How to use ICC12 with Adapt812DX and FLASH Loader

This document will show and demonstrate the use of ImageCraft ICC12 Latest **Version 6** with Technological Arts' Adapt812DX module.

The MXFLash Loader is a utility to program the external FLASH of Adapt812DX ,DXLT or Adapt812 + MX1 combo module. The file can be downloaded at the link below.

http://www.interlog.com/~techart/myfiles/files/mxflash.zip

It will be used here to erase and program FLASH after the compilation of a test program.

This document assumes that the user is familiar with C and so will not teach how to program C here.

ImageCraft Links:

About		×
	ICC12 Version 6.16A Built Jan 26 2004 21:58:29 (650) 493-9326 FAX: (650) 493-9329	
Copyright (C) 199	18-2002 ImageCraft Creations Inc.	
Web Site:	http://www.uragonsgate.net/mainnan/listinio	
Demo/Updates: Email Support:	http://www.imagecraft.com/software/demos.html	
License Reques	t: mailto:license@imagecraft.com	
Savlo tiny URL:		
	OK	

http://www.imagecraft.com/software/ http://www.ece.utexas.edu/%7Evalvano http://www.dragonsgate.net/FAQ/cache/20.html http://www.imagecraft.com/software/mdevtools.html http://www.dragonsgate.net/mailman/listinfo

Technological Arts Links:

http://www.technologicalarts.com/myfiles/ad812dx.html http://www.technologicalarts.com/myfiles/812dxlt.html

Getting Started:

Double click on the ICC12 icon. If a user has not read the ICC12 manual and just open the IDE one will wonder what to do next. Well wonder no more.

Note the 3 window panes. The top left most is greyed out and the right is the project window. The left bottom pane is where the error messages are displayed during compilation.

Before creating a new Project, the hardware target in the Compiler Options must be setup properly for the target MCU. This is to ensure that the compiler will setup the type of MCU the C program will compile for. In this example it is the Adapt812DX, Adapt812DXLT or the Adapt812 + MX1 card.



Compiler Setup:

Click on Project Menu – Options – Target Tab.

🕌 ImageCraft IDE for ICC	12 (PROFESSIONAL)			
File Edit Search View F	Project RCS Tools Terminal	Help	 	
	New Open	Ctrl+F11	Project Browser	
	Close All Files			
	Reopen	•		
	Make Project Rebuild All	F9 Shift+F9		
	Add File(s) Add Topmost Opened File Remove Selected File(s)	Shift+F11		
	Options Manual Sort Browser Window			
	Close Save As			
	[No Open File]		[No Open Project]	

Please note the Device Configuration. Click on the pull down arrow to change the device type.

Compiler Options	×			
Paths Compiler Target				
Device Configuration 9S12Dx256 / Ax256 / H256 9S12Dx256 / Ax256 / H256 9S12DP256 4K EEPROM Mode 9S12DP256 12K RAM Mode 9S12Dx512 / Ax512 912B32 912B32 Internal EEPROM 912D60/A 912D60 Internal EEPROM © Enable Addr OxC0000.0xF7FFF © Make Paged Functions Default S2 Record Type © Linear © Map Vector Page © CPU / Banked Address	PRINTF Version Small (int only, no modifier) Iong (+ long, and modifiers) float (full function) Additional Lib. Word Alignment Advanced Other Options No Startup/Lib Non-default Startup			
For Expanded Memory, "Linear S2" and "Map Vector Page" should generally be used. You may need to use the SRecCvt program. Click Help for details. STD and Demo version can access up to 64K of expanded memory. PRD version has no				
OK Cancel Set As Default	Load Default <u>H</u> elp			

Scroll up or down to select Custom as shown. Note that an Adapt812DX device Configuration does already exist. Unfortunately, the addresses are not setup properly, therefore the Custom configuration must be selected and the memory parameters are changed for proper operations.

In this example an *Adapt812DX with 128Kbyte FLASH* is used. Process are the same for different FLASH sizes.

Compiler Options	×			
Paths Compiler Target				
Device Configuration 9512DP256 12K BAM Mode Small (int only, no modifier)				
Custom Image: Custom indication of the second system of the second				
For Expanded Memory, "Linear S2" and "Map Vector Page" should generally be used. You may need to use the SRecCvt program. Click Help for details. STD and Demo version can access up to 64K of expanded memory. PRO version has no				
OK Cancel Set As Default Load Default <u>H</u> elp				

Device Configuration:

Program Memory: **0xC000** Data Memory: **0x7000** Stack Pointer: **0x0C00**

The external memory area from 0xC000 to \$FFFF is where the start of code and the ISR must reside. Any interrupt routine must be in this area. It is possible to call routine in another PAGE while servicing the ISR.

The external memory area 0x7000 to 0x7FFF is RAM. The DPAGE is enabled and is used to access the RAM memory in paged window.

The RAM memory area 0x0800 to 0x0C00 is internal to the MCU. The stack pointer will use this area.

