

CodeWarrior™ QUICC Engine Utility

Overview

The CodeWarrior™ QUICC Engine Utility is a GUI tool that speeds and simplifies initialization and configuration of the drivers and communications protocols managed by the QUICC Engine module. The tool provides an easy-to-use environment for handling common QUICC Engine programming tasks and offers immediate access to relevant documentation. It is designed to help developers quickly

initialize the QUICC Engine device so they can focus on application creation and other tasks that differentiate their end products in the marketplace.

The utility works with any tool set and supports implementation of a variety of protocols. It includes functionality that discovers resource collisions and issues an alarm when several drivers simultaneously share common device resources.

Features

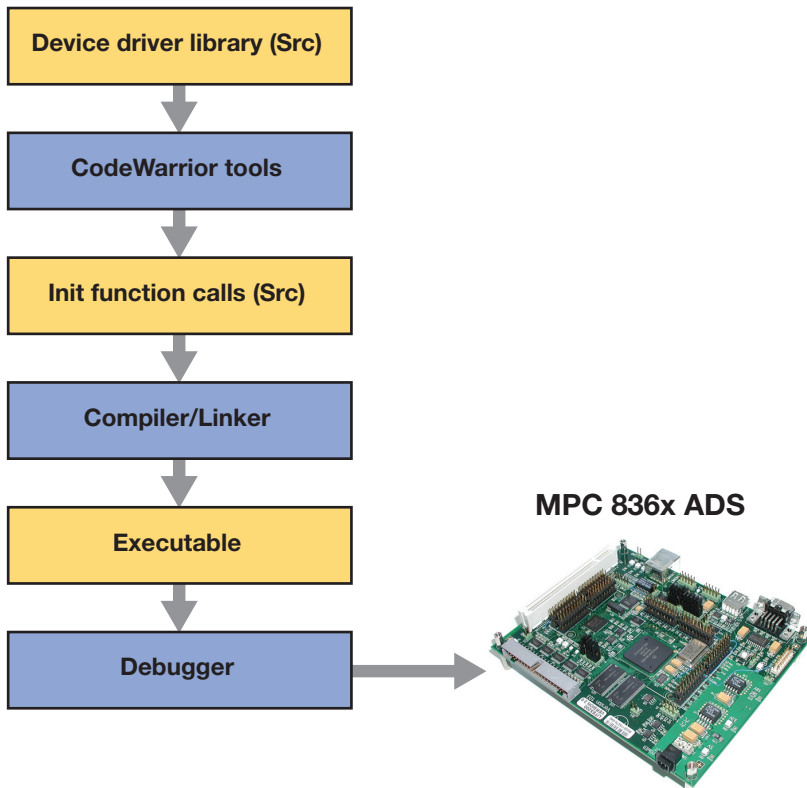
Protocol Configuration

- > Point-and-click device initialization
- > Initializes devices using a standard driver API
- > GUI acts as the driver interface, generating API calls that configure and initialize the device to designer requirements
- > Supports multiple protocols
- > Available with any tool set
- > Document window with access to pertinent section of QUICC Engine document

Hierarchical GUI

- > Three-level hierarchical GUI featuring standard-, device- and driver-aware functionality
 - Standard-level protocol GUI: for designers who understand the protocol but don't want to invest time setting detailed features
 - Hardware and uCode implementation-level GUI: supports device-specific hardware and uCode implementations as well as default values; provides device parameters necessary to improve application implementation. Users can define proprietary features such as auto VC off and amount of Rx buffer
 - Driver-level GUI: provides specific driver features and default values. Allow designers to take full advantage of driver API strength without driver API complexity

Continued >



Register View

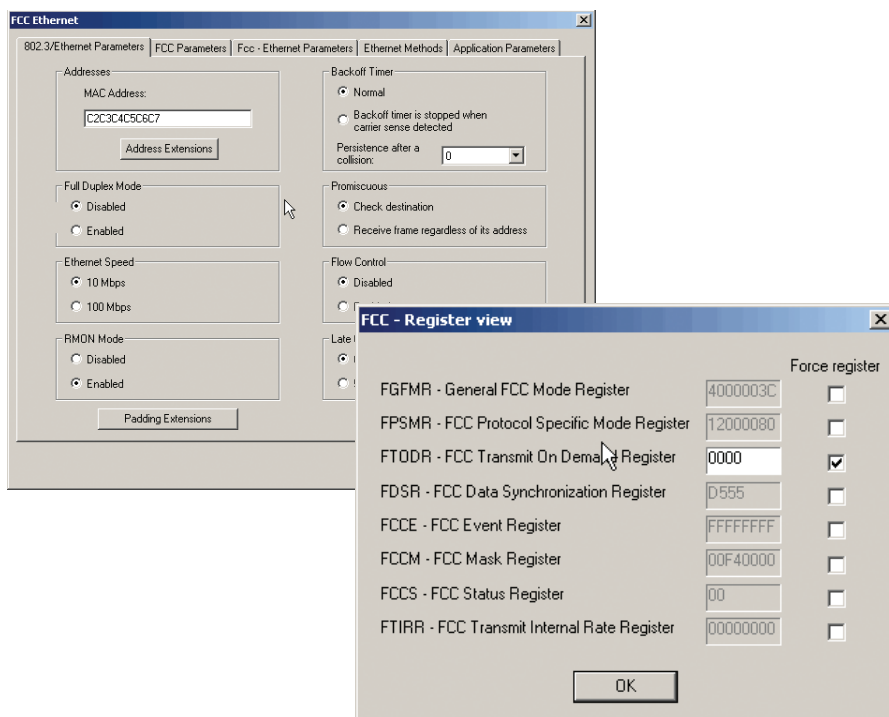
- > Details the registers and memory views of a specific protocol
- > Calculates values of the driver API with a single click
- > Screen is visible throughout configuration setup — providing a view of registers being initialized by a particular driver API
- > Designers can force specific values of registers initialized by a specific driver

Warning and Collision Alarms

- > Allows for manual changes to utility GUI default parameters, as well as integration of several protocols to run on the same device
- > Two types of alarms are available
 - A warning — detailing a problematic parameter — appears if a manually defined parameter causes an issue, but the device can still run the application
 - A collision alarm — detailing a problematic parameter — appears if a manually defined parameter causes an issue that prevents device setup
- > Collisions are checked at the system level across different protocols; a comparison is then made against the device's available resources and parameters

Online Documentation

- > Significantly minimizes the need for documentation during the initial stages of device setup
- > Document window offers access to specific, pertinent sections of QUICC Engine documentation



Supported communication protocols

- > 802.3 Ethernet
- > L2 Switch
- > Transparent
- > ATM
 - AALO
 - AAL1
 - AAL2
 - AAL5
- > IMA
- > Multilink multiclass PPP
- > POS
- > TDM
- > HDLC
- > UART
- > QMC (QUICC Multi-Channel Controller)

- > SPI – Serial peripheral controllers
- > USB
- > I²C

Security protocol

- > Encryption

Other interfaces

- > PCI
- > Local Bus
- > Interrupt controller
- > DDR memory controller
- > MMU
- > I/D cache

Other specifications

- CPUs (target platform) supported**
- > Freescale™ MPC836x

Host Platform

- > Microsoft® Windows® XP/2000

Learn More: For more information about Freescale products, please visit www.freescale.com/codewarrior