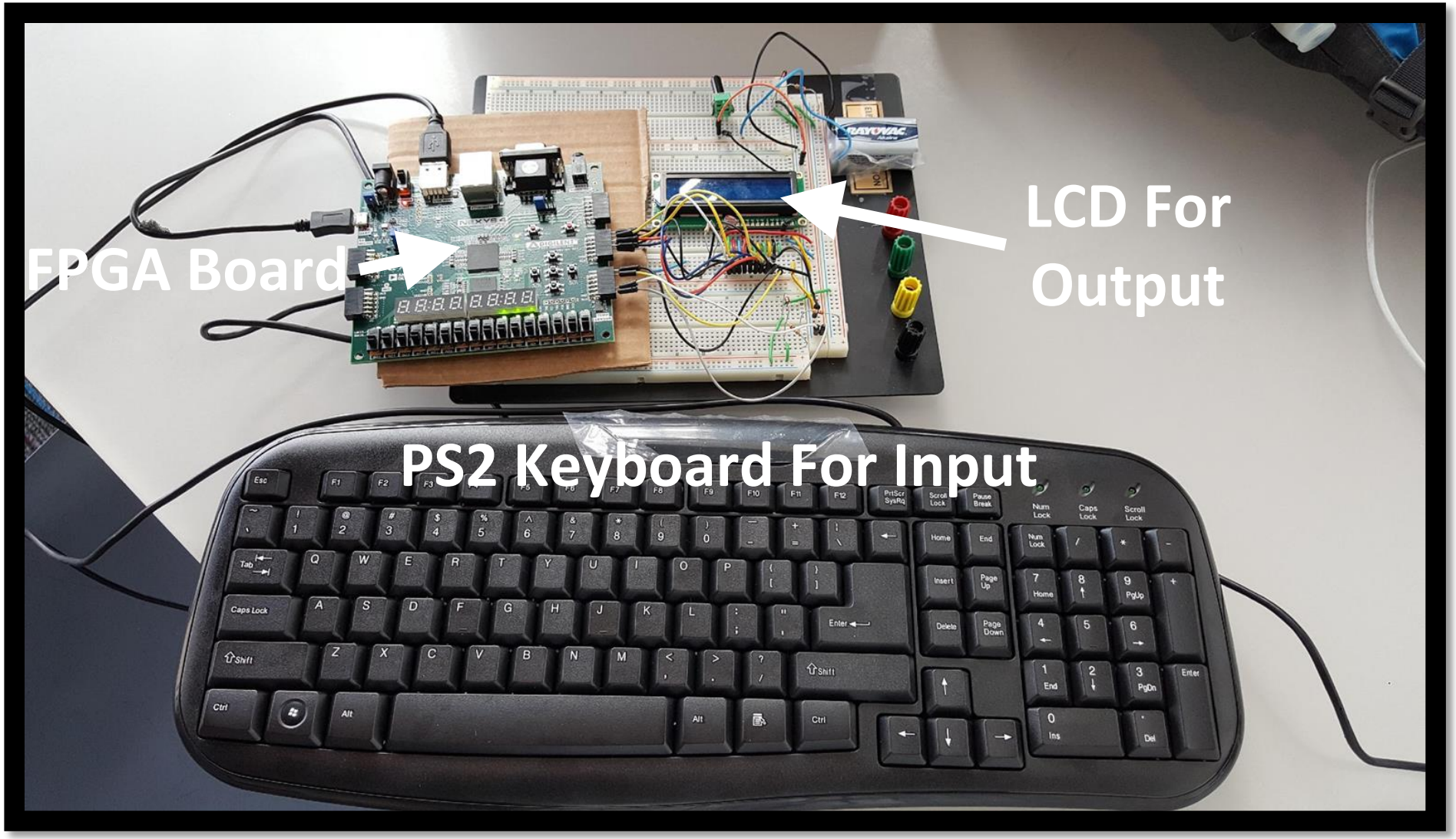
ECE 508, W2017
Santosh Epuri
Jacob Morales



