IIIT BANGALORE

VLS 864
Embedded Systems Design

Project Report

Submitted By:

Omkar Vijay Gavandi- MT2024524

Rachaputi Likithkumar – MT2024528

Vaibhav Naresh Dachewar- MT2024539

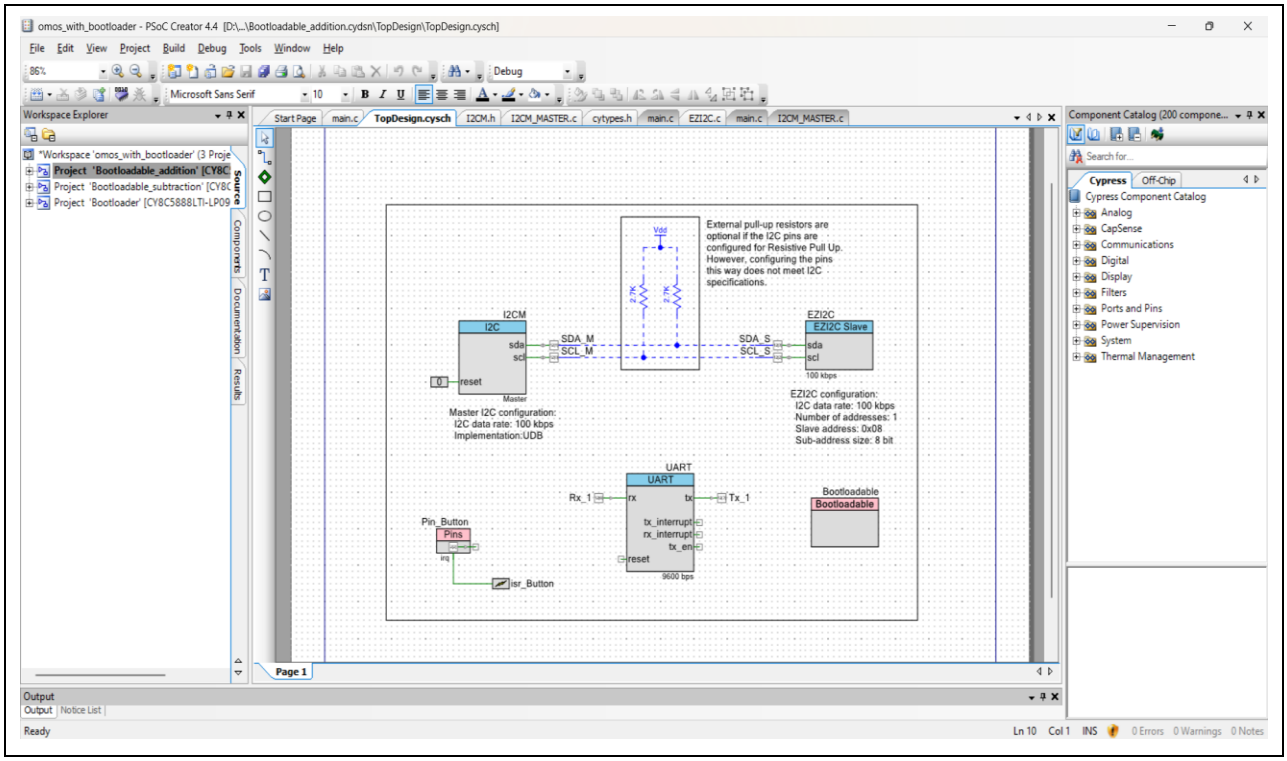
Submitted To:

Prof. Kurian Polachan

VLSI Lab,

International Institute of Information Technology- Bangalore

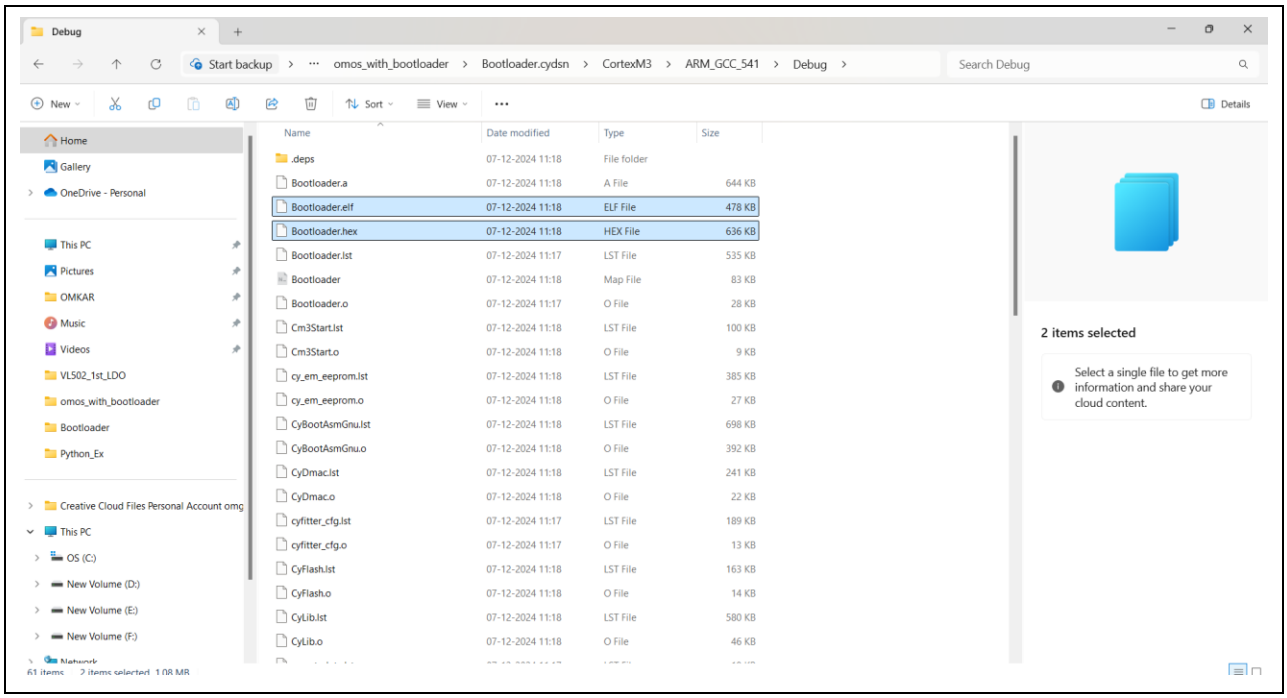
Schematic:-



We have total three projects. The first one is for the bootloader and the rest are for performing operations like addition and subtraction of the buffer which is defined initially.