Expanded Memory:

Note the address range is **0x10000. 0X1FFFF** for a 128Kbyte FLASH. For 512Kbyte Flash the address range is **0x10000. 0X7FFFF.**

S2 Record Type:

Select Linear and check marked the Map Vector Page. ICC12 will output Linear S2 record.

Compiler Options 🛛 🗶					
Paths Compiler Target					
Device Configuration Custom Memory Addresses Program Memory Data Memory 0x7000 Stack Pointer 0x0C00 Expanded Memory ✓ Enable Addr 0x10000.0x1FFFF ✓ Make Paged Functions Default S2 Record Type ✓ Linear ✓ Map Vector Page CPU / Banked Address	PRINTF Version small (int only, no modifier) long (+ long, and modifiers) float (full function) Additional Lib. Word Alignment Advanced Other Options No Startup/Lib Non-default Startup				
For Expanded Memory, "Linear S2" and "Map Vector Page" should generally be used. You may need to use the SRecCvt program. Click Help for details. STD and Demo version can access up to 64K of expanded memory. PRO version has no OK Cancel Set As Default Help					

On the compiler tab there are several choices of S-record output as shown. Select which one that suits you.

Compiler Options
Paths Compiler Target
Strict ANSI C Checkings
Accept Extensions (C++ comments, binary constants)
🔲 int size enum (for backward compatibility)
Macro Define(s): Undefine(s):
Output Format Motorola S19
Motorola S19
S19 with Source Level Debugging S19 with ASM/Source Level Debugging
Intel HEX
NOTE: Debug information for structure
members is only generated by the
PRU version
Execute Command After Successful Build:
OK Cancel Set As Default Load Default <u>H</u> elp

Starting a new Project:

Once the compiler options are setup, a new project can be created. Click Project menu – New.

ImageCraft IDE for ICC12 (PROFESSIONAL)				
File Edit Search View	Project RCS Tools Termina	l Help		
	Open Open All Files	Ctrl+F11	Project	Browser
	Close All Files			NO PROJECT OPEN
	Reopen	+		I
	Make Project Rebuild All	F9 Shift+F9		
	Add File(s) Add Topmost Opened File Remove Selected File(s)	Shift+F11		
	Options Manual Sort Browser Window	,		
	Close Save As			
	[NoOpen File]		[No Open Project]	S19

The ICC12 will prompt to save the new project. The user should decide whether to create a new directory to save the new project. In this example a new directory called *Test* is created and the file is saved as file *test.prj*.

Save New Pro	ject As					? ×
Save in: 🕯	Local Disk (C:)		•	ڭ 🖻	•	
🚞 Temp	(🚞 usr				
🚞 Temp1	(🚞 WINDOWS				
🚞 Temp2	(🚞 WUTemp				
Contraction temp3	(📄 Test				
C Temp4						
C Temp5						
•						►
File name:					Oper	n
Save as type:	Project Files (*.prj))		-	Canc	el

Type the filename as *test.prj* and click on the Save button.

Save New Pro	ject As	? ×
Save in: 🔀	Test 🔽 🗢 🛍 🚟 🎫	
I		_
File name:	test.pri Save	e
Save as type:	Project Files (*.prj)	el

Note that the project window has changed to add Files, Headers and Documents.



Creating a new file to the project:

To add files to the project, click on the File menu – new as shown.

🛃 ImageCraft IDE for ICC	12 (PROFESSIONAL)		<u>-0×</u>
File Edit Search View P	Project RCS Tools Terminal Help		
Reopen			Project Downey
Reopen Reopen Open Reload Save Save As Close Compile File Save All Print Exit		2]	Project Browser
	[No Open File]	C:\Test\test.prj	S19

Note that ICC12 created an untitled file. Save the file as BlinkLED.C.

🐸 ImageCraft IDE for ICC12 (PROF	ESSIONAL)	
File Edit Search View Project R	25 Tools Terminal Help	
1 🗗 🖬 🚳 😹 💼 🗉	3 🕾 🏩 🎟 🖬 📾 📼	
		Project Browser
1:1	C:\Test\test.prj	S19 //.

To save, click on File menu - Save As

File Edit Search View F	12 (PROFESSIONAL)	
New		
Reopen >		Project Browser
Reopen Open Reload Save Ctrl+S Save As Close Compile File Save All Close All Print Exit		Project Browser
1:1	C:\Test\test mi	 \$19

ICC12 will open an explorer window to help save the file. Type BlinkLED.c then press the save button.



ImageCraft IDE for ICC12 (PROFESSIONAL) File Edit Search View Project RCS Tools Terminal Help	
🗓 💪 🗆 🎻 🗶 🖻 🥵 🕰 🏩 📰 🔤	
BlinkLED.c	Project Browser
	TEST Headers Documents
1: 1 C:\Test\BlinkLED.c C:\Test\vest.prj	S19

Note that ICC12 has renamed the file to BlinkLED.c.