TOPICS TO BE COVERED TODAY

- Overview of hardware
- Calculator Input Output Specifications
- Keyboard input of operands
- Mathematical functions (addition, subtraction, multiplication, division)
- LCD output of result
- Project challenges
- Live Demonstration

OVERVIEW OF HARDWARE

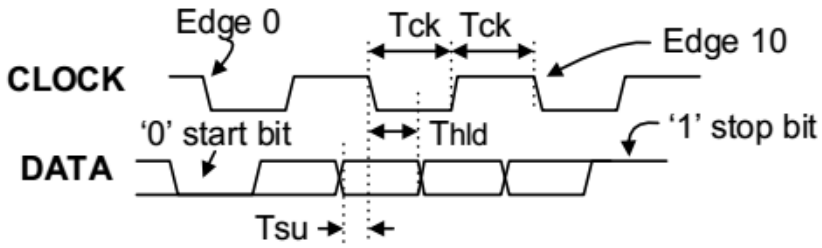


Calculator Input Output Specifications

- Resolution: $2^{-16} = 0.000015259$
- Input range: $[-2^{47}, 2^{47}-2^{-16}]$ =
  - -140,737,488,355,328 to +140,737,488,355,327.999984741
- Output range: $[-2^{95}, 2^{95} - 2^{-16}]$ =
  - -3.961408125713217e+28 to +3.961408125713217e+28

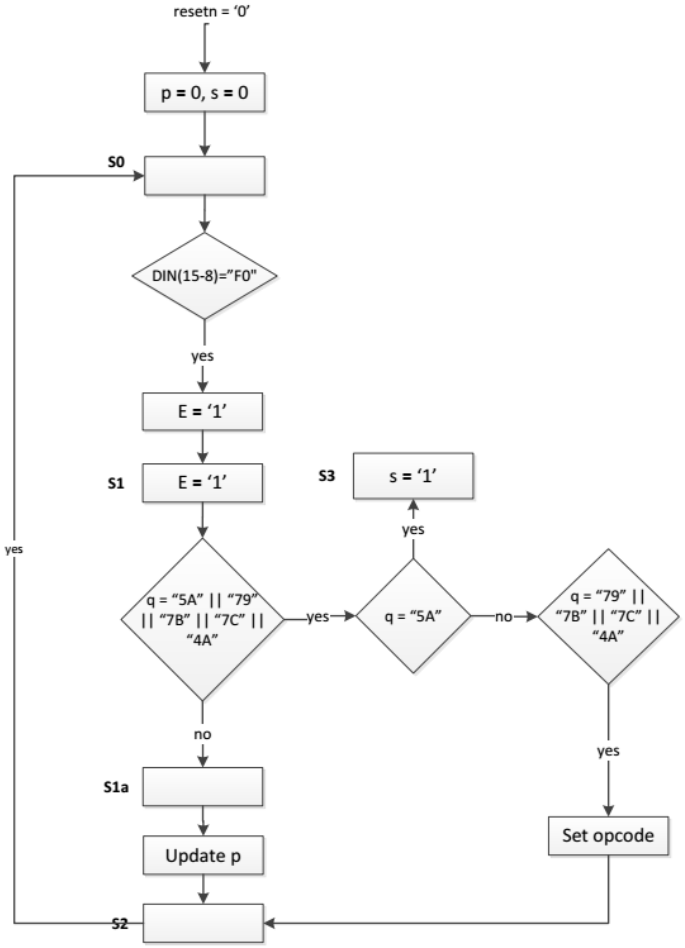
KEYBOARD INTERFACE

- PIC μC hides the USB HID protocol and emulates PS/2 bus
- PS/2-style keyboards use scan codes to communicate key press data
- If the key is held down, the scan code will be sent repeatedly about once every 100ms
- When a key is released, an F0 key-up code is sent, followed by the scan code of the released key