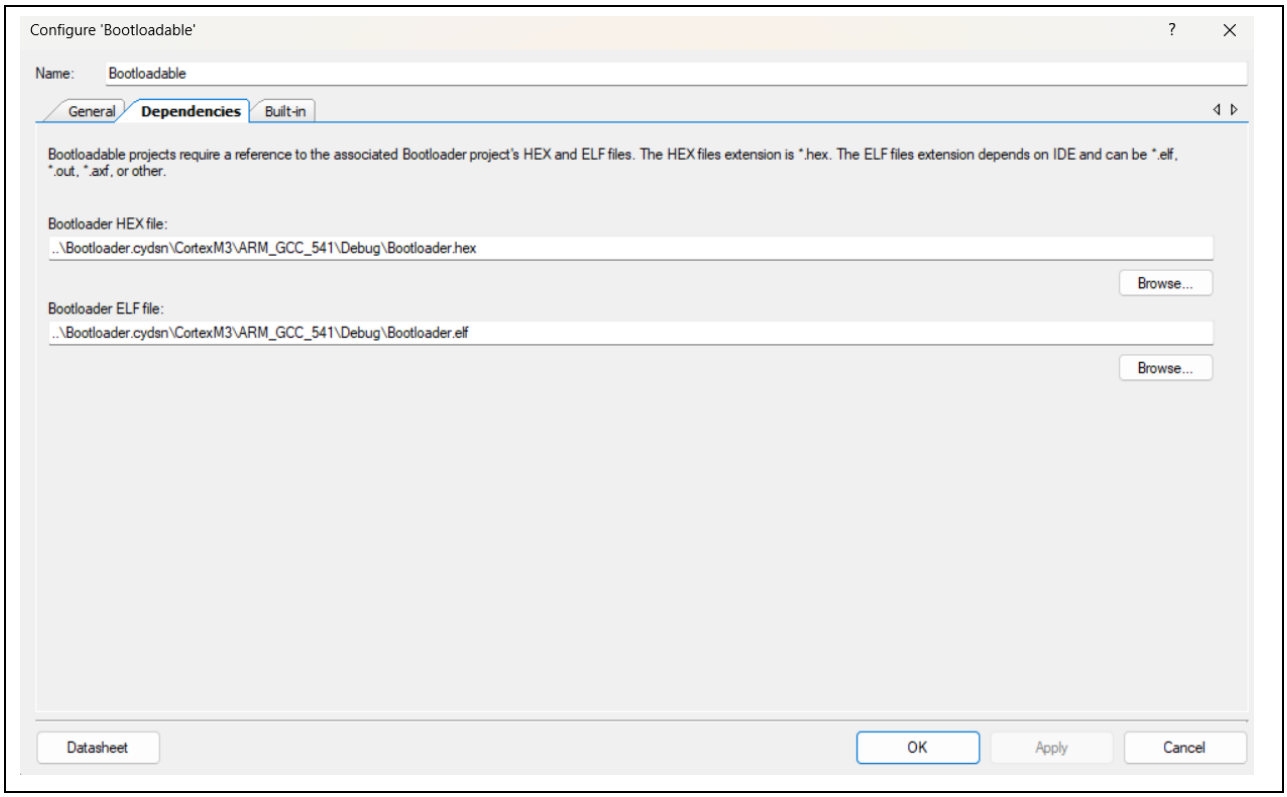
Generation of .hex and .elf file:-

We build the bootloader project to generate the hex and the elf files. The hex file consists of the binary data of the firmware which we have written in the bootloader. Similarly, the .elf file consists of the memory layout of the design.



