TECHNICAL NOTE

TN273

Updating Your Application from the RCM5700 to the RCM6700 Rabbit® MiniCore™ Family

Introduction

This technical note describes some of the common considerations and concerns that may be encountered in updating a Rabbit application from the RCM5700/RCM5710/RCM5750/RCM5760 (collectively, RCM57xx) MiniCore family to the RCM6700/RCM6710/RCM6750/RCM6760 (collectively, RCM67xx) MiniCore family.

Comparing the RCM57xx and the RCM67xx Rabbit MiniCores

Dynamic C® Versions

RCM57xx and RCM67xx MiniCores are supported by the following Dynamic C versions:

MiniCore Model	Supported by Dynamic C Version
RCM5700/RCM5710	10.44 and later
RCM5750/RCM5760	10.56 and later
RCM6700/RCM6710	10.64 and later
RCM6750/RCM6760	10.66 and later*

* Although support for RCM6750/RCM6760 boards is first mentioned in the Dynamic C 10.64 release notes, a pre-production change to RCM6750/RCM6760 hardware requires corresponding software changes that are supported by Dynamic C 10.66 and later versions.

Binary Incompatibility

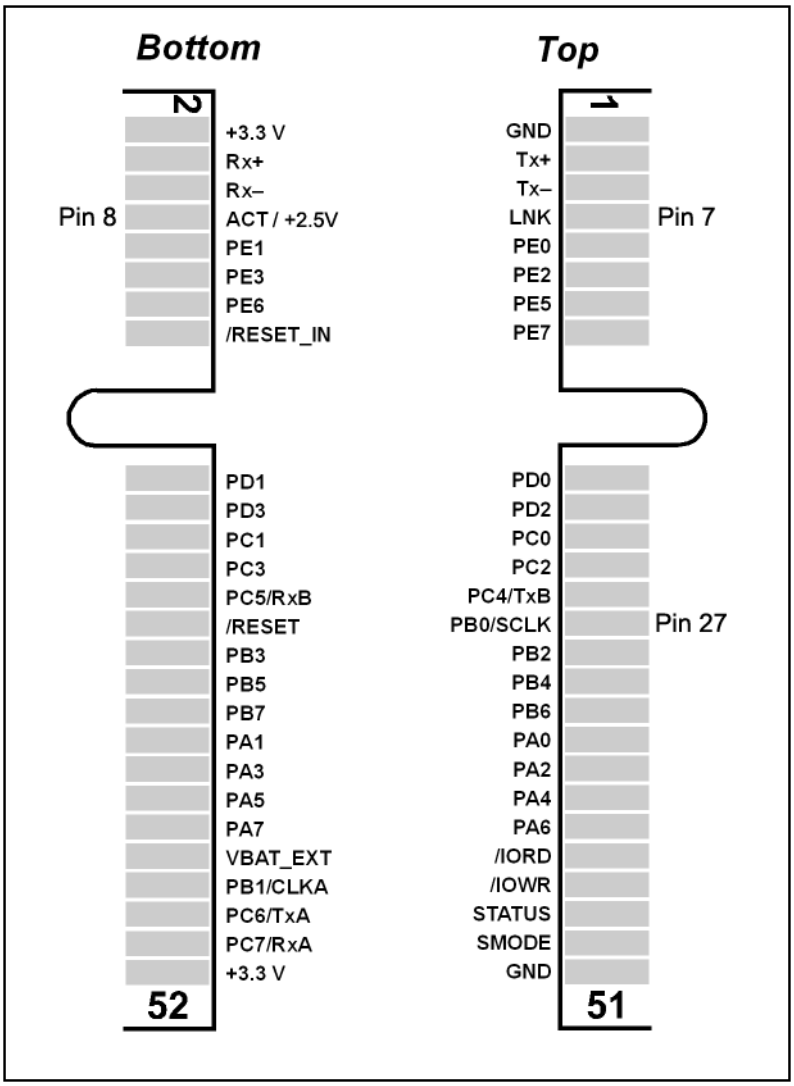
RCM57xx applications are not binary compatible with the corresponding RCM67xx board. This binary incompatibility is due to a number of factors, with the primary reason being the difference between the program storage flash devices. Where the RCM57xx stores and runs its application in its parallel program flash, the RCM67xx stores its application in serial program flash buts runs the application in its internal fast RAM.