To add BlinkLED.c to the Project, click on the Project menu - Add File(s)

🛃 ImageCraft IDE for IC	C12 (PROFESSIONAL)			
File Edit Search View	Project RCS Tools Termina New Open	Ctrl+F11	ERT	Brained In 1
BlinkLED.c	Open All Files Close All Files			Project Browser
Re	Reopen	•		Files Headers
	Make Project Rebuild All	F9 Shift+F9		Documents
	Add File(s) Add Topmost Opened File Remove Selected File(s)	Shift+F11		
	Options Manual Sort Browser Window			
	Close Save As			
1: 1	C:\Test\BlinkLED.	,	C:\Test\test.prj	519

ICC12 will open an explorer window to help and locate the file of interest.

Add Files		<u>?</u> ×
Look in: [Test 🔽 🗲 🛍 🗰 🕇	
BlinkLED.c		
J. File name:		
riie name.	BIINKLED.C	
Files of type:	Source Files (*.c, *.s, *.h)	el
	Dpen as read-only	

Note that the right window pane has changed to include BlinkLED.c under the Files Project.

ImageCraft IDE for ICC12 (PROFESS	SIONAL)		<u>_ </u>
File Edit Search View Project RCS	Tools Terminal Help		
🕒 🖻 🖬 🗳 🗟 🕵	🛛 💐 🔔 💷 🖬 📠	ERR	
BlinkLED.c			Project Browser
2: 1 Modified C:V	\Test\BlinkLED.c	C:\Test\test.prj	<u>,</u> \$19 <i>//</i> ,

Locate **vectors.c** and copy file to Test directory. The major reason why this must be done is because of project to project dependency. It is not good to keep editing a single **vectors.c** if other projects are using this same file. It becomes a problem to keep track of the changes made to the different projects.

To add vectors.c to the Project, click on the Project menu – Add File(s)

Add Files		? ×
Look in: 🗀	Test 💌 🗢 🛍 📸 💷 -	
BlinkLED.c		
vectors.c		
File name:	vectors c	
Files of type:	Source Files (*.c, *.s, *.h)	
	🔲 Open as read-only	

Note that ICC12 has changed to include *vectors.c.* Note that the vectors.c was written for the 68HC912B32 and 812A4 MCUs.



Write the codes below into BlinkLED.c file. Once it is written we can then compile/make/build the code.

```
File Edit Search View Project RCS Tools Terminal Help
                                                                                                                       _ 🗆 🗵
🛅 📂 🖬 🕩 | ៩ 🛍 🕵 | 🖓 | 🚣 💷 📰 📼
                                                                                              Project Browser 

TEST

Files

Vectors.c

Headers

Documents
 vectors.c BlinkLED.c
     #include "HC12.h"
                                                                                          -
     void blink_delay(void);
void main()
          int i;
          PEAR = 0x04 ; // Enable read/write signal
MODE = 0x80 ; // Normal exp. narrow
WINDEF = 0xC0 ; // Enable data ¢ program windows
MXAR = 0xO7 ; // Enable A16, A17, A18
CSCTL0 = 0x30 ; // Enable CSD ¢ CSP0
CSCTL1 = 0x10 ; // CSD covers lower half of memory map
          DDRT = 0×FF;
PORTT = 0×FF;
           blink_delay();
           while(1)
           PORTT = 0xFF;
blink_delay();
PORTT = 0x00;
blink_delay();
                                       //LED on
                                        //LED off
                         C:\Test\BlinkLED.c
          Modified
                                                                C:\Test\Test.pri
22: 18
                                                                                                        S19
 #include "HC12.h"
 void blink_delay(void);
 void main()
 {
            int i;
            COPCTL = 0x00 ; // Disable COP
            PEAR = 0x04 ; // Enable read/write signal
            MODE = 0xB0 ; // Normal exp. narrow
            WINDEF = 0xC0 ; // Enable data & program windows
            MXAR = 0x07 ;// Enable A16, A17, A18
            CSCTL0 = 0x30 ; // Enable CSD & CSP0
            CSCTL1 = 0x10 ; // CSD covers lower half of memory map
            DDRT = 0xFF;
            PORTT = 0xFF;
            blink_delay();
            while(1)
    {
            PORTT = 0xFF;
                                           //LED on
            blink_delay();
            PORTT = 0x00;
                                            //LED off
            blink_delay();
           }
 }
 void blink_delay(void)
  int i:
           for(i=0;i<64000;i++)
            {
                                       ;
           }
 Compiling/Build/Make the file:
```

To make the file click Project menu – make project as shown.

🛃 ImageCraft IDE fo	r ICC12 (PROFESSIONAL)		
File Edit Search Vie	ew Project RCS Tools Terminal Help		
	New Open Ctrl+F11	ERR	
vectors.c BlinkLED.c	Open All Files		Project Browser
#include	"He Close All Files		
void blin	k_(Reopen)		Files
void main	() Make Project F9		Headers
(Rebuild All Shift+F9	-	Documents
Int	Add File(s) Shift+F11		
DDRP	Add Topmost Opened File		
PTP	= (-	
	Options		
blin	k_c Manual Sort Browser Window		
	Close		
While /	e (. Save As		
		-	
PTP	= OxFF; //LED on		
blin	k_delay();		
PTP	= 0x00; //LED off		
blin	k_delay();		
,			
· · ·			
void blin	k_delay(void)		
(
int i;	-0.4.40000.4.4.4		
for (1)	=0;1<1000;1++)		
ì ì			
	· · · · · · · · · · · · · · · · · · ·		
JI.			<u></u>
1:1	C:\Test\BlinkLED.c	C:\Test\test.prj	S19

Note the bottom window pane will show messages to display how the build progressed. Any errors, if any, are shown in this window. The build was without error so we can progress to erasing and programming the Adapt812DX.

ImageCraft IDE for ICC12 (STANDARD) [WARNING: 45 Days EVALUATION version]	
File Edit Search View Project RCS Tools Terminal Help	
🗓 🖻 🖬 🗳 🖟 🖻 🥱 💐 🔹 🖮 🔜 📼	
vectors.c BlinkLED.c	Project Browser
#include "HC12.h"	E-III TEST
	BlinkLED.c
void main()	vectors.c
	- Headers
int i;	Documents
PFND = 0x04 · // Enchle read/write signal	
MODE = 0x80; // Normal exp. narrow	
WINDEF = 0xC0 ; // Enable data & program windows	
MXAR = 0x07 ; // Enable A16, A17, A18	
CSCTLO = 0x30 ; // Enable CSD & CSP0	
CSCTL1 = 0x10 ; // CSD covers lower half of memory map	
DDRT = 0xFF;	
PORTT = OxFF;	
blink_delay();	
while (1)	
PORTT = 0xFF; //LED on	
blink_delay();	
PORTT = 0x00; //LED off	
blink_delay();	
3	
icc12w -c -IC:\icc\include\ -e -1 -Wf-cpdon C:\Test\vectors.c	
icc12w -o Test -LC:\icc\lib\ -btext:0xC000 -bdata:0x7000 -bextcode:0x00000.0x1	
pone.	
25: 43 C:\Test\BlinkLED.c C:\Test\Test.prj	S19

Note the other extraneous files are created after a make.

C:\Test					
File Edit View	Favorites Tools Help				-
G Back 👻 🕥	- 🏂 🔎 Search 🔀 Folders	5	B D 1	3 🗙 🖽 -	
Address 🙆 C:\Test				-	→ Go
Folders X	Name 🔺	Size	Туре	Date Modified	Date Cre
	BlinkLEDc	1 KB	_C File	10/21/2004 3:	10/21/20
	BlinkLED.c	1 KB	C source file	10/21/2004 3:	10/21/20
	🛅 BlinkLED, dp2	1 KB	DP2 File	10/21/2004 3:	10/21/20
	🗐 BlinkLED.lis	З КВ	Text Document	10/21/2004 3:	10/21/2(
	🗟 BlinkLED.o	1 KB	O File	10/21/2004 3:	10/21/20
	BlinkLED.s	2 KB	ASM File	10/21/2004 3:	10/21/20
	🖻 test.lk	1 KB	LK File	10/21/2004 3:	10/21/20
	💼 test.lst	4 KB	list file	10/21/2004 3:	10/21/20
H H	👼 test.mak	1 KB	MAK File	10/21/2004 3:	10/21/20
μ Ω Τ	🕑 test.mp	2 KB	MP File	10/21/2004 3:	10/21/20
φ 🗛 τ	🕥 test.prj	1 KB	EmbeddedGNU Proj	10/21/2004 3:	10/21/2(
ф. <mark>С</mark> т_	🖬 test.s19	1 KB	S19 File	10/21/2004 3:	10/21/2(
	TEST.SRC	1 KB	SRC File	10/21/2004 2:	10/21/20
	💼 vectors.c	2 KB	C source file	7/17/2002 2:3	10/21/20
	🔂 vectors.dp2	0 KB	DP2 File	10/21/2004 3:	10/21/20
	🗊 vectors.lis	2 KB	Text Document	10/21/2004 3:	10/21/20
	vectors.o	1 KB	O File	10/21/2004 3:	10/21/20
_	• vectors.s	1 KB	ASM File	10/21/2004 3:	10/21/20
	• test.prjdefICC12.prj	1 KB	EmbeddedGNU Proj	10/21/2004 3:	10/21/20
	• [•

Using WordPad to check the content of *test.s19* file. Note that the S-records are of different lengths. In reality the test.s19 record is an S2 type.

S20F01C000CF0C0016C07387CE70008EB8 S21101C00B700027056A000820F6CEC078CD2B S21201C01870008EC0782706180A307020F516C4 S20801C026C02A20FE08 S21101C02A34B7751B9EC6047B000AC6B07BAA S21201C037000BC6C07B0037C6077B0038C6303C S21201C0457B003CC6107B003DC6FF7B00AFC6ED S21101C053FF7B00AE4A8000042010C6FF7B74 S21001C06000AE4A8000047900AE4A800061 S20B01C06C0420EEB757303D3A S21201000034B7751B9ECC00006C1E2007EC1E4C S21201000EC300016C1EEC1E8CFA0025F2B757DB S20601001C300AA2 S20A01FFFAFFFFFFFC0003F S20901C0731D0016073D4B S903C0003C

Programming the Adapt812DX:

Any terminal program can be use to communicate to the FLASH Loader. In this example ICC12 is used. Connect Serial cable to any available PC COM port and the other end to Adapt812DX.

🕌 ImageCraft II	E for ICC12 (STANDAR	RD) [WARNING: 45 Days EVALUATI	[ON version]	
File Edit Search	View Project RCS	Tools Terminal Help		
🕒 🖻 🖬 🛛	🍰 🚽 💼 🕵	Show Terminal Window		
vectors.c BlinkL	ED.c	Clear Window		Project Browser
		Capture		E-IC TEST
DI	RT = OxFF;			E-Files
P	RTT = OxFF;			BlinkLED.c
				Vectors.c
b	.ink_delay();			Documents
P	ORTT = OxFF;	//LED on		
b	.ink_delay();			
P	RTT = 0x00;	//LED off		
b	.ink_delay();			
)				
/				
void b	ink delav(void))		
(•		
int i.				
fo	(i=0;i<64000;i+	++)		
(_	
icc	12w -c -IC:\icc	\include\ -e -l -Wf-cp	don C:\Test\vectors.c	
ico	12w -o Test -LC	:\icc\lib\ -btext:0xC00	0 -bdata:0x7000 -bextcode:0x000	I I
Done.				
				I I
				<u> </u>
39: 1	C:V	\Test\BlinkLED.c	C:\Test\Test.prj	\$19

Click Terminal menu – Show Terminal window as shown

Note that ICC12 will immediately change to show terminal window.

MageCraft IDE for ICC12 (STANDARD) [WARNING: 45 Days	EVALUATION version]	<u>_ X</u>
File Edit Search View Project RCS Tools Terminal Help		
1 🗗 🗁 🖬 🗳 😹 💼 🖼 🖉 🚘	ERR ERR	
	Download File:	Project Browser Image: TEST Image: TES
	Download	Documents
	Dpen Com Pot	
	Show Editors	
		I I
icc12w -c -IC:\icc\include\ -e -1 icc12w -o Test -LC:\icc\lib\ -btext Done.	-Wf-opdon C:\Test\vectors.c :0xC000 -bdata:0x7000 -bextcode:0x000	
Terminal Window	C:\Test\Test.prj	S19

Setup the terminal BAUD by clicking on Tools menu – Environment options as

shown.

ImageCraft IDE for ICC12 (STANDA)	ARD) [WARNING: 45 Days E	WALUATION version]	
File Edit Search View Project RCS	Tools Terminal Help		
🖺 🖻 🖬 🗳 👘 🥵	Environment Options		
F	Editor and Print Options	wnload File:	Project Browser
	Configure Tool		E-IC TEST
	Kun		
		<u>B</u> rowse	vectors.c
		Download	🔁 Headers
			Documents
		Open <u>C</u> om Port	
		Show Editors	
	_		
•	▼ ►		
iccl2w -c -IC:\ic iccl2w -o Test -L	c\include\ -e -l C:\icc\lib\ -btext	-Wi-cpdon C:\Test\vectors.c	
Done.			
			<u> </u>
T	erminal Window	C:\Test\Test.prj	S19

Select the Terminal Tab and the set the BAUD = 9600, Flow control = None, ASCII Transfer Protocol = None. Select the COM that the serial cable is connected to at the back of the PC.

Environment Options	×
Preferences Terminal	
COM Port COM 1 COM 2 COM 3 COM 4 Baudrate: 9600 Terminal Font	Flow Control None Hardware (CTS/RTS) Software (^S/^Q) Keep DTR Active ASCII Transfer Protocol None Line delay (ms.) 10 Wait for ^{***} (Flash)
OK Cance	el <u>H</u> elp

In the middle of ICC12 is the Open COM Port button. Push the button to open COM port as shown. Note that it will change to Close Com Port.

ImageCraft IDE for ICC12 (STANDARD) [WARNING: 45 Days	EVALUATION version]	
File Edit Search View Project RCS Tools Terminal Help		
🗓 🗗 🖬 🗳 😹 🛍 🥵 🖉 🎎 💷 🔜	ERR ERR	
	Download File: Browse Download!	Project Browser
	Close Com Port	
	Show Editors	
icc12w -c -IC:\icc\include\ -e -1 icc12w -o Test -LC:\icc\lib\ -btext Done.	-₩f-cpdon C:\Test\vectors.c ▲ ::0xC000 -bdata:0x7000 -bextcode:0x000	
Terminal Window	C:\Test\Test.prj	S19 //

The MCU must boot up in single chip mode in order for the Loader to work. Set the various jumpers as indicated below.

