Assembler Code review:

A list of all ARM assembler instructions we have covered in class. The midterm will use the math and the ARM assembler so you can think of this part of the assignment as being study preparation for the midterm.

Lists of Assembly Commands ------------------------------------

Registers & Numbers:

R1, R2, R3,...,R15 [Registers that hold int values]

F1, F2, F3,...,F15 [Registers that hold floating point values]

D1, D2, D3,...,D15 [Registers that hold double precision values]

#1, #2, … [Calling a numerical value you want to call]

Loading variables:

MOV R0, R1 [“Moves” value of R1 into R0]

[is used to move data from one CPU register to

another CPU register.]

LDR R0, R1 [“Loads Register” R0 with value of R1]

[ is used to move data from memory (usually RAM) into a CPU register.]

Mathematical Operations:

[ (Operation) <Destination> <Value 1> <Value 2> ]

[<Values> are interchangeable with “Registers” or “Numerical Values” ]

ADD R0, R0, R1 [“Addition” function R0 = R0 + R1]

SUB R0, R0, R1 [“Subtraction” function R0 = R0 - R1]

MUL R0, R0, R1 [“Addition” function R0 = R0 \* R1]

DIV R0, R0, R1 [“Division” function R0 = R0 / R1]

MLA R0, R0, R1, R2 [“Multiply and Add” function R0 = R0 \* R1 +

R2]

Logical Commands:

\*Command that must be first when comparing values and making decisions

[ (Commands) <value 1> <value 2>]

CMP R0, R1 [“Compares” R0 with R1]

\*Commands that come after CMP, that determines how R0 and R1 are being compared, then calls function to do other processes

[ (Commands) {Call function} ]

BGE [“Branch Greater Than or Equal To” === “>=” ]

BGT [“Branch Greater Than” === “>” ]

BLT [“Branch Less Than” ==== “<” ]

BLE [“Branch Less Than or Equal To” === “<=” ]

BNZ: !=0 [“Branch Not Equal To 0” === “!= 0” ]

BZ: equal zero [“Branch Equal To 0” === “=0” ]

BEQ: == [“Branch Equal To” === “=” ]

NF: != [“Branch Not Equal To” === “!=” ]

Bit Manipulation:

\*1 = True, 0 = false

ADC [“Add And Carry”]

ADCS [“Add And Carry Special”]

AND [“And” === true only when 1&1 ]

OR [“Or” === true when 1&0, 0&1, 1&1 ]

XOR [“Exclusively Or” === true when 1&0, 0&1 ]

Return Function:

\*when returning, must LDR or MOV answer into R0 to print out answer

BX LR

Commenting Out:

\*Way of writing notes in code

@ [Single line commenting]

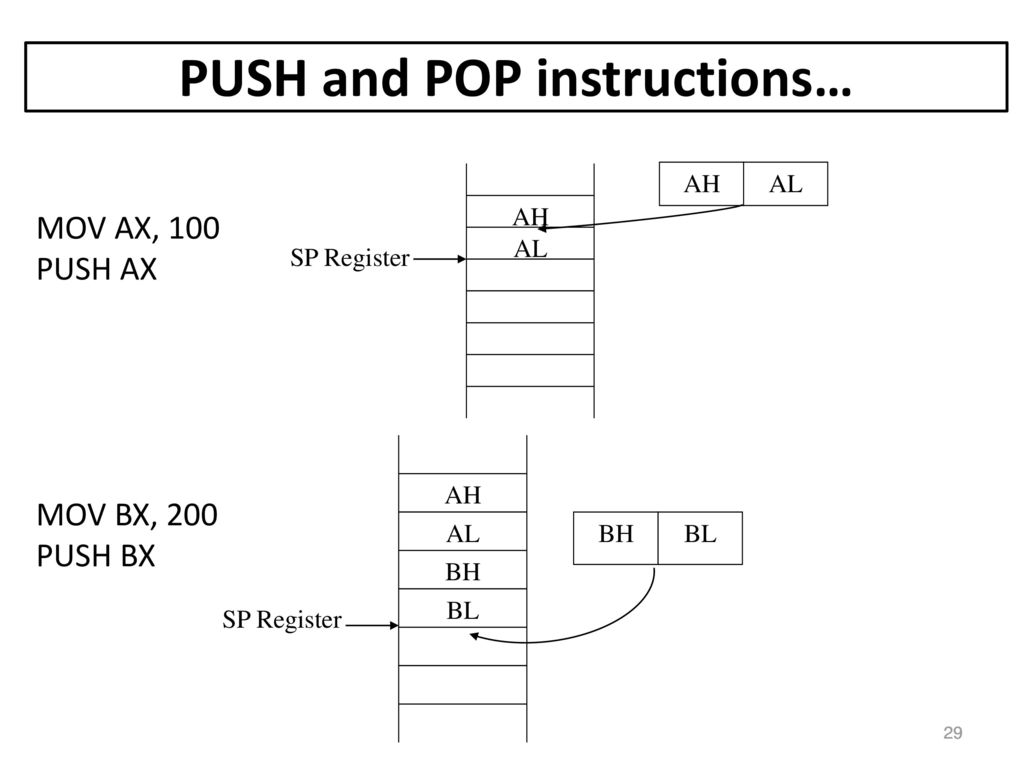
/\* <notes> \*/ [Multiple line commenting]

Push and Pop mean:

Push: go down in memory

Pop: take value of register back and push it back to original value

Diagram:



Additional ARM Command Resources:

