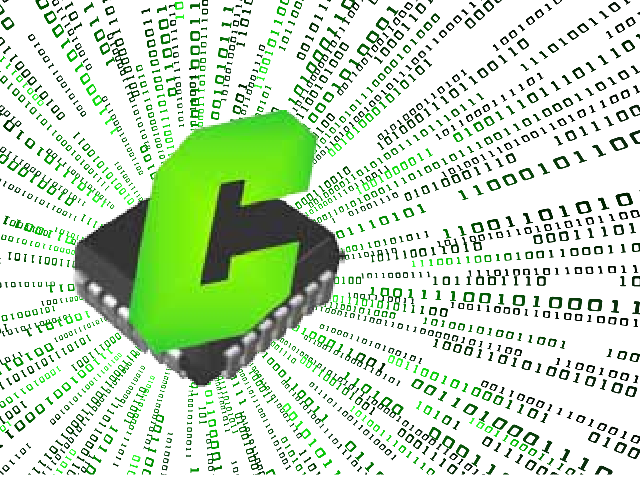


May 2020

<Bits & Bytes Newsletter/>

TECH NOTE: Review of the new PIC18FxxQ10 Family

TECH NOTE: Compiler Feature Highlight

TECH NOTE: Debugging Support for Dual Core dsPIC33CH Family

SPECIAL OFFERS: COVID-19 Response

CCS Inc

Email:
support@ccsinfo.com

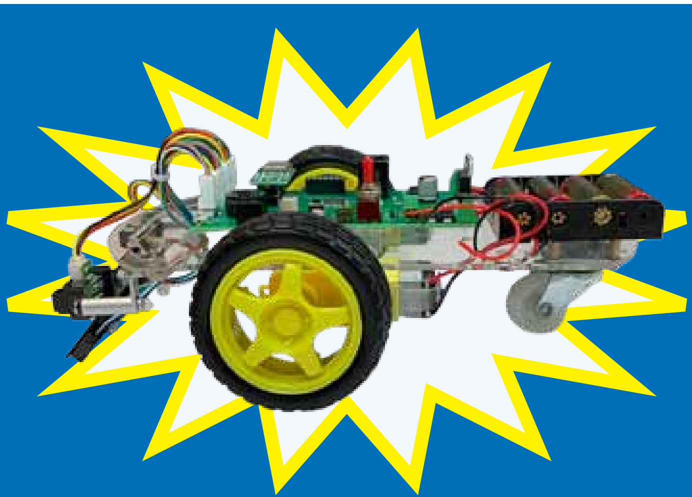
sales@ccsinfo.com
Ph: 262-522-6500

www.ccsinfo.com

New Compiler Versions
>Downloads>Compiler Software

Easy License Renewal
>Support>Renewals

Product Spotlight



Robot Car Development Kit!

Review of the new PIC18FxxQ10 Family

The CCS C Compiler supports the new PIC18FxxQ10 family of microcontroller, which are low-cost alternative to similar general purpose devices.

