LatticeMico32 System Platform for uClinux and U-Boot

Lattice Semiconductor Corporation
5555 NE Moore Court
Hillsboro, OR 97124
(503) 268-8000

February 2008

Copyright

Trademarks

Disclaimers

to correct any errors contained herein or to advise any user of this document of any correction if such be made. LSC recommends its customers obtain the latest version of the relevant information to establish, before ordering, that the information being relied upon is current.

Type Conventions Used in This Document

Convention	Meaning or Use
Bold	Items in the user interface that you select or click. Text that you type into the user interface.
<*Italic*>	Variables in commands, code syntax, and path names.
Ctrl+L	Press the two keys at the same time.
`Courier`	Code examples. Messages, reports, and prompts from the software.
`...`	Omitted material in a line of code.
`.` `.` `.`	Omitted lines in code and report examples.
`[ ]`	Optional items in syntax descriptions. In bus specifications, the brackets are required.
`( )`	Grouped items in syntax descriptions.
`{ }`	Repeatable items in syntax descriptions.
`|`	A choice between items in syntax descriptions.

Contents

LatticeMico32 System Platform for uClinux and U-Boot

This document describes the LatticeMico32 platform used for the uClinux and U-Boot port and development.

Platform Prerequisites for U-BOOT and uClinux

The minimum platform prerequisites for working with the Lattice U-Boot and uClinux port distribution are the following:

- LatticeMico32 processor with the following:
  - Divide enabled
  - Multiplier enabled
  - Sign-extend enabled
  - Shifter enabled
  - Debug interface enabled
  - Location of exception handlers at 0x04000000
- 64-MB read/write memory, starting at address 0x08000000
- At least one LatticeMico32 UART instance
- At least one LatticeMico32 Tri-Speed Ethernet MAC instance

Note

Use of the DDR controller and Tri-Speed Ethernet MAC IP in new platforms requires recompilation in IPexpress and an IP core license. Free evaluation of the IP on an FPGA is provided with about a four-hour timeout before you purchase a license. Other IPs are free of charge.

- At least one LatticeMico32 timer instance

- One DDR controller for the main 64-MB memory

The platform provided with U-Boot and the uClinux port distribution uses each of the IPs just listed.

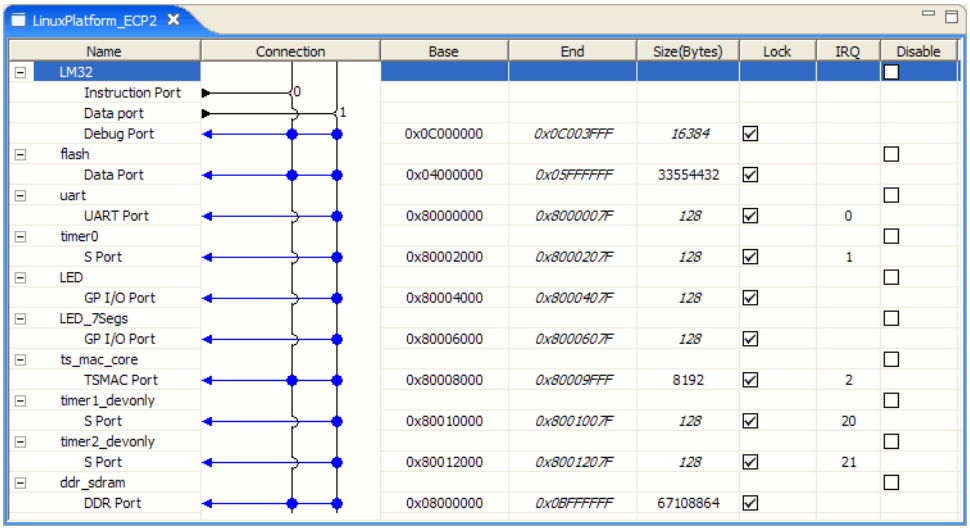
This platform has been tested on the following two DDR SDRAM memory modules:

- Infineon DDR SDRAM memory SODIMM 512M DDR333-2700
  P/N HYS64D64020GBDL-6-B
- Corsair memory SODIMM 512MB DDR266-2100
  P/N: VS512SDS266

Platform Design

Figure 1 shows the platform design used for U-Boot and uClinux development.

