

# LAB 3-A

## STACK MEMORY

---

### OBJECTIVES:

- To examine the stack.

### MATERIAL:

- Atmel Studio.

### WEB SITES:

- [www.microchip.com](http://www.microchip.com) for Atmel Studio Software

### ACTIVITY 1

Write and assemble a program to load values \$20, \$31, \$42, \$53, and \$64 into each of registers R20 to R24 and then push each of these registers onto the stack. Single-step the program, and examine the stack and the SP register after the execution of each instruction.

### ACTIVITY 2

Write and assemble a program to:

- Set SP = \$29D,
- Store (without using push operation) a different value 6, 5, 4, 3, 2, 1 in RAM locations \$29D, \$29C, \$29B, \$29A, \$299, and \$298, respectively
- POP each stack location into registers R20 – R24.
- Use the simulator to single-step and examine the registers, the stack, and the stack pointer.

**From Activity 1 and 2, answer the following questions:**

- Upon reset, what is the value in the SP register? **RAMEND**
- Upon pushing data onto the stack, the SP register is \_\_\_\_ **Decrementd.** \_\_\_\_ (decremented, incremented).
- Upon popping data from the stack, the SP register is \_\_\_\_ **Incremented.** \_\_\_\_ (decremented, incremented).

- Yes, you can change it by writing to the SPH and SPL registers** To support multitasking in operating systems where each task needs its own separate stack for context switching. It is also used to organize memory by placing the stack in a safe, reserved area to prevent it from accidentally overwriting critical variables.

Processor Status	
Name	Value
R12	0x00
R13	0x00
R14	0x00
R15	0x00
R16	0x9D
R17	0x00
R18	0x00
R19	0x00
R20	0x01
R21	0x02
R22	0x03
R23	0x04
R24	0x05
R25	0x00
R26	0x00
R27	0x00
R28	0x00
R29	0x00

  

Memory 4	
Memory:	data IRAM
data 0x0248	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0254	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0260	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x026C	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0278	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0284	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0290	00 00 00 00 00 00 00 00 00 01 02 03 04 .....
data 0x029C	05 06 00 00 00 00 00 00 00 00 00 00 .....
data 0x02A8	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02B4	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02C0	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02CC	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02D8	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02E4	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02F0	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x02FC	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0308	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0314	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0320	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x032C	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0338	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0344	00 00 00 00 00 00 00 00 00 00 00 00 .....
data 0x0350	00 00 00 00 00 00 00 00 00 00 00 00 .....