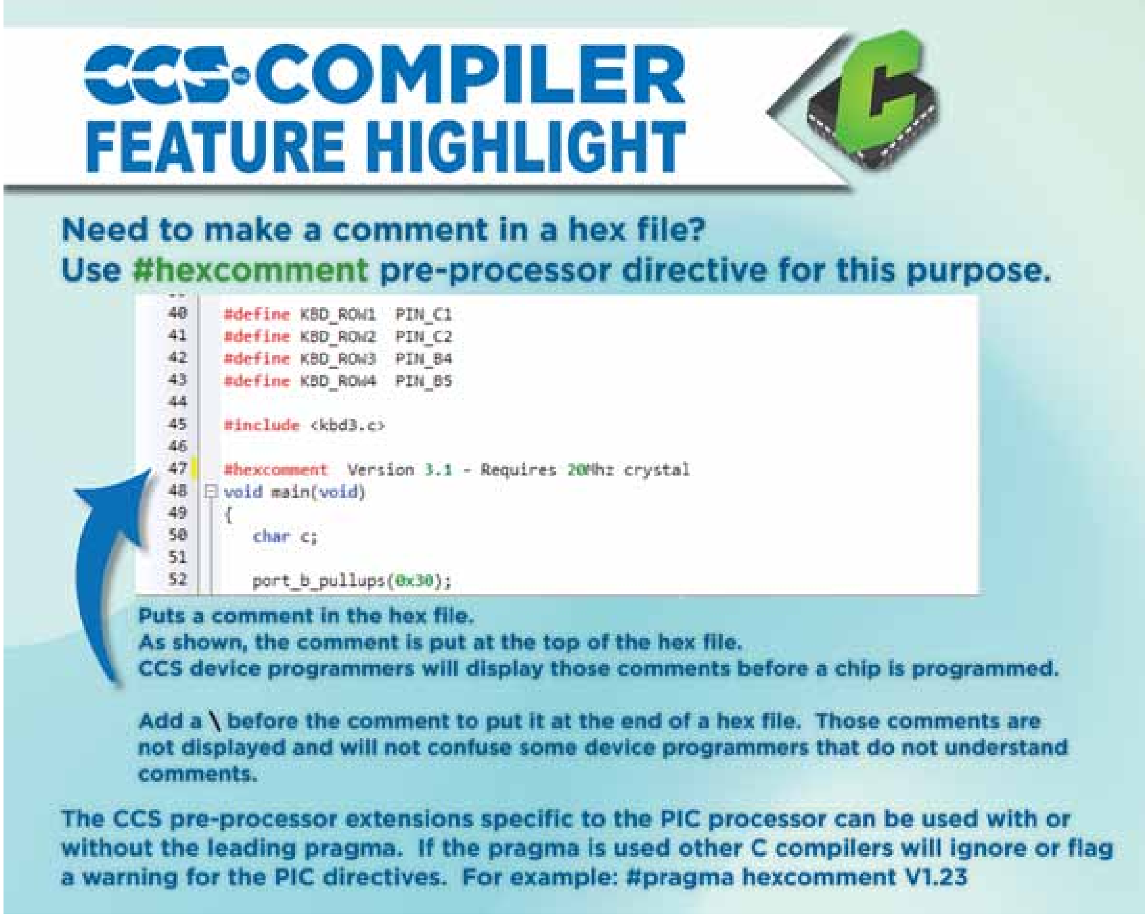
The PIC18FxxQ10 family come in 28, 40 and 44 pin packages of which up to 36 of them can be used as I/O pins. Additionally has up to 128 K Bytes, 65536 instructions, of Program Flash Memory, up to 1024 Bytes of Data EEPROM and up to 3615 Bytes of Data SRAM.

The PIC18FxxQ10 family has a wide array of Analog and Digital peripherals that can be used with it including:

- 10-bit ADC with Computation module on up to 35 external channels
- 5-bit DAC, two High-Speed Analog Comparators
- Up to 7 Timers (3 8-bit/4 16-bit)
- Windowed Watchdog Timer, a Cyclic Redundancy Check (CRC) module
- Two Capture/Compare/PWM (CCP) modules
- Two 10-bit PWM modules
- Zero-Cross Detect (ZCD) module
- Up to one Complementary Waveform Generator (CWG) module
- Up to eight Configurable Logic Cell (CLC) modules
- Low Voltage Detect (LVD) module
- Up to two Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART) modules
- Up to two Master Synchronous Serial Port (MSSP) modules

Additionally almost every I/O pins for the PIC18FxxQ10 family can be assigned to almost any peripheral, using the CCS C Compiler's #pin_select directive, making it highly configurable for your specific hardware implementation.

If you plan to use this family of devices, make sure you have the latest compiler. Check the status of your download rights on our website: www.ccsinfo.com/renewals



Need to make a comment in a hex file?
Use #hexcomment pre-processor directive for this purpose.

```
#define KBD_ROW1  PIN_C1
#define KBD_ROW2  PIN_C2
#define KBD_ROW3  PIN_B4
#define KBD_ROW4  PIN_B5

#include <kbd3.c>

#hexcomment  Version 3.1 - Requires 20Mhz crystal
void main(void)
{
   char c;

   port_b_pullups(0x30);
```

Puts a comment in the hex file.
As shown, the comment is put at the top of the hex file.
CCS device programmers will display those comments before a chip is programmed.

Add a \ before the comment to put it at the end of a hex file. Those comments are not displayed and will not confuse some device programmers that do not understand comments.

The CCS pre-processor extensions specific to the PIC processor can be used with or without the leading pragma. If the pragma is used other C compilers will ignore or flag a warning for the PIC directives. For example: #pragma hexcomment V1.23

Product Spotlight
Robot Car Development Kit

NEW!

