Task-1 Introduction to Embedded C, digital logic and AVR Studio 4.17

Experiment-1: Getting Familiar with AVR Studio and code debugging

The aim of this experiment is to get you familiar with (Integrated Development Environment) IDE and debugging syntax errors. We will use the AVR Studio IDE to open an existing project (provided in Experiment-1 folder). This project has some syntax errors, deliberately introduced in the code.

Your task is to locate and rectify these errors.

Note:

Check the version of AVR Studio installed on your machine. To check the version of AVR Studio installed on your machine, open AVR Studio and go to Menu bar   Help   About AVR Studio … A new window will open. Make sure Version is 4.17.

In case, you have some older version of AVR Studio installed on your machine, update to AVR studio 4.17 by installing AVR studio 4.17 included in e-Yantra DVD inside Software and Drivers folder.

If you have a newer version of AVR Studio installed on your machine, you can uninstall the newer version and install the AVR studio 4.17 included on the Resources page of the TBT portal.

Procedure:

Step-1: Open Experiment-1 folder. Open the given project in AVR Studio.

Step-2: Build the code. Check the build window.

You will find number of errors and warning which are deliberately introduced in the code.

Step-3: Debug the code and remove all the errors and warnings.

Step-4: Save the project.

Step-5: Submit this entire project folder on our portal.

Grading:

There are 5 unique errors deliberately introduced in this code. Your submitted code must have 0 error and 0 warning.

Removing each error will fetch you 2 marks. **Maximum marks for this experiment is 10**.

Note:

You may get more than 5 errors when you build the code for first time. Many of these errors are dependent i.e. errors appear because of another error. So removing one error may remove multiple dependent errors.

Experiment-2: Write your C-program

The motive of this experiment is to understand functions, variables and decision statements in C. You are expected to write a function which returns 1, if the given number is

multiple of 3 else return 0

In this experiment you have to use the code skeleton given in Experiment-2. You have to use the – function calls:

Unsigned int multiple_of3(unsigned int Number)

> This function will be called from main().
> Function will have only one   argument, viz. number.
> Number is a type of unsigned integer.
> Function arguments i.e. Number can be any whole number from 0 to 65535.
> Function should store 1 in variable "result", if the number is multiple of 3.
> Function should store 0 in variable "result", if the number is not a multiple of 3 .
> This variable "result" will be returned to the main function.

Note:

> You are required to make changes inside the function multiple_of3().

Procedure:

Step-1: Open Experiment-2 folder. Open the given project in AVR Studio.

Step-2: You will notice a function named "multiple_of3" and a comment inside it saying "WRITE YOUR CODE HERE".

Step-3: Complete this function by writing a code to find if the given number is multiple of 3 or not. In short, you have to write a code which stores 1 in variable "result", if the given number is multiple of 3 else stores 0 in variable "result".

Step-4: If required, debug the code to ensure zero errors and warnings.

Step-5: Save your project.

Step-5: Submit this entire project folder on our portal.

Grading:

Your function will be passed through test cases randomly generated. Marks will be given on basis of percentage of successful pass in test case.

100% Pass – 10 marks

<100% Pass – 5 marks