

Migration guide for EW430 2.21B

This is a quick step by step description on how to migrate code and embedded workbench projects from version 1.x to version 2.x of EW430. More detailed information about the migration process is available in the migration part in the [MSP430 IAR C/EC++ Compiler Reference Guide](#).

The migration process

The project file

1. Start version 2.x of EW430 and create a new workspace by selecting **File>New** and then **Workspace**.
2. From the Project menu select **Insert Project into the Workspace** to include your old project into the workspace. This step will create two new project files with the same name as the old one, but with the extensions .ewp and .ewd. The .ewp file contains all settings required to build the application while the .ewd file contains all settings related to the debugger. The old project file will remain untouched.
3. The **Device** option on the **Target** options page in the **General** category must be set by hand, as it is new. This option will select the correct linker command file and device description file.
4. The following segment names have been changed:

Old name	New name
CDATA0	DATA16_ID
CONST	DATA16_C
CSTR	DATA16_C
IDATA0	DATA16_I
NO_INIT	DATA16_N
UDATA0	DATA16_Z

5. The specification of runtime library has, together with the possibility to ignore cstartup in the library, been moved from the linker command file. This means that it is possible to use the same linker command file for both C and assembler code. The library and the **Ignore CSTARTUP in library** option can be set on the **Include** options page in the **XLINK** category.

C source code

1. Replace or modify used extended keywords according to the migration part in the [MSP430 IAR C/EC++ Compiler Reference Guide](#). The include file for the old intrinsics, in430.h, defines some of the old extended keywords and can be used if you want to keep the old syntax. However, the syntax of the following extended keywords must be changed:

- **Interrupt**

Old interrupt syntax:

```
interrupt[VECTOR] void my_function(void)
```

New interrupt syntax:

```
#pragma vector=VECTOR
__interrupt void my_function(void)
```

If you want to keep your source compatible with earlier version of the compiler the following code can be used:

```
#if __VER__ < 200
interrupt [ VECTOR ] void my_function( void )
#else
#pragma vector=VECTOR
__interrupt void my_function( void )
#endif
{
}
```

The following error messages will be displayed if the old interrupt syntax is used:

```
Error[Pe077]: this declaration has no storage class or type specifier
Error[Pe065]: expected a ";"
```

- **sfrb/sfrw**

These extended keywords have been removed and must be replaced:

```
sfrb name = address;
```

must be replaced with

```
__no_init volatile unsigned char name @ address;
```

and

```
sfrw name = address;
```

must be replaced with

```
__no_init volatile unsigned short name @ address;
```

2. Replace or modify used pragma directives according to the migration guide.
3. Make sure that you do not use nested comments in your code, in ICC430 version 2, nested comment are never allowed.
4. If your are using a I/O include file named io430xxx.h, you must change the name of the include file to msp430xxx.h. The new set of io430xxx.h files are using bit fields to define individual bits and will require that your code is changed.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		Applications	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265