Have fun learning about embedded programming with the CCS Robot Car development kit. The development kit includes the prototyping board to attach to the robot car once it is assembled. The sensors provided as part of the assembly will be used directly in the example programs from the Exercise Book and to be able to compete autonomously.

Infared sensors can detect white or dark under the car and can be used for line sensing or line following. Proximity sensors can detect if it is approaching an object. Magnetic compass can be used to detect current heading. The car is controlled by two direct DC motors. This kit is a great learning tool for showing how an embedded system can read analog and digital sensors, then react to those sensors.

This kit has an optional EZ App Lynx connector. If an EZ App Lynx is attached, the car can be controlled over Bluetooth® using an Android or Apple iOS device. All sensor readings can be displayed on the Android or Apple iOS device.

CCS designed a Robot Soccer field for Maker Faire. Kids, and adults, enjoyed playing Robot soccer with tablets. See a video here:
https://www.ccsinfo.com/content.php?page=video_tutorials

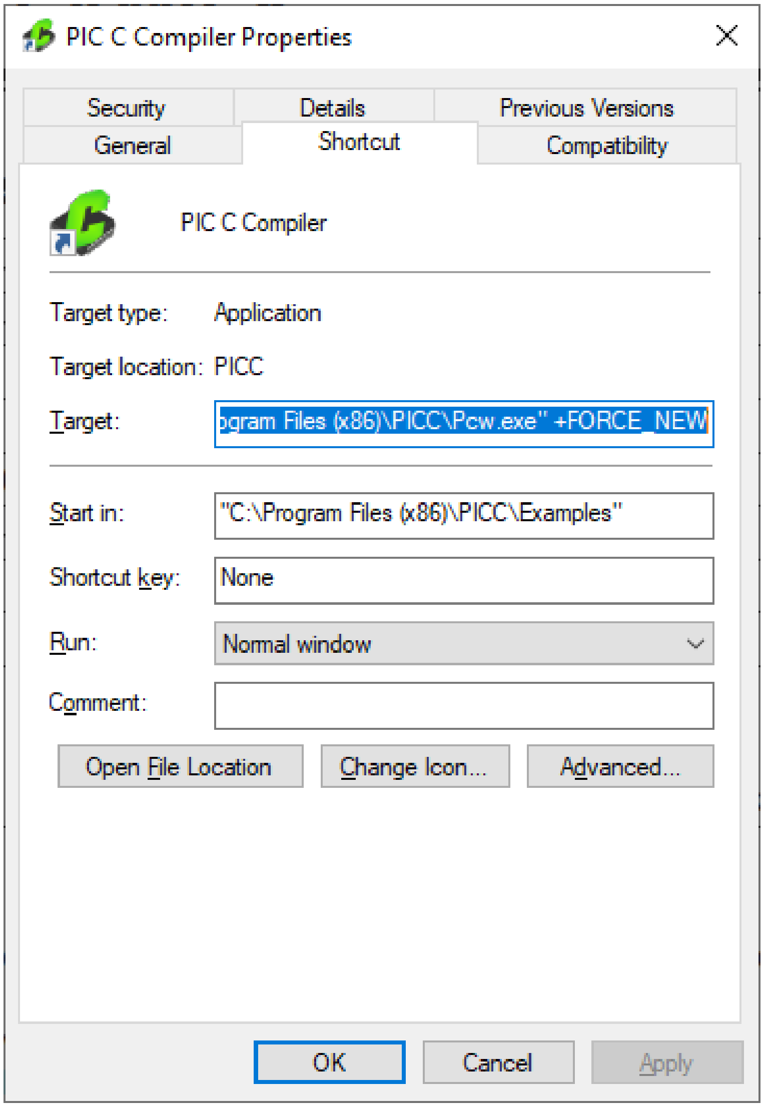
Robot Car Development Kit is available Now!

Debugging Support for Dual Core dsPIC33CH Family

Introducing support added for debugging dsPIC33CH devices for both the master and slave cores to the CCS C Compiler and IDE using the CCS ICD-U80 or ICD-U64 device programmers. This support will be available starting with Compiler version 5.094 and ICD firmware version 3.42 and newer. In addition to debugging the slave core with the CCS tools the slave core can also be programmed starting with the previously mentioned version. Both of these features will aid in developing code for the dsPIC33CH dual core devices.

