# PEEK AND POKE

## ASSIGNMENT-04

BASATI SIVAKRISHNA

22976 (MTech EPD)

**Code Explanation:**

From the previous assignment, we are adding two more features

1. Peek mem\_loc

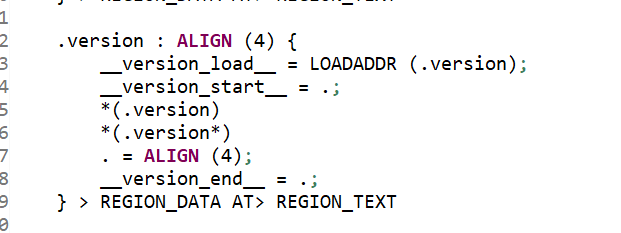
This command should be useful to search the given location from the memory and show the data in Serial monitor. Mem\_loc should be hex value.

1. Poke mem\_loc “\_data\_”

This command should replace the data from the mem\_loc to size of data+mem\_loc with “\_data\_”. If the mem\_loc+data size exceeds the limit of aur accessible data then it should not replace and should show the Memory overflow.

**Creating Symbol in Linker script:**

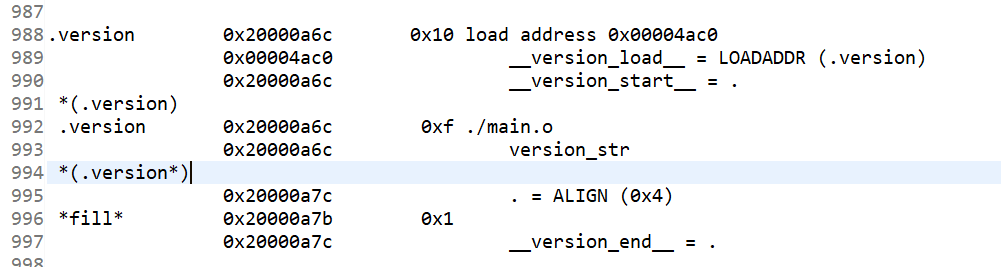
To achieve this, we should create a symbol in linker script to get a memory allocation of dynamic number of bytes. Here iam declaring the section in SRAM.



Once we declare a variable with this attribute then there will be memory allocation according to the size of the variable in .map file



Respective .map file for the above declarations in linker script and main.c file(16 bytes has allotted in this case).



**LCD initialization and Printing Data on LCD screen:**

To initialize LCD, we need to send series of commands and data bytes via data pins of LCD. To switch from command mode to data mode we should pull the RS pin accordingly.

**void** **LCD\_Initialization**();

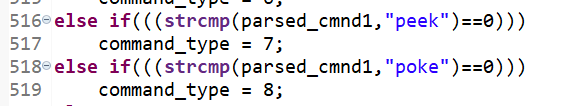
**void** **writeStringLCD**(**char** s[], **int** line);

These two functions will be useful to initialize and show the data on screen.

The line argument in writestring() will take 1 or 2 as input and prints the data on first line or 2nd line of the 16\*2 LCD.

**PEEK AND POKE:**

Based on the previous assignment, these commands has assigned with the command\_type of 7 and 8 respectively.



1. peek mem\_loc

Given mem\_loc(in ascii format) will be converted to hex value and typecasted as char pointer. Data from that location is accessed using dereferencing operator and respective byte will be printed on the serial monitor.

Output from the serial monitor for valid peek command:

A screenshot of a computer program

Description automatically generated

For invalid address:

If the entered location after peek is not in our defined range the serial monitor should print the accessible range.

A black text with a white background

Description automatically generated with medium confidence

1. Poke mem\_loc “\_data\_”

Given mem\_loc(ascii value) will be converted to the hex value and typecasted to the char pointer. And the \_data\_ will be parsed and stored in the poke\_data variable. And the size \_data\_ will be stored in the poke\_index variable.

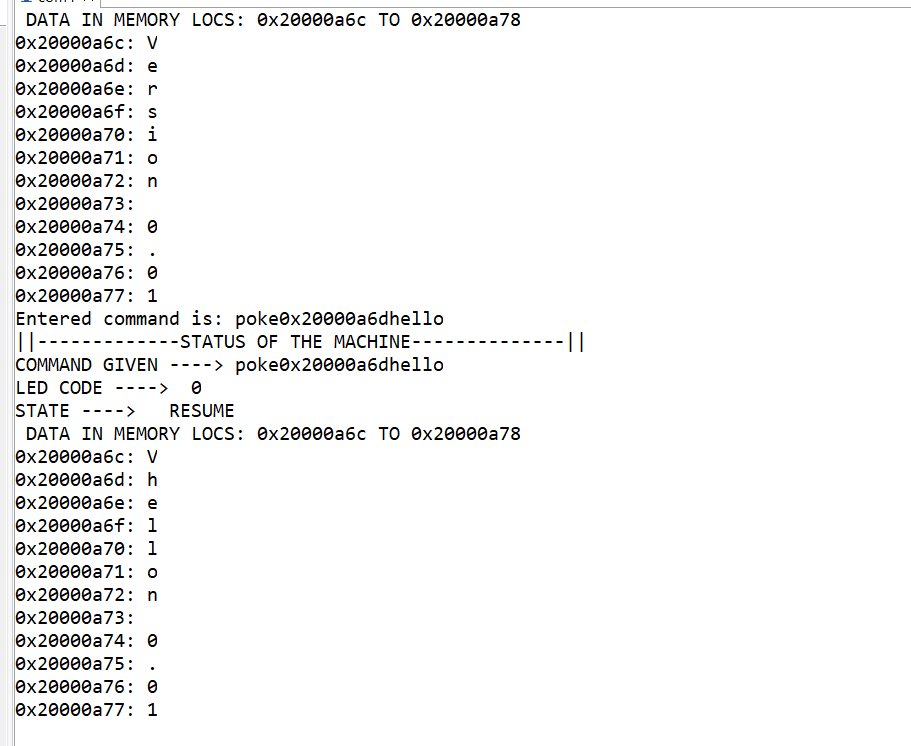
The parsed \_data\_ will be written from the mem\_loc to

mem\_loc+ size of data.

If the mem\_loc \_ size of data exceeds the memory that we a lot for the .version section than serial monitor will show the “DATA OVERFLOW” error.

Note: Here we are not giving the size of data in command, it is getting calculated in the program itself.

Poke command(valid):



Invalid poke command:

A white screen with black text

Description automatically generated

LCD output:

Before poke command

A yellow rectangular object with a black background

Description automatically generated with medium confidence

After poke command

A yellow rectangular object with a black background

Description automatically generated