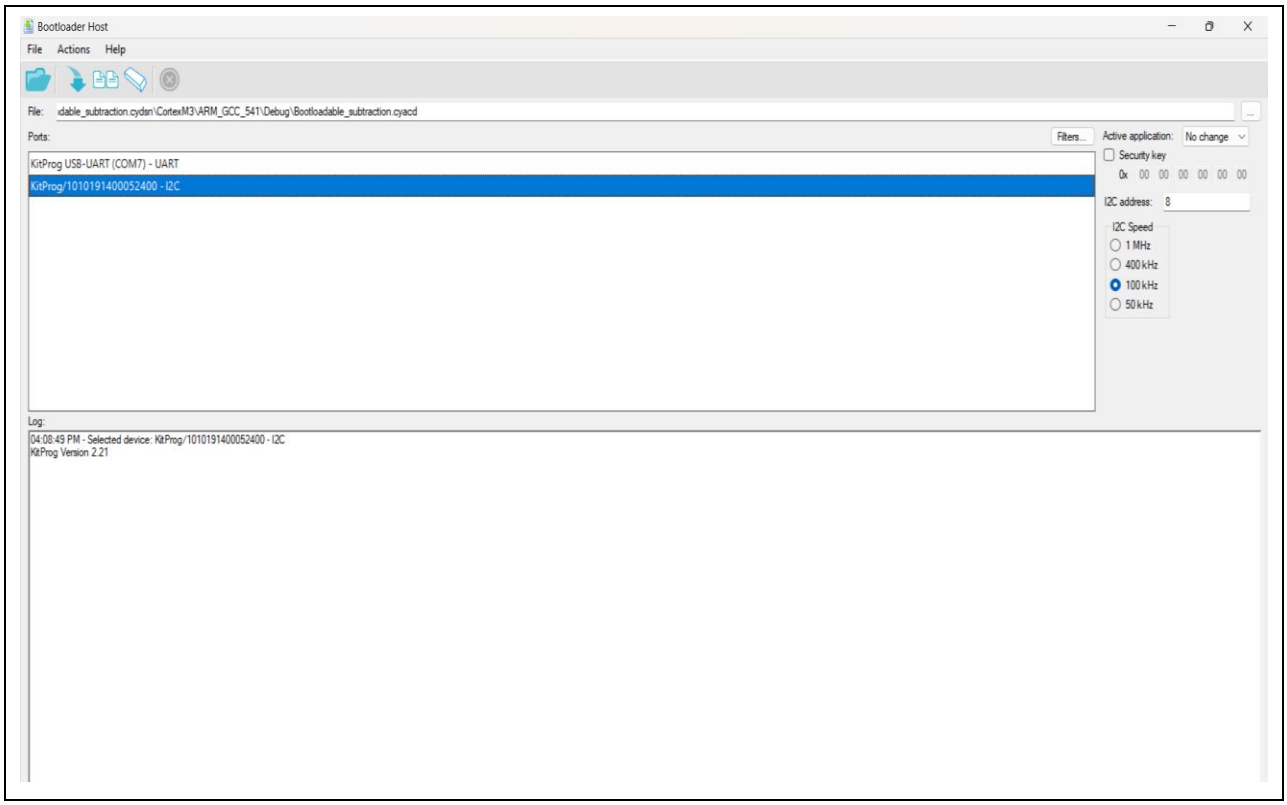
Loading the .hex and .elf file of the bootloader into the bootloadable project:-

Since we need to make sure that there is no overlap in the flash memory we are loading the .hex and the .elf file into the bootloadable project. After this we can make sure that there will be no overlap as we have defined fixed boundaries in the flash and when we build the bootloadable project the flash will be updated with the .hex and .elf file of the bootloadable in such a way that it does not overlap with the .hex and .elf data of the bootloader. After building the bootloadable file we will get the .cyacd file which will be programmed through the bootloader host whenever an interrupt is initiated through the switch press on the psoc.

