

# COL216 Computer Architecture

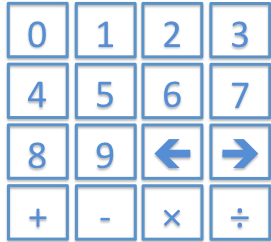
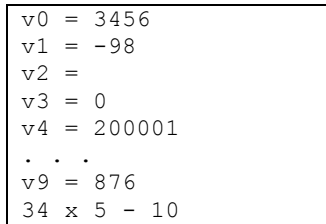
## Lab Assignment 2

ARMSim# simulator has a plug-in called “Embest Board Plug-in” which simulates a particular ARM processor board. It provides the following virtual input/output devices.

1. One 8-segment display (output).
2. Two red LED lights (output).
3. Two black buttons (input).
4. Sixteen blue buttons arranged in a keyboard 4 x 4 grid (input).
5. One LCD display screen, which is a grid of 40 columns by 15 rows.

How to use these devices is explained in the user guide already posted on moodle.

Using this plug-in, create a simple 4 function calculator with 10 variables. The specifications are given below.

- The calculator evaluates expressions entered using the blue buttons and assigns the result to a variable. The expressions involve operands, four types of operators : add (+), subtract (-), multiply ( $\times$ ), and divide ( $\div$ ). The operands can be signed integer constants or variables  $v_0, v_1, \dots, v_9$ .
- The blue keys are designated as digits ‘0’ to ‘9’, operators ‘+’, ‘-’, ‘ $\times$ ’, ‘ $\div$ ’, *assign* ‘ $\leftarrow$ ’ and *recall* ‘ $\rightarrow$ ’, as shown in the figure. The digit and operator keys have their usual meaning. The *assign* key is pressed at the end of an expression. It causes the expression to be evaluated and the result assigned to a variable. The *recall* key is pressed while entering an expression to bring in the value of a variable as an operand.
- The two black buttons select the variable to which a value is assigned or whose value is recalled. The variables have indices 0 to 9. Pressing the right button causes the variable with the next higher index to be selected and pressing the left button causes the variable with the next lower index to be selected. Index of the selected variable is displayed on the 8-segment display.
- Current values of all the variables are displayed in the 40 $\times$ 15 screen as shown in the figure. Values of the variable that have never been assigned so far are considered as undefined and are shown as blanks. The current expression is displayed in the 11<sup>th</sup> line as it gets entered on the keyboard.
- The LEDs indicate two kinds of errors. The left LED is turned on to indicate an attempt to recall an undefined value and the right LED is turned on to indicate an overflow in the current computation.

- The operators have their usual precedence. That is, unary minus and plus, followed by multiply and divide, and lastly add and subtract.
- Note that you would need to implement divide operation. Secondly, the constant operands entered in decimal will need to be converted into binary and the computed results will need to be converted into decimal before displaying.

Submission schedule:

a) 5<sup>th</sup> February: High level design

b) 19<sup>th</sup> February: Final report