

# Lab4 (complicated)

**Due** Oct 21, 2022 by 11:59pm


**Points** 100

**Submitting** a file upload

**File Types** s

Hi,

here is our lab: [Lab5C](https://www.engr.scu.edu/~dlewis/book3/labs/Lab5C.pdf)  (<https://www.engr.scu.edu/~dlewis/book3/labs/Lab5C.pdf>)

Empty assignment file: [Lab5C-Empty.s](https://camino.instructure.com/courses/83789/files/6295557?wrap=1) (<https://camino.instructure.com/courses/83789/files/6295557?wrap=1>)  ([https://camino.instructure.com/courses/83789/files/6295557/download?download\\_frd=1](https://camino.instructure.com/courses/83789/files/6295557/download?download_frd=1))

## Step1: original C-code -----

```
// Function to implement in assembly
int32_t __attribute__((weak)) MxPlusB(int32_t x, int32_t mtop, int32_t mbtm, int32_t b){
    int32_t rounding ;

    rounding = ((( (mtop*x*mbtm) >> 31) * mbtm) << 1) + mbtm) / 2 ;
    return (mtop*x + rounding) / mbtm + b ;
}
```

## Step2: break the operations up, and use register names as variables: -----

```
// Function to implement in assembly
int32_t __attribute__((weak)) MxPlusB(int32_t R0x, int32_t R1mtop, int32_t R2mbtm, int32_t R3b){
    int32_t R4 =
        (
            (
                (
                    (
                        R1mtop * R0x * R2mbtm
                    ) >> 31
                ) * R2mbtm
            ) << 1
        ) + R2mbtm
    ) / 2 ;
    return (R1mtop * R0x + R4) / R2mbtm + R3b ;
}
```

## Step3: Make it almost look like assembler: -----

```
// Function to implement in assembly
int32_t __attribute__((weak)) MxPlusB2(int32_t R0x, int32_t R1mtop, int32_t R2mbtm, int32_t R3b){
    int32_t R4 = R1mtop * R0x;
    int32_t R5 = R4 * R2mbtm;
    int32_t R6 = R5 >> 31;
    int32_t R7 = R6 * R2mbtm;
    int32_t R8 = R7 << 1;
    int32_t R9 = R8 + R2mbtm;
    int32_t R10 = R9 / 2 ;
    int32_t R11 = R1mtop * R0x;
    int32_t R12 = R11 + R10;
    R0x = R12 / R2mbtm;
    return R0x + R3b;
}
```

Step4: finally translate it line by line into assembler commands, that's it :)