Figure 1: LatticeMico32 uClinux Platform in MSB

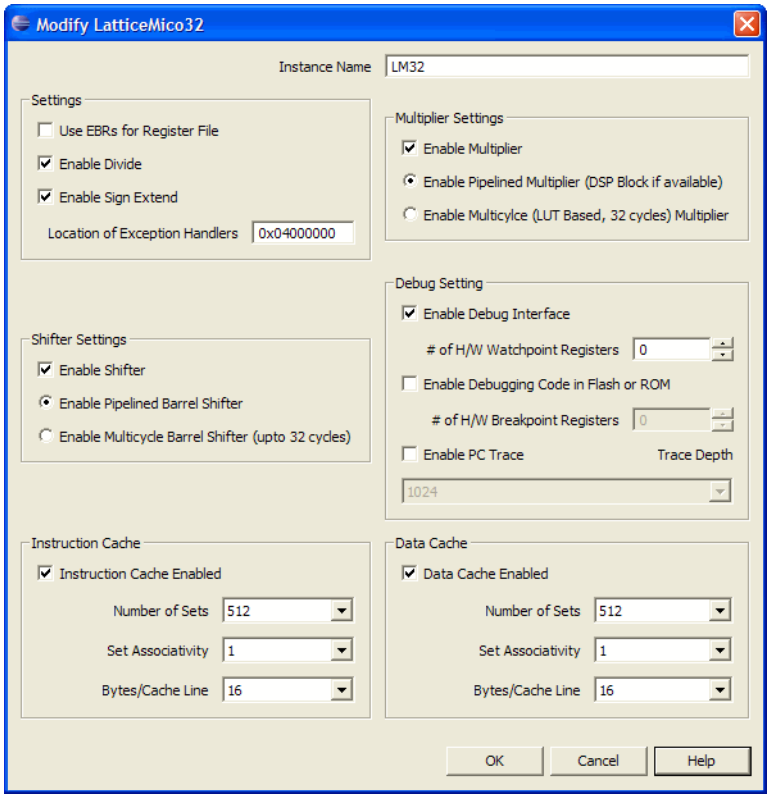


The following sections detail the component configurations used in the platform.

LatticeMico32 Processor

Figure 2 shows the component configuration of a LatticeMico32 processor in MSB.

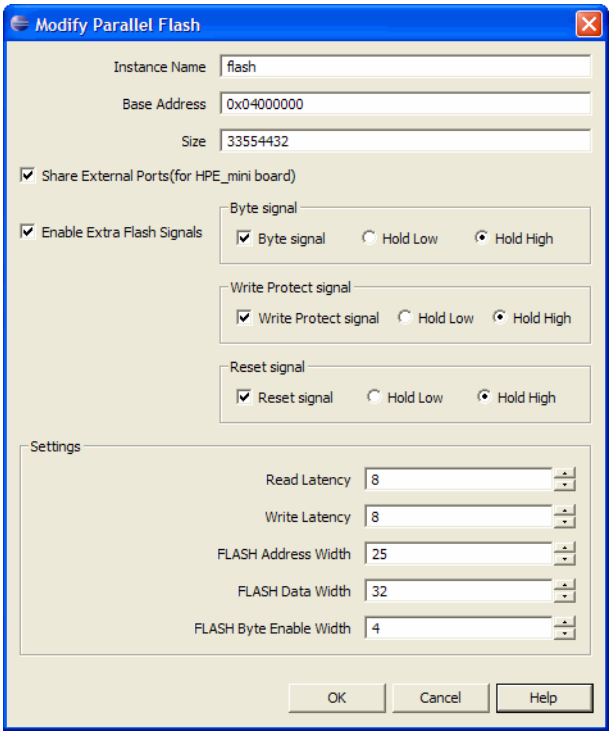
Figure 2: Configuration of LatticeMico32 Processor (Instance Name: LM32) in MSB



Parallel Flash

Figure 3 shows the component configuration of a parallel flash in MSB.

