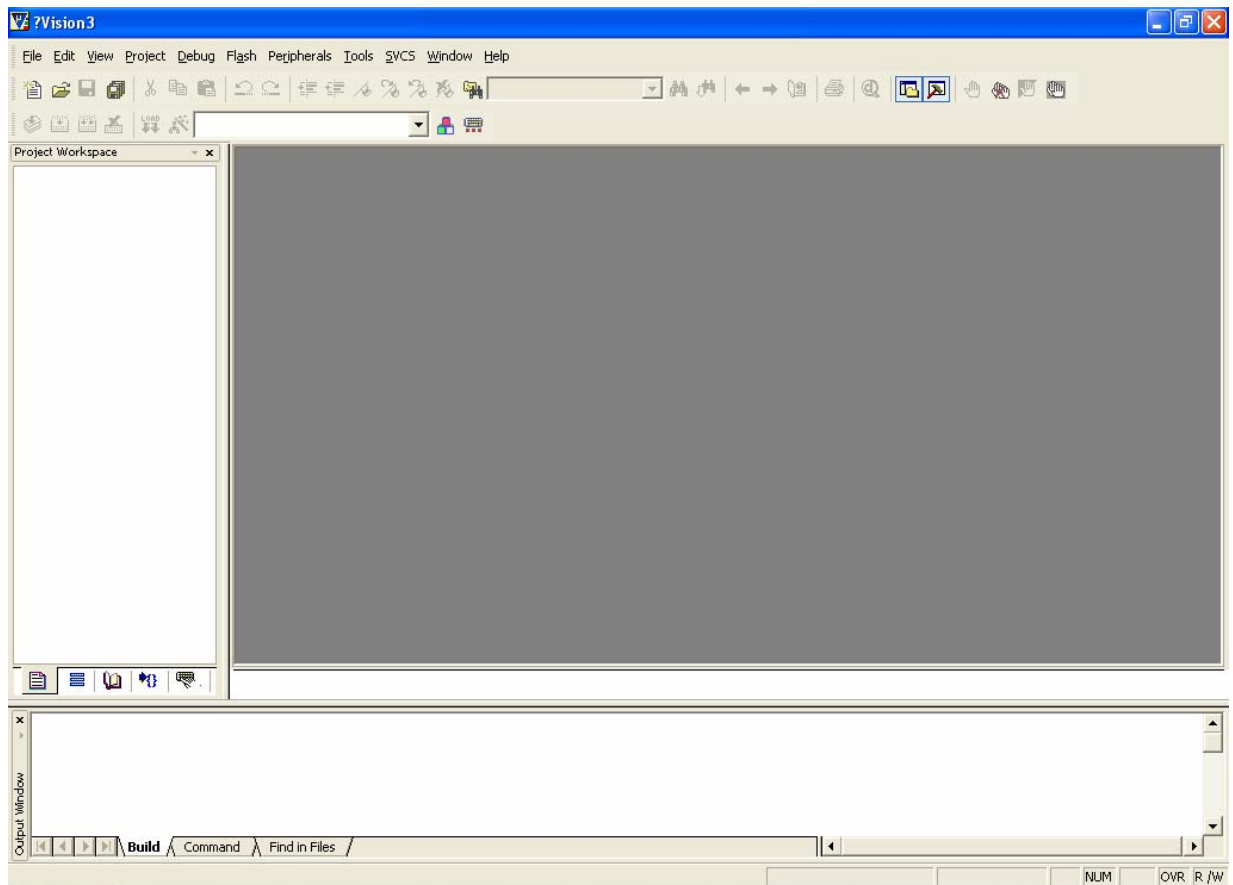


## An Example of using Keil uVision3 For creating Keil CARM's Project File

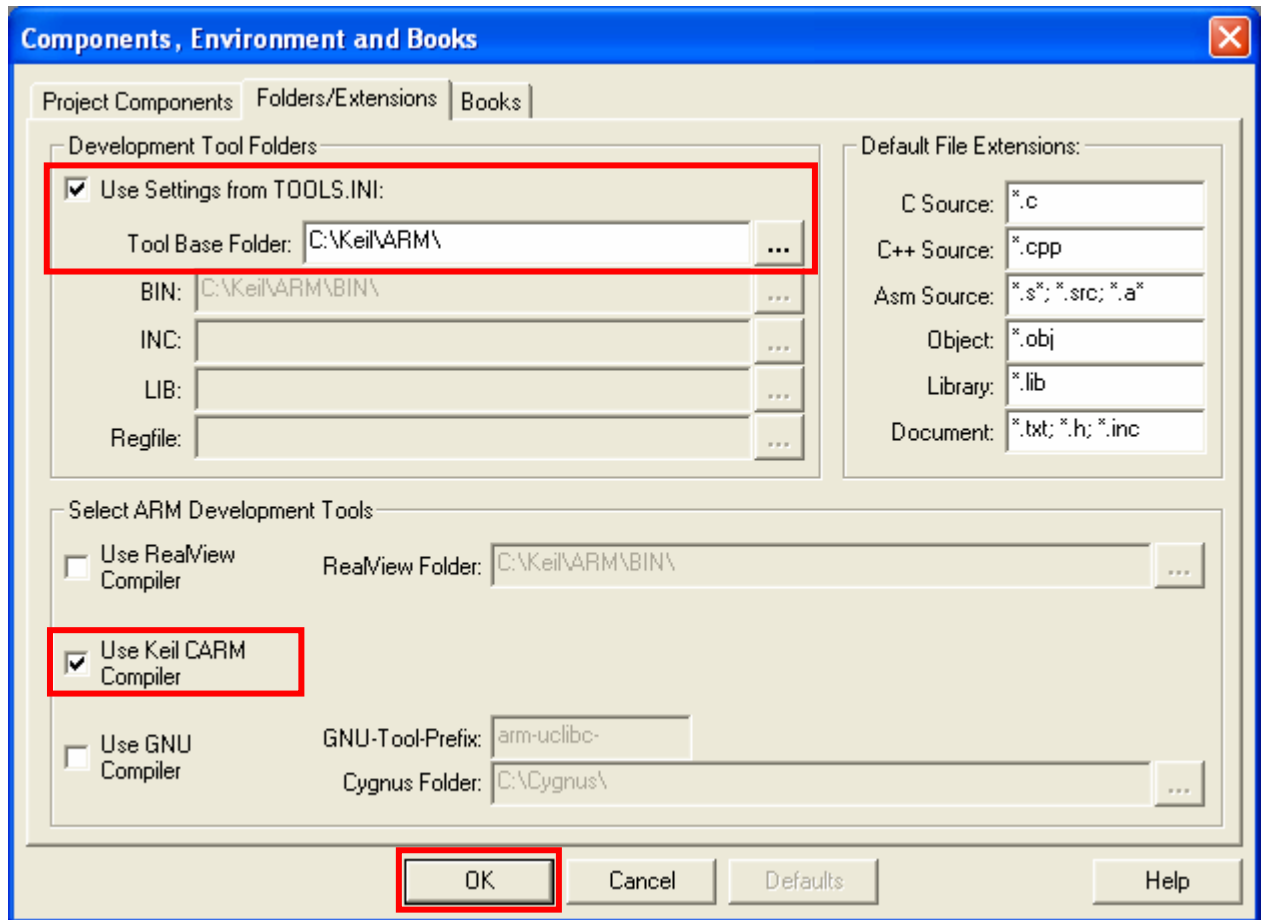
In this chapter, represent how to write C Language Program via Keil-CARM for translating orders under Text Editor Program of Keil (Keil uVision3). We would explain the method of setting Option for connecting orders to translate Program with Keil-CARM through Keil uVision3 only. Other detailed instruction and how to write Keil-CARM, users can learn from user manual of Keil-CARM. Method of setting default value of Keil uVision3 to use with Keil-CARM is;