When debugging there are three setups that can be done. First debug only the master core, second debug only the slave core, or third debug both the master and slave core at the same time. When debugging the slave core the master core's configuration bit need to be set so that the slave core can be debugged. At minimum the S1_ISOLAT, S1_DEBUG and S1_ICSPx, x being the debug pin to use for the slave core, should be set. The S1_ISOLAT configuration fuse allows the slave core to operate even if the master core hasn't set the SLVEN bit in the MSI1CON register, enabling the slave core to run, the S1_DEBUG configuration fuse enables the slave core debugger and the S1_ICSPx configuration fuses sets which S1MCLRx, S1PGCx and S1PGDx pins are being used to debug the slave core.

When debugging both the master core and slave core at the same time it requires that two instances of the CCS C Compiler and IDE be running, two device programmers, and the development board will have to have two ICD connectors on it connected to different MCLR, PGC and PGD pins, one for the master core and one for the slave core. To run two instances of the CCS C Compiler and IDE +FORCE_NEW needs added to the target line of the icon used to launch the IDE, for example:



Target type: Application

Target location: PICC

Target: ogram Files (x86)\PICC\Pcw.exe" +FORCE_NEW

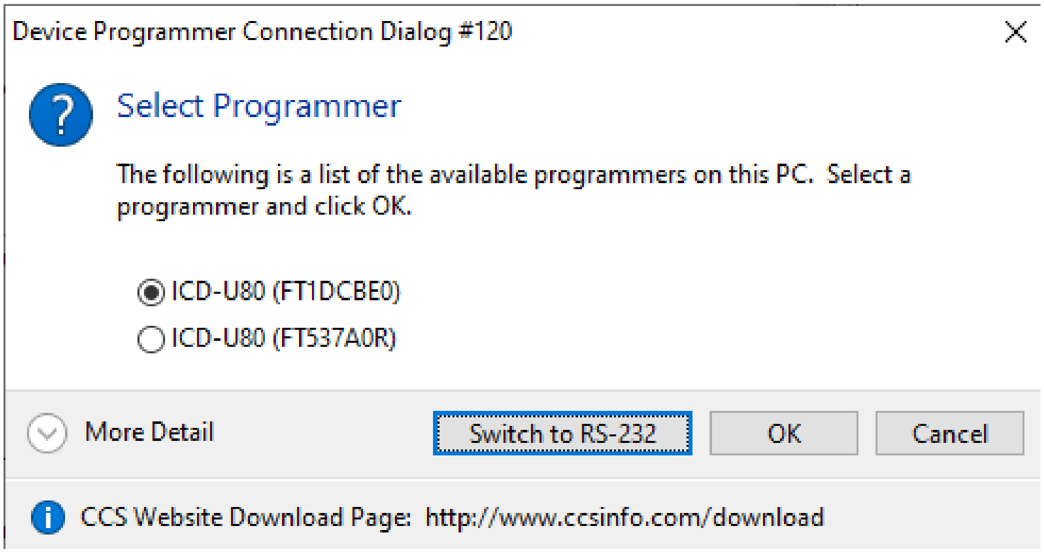
Start in: "C:\Program Files (x86)\PICC\Examples"

Shortcut key: None

Run: Normal window

Comment:

With the above change when using that icon to start the IDE will cause a new instance of the IDE to start allowing for more the one instance to be running. Once there are two instances of the IDE running the master core project should being opened in one and the salve core project should be opened in the other. Next starting with the master core project, build it with the appropriate slave core configuration fuses set and start the debugger. Assuming both ICDs are connected to the PC a selection box similar to the following should pop up:



Device Programmer Connection Dialog #120

Select Programmer

The following is a list of the available programmers on this PC. Select a programmer and click OK.

● ICD-U80 (FT1DCBE0)

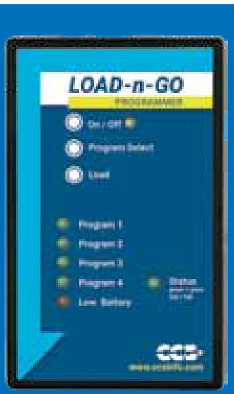
○ ICD-U80 (FT537A0R)