Main Oscillator Maximum and Minimum Clock Rates

Another significant reason for binary incompatibility is the difference between Rabbit 5000 and Rabbit 6000 CPU clocks. With regard to maximums, an RCM57xx board is limited to its 25 MHz main oscillator clock being doubled to 50 MHz. In contrast, an RCM67xx board's PLL is by default set to multiply the 25 MHz main oscillator rate to provide a 187.5 MHz CPU clock frequency.

On the minimums side, RCM57xx applications may operate in one of the Rabbit 5000 CPU's low-power "sleepy" modes, potentially using as low as a 2.048 KHz CPU clock frequency. However, an RCM67xx's Rabbit 6000 CPU can only operate from its internal 1MByte fast RAM at a minimum clock frequency of about 12 MHz; below this CPU clock rate the internal fast RAM content (including application code) is not refreshed often enough to be retained.

Edge Connector Changes



Pinout Diagram

Edge Connector	Board	Default Use	GPIO Use
Pin 7	RCM57xx	Ethernet link	No
	RCM6700/ RCM6750	Combined Ethernet link plus activity output signal	PG0 when DISABLE_ENET_STATUS is defined. See “Edge Connector Pin 7” below.
	RCM6710/ RCM6760	None	PG0
Pin 8	RCM57xx	Ethernet activity output signal	No
	RCM67xx	2.5VDC Ethernet supply voltage	No
Pin 27	RCM5700/ RCM5710	None	PB0/SCLKB
	RCM5750/ RCM5760	Synchronous serial clock connected to the on-MiniCore data storage serial flash	PB0/SCLKB is a shared resource. See “Edge Connector Pin 27” below.
	RCM67xx	Synchronous serial clock connected to the on-MiniCore serial boot flash	PB0/SCLKB is a shared resource. During boot-up the Rabbit CPU requires exclusive, unimpeded access to SCLKB. See “Edge Connector Pin 27” below.

Edge Connector Pin 7

Edge connector pin 7 on an RCM57xx board carries only an Ethernet link output signal.The RCM6700/ RCM6750 boards' edge connector pin 7 carries a combined Ethernet link plus activity output signal by default, via the Rabbit CPU's PG0. On RCM6700/RCM6750 boards, PG0 may be used instead for GPIO on edge connector pin 7 only when the DISABLE_ENET_STATUS macro is defined in Dynamic C's Project Options' Defines box. On RCM6710/RCM6760 boards, PG0 is always available for GPIO on edge connector pin 7.

Edge Connector Pin 8

On an RCM57xx board, edge connector pin 8 carries an Ethernet activity output signal. RCM67xx boards' edge connector pin 8 carries a 2.5VDC Ethernet supply voltage.

Edge Connector Pin 27

Edge connector pin 27 on an RCM57xx is direct connected (via a 0-Ohm jumper) to Rabbit's PB0/SCLKB pin and is generally available for GPIO, with the following caveat:

- On RCM5750/RCM5760 boards only, a 49.9 Ohm isolation resistor is placed between the serial flash's SCK input and the other on-MiniCore PB0/SCLKB connections. An RCM5750/RCM5760 application which uses PB0/SCLKB for both off-MiniCore connections and on-MiniCore serial flash communication is responsible for implementing a suitable mutex to protect the shared PB0/SCLKB resource.

An RCM67xx's edge connector pin 27 is isolated by a 49.9 Ohm resistor from the other on-MiniCore PB0_SCLKB connections. In particular, an RCM67xx's serial boot capability relies on the direct connection between the serial flash's SCK and the Rabbit CPU's PB0_SCLKB together with their isolation from edge connector pin 27. Because the PB0_SCLKB output is extremely active during the RCM67xx's boot-up from serial flash, if edge connector pin 27 is externally connected at all, it is strongly recommended that its use be restricted to that of a serial clock for additional external SPI devices. Note that if PB0_SCLKB is shared between the on-MiniCore serial boot flash and any other external use, that the RCM67xx application is responsible for implementing a suitable mutex to protect the shared PB0_SCLKB resource.

SPI Port Sharing Mutual Exclusion (Mutex) Examples

Introduced in Dynamic C 10.66, the Samples\RCM6700\Shared_SPI folder contains example programs that demonstrate use of an extensible mutex to allow sharing of the SPI serial port (B) between several different devices. Standard samples are provided for both cooperative and preemptive multitasking.