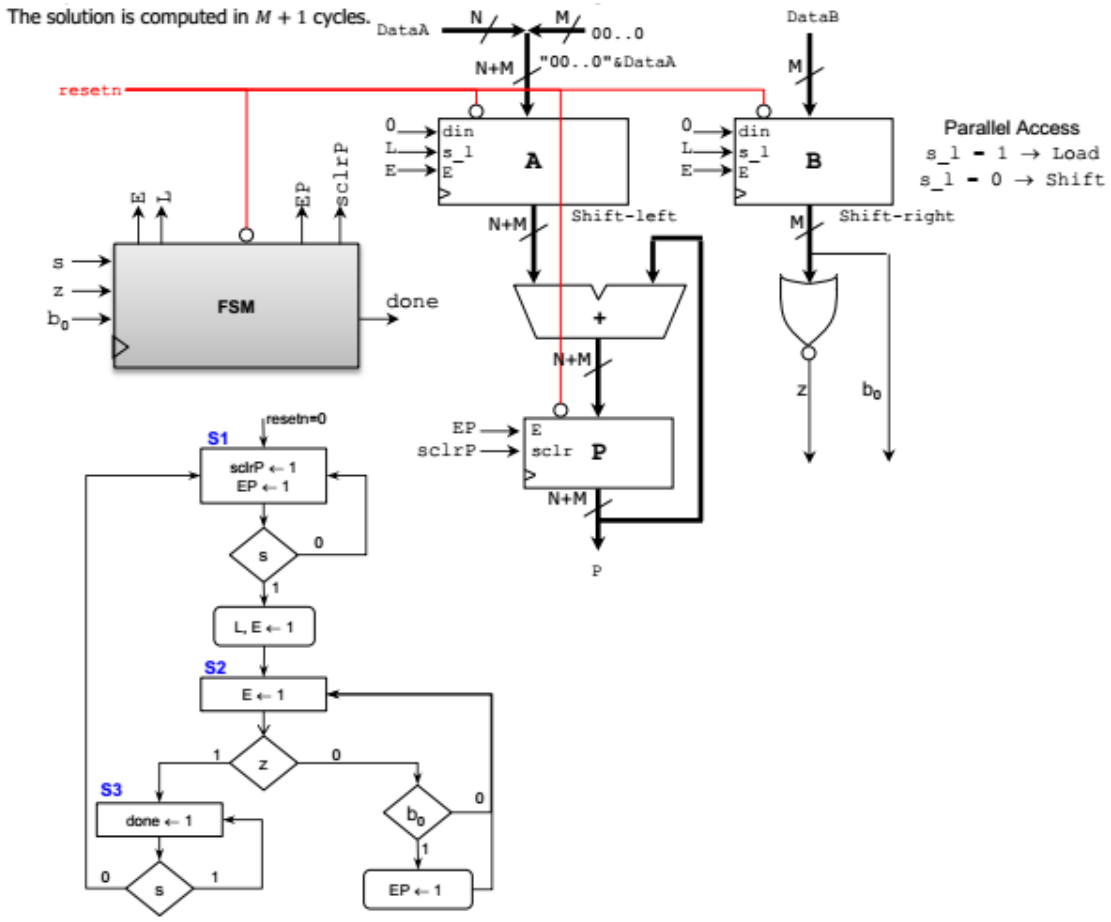
Symbol	Parameter	Min	Max
T_{CK}	Clock time	30us	50us
T_{SU}	Data-to-clock setup time	5us	25us
T_{HLD}	Clock-to-data hold time	5us	25us