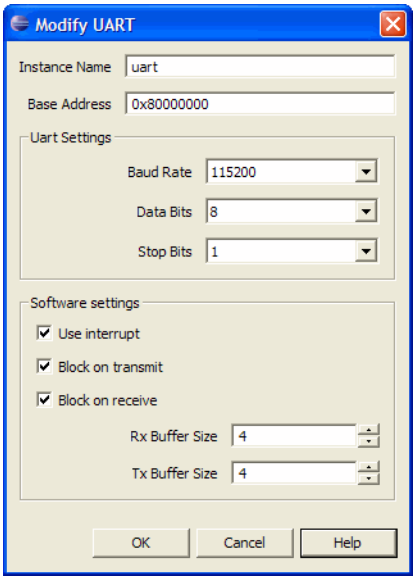
Figure 3: Configuration of Parallel Flash (Instance Name: flash) in MSB



UART

Figure 4 shows the component configuration of a UART in MSB.

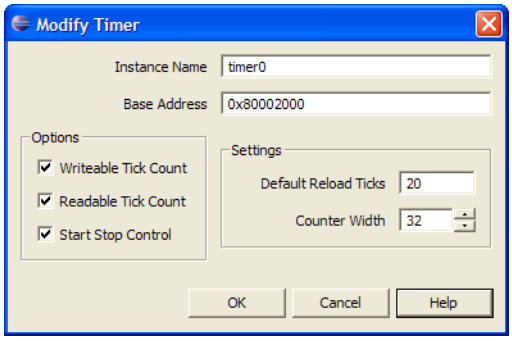
Figure 4: Configuration of UART (Instance Name: uart) in MSB



Timer

Figure 5 shows the component configuration of a timer instance called timer0 in MSB.

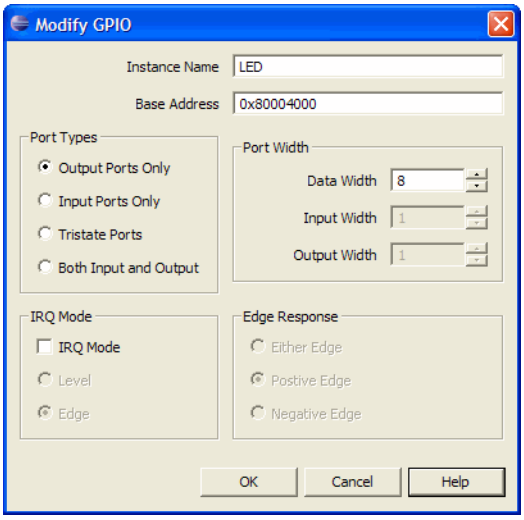
Figure 5: Configuratino of Timer (Instance Name: timer0) in MSB



GPIO

Figure 6 shows the component configuration of a GPIO instance called LED in MSB.

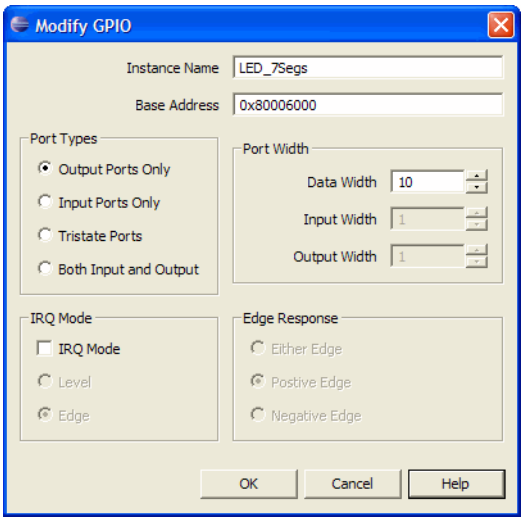
Figure 6: Configuration of GPIO (Instance Name: LED) in MSB



GPIO

Figure 7 shows the component configuration of a GPIO instance called LED_7Segs in MSB.

Figure 7: Configuration of GPIO (Instance Name: LED_7Segs) in MSB



Tri-Speed Ethernet MAC v2.5

Figure 8 shows the component configuration of a Tri-Speed Ethernet MAC v2.5 instance in MSB.