CCS Website Download Page: http://www.ccsinfo.com/download

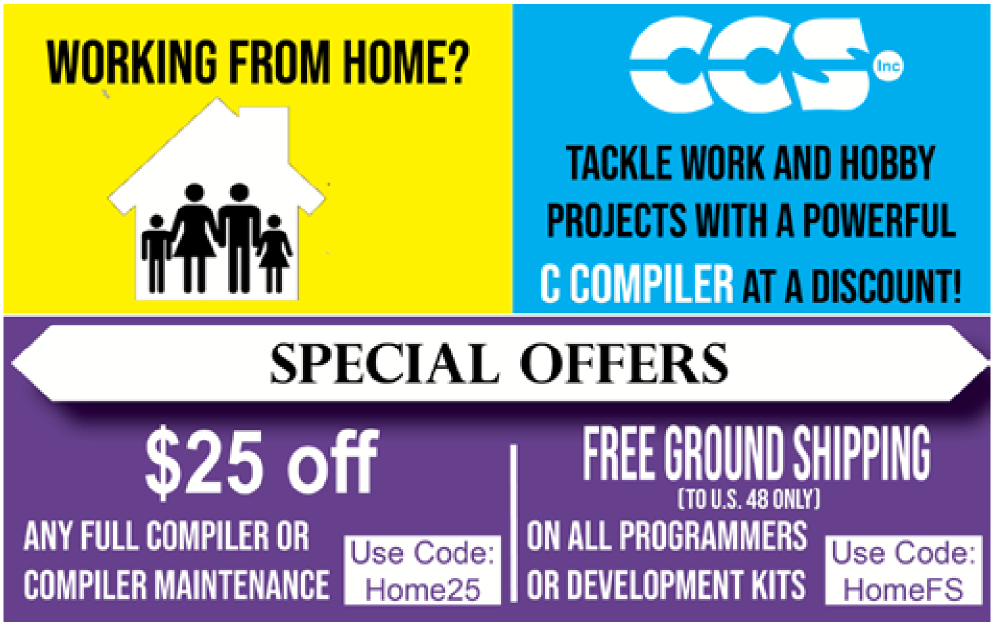
Select the ICD that is connected to the master core's programming/debugging pins and select 'OK'. If you're not sure which ICD is connected to the master core you may want to disconnect the ICD connect to the slave core debugging pins from the PC and then start the master core's debugger, then when the master core's debugger is loaded the slave core ICD can be reconnected to the PC. Once the master code program debugger is loaded, switch to the CCS IDE instance with the slave core project, build it and start its debugger. Once it's done loading both the master and slave core can be debugged at the same time.

One thing to be aware of the master code debugger using the MCLR pin of the PIC. If the master core's debugger pulls that pin low to reset the master core it will also reset the slave core. So if the master core debugger is reset or the program if reloaded when the slave core is also being debugged, the slave core's debugger needs to be reset to synchronize it with the slave core.

Check out the popular CCS Programmer / Debuggers



COVID-19 RESPONSE


WORKING FROM HOME?
CCS Inc
TACKLE WORK AND HOBBY PROJECTS WITH A POWERFUL C COMPILER AT A DISCOUNT!
SPECIAL OFFERS
$25 off
ANY FULL COMPILER OR COMPILER MAINTENANCE
Use Code: Home25
FREE GROUND SHIPPING
(TO U.S. 48 ONLY)
ON ALL PROGRAMMERS OR DEVELOPMENT KITS
Use Code: HomeFS

During this time of global uncertainty and change, we want to assure you that we are taking every precaution to ensure that we can safely support our customers during this time.

Despite these challenges, CCS staff is continuing to provide technical support, as well as processing orders. It is essential customers have the tools they need to provide the development of existing or new products that may be necessary in the fight of Covid-19.

Many of our existing customers are having to work from home and we want to remind everyone of our Software Licensing Agreement. We pre-register all compilers in a user's name. You can install your compiler on your home PC and laptops. If you do not have access to the registration files and installer, contact customer service for assistance.

CCS wants to help further embedded development by customers, and is offering a discount on any new compilers or maintenance plan purchases. The customers that need development boards, and programmers, we are offering Free Ground shipping (to the U.S.48) so you can get the tools you need to continue working from home.

Most importantly, as we work together in this unique and rapidly changing environment, we do so with confidence that we will overcome this challenge. Until then, we hold our enduring commitment to the health and well-being of our employees and customers.

Please let us know how we can help you. Stay healthy.

More than 25 years experience in software, firmware and hardware design and over 500 custom embedded C design projects using a Microchip PIC® MCU device. We are a recognized Microchip Third-Party Partner.

www.ccsinfo.com

Follow Us!