Jumper settings for MODA and MODB for single chip mode.

JB1 – pin 2 and 3 shunted – MODB JB5 – shunted – MODA SW3 to SGL (single chip) position.

Jumper settings for FLASH and RAM with REV0 to REV2A

JB3 – pin 1 and 2 shunted JB4 – CSP0* and F shunted JB6 – CSD* and R shunted

Jumper settings for FLASH and RAM with REV3

JB3 – pin 1 and 2 shunted (CSD* RAM enable) JB6 – pin 1 and 2 shunted (CSP0* FLASH select) JB7 – pin 1 and 2 shunted (RAM write enable)

Connect Serial Cable to COM 1 at the back of PC and to the Adapt812DX. Slide SW2 Run/Boot switch to Run position. Make sure SW3 is in SGL position. Power up the Adapt812DX board with a known good power supply, making sure the PWR LED is on. Once the board is powered, ICC12 Terminal should display the MXFlash Loader menu as shown.

Notice that it detected the FLASH to be **AMD Flash Device**. If the Loader cannot detect a FLASH memory it will display unknown Memory and will assume that the memory is RAM.

ImageCraft IDE for ICC12 (STANDARD) [WARNING: 45 Days EVALUATION vers	sion]	
File Edit Search View Project RCS Tools Terminal Help		
1 🗗 🗗 🗳 🛷 🖻 🥵 🟹 🝰 💷 📰 📼		
AMD Flash Device	Download File:	Project Browser
812 NARROW MODE AMD/ST FLASH UTILITY VER. 2.01	Browse	⊟@> Files
		vectors.c
F = b Dump 16K Block Program Memory from Requested Page G => Execute Program Pointed to by Reset Vector J => Toggle Unknown Proram Memory size - 128K/512K N => Toggle Banked/Non-banked Program Memory Mode N => Steps Menu:	<u>D</u> ownload!	Documents
E => Erase entire Flash Memory P => Program S1 or S2 record into Program Memory U => Uerify S-records in Program memory Baud choice A->9600, B->14K4, C->19K2, D->38K4	Close <u>C</u> om Port	
?		
	Show Editors	
	-	
•		
icc12w -c -IC:\icc\include\ -e -1 -Wf-cpdon C:\Test\vectors.c icc12w -o Test -LC:\icc\lib\ -btext:0xC000 -bdata:0x7000 -bextcode:0x000 Done.		
Terminal Window	C:\Test\Test.prj	\$19

There are several things to check if the Menu will not show up as shown above.

- Do not use NULL cable. Use a straight thru cable.
- Check and verify the cable is connected to the correct COM port.
- Re-boot the PC if necessary as other application may have corrupted the OS.
- Use known good serial cable.
- If there are other problems, test the COM port with another device that is known to work.
- If the FLASH type is not recognized re-check jumper settings. If still unrecognized then the board has a problem.

Erasing:

To erase the FLASH the command is *E*.

ImageCraft IDE for ICC12 (STANDARD) [WARNING: 45 Days EVALUATION version]		
File Edit Search View Project RCS Tools Terminal Help		
🗓 💪 🗆 🧳 😹 🛍 🥵 🖗 🍰 🔤 🖬 📾		
AMD Flash Device 812 NARROW MODE AMD/ST FLASH UTILITY UER. 2.01 F => Dump 16K Block Program Memory from Requested Page G => Execute Program Pointed to by Reset Uector J => Ioggle Unknown Proram Memory size - 128K/S12K M => Ioggle Banked/Non-Danked Program Memory Mode H => Show Menu E => Erase entire Flash Memory P => Program S1 or S2 record into Program Memory Baud choice A->9600, B->14K4, C->19K2, D->38K4 ? e Flash erased OK ? Iccl2w -c -IC:\icc\include\ -e -1 -Wf-cpdon C:\Test iccl2w -o Test -LC:\icc\include\ -btext:OxCOOO -bdata:Ox7	Download File: Browse Download Close Com Port Show Editors	Project Browser
Done.		
	Þ	<u> </u>
Terminal Window	C:\Test\Test.prj	J\$19 //

Programming:

There are 2 options for programming the FLASH. The 1^{st} is whether the S-record is BANKED. The command is **N** to toggle from BANKED to NON-BANKED. ICC12 creates mixed S1 and S2 record. This is generally called BANKED mode. One should always check the S19 record to see if there are S1 and S2 mixture. Other compiler creates BANKED S2 too. One should experiment whether to program the S-record as BANKED or NON-BANKED.

By choosing the **N** command the loader will always toggle from previous state to the next. By default it is set for BANKED mode.