Figure 8: Configuration of Tri-Speed Ethernet MAC v2.5 (Instance Name: ts_mac_core) in MSB

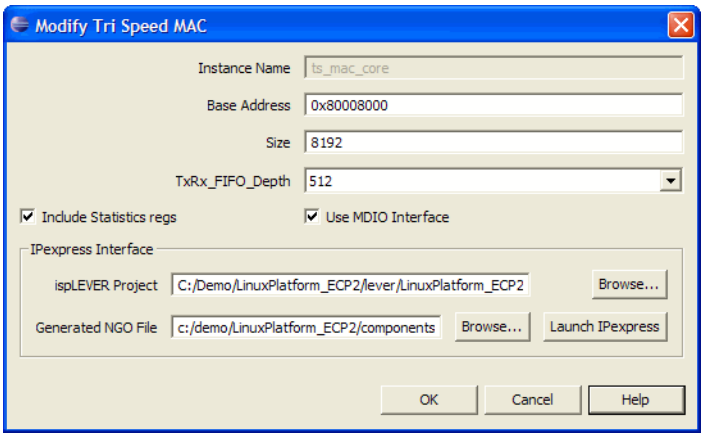
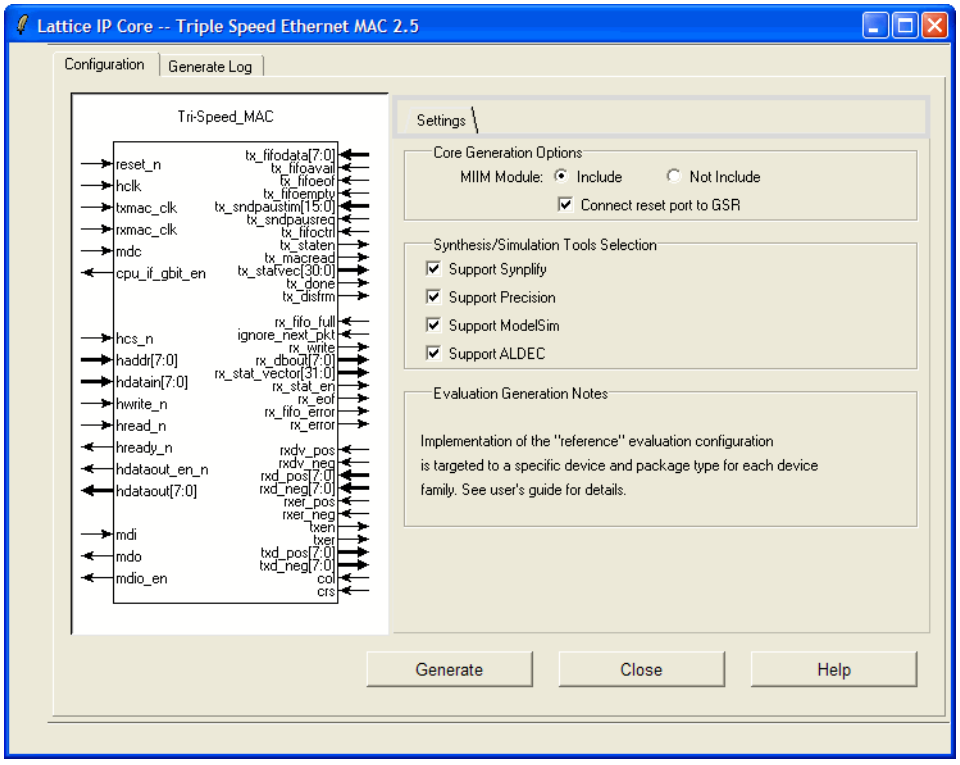


Figure 9 shows the configuration of the Tri-Speed Ethernet MAC in IPexpress.

Figure 9: Configuration of Tri-Speed Ethernet MAC v2.5 (Instance Name: ts_mac_core) in IPexpress



DDR SDRAM Controller v6.4

Figure 10 shows the component configuration of a DDR SDRAM controller instance called ddr_sdram in MSB.