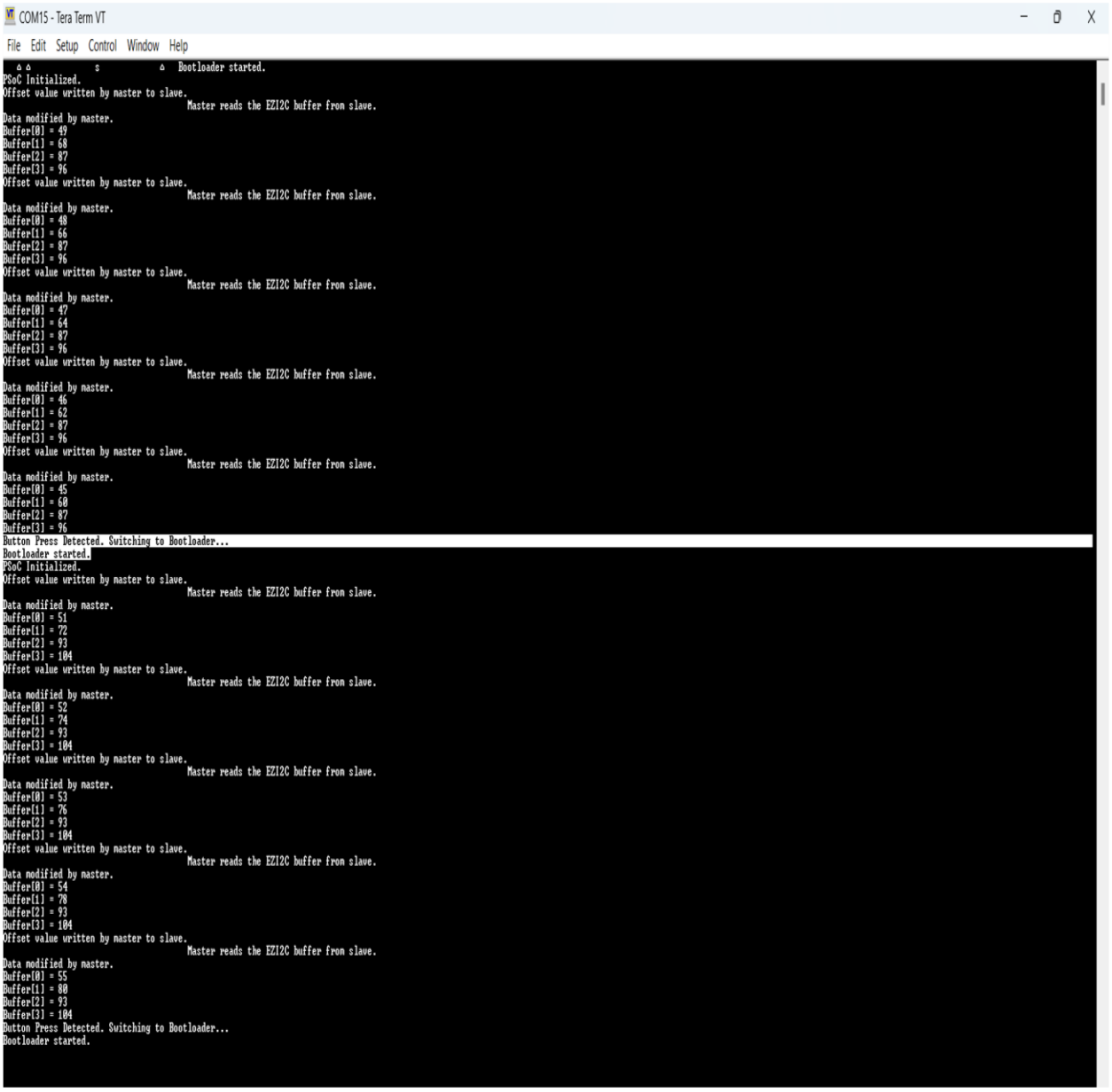


Loading the .cyacd file into the Bootloader Host:-

After building the bootloadable project the .cyacd files are generated in the directory. After switch press, the ISR is triggered and the bootloader is invoked. After this we need to put the generated .cyacd file based on the application which we need to perform (addition or subtraction) on the defined buffer in the ezi2cslave. After selecting the file we have to program the psoc. This means we are trying to update the flash memory of the psoc with the new .hex files and the task will be executed and the result will be displayed on tera term.


Bootloader Host
File Actions Help
File: dable_subtraction.cydsn\CortexM3\ARM_GCC_541\Debug\Bootloadable_subtraction.cyacd
Ports:
Filters...
Active application: No change
Security key
0x 00 00 00 00 00 00
I2C address: 8
I2C Speed
1 MHz
400 kHz
100 kHz
50 kHz
KitProg USB-UART (COM7) - UART
KitProg/1010191400052400 - I2C
Log:
04:08:49 PM - Selected device: KitProg/1010191400052400 - I2C
KitProg Version 2.21

Result (on Tera-Term) :



Here we have initialized the the EZIC slave buffer as [ezi2cBuffer[EZI2C_SIZE] = {50, 70, 90, 100}, now after programing the psoc with Bootloadable_Subtraction we can see the result on upper section of Tera-Term as shown in above image.

To enable the Bootloader we'll press the switch and we can observe "Button Press Detected. Switching to Bootloader " on Tera-Term.

Now to operate psoc without making any changes to original firmware with other firmware of addition, we will program the Bootloadable_Addition via Bootloader Host and it's reponse can be observed on the lower section of Tera-Term.