1. Open Keil uVision3 Program which is Text Editor of Keil-CARM uses for writing C Language Source Code Program as in the picture

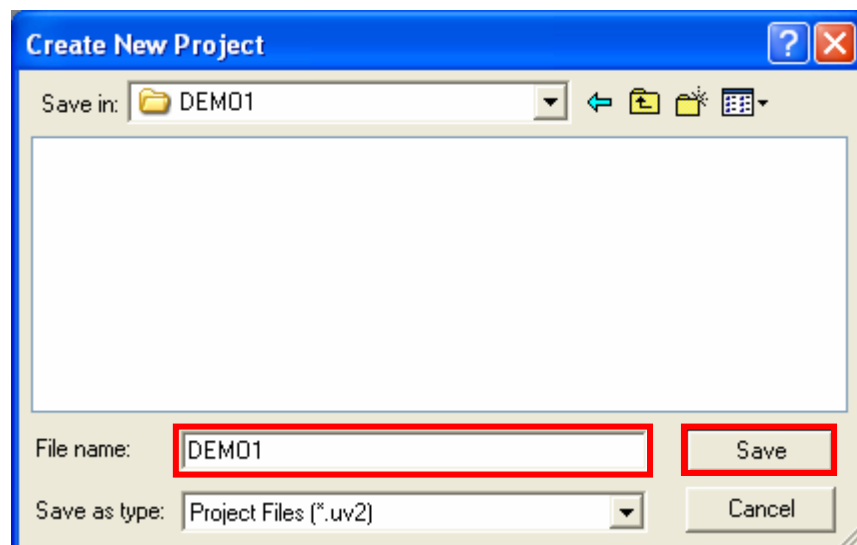


2. Set default value to translate uVision3 Code to use with Keil uVision3 Program and Keil-CARM. Click **Project → Components, Environment, and Books** then select default value to use Compiler titled **Select ARM Development Tools**. There are 3 default values of them such as **Use Keil ARM Tools**, **Use GNU Tools**, and **Use ARM Tools**. In this case, select "**Use Keil ARM Tools**" then assign position of folder to save default value of Keil-CARM Program. Generally, it is "C:\Keil\ARM\", but if you

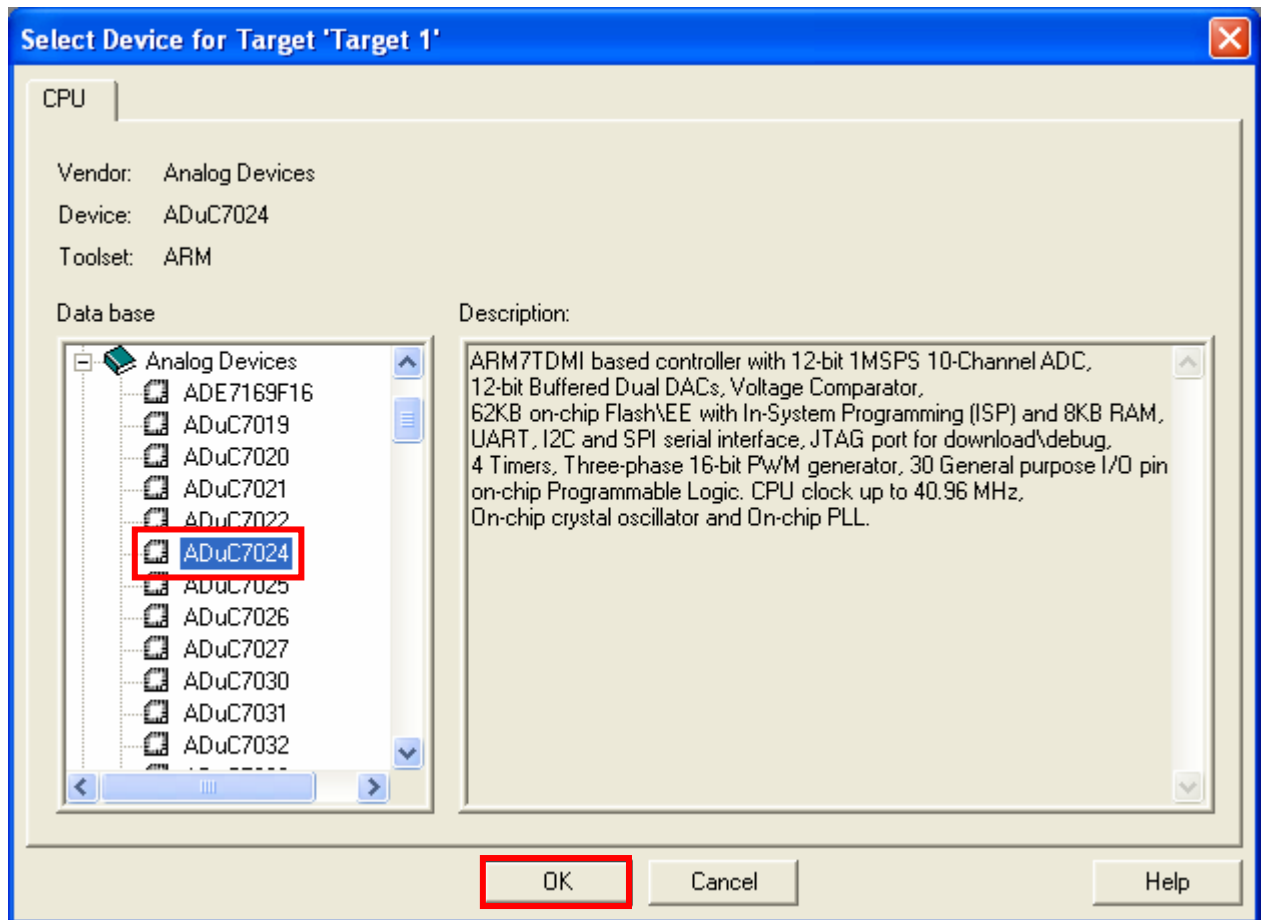
install Keil in other position, you need to change correctly as in the picture



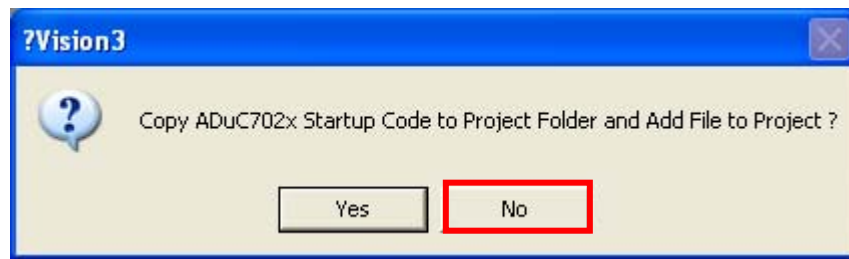
3. Create new Project File. Click **Project** → **New Project** then set the position of folder that you want to save with its title. For example, if you want to create Project File named DEMO1 and save in folder named DEMO1, you can set them by yourself. After you fill its name in File name blank, click **Save** project File as in the picture