Figure 10: Configuration of DDR SSDRAM Controller v6.4 (Instance Name: ddr_sdram) in MSB

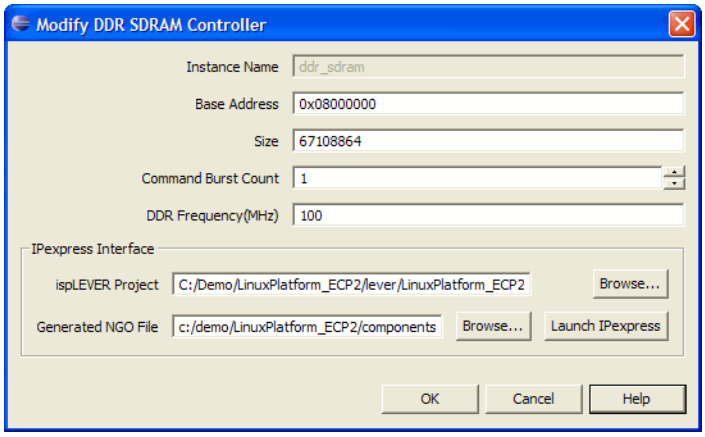


Figure 11 through Figure 16 show the configuration of a DDR SDRAM controller in each tab of the Lattice IP Core dialog box in IPexpress.

Figure 11: Configuration of DDR SDRAM Controller v6.4 (Instance Name: ddr_sdram) in Mode Tab of IPExpress

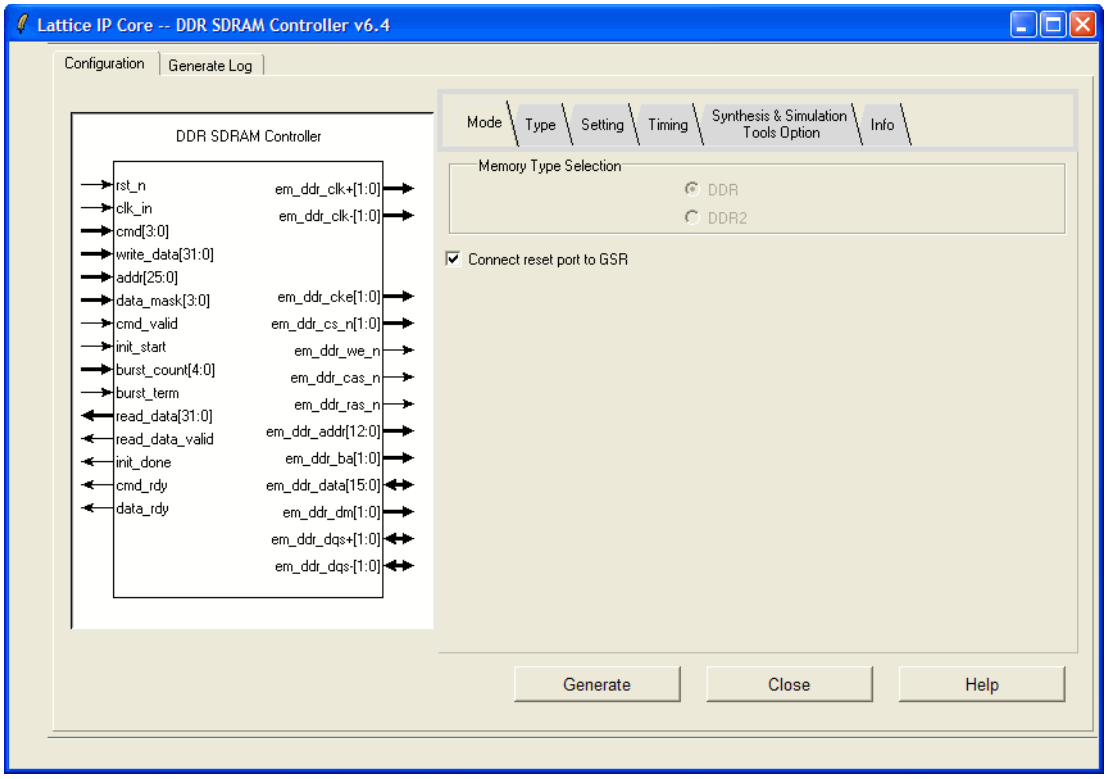


Figure 12: Configuration of DDR SDRAM Controller v6.4 (Instance Name: ddr_sdram) in Type Tab of IPexpress

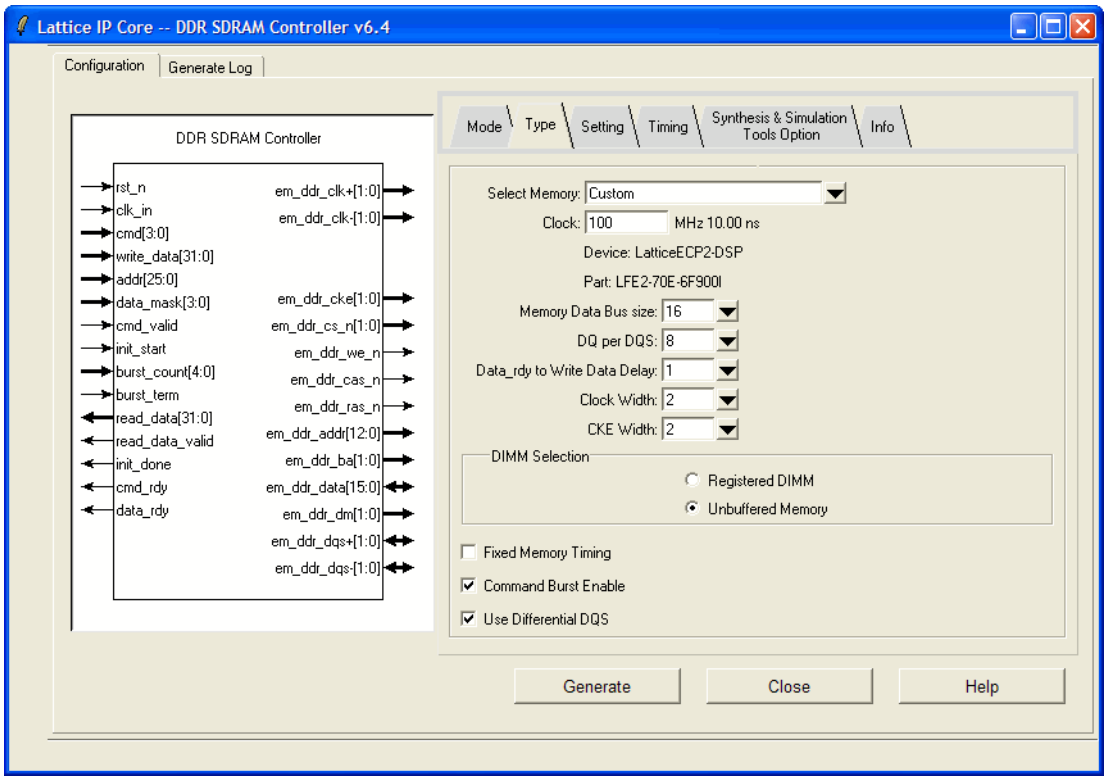


Figure 13: Configuration of DDR SDRAM Controller v6.4 (Instance Name: ddr_sdram) in Setting Tab of IPexpress

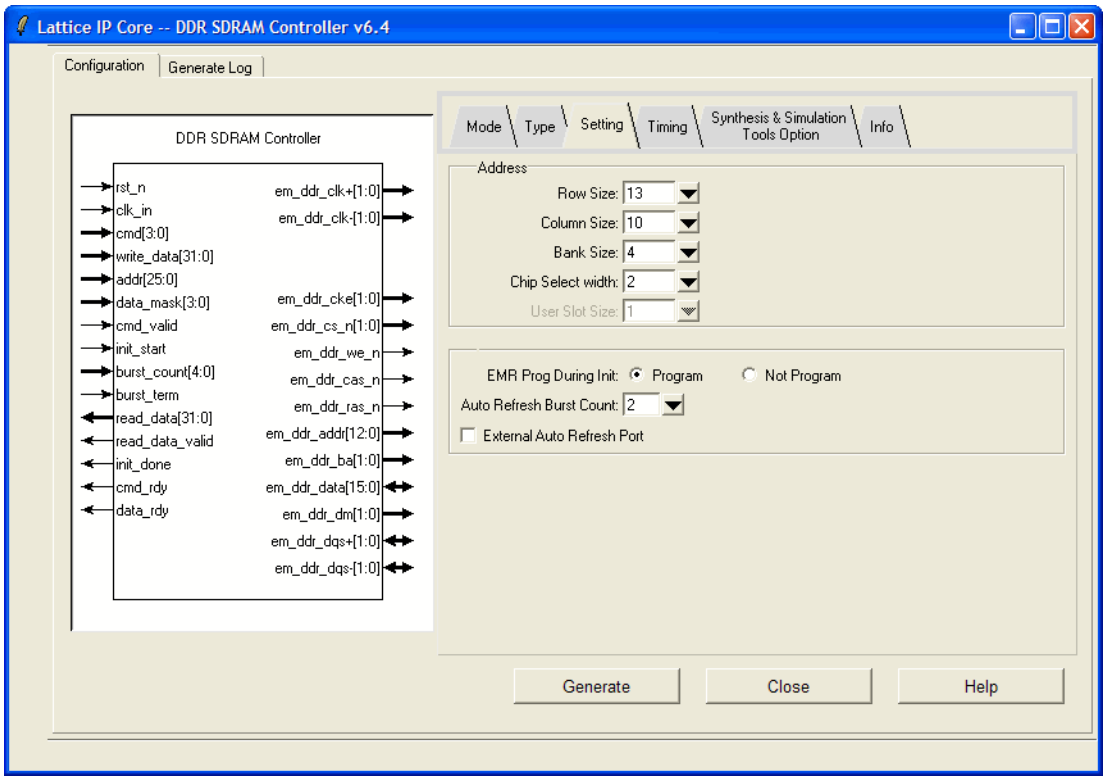


Figure 14: Configuration of DDR SDRAM Controller v6.4 (Instance Name: ddr_sdram) in Timing Tab of IPexpress

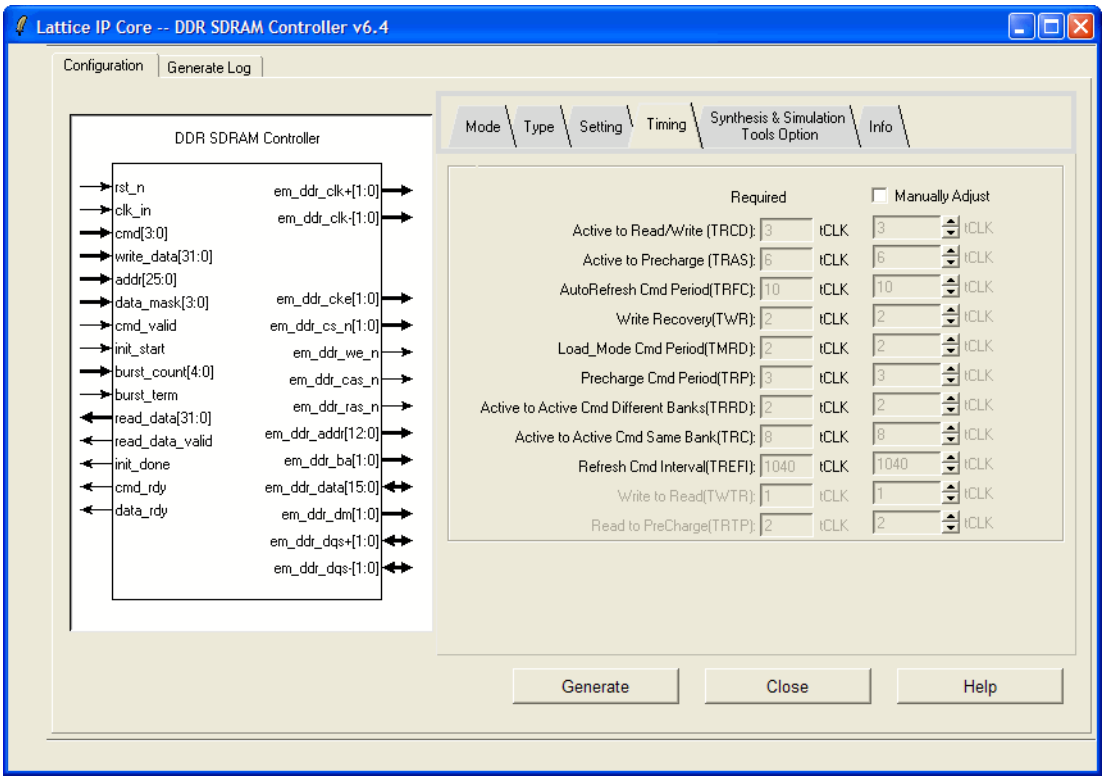


Figure 15: Configuration of DDR SDRAM Controller v6.4 (Instance Name: ddr_sdram) of Synthesis & Simulation Tools Option Tab of IPexpress