If the S-record contain only addresses from \$C000 to \$FFFF then it should always be programmed as NON-BANKED. Linear S2 record must always be programmed as NON-BANKED.

Note: For ICC12 the S2 record is programmed as BANKED.

Type P to initiate program.

ImageCraft IDE for ICC12 (STANDARD) [WARNING: 45 Days EVALUATION ver	sion]	
File Edit Search View Project RCS Tools Terminal Help		
1 🗗 🗳 🚽 🛍 🥵 🔍 🚘 🔤 📰		
AMD Flash Device	Download File:	Project Browser
812 NARROW MODE AMD/ST FLASH UTILITY UER. 2.01	Browse	BlinkLED.c
F => Dump 16K Block Program Memory from Requested Page G => Execute Program Pointed to by Reset Vector J => Toggle Unknown Proram Memory size 128K/512K M => Toggle Banked/Non-banked Program Memory Mode	Download!	Vectors.c Headers Documents
M => Show Menu E => Erase entire Flash Memory P => Program S1 or S2 record into Program Memory U => Verify S-records in Program memory Baud choice A->9600, B->14K4, C->19K2, D->38K4 ? e Flash erased OK	Close Com Port	
? p Send S-record file now		
	Show Editors	
icc12w -c -IC:\icc\include\ -e -1 -Wf-cpdon of icc12w -o Test -LC:\icc\lib\ -btext:0xC000 -bd. Done.	C:\Test\vectors.c	
Terminal Window	C:\Test\Test.prj	\$19

Use the *Browse* button to help locate the file to be uploaded.

Open		? ×
Look in: [Test 💌 🗢 🖻 📸 🎟 -	
Test.s19		
J File name:	Task e10	-1
riie name.	Upen Upen	
Files of type:	s19 (*.s19) Cancel	
	Dpen as read-only	
riles of type.	Open as read-only	

Select the Test.s19 file then press the Open button.

Once selected the *Download* button will now become active.

ImageCraft IDE for ICC12 (STANDARD) [WARNING: 45 Days EVALUATION version	1	
File Edit Search View Project RCS Tools Terminal Help		
🗓 😅 🖬 🗳 🤯 🖻 🥵 👰 🔬 💷 📰 📰 🚺		
AMD Flash Device 812 NARROW MODE AMD/ST FLASH UTILITY UER. 2.01 F => Dump 16K Block Program Memory from Requested Page G => Execute Program Pointed to by Reset Uector J => Toggle Unknown Proram Memory size - 128K/512K N => Toggle Banked/Non-banked Program Memory Mode M => Show Menu E => Erase entire Flash Memory P => Program S1 or S2 record into Program Memory U => Uerify S-records in Program memory Baud choice A->9600, B->14K4, C->19K2, D->38K4 ? e Flash erased OK ? p Send S-record file now Flash programmed OK ?	Download File: C:\Test\Test.s19 Browse Download! Close Com Port	Project Browser
? Done. ASCII Downloading C:\Test.s19 Done	Show Editors	
Terminal Window	C:\Test\Test.prj	S19 //

After programming, slide SW3 to EXP (MODA = 1) then press the RESET button. The programmed application in now running in expanded NARROW mode.

Checking DATA in Load mode:

The command is *F* for memory dumping DATA from FLASH. This is used to check certain parts of the memory.

For example, one may want to check the address from \$C000 to \$FFFF. How is this possible? There are few things that needs to be understood. The memory block \$C000 to \$FFFF is always present in the memory map. This is where the ISR and start of code resides. When servicing an ISR it is possible to access a different PPAGE. With 128Kbyte Flash, the \$C000 to \$FFFF can be access at PPAGE = \$07 in the page window \$8000 to \$BFFF.

For 512Kbyte Flash, the \$C000 to \$FFFF can be access at PPAGE = \$1F in the page window \$8000 to \$BFFF.

Here is an example to dump \$C000 to \$FFFF with a 128Kbyte Flash. Select *F* command then type the value *07*

ImageCraft IDE for ICC12 (STANDARD) [WARNING: 45 Days EVALUATION version]		
File Edit Search View Project RCS Tools Terminal Help		
1 🔁 🖬 🗳 😹 🛍 🕵 🖓 🎎 💷 🖬 🖬 📕		
7 AMD Flash Device	Download File: C:\Test\Test.s19	Project Browser
812 NARROW MODE AMD/ST FLASH UTILITY VER. 2.01	Browse	BlinkLED.c
F => Dump 16K Block Program Memory from Requested Page G => Execute Program Pointed to by Reset Uector J => Toggle Unknown Prome Memory size - 128K/512K N => Toggle Banked/Non-banked Program Memory Mode M => Show Menu	<u>D</u> ownload!	Headers Documents
E => Erase entire Flash Memory P => Program S1 or S2 record into Program Memory U => Verify S-records in Program memory Baud choice A->9600, B->14K4, C->19K2, D->38K4 7 e	Close Com Port	
Flash erased OK		
Send S-record file now		
?		
2	Show Editors	
? f		
Page XX?	1	
Done		
ASCII Downloading C:\Test\Test.s19		
ASCII Downloading C:\Test\Test.s19		
Done		
ASCII Downloading C:\Test\Test.s19		
Pone		
Terminal Window	C:\Test\Test.prj	S19 //

