How to Use and Integrate AVR STUDIO With AVR GCC Complier Step by Step

(For ATMEL AVR 8-Bit Series MCUs)



Step1: After installing AVR Studio and AVR GCC. Click on **AVR Studio 4** shown below.



Step 2 : Click on New Project.

Welcome to AVR Studio 4							
	New Project Open						
	Recent projects	Modified					
Studio	 E: AVR\ AVR C Programs/AVR LCD\lcd1.aps E: AVR\ AVR C Programs/AVR TIMER\lcd1.aps E: AVR\ AVR C Programs/AVR SS2\AVR.aps E: AVR\ AVR SERIAL COMM ARRAY\AVR.aps E: AVR\ AVR LCD SERIAL 2\lcd1.aps E: AVR\ AVR LCD SERIAL COMM\lcd1.aps 	04-0ct-2010 22:04:51 05-0ct-2010 13:36:08 22:Sep-2010 21:26:46 28:Sep-2010 00:09:22 04-0ct-2010 22:04:51 05-0ct-2010 13:36:08					
Ver 4.18.684 🗹 Show di	alog at startup K< Back	<u>Cancel</u> Help					

Step 3: Select Complier AVR GCC, Set Path and Project Name, and then Click Next.

NOTE: Please note WINAVR must install before AVR GCC Complier.

Welcome to AVR Studi	io 4	Write Project Name
	Create new project	
	Project type:	Project name:
	Atmel AVR Assembler	LED
80	Choose : AVR GCC	Create initial file Create folder
		LED
\leq \Im	Location:	
	C:VAVR Test	
		Set Path
Ver 4.18.684		
	<< Back Next >>	Finish Cancel Help

Step 4: Select AVR Simulator and ATmega8515 MCU, then click Finish.

Welcome to AVR Stud	io 4			
Sinding 4	Select debug platform and Debug platform: AVR Dragon AVR ONE! AVR Simulator AVR Simulator 2 ICE 200 ICE 40 ICE 50 JTAG ICE JTAGICE mkII Proteus VSM Viewer	device	Device: ATmega64C1 ATmega64M1 ATmega8 ATmega8535 ATmega888 ATmega88P ATmega88PA ATmega8AA ATmega8AA ATmega8HVA ATmega8U2	
Ver 4.18.684	<< <u>B</u> ack	Open platform <u>N</u> ext >>	options next time debug	mode is entered

Step 5: After that you see **LED.c** already added.



Step 6: After that you see **LED.c** already added. Write or Copy paste code in LED.C

```
// All LEDs are Connected to PORTA
// Author: embeddedcraft.org
// Company : IMBUENT TECHNOLOGIES PVT. LTD.
#include<avr/io.h>
                               //HEADER FILE FOR AVR INPUT OUTPUT
#include<util/delay.h>
                               //HEADER FILE FOR DELAY
#define F_CPU 8000000UL
#define LED_PORT PORTA
                               // Crystal Frequency 8 MHz
                               // All LEDs to PORTC
int main(void)
{
                               //Configure PORTA as Output Port
DDRA = 0xFF;
while(1)
      {
            LED PORT= 0xFF ; //All LED ON
            _delay_ms(250);
                                                        //DELAY IN
            _delay_ms(250);
                                                        //DELAY IN
            LED_PORT= 0x00; //All LED OFF
            _delay_ms(250);
                                                        //DELAY IN
                        ATMEL AVR Training at:
                IMBUENT TECHNOLOGIES PVT.LTD.
                        Ludhiana (Punjab) India
                             www.imbuent.com
```

```
For Latest Embedded Technologies Log on to

www.emebddedcraft.org

__delay_ms(250); //DELAY IN

}

return(0);

}
```

Step 7:

: 🖹 File Project Build Edit View Iools Debug Window Help : D 📂 🛃 🗐 🙂 X ங 🕾 🗃 🦘 🔍 🗟 🦏 : 🚧 💉 % % 漆 譯 壽 : 🧱 🗙 🧃 ன 🚱 : 🕨 🗉 🗒 II	v Help • ≫ ≫ ≫ ≢ ≢ : ¥ 🔋 ar 🕹 : ► 🗆 🗉 u u
: 🗋 😂 🛃 🖉 🙂 🕺 🛍 🔠 🥙 🕅 🔁 🧌 : 🛤 🦽 🌾 % % 淳 宗 宗 部 🗑 : 🕨 💷 💷 🗉	• 🔏 🖄 淳 宰 🗄 🙀 🖠 🖬 🖬 🖓 🖏 🕷
: Trace Disabled 💦 🔨 🛠 🖓 💷 本 🕴 📾 🎟 🛲 🤫 🧏 🕬 🗄 🏙 🎒 🏈 🗙 🥺	Ye Ye 🗤 🛛 🔛 🚯 🗙 🛞
AVR GCC	<pre>icted to PORTA aft.org TECHNOLOGIES PVT. LTD </pre>

Step 8: Now Click on Build.

AVR Studio - [C:\AVR Test\LED.c]							
E File Project	Build	d Edit	View	Tools Debu	ug Window H	Help	
: 🗋 💕 🔛 💋 👔		Build		F7	1 M A 7	1 % % 律 🗐 🚟	
Trace Disabled AVR GCC Survey LED (default) LED.cc LED.cc	∰ ⊗ ×	Rebuild Build an Compile Clean Export I	All d Run Makefile	Ctrl+F7 Alt+F7 F12	e Connecte eddedcraf: MBUENT TE(o.h> delay.h>	ed to PORTA t.org CHNOLOGIES PVT. L' //HEADER FILI //HEADER FILI	
 Inclusion in the second second		ide	<pre>#define F_CPU #define LED_PO int main(void) {</pre>		8000000UL DRT PORTA)	// Crystal 8 1 // Al:	
			DDRA =	= 0xFF;		//Configure I	

Step 9: Now you see Build Succeeded with 0 Warning...

EL:\AAK IESC\LED.C < Build Program: 96 bytes (1.2% Full) (.text + .data + .bootloader) Data: 0 bytes (0.0% Full) (.data + .bss + .noinit) Build succeeded with O Warnings...

Step 10: Now Open C:\AVR Test\default you will see LED.hex is created. Download Hex file in Target Board.



For Testing on AVR Simulator

Step 11: Because we have also select AVR Simulator at Step4.

We can also test code without Hardware on Simulator.

Go to >> View >> Toolbars>> I/O.

File Project Build Ed	View Tools Debug Window	w Help
	Toolbars	Standard Toolbar
VR GCC LED (default) CLED (default) CLED.c Header Files CLED.c	Disassembler Watch Alt+1 Memory Alt+4 Memory 2 Memory 3 Register Alt+0 DDRA = 0xFF; while(1)	✓ Debug ✓ Debug Windows ✓ MDI Tabs ✓ AVRGCCPLUGIN ✓ STK500 ✓ TraceBar I/O Alt+5 Processor A as ✓ Build Output
	LED_PORT= 0x _delay_ms(25 _delay_ms(25 LED_PORT= 0x _delay_ms(25) _delay_ms(25) } return(0);	 Message Output Find Output Breakpoints and Tracepoints AVR GCC AVR GCC AVR DELAY



Step 12: Following Windows shown to you.

AVR Studio - [C:\AVR T E] File Project Build E	estVLED.c] dit View Tools	Deb	ua Window Help		
: D 📬 🗐 🕼 U X 🗈	A P B A		Start Debugging	Ctrl+Shift+Alt+F5	α
Trace Disabled	致然,日上:		Stop Debugging	Ctrl+Shift+F5	12
AVR GCC 👻 🗙	// All LED	101	Run	F5	
E 🛱 LED (default)	// Author:	11	Break	Ctrl+F5	
E Cource Files	#include <a< td=""><td>1</td><td>Reset</td><td>Shift+F5</td><td>FOR AVI</td></a<>	1	Reset	Shift+F5	FOR AVI
Header Files	#define F	1	Step Into	F11	7
🗄 🔄 External Depende	#define LE	["=	Step Over	F10	LEDs to
	<pre>int main(v {</pre>	1	Step Out	Shift+F11	
	DDRA = 0xF	*()	Run to Cursor	Ctrl+F10	RTA as





Thanks query@embeddedcraft.org