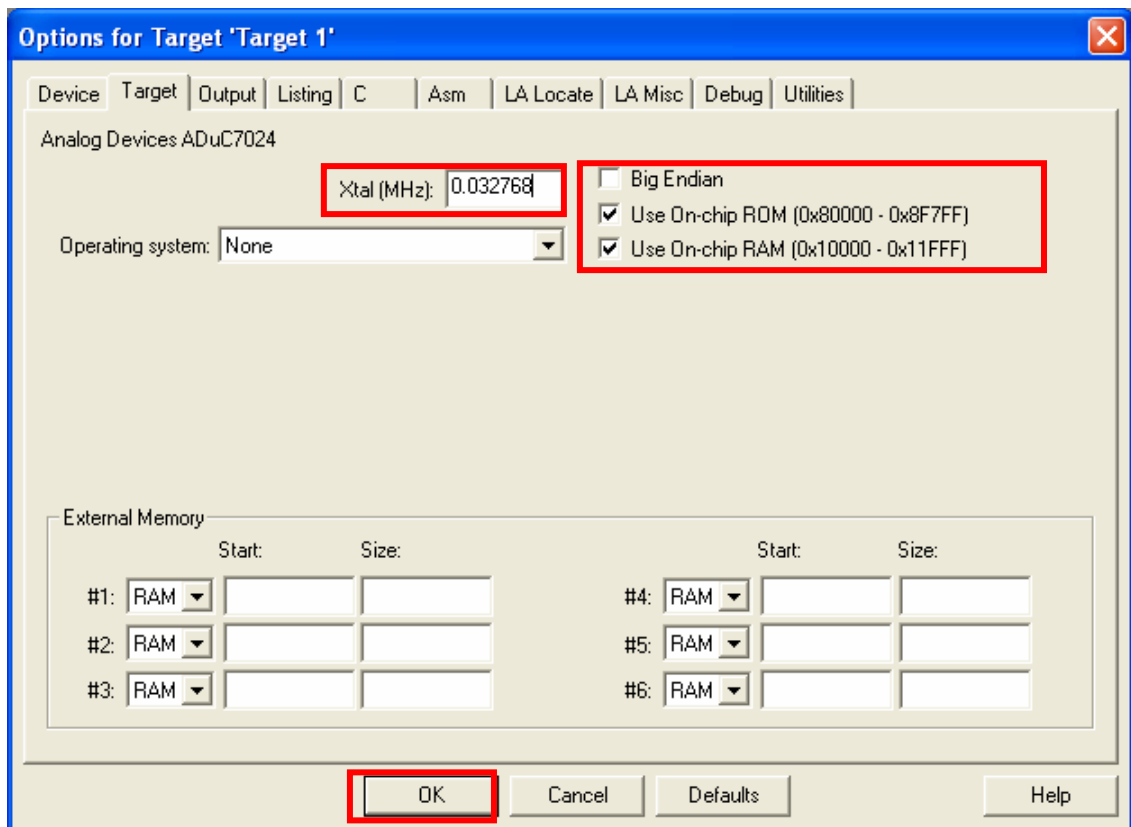
Program proceeding waits for users to set No. of MCU after you saved Project File. No. of Philips MCU in ET-ARM BASE7024 Board is ADuC7024 then click **OK** as in the picture



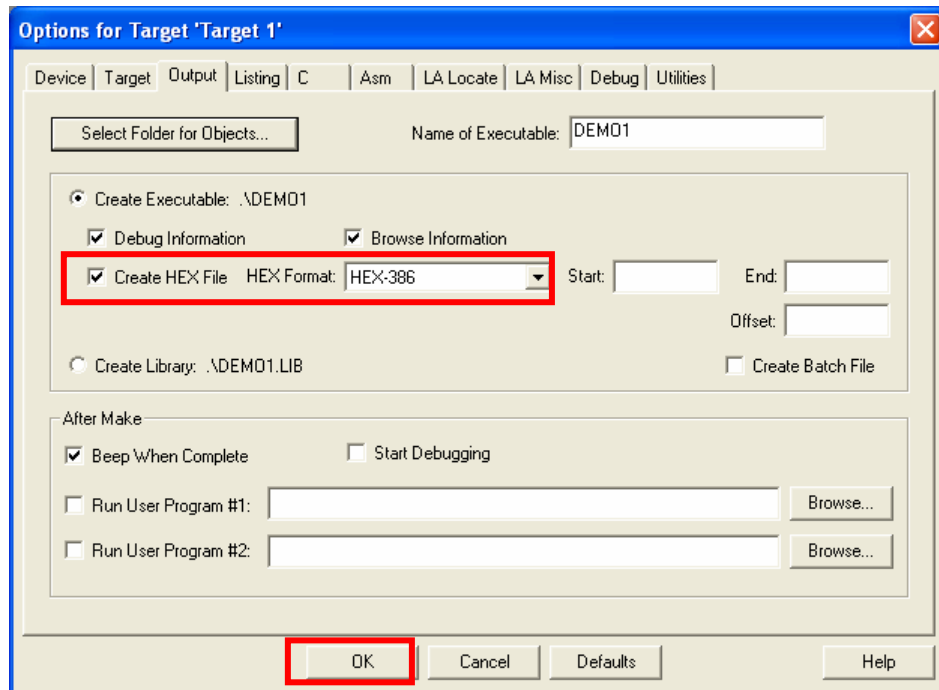
Next, users need to confirm to copy File Startup of Keil to use with Analog Devices' MCU or not because Startup File is the beginning value of MCU such as Stack values and default value to use with Phase Lock Loop (PLL). You should set values of them before starting to run your written program; otherwise MCU is added orders to run by the written program. Startup File of Keil-CARM is Assembly Language and is set to work with Keil Development. So, there are some different specifications to use. You can't use Startup File with ET-ARM BASE7024 Board immediately; you need to change some default value. Using Keil-CARM Program to translate orders, users need to correct Startup File to set value as well as ET-ARM BASE7024 Board form. In this case, select **"No"** to protect Keil uVision3 copy Startup File of Keil-CARM to use in this Project



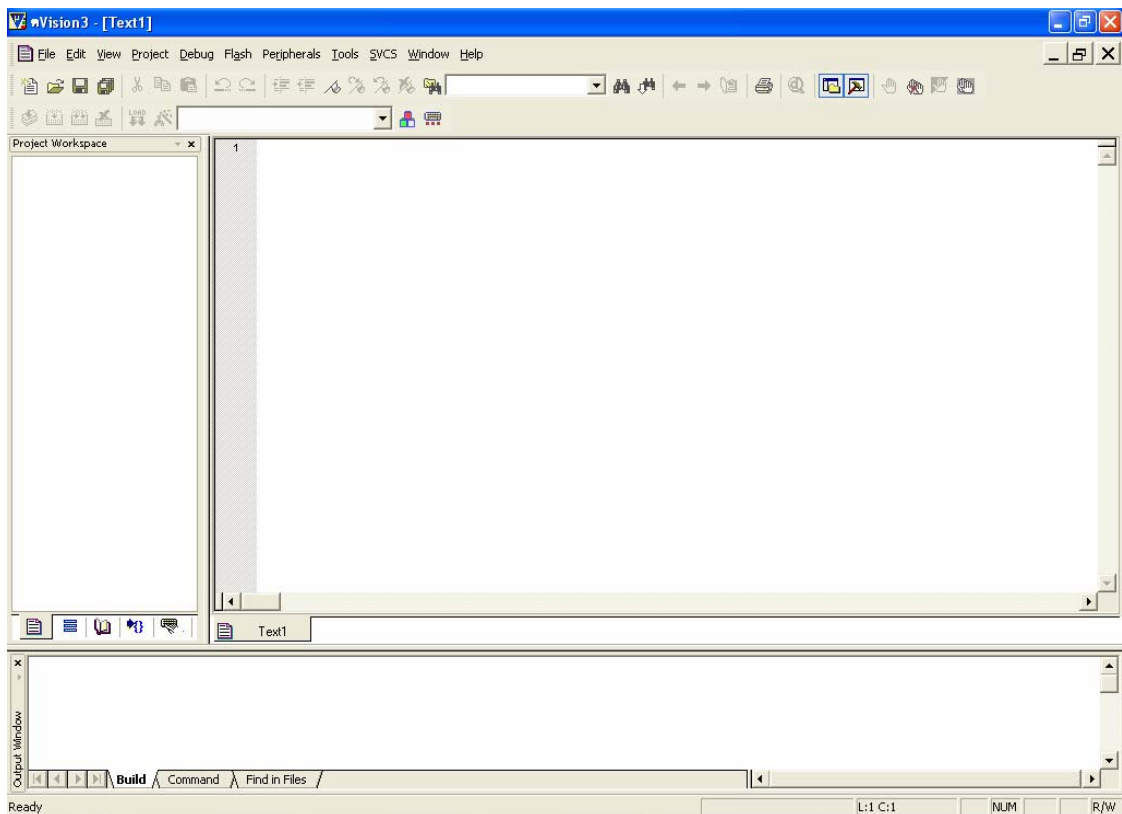
4. Copy File named "Startup.s" that is in ETT's CD-ROM in folder named "SOURCE\_KEIL" into the same position of New Project File File named "Startup.S" which is contained orders of ARM7 in Assembly Language uses to set the necessary default value of MCU such as setting Stack value, Initial Phase-Lock-Loop, MAM Function, and MCU's Vector. If you add "Startup.s File" from Keil or copy file from other source to use with "ET-ARM BASE7024 Board", its program in Startup is different, and it will effect on written program
5. Select Option of Project File. Click **Project → Option for target 'target 1'** then select Tab of Target for setting MCU Target value. The proceeding is;
  - 5.1 X-TAL is 32.768 KHz (0.032768 MHz) and set the MCU Memory is condition to translate Keil-CARM Program as in the picture



- 5.2 **Output**, click the default value of Create HEX File and set HEX Form as HEX-386 then click OK as in the picture



6. Start to write Source Code with C Language by click **File** → **New...** The Text File is displayed. In the first time, you should set its name as "Text1" as in the picture



In this step, you should write Source Code with C Language in the Keil-CARM's form as in the example

```

/*****
/* Examples Program For "ET-MASE ARM ADUc7024" Board */
/* Target MCU   : Analog Device ADUc7024          */
/*              : X-TAL : 32.768 KHz                */
/*              : Run Speed 41.78 MHz (With PLL)     */
/* Keil Editor   : uVision3 V3.03a                 */
/* Compiler      : Keil CARM V2.50a                 */
/* Function      : LED Blink on P4.7 Output Mode    */
*****/

#include <ADUc7024.H>                                // ADUc7024 MPU Register

/* pototype section */
void delay(unsigned long int);                       // Delay Time Function

int main(void)
{
    // Config Port P4 = GPIO                          // P4 = GPIO
    GP4CON = 0x00000000;

    // Config GPIO Direction (Bit[31..24])
    // Config GPIO Data Output (Bit[23..16])
    GP4DAT = 0x80000000;                             // P4[7] = Output

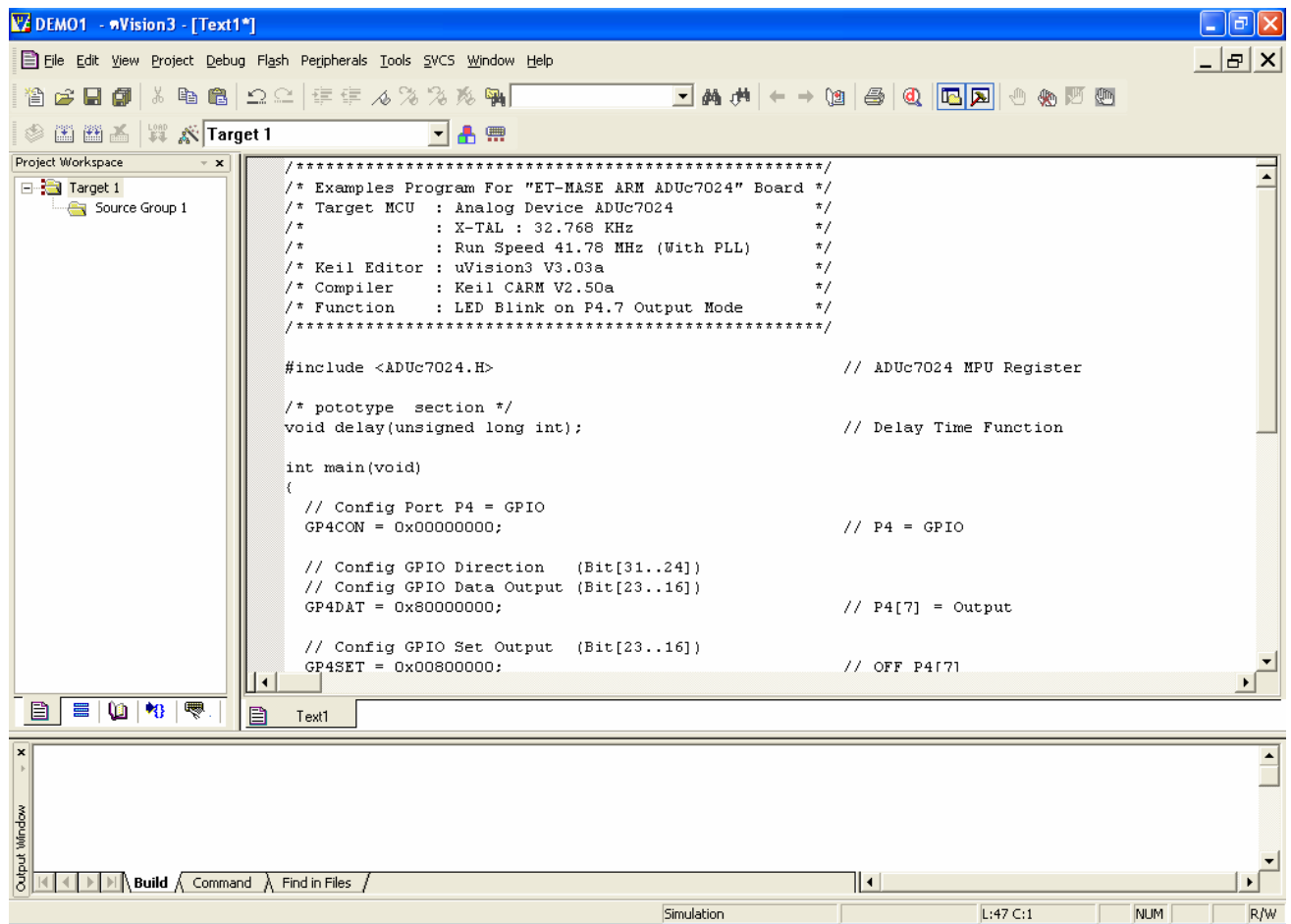
    // Config GPIO Set Output (Bit[23..16])
    GP4SET = 0x00800000;                             // Set Output P4[7]

    while (1)                                         // Loop Continue
    {
        // Set/Reset Output (Bit[23..16])
        GP4CLR = 0x00800000;                         // Reset Output P4.7
        delay(500000);                               // Delay LED Display
        GP4SET = 0x00800000;                         // Set Output P4.7
        delay(500000);                               // Delay LED Display
    }
}

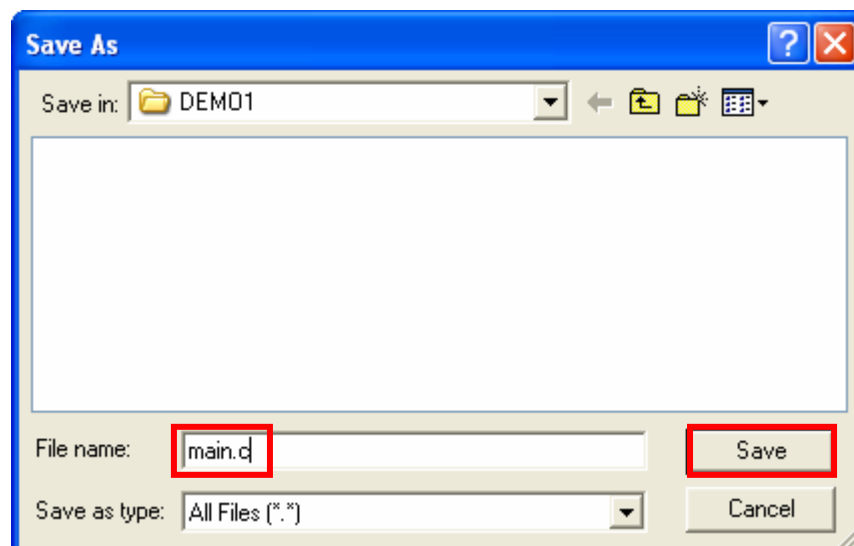
/*****
/* Delay Time Function */
/*      1-4294967296      */
*****/
void delay(unsigned long int count1)
{
    while(count1 > 0) {count1--;}                    // Loop Decrease Counter
}

```

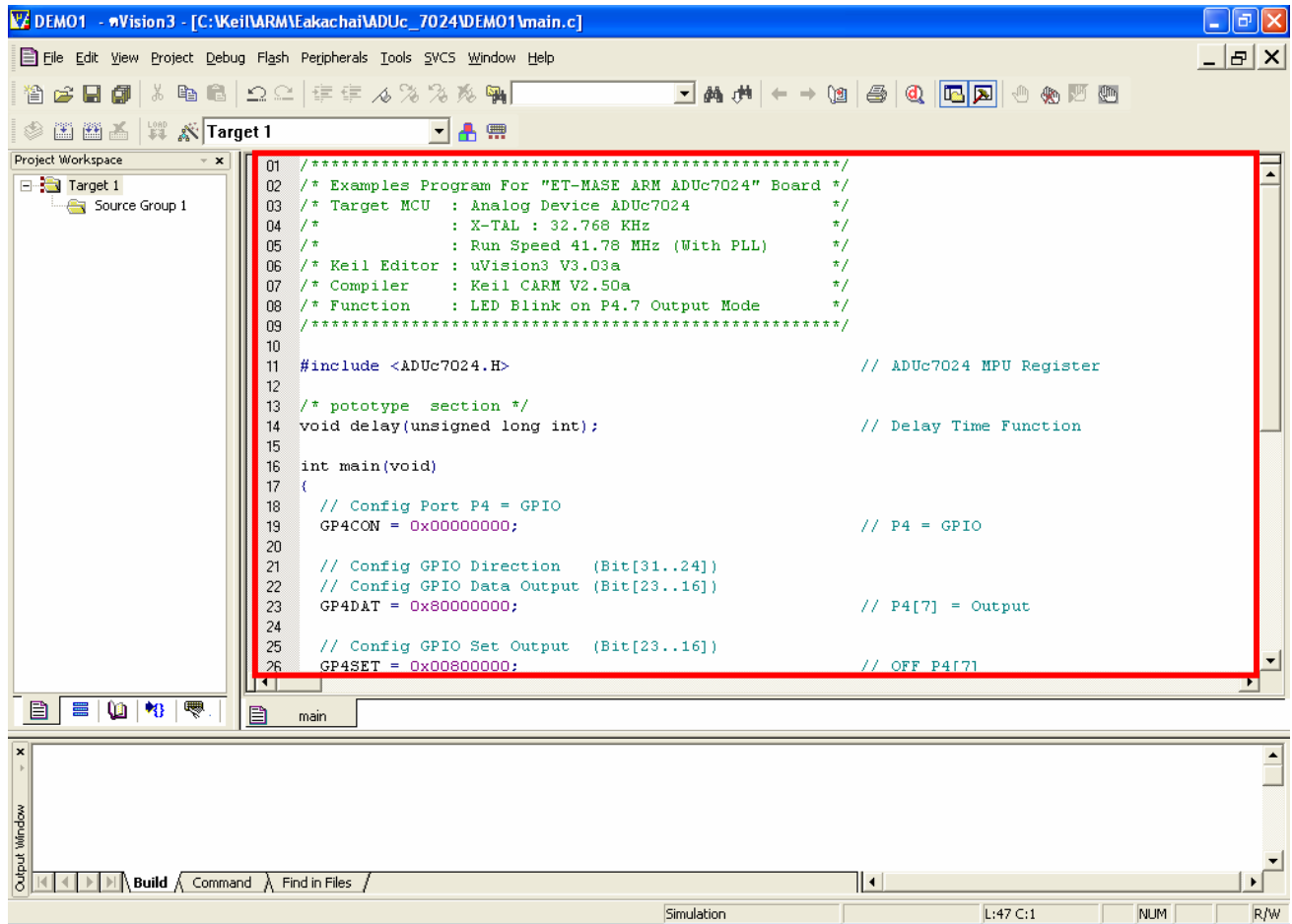
### An example of a blinker Program at P4.7



After finished to write program in C Language, you click save this file with its surname as ".C". For example, **File** → **Save As...** and then set its file name as ".main.c" as in the picture



After saved file, colours of letters in this program are changed with its function such as Comment, Variable, and Order. It is the good point of Keil uVision3 program and users use it easily as in the picture