Immediately after it would begin to dump the DATA beginning at \$8000 to \$BFFF. Please note that PPAGE = \$07 at \$8000 to \$BFFF is the same as \$C000 to \$FFFF

ImageCraft IDE for ICC12 (STANDARD) [WARNING: 45 Days EVALUATION version]		
File Edit Search View Project RCS Tools Terminal Help		
<u> 1 🗗 🖬 🎯 ở 🖬 5, 🔍 🎎 🖤 🖬 🕮 🖬 💶 🗖 🚺 🖬 🗰 🖬 🖿 👘 👘 🖬 👘 👘 🖬 👘 🖬 👘 👘 👘 👘 👘 👘 👘 👘 👘 👘 👘 👘 👘 </u>		
? f Page XX? 8000 CF0C 0016 C073 87CE 7000 8E70 0027 056A 8010 0008 20F6 CEC0 78CD 7000 8EC0 7827 0618 8020 0130 7020 F515 C022 A20F5 3H57 751B 9EC6 8030 047B 000A C6B0 7B00 0BCC C07B 0037 C607 8040 7820 007 C6F7 7B00 AE4A 8000 0420 10C6 F72B 8050 000F C6FF 7B00 AE4A 8000 0420 10C6 F72B 8060 000F 4A80 0004 7700 AE4A 8000 0420 EEF7 8070 573 3D1D 0016 0730 FFFF FFFF FFFF FFFF	Download File: C:\Test\Test.s19 Browse Download	Project Browser
8080 FFFF FFFF FFFF FFFF FFFF FFFF FFFF	Close Com Port	
8140 FPFF FPFF FFFF FFFF FFFF FFFF FFFF FF	Show Editors	
Done		
Done		
ASCII Downloading C:\Test\Test.s19		
ASCII Downloading C:\Test\Test.s19 Done		
Terminal Window	C:\Test\Test.prj	S19 //

Below is at the end of DATA dump.

ImageCraft IDE for ICC12 (STANDARD) [WARNING: 45 Days EVALUATION version]		
File Edit Search View Project RCS Tools Terminal Help		
<u>🕆 🖻 🚳</u> 🖈 💼 🕵 👰 <u></u> 🖬 🔤 🖬 🗖		
BE10 PFPF PFFF PFFF	Download File: C:\Test\Test.s19 Browse Download! Close Com Port Show Editors	Project Browser
?		
Done ASCII Downloading C:\Test\Test.s19 Done ASCII Downloading C:\Test\Test.s19 Done ASCII Downloading C:\Test\Test.s19 Done	A	
		<u> </u>
Terminal Window	C:\Test\Test.prj	\$19

Note that at address \$BFFE:BFFF is equal to \$FFFE:\$FFFF. The value is seen to be \$C000 the start of code.

This tool is implemented to help diagnose where are the DATA being programmed too from an S-record file.

Executing Program in Load Mode:

The command is G to execute the already program code. The Loader will fetch the address at \$BFFE:\$BFFF (PPAGE set at the Last page) then start executing the code at that address. For example, if the value was \$C000, the loader removes the EEPROM out of the memory map, read the vector address and make Register X equal \$C000 and jump to \$C000 with Register X.

Note:

In load mode certain registers are initialized. The stack is set to \$0C00, MODA:MODB in expanded narrow mode including the various expanded register controls. Lastly the COP control is disabled.

Problems:

Q. Programmed okay but will not appear to run.

A. Toggle from BANKED to NON-BANKED or vice-versa then erase and reprogram again.

Q. Tried to program in BANKED or NON-BANKED, Flash programmed OK but will not work.

A. There are 3 things to check.

1. COP control needs to be disable if not being serviced.

2. Check if the Stack is initialized

3. Check if the Power ON reset vector is correctly pointing to start of code.

Q. Programmed OK and runs with G command but not in Expanded mode.A. Check the stack is initialized and COP control disabled if not being serviced. In load mode the stack is initialized and COP disabled. Make sure to slide SW3 to Expanded Narrow mode.

Q. During programming the red LED comes on

A. The Loader is unable to program into a memory location. The reasons are many. The S-record may contain code that are manipulating the registers. The S-record may contain addresses that are outside the bound of the FLASH chip. Lastly, the FLASH memory maybe faulty.

Q. Programming Rev3 is different from Rev2A.

A. Verify the jumper settings FLASH and RAM

JB3 – pin 1 and 2 shunted (CSD* RAM enable) JB6 – pin 1 and 2 shunted (CSP0* FLASH select)

JB7 – pin 1 and 2 shunted (COI 0 1 LASH select JB7 – pin 1 and 2 shunted (RAM write enable)