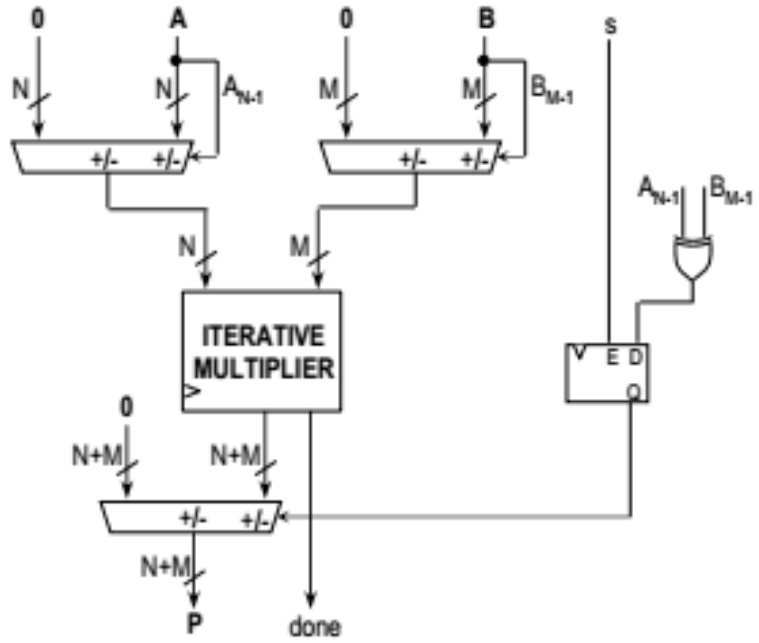
KEYBOARD INPUT OF OPERANDS



MATHEMATICAL FUNCTIONS

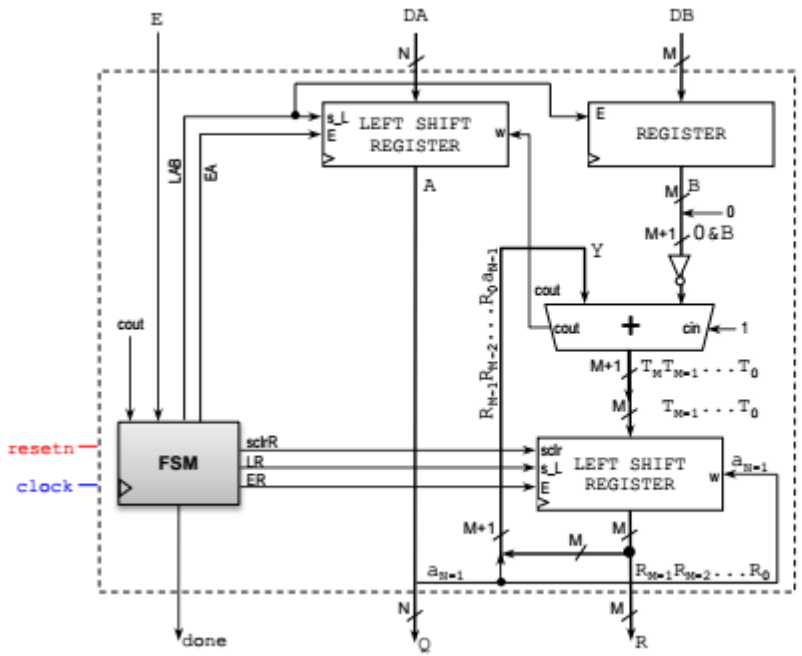
- Multiplication

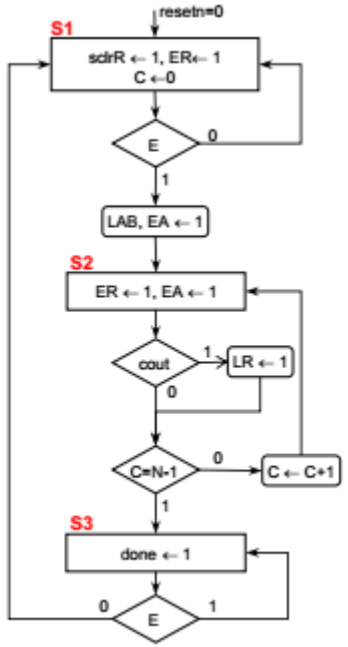


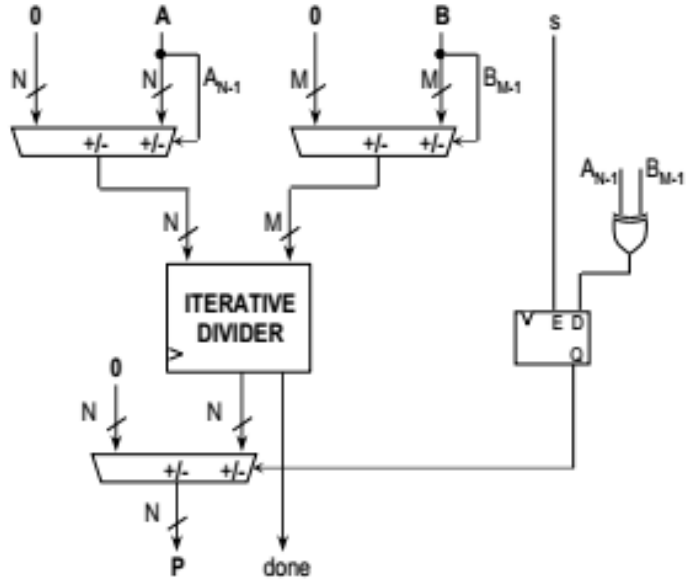


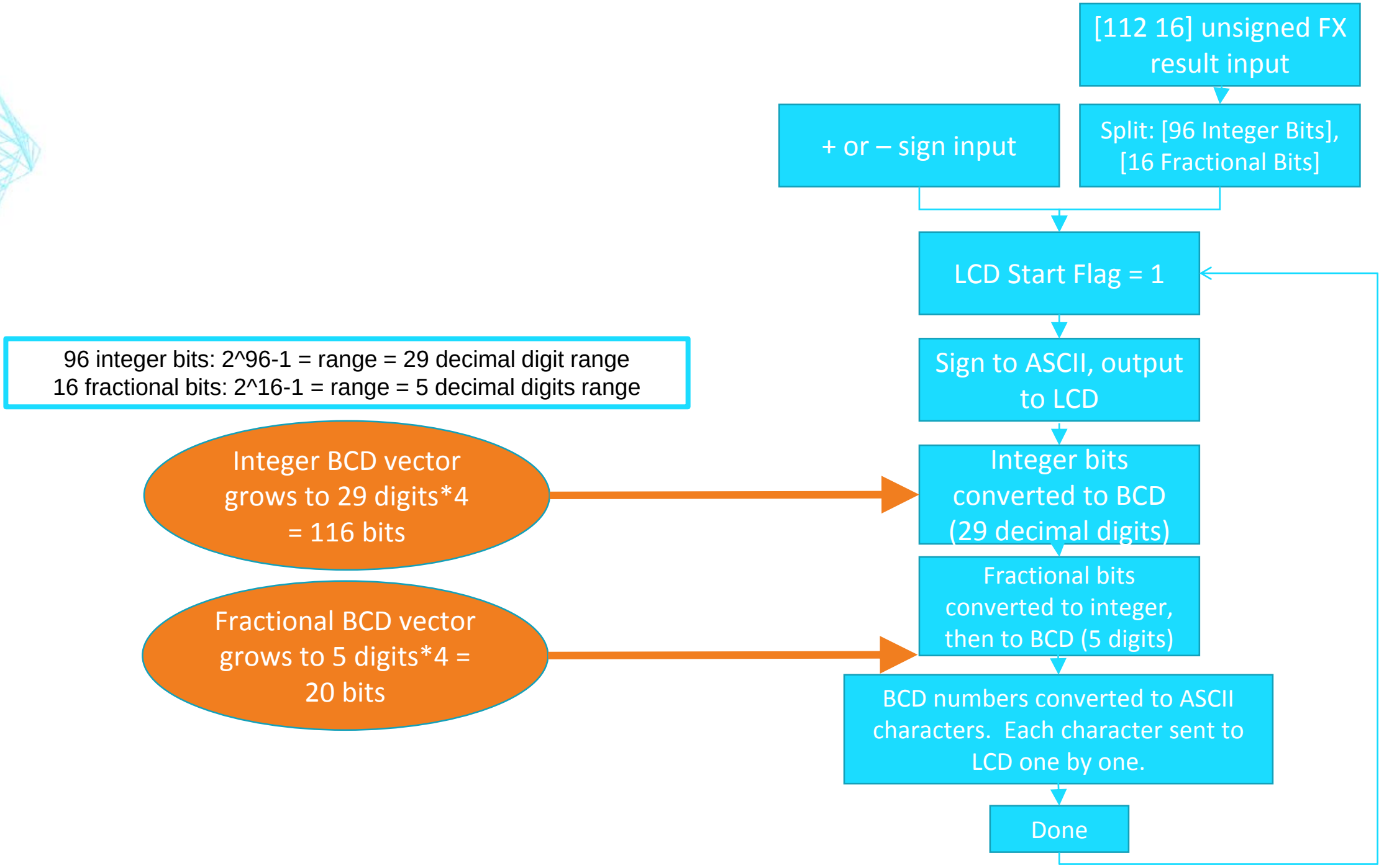
MATHEMATICAL FUNCTIONS

- Division (Appended 16 0's to the dividend for improved resolution)








[112 16] unsigned FX result input
+ or – sign input
Split: [96 Integer Bits], [16 Fractional Bits]
LCD Start Flag = 1
Sign to ASCII, output to LCD
Integer bits converted to BCD (29 decimal digits)
Fractional bits converted to integer, then to BCD (5 digits)
BCD numbers converted to ASCII characters. Each character sent to LCD one by one.
Done
96 integer bits: 2^96-1 = range = 29 decimal digit range
16 fractional bits: 2^16-1 = range = 5 decimal digits range
Integer BCD vector grows to 29 digits*4 = 116 bits
Fractional BCD vector grows to 5 digits*4 = 20 bits

PROJECT CHALLENGES

- Matlab Quantizer not capable of supporting full [64 16] range
