Homework Assignment #1

Due Friday Feb 6th @ 5PM

1. Important Information

- This is an individual assignment. Any evidence that code was shared between individuals will be treated as academic misconduct.
- All code should have meaningful comments. A 5 point penalty will be levied against each problem that is not commented sufficiently.
- Any code that does not assemble/compile will receive no points
- All code is to be written in ARM assembly using the Keil uVision IDE.
- The maximum point total for the assignment is 50 points
- A basic test bench has been provided for you. If your code passes this test bench, you are guaranteed to receive 40 out of the possible 50 points. Your code will be examined against a more exhaustive test bench. Feel free to add any additional tests to the test bench to verify your code.

2. Problem Overview

In this homework, you will use ARM assembly instructions to calculate the total deposits and withdrawals of electronic banking records. After the last transactions is encountered, the overall total of the transactions will be saved to a specific address in the SRAM.

The application code will need to convert the transaction logs from ASCII strings to their binary equivalent to perform the required operations to sum all of the transactions. Once a total value has been calculated, you will store the total as an ASCII string. The formats of the transactions and the final calculation are defined below.

a. Bank transactions are stored at a label called TRANSACTIONS. The maximum deposit/withdrawal is $999.99. Each of the transactions is 8 bytes long and have the following format

Table 1: Transaction Format

Byte Number	Description
0	Indicates if the transaction was a deposit or withdrawal. The valid values are ASCII characters '+', '-', or binary value 0.
1	ASCII Character between 0 and 9
2	ASCII Character between 0 and 9
3	ASCII Character between 0 and 9
4	Decimal Point
5	ASCII Character between 0 and 9
6	ASCII Character between 0 and 9
7	NULL Character (binary value of 0)

Table 2: Transaction Examples

Transaction Description	Transaction Data
Deposit $100.00	ASCII String: "+100.00"
Deposit $1.25	ASCII String: "+001.25"
Withdraw $54.12	ASCII String: "-054.12"
Withdraw $900.00	ASCII String: "-900.00"

b. The transactions will be initialized by the test bench. You can open a serial debug interface (8N1, 115200 Baud) to observe the output of the test bench.

3. Requirements

a. Fill in your in the FLASH area of ece353_hw1.s

b. Allocate the required space in the SRAM for 1000 transactions. The name of this space **must** be TRANSACTIONS.

c. Validate each transaction. Each transaction must adhere to the requirements found in Table 1. If there is an error in **any** byte of **any** transaction, set the TRANSACTION_TOTAL global variable to the ASCII string "CALC ERROR". Remember to terminate all strings with the NULL character (binary value of 0).

d. The sequence of transactions is terminated when the first byte of **any** transaction is a binary value of 0.

e. Allocate 12 bytes in the SRAM for **TRANSACTION_TOTAL** global variable. This global variable will be used to store the total of all the transactions OR to indicate that there was a formatting error in the sequence of transactions.

   The data stored at this label is an ASCII string indicating the transaction total or the error message described in item c above. (Examples +$99,123.00 or -$54,678.12):

Table 3

Byte Number	Description
0	Indicates if the total was positive or negavite. The only two valid characters for byte 0 is a ‘+’ or a ‘-‘
1	ASCII Character ‘$’
2	ASCII Character between 0 and 9
3	ASCII Character between 0 and 9
4	ASCII Character ‘,’
5	ASCII Character between 0 and 9
6	ASCII Character between 0 and 9
7	ASCII Character between 0 and 9
8	ASCII Character ‘.’
9	ASCII Character between 0 and 9
10	ASCII Character between 0 and 9
11	NULL Character (Binary value of 0)

f. All of your ARM assembly code must be written after the hw1 label in ece353_hw1.s. **Do not add BL, BLX, PUSH, POP, or code that calls or returns from a function**. This has already been provided for you.

4. Assignment Submission

You will submit ece353_hw1.s to the drop box on the course website.akshay