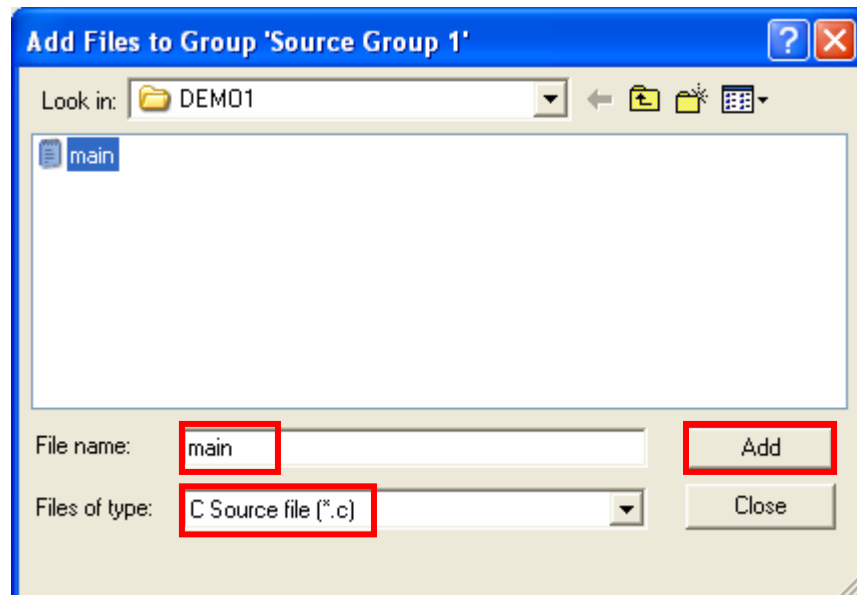
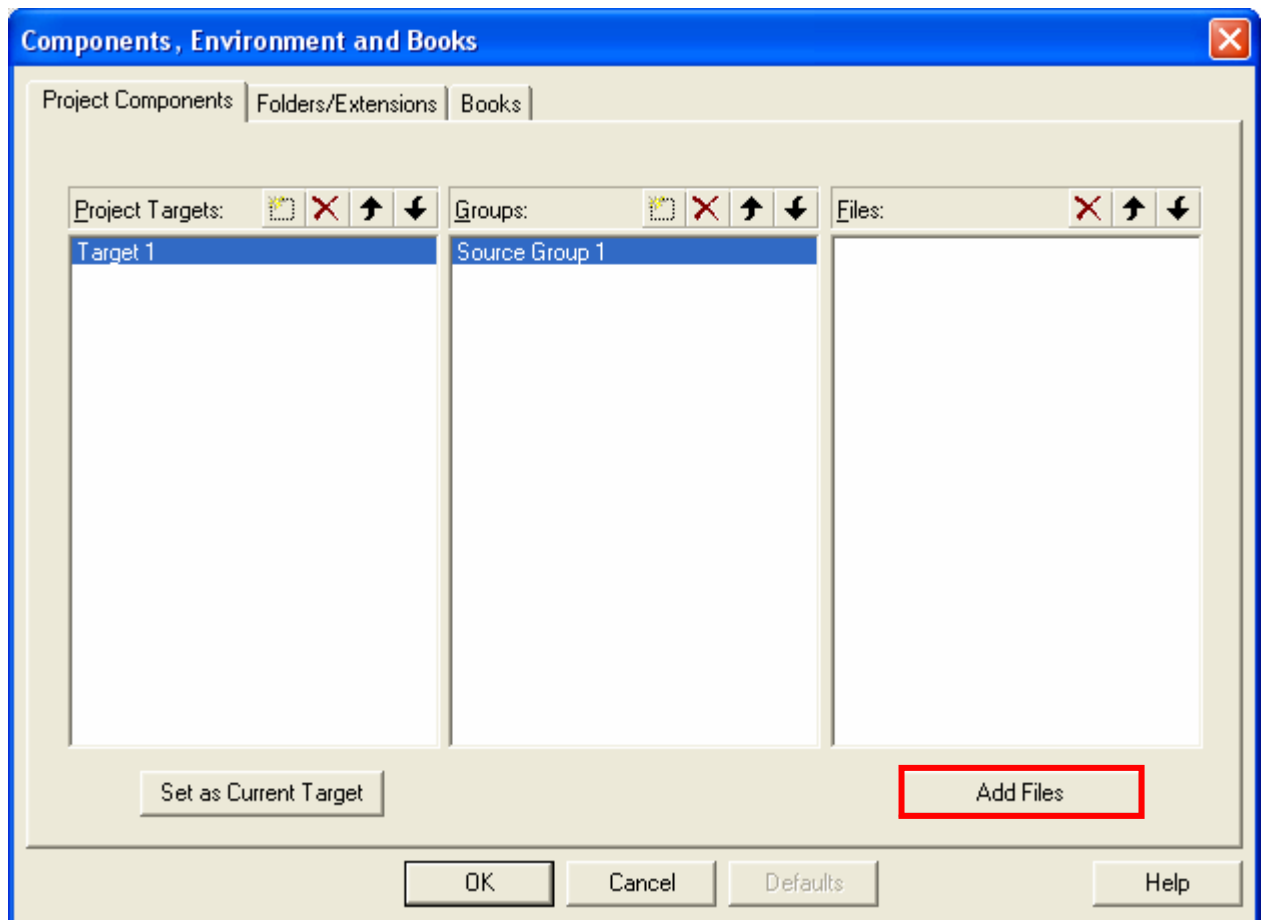
7. Add File with Project File by click **Project** → **Components, Environment, BOOKs...** then select Tab **Project Components** and then select **Add File**

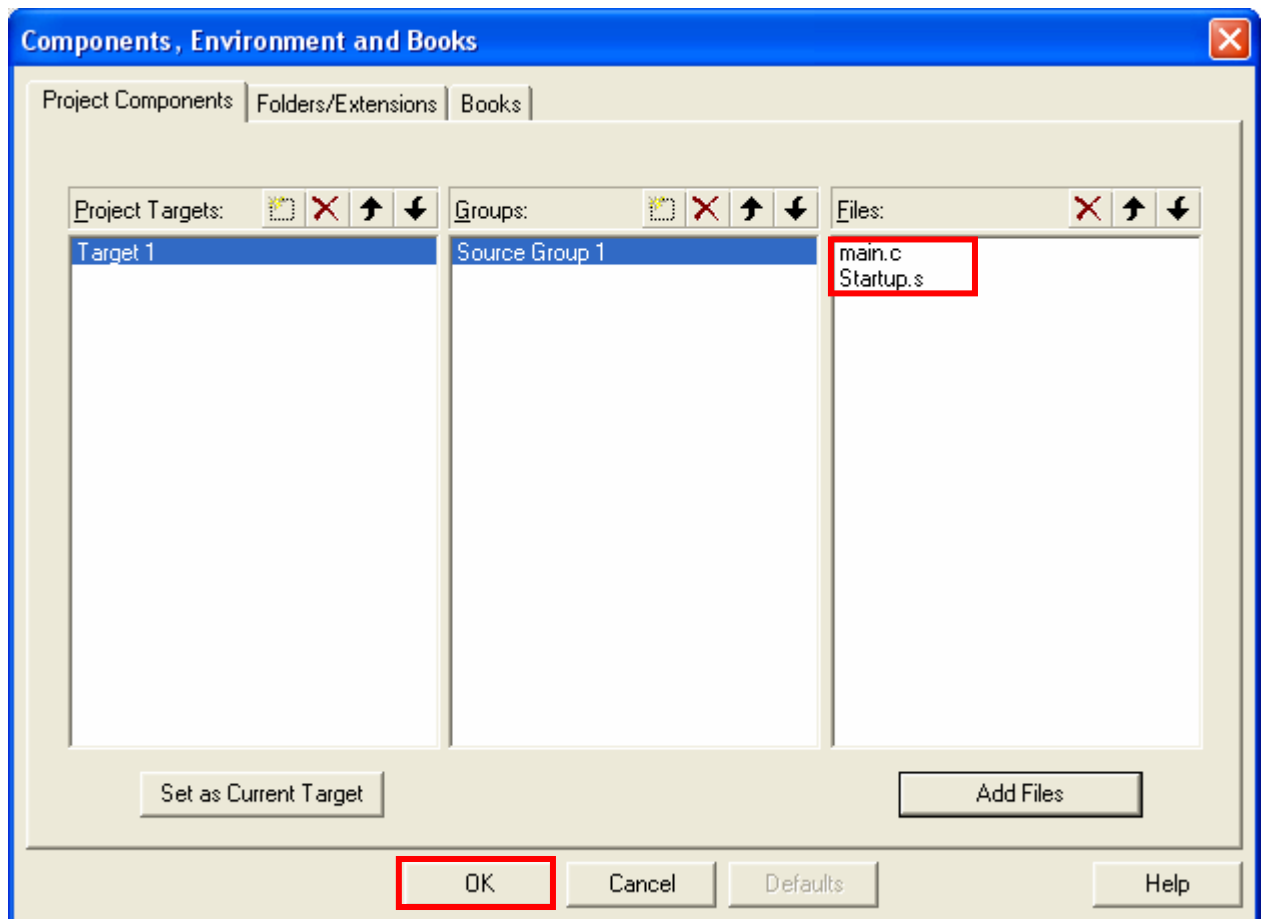
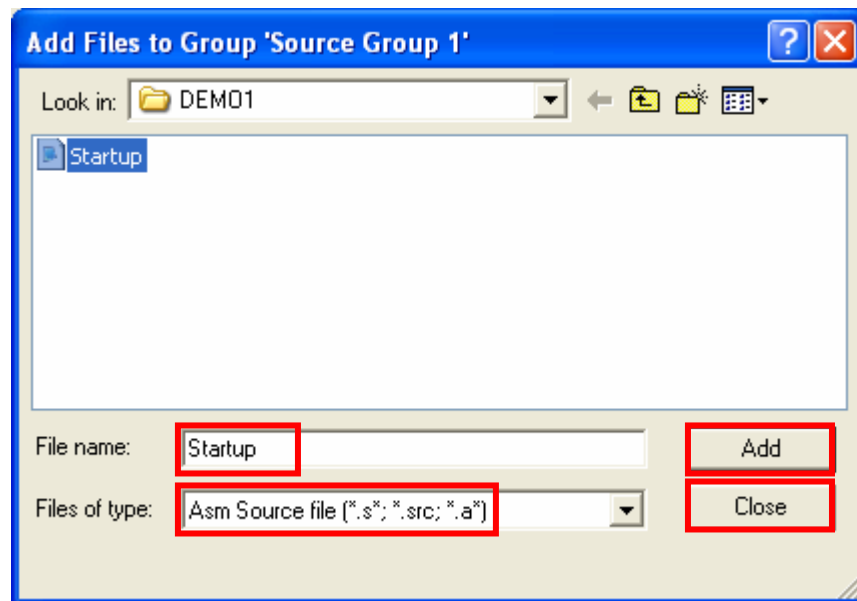
In the first time, you should select Files of type as "C Source files(\*.c)" because it shows other files name in C Source Code. Click icon file named "main.c" then select **Add** file named "Startup.s" to Project File

Set new File of type as "ASM Source files (\*.s;\*.src;\*.a)". Its file named Startup.s is shown in file name blank, click icon file "Startup.s", and then selects **Add**

After Add file named "main.c" and "Startup.s" to Project File completely, select **Close** as in the picture

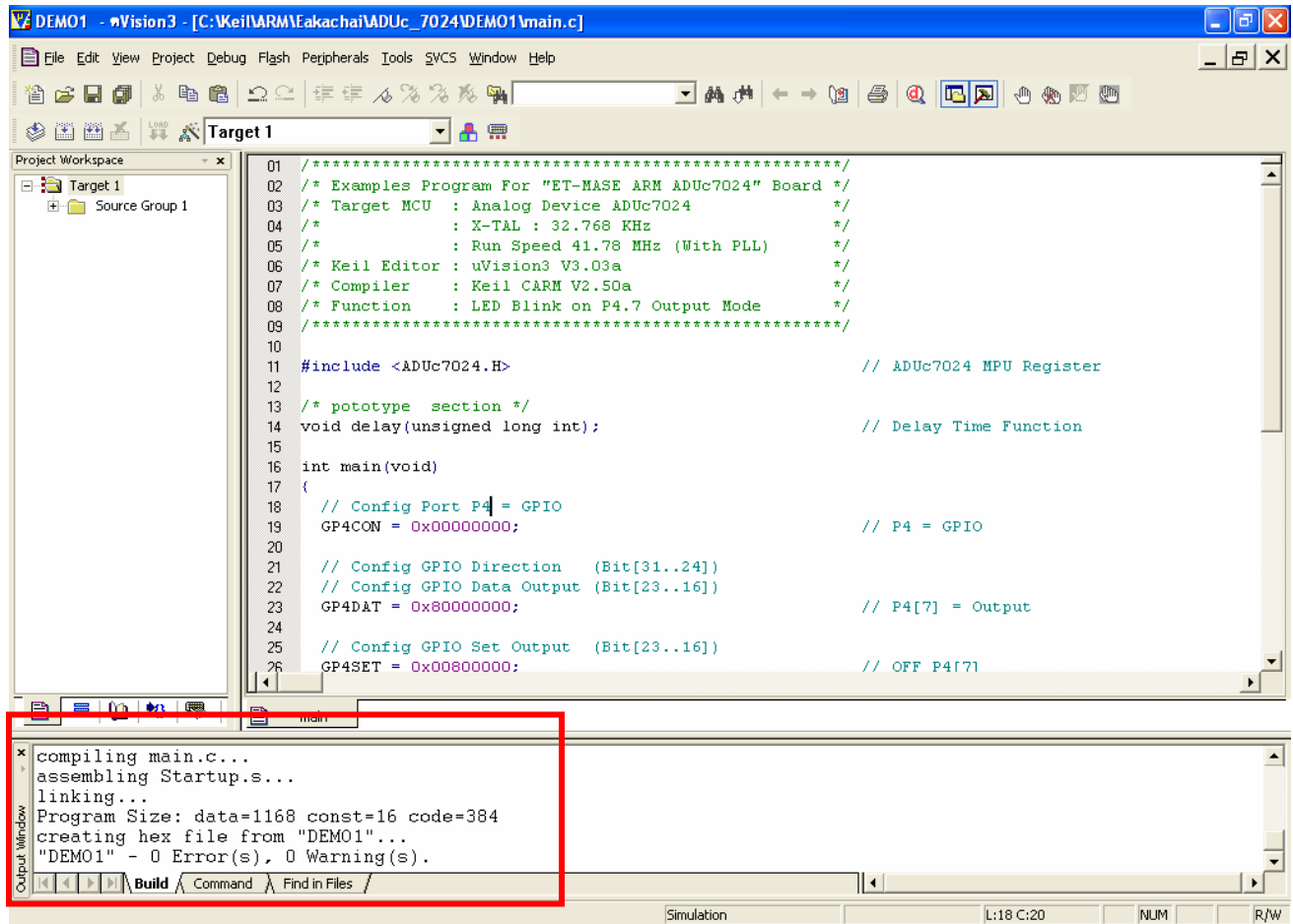






After both of file named "main.c" and "Startup.s" are added to Project File, in the Tab of File displays their names

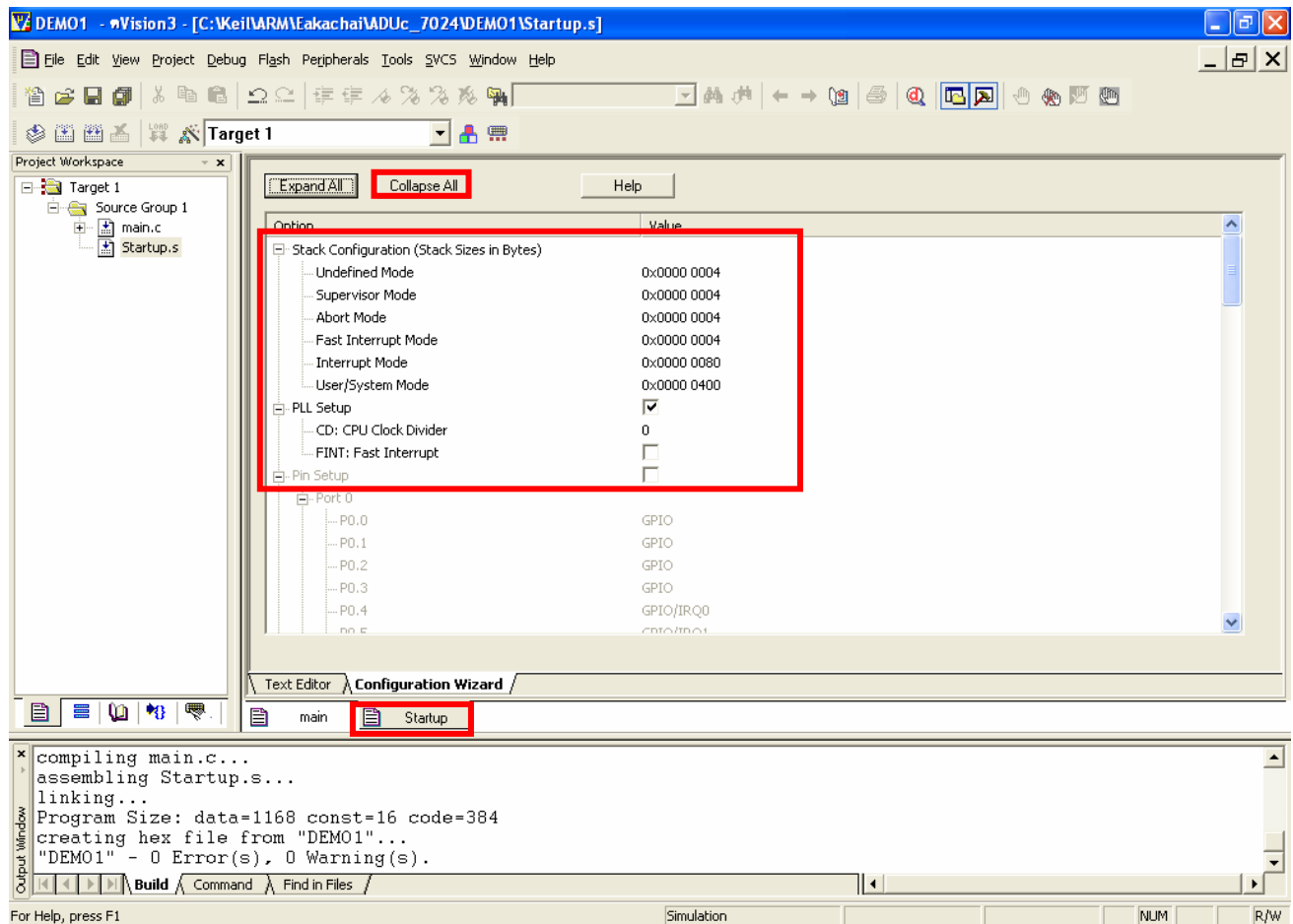
8. Translate program by click **Projects** → **Rebuild all target files**. Keil uVision3 Program can order Keil ARM Program to translate program immediately

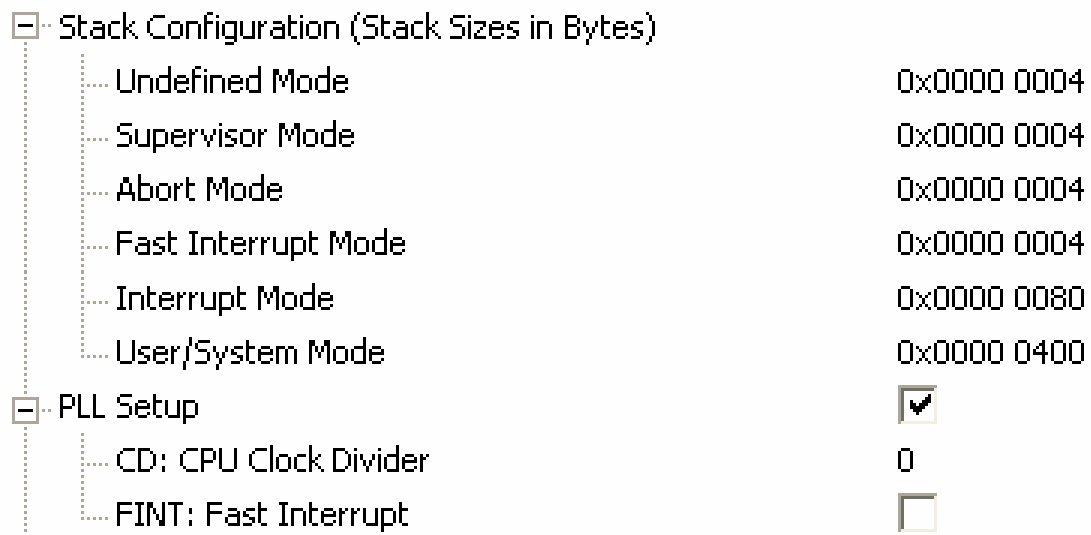


After translated program correctly without any mistake (0 Error and 0 Warning), you will get the HEX File name as same as the Project File name. Users can download HEX File to MCU immediately

## Verification of Startup File

Function of Startup File is contained beginning orders of MCU before start to run with written program by yourself. Function of Program in Startup File is Initial MCU in the necessary part and then jump to run in written Program in C Language. The method to verify Startup is clicking Tab of Startup File then selecting "Expand All". The correct default value of Startup File is like in the picture





Picture shows setting the default value of Startup File in ADUc7024