  - Matlab Quantizer uses double floating point arithmetic to convert from decimal to hex, therefore the function will do strange rounding to your input number and the binary output will not be the same as the input.
  - 1 week was spent trying to troubleshoot the binary to BCD converter due to this issue. BCD output values did not match quantizer decimal input.

BE CAREFUL!

```
0101100111010011100111100111111100111011010000000000000000000000

>> b = num2bin(a,098765432109887.9980)

b =

0101100111010011100111100111111100111011010000000000000000000000

>> b = num2bin(a,098765432109887.9982)

b =

0101100111010011100111100111111100111011010000000000000000000000

>> b = num2bin(a,098765432109887.9984)

b =

0101100111010011100111100111111100111011010000000000000000000000

>> b = num2bin(a,098765432109887.9986)

b =

0101100111010011100111100111111100111011010000000000000000000000

>>
```

- Syntax issues
  - Assignment is NOT the same as equality check (Vivado will not flag an error)
    - b <= "1000" when a = "00" else **not the same as** b <= "1000" **when** a <= "00"
- Keyboard input functionality troubleshooting
  - We spent hours troubleshooting the keyboard input after fixing the VHDL code issues and it turned out that power cycling the FPGA board solved the problem.

Live Demonstration!

User Instructions:

1. Power on calculator (FPGA) and wait for initialization
2. Input first operand using keyboard (16 hex values 0-F)
3. Input operator using keyboard (+,-,/,*)
4. Input second operand using keyboard (16 hex values 0-F)
5. Press Enter
6. Press BTNU on FPGA to print result.
7. Press BTNL and BTNR to scroll display left and right to see all digits.

Note the following:

1. 16 hex values * 4 bits = 64 bit input.
2. Input is shifted in from the right so input your operands as you would read them from left to right (MSB to LSB).
3. Last 4 hex values represent fractional bits (4*4 = 16) [64 16]
4. 7 segment display will show the 64 LSB values of a 128 bit vector signal which holds both operands as they are shifted in from the right.
5. Output from mathematical operation is in [112 16] format. 16 LSB were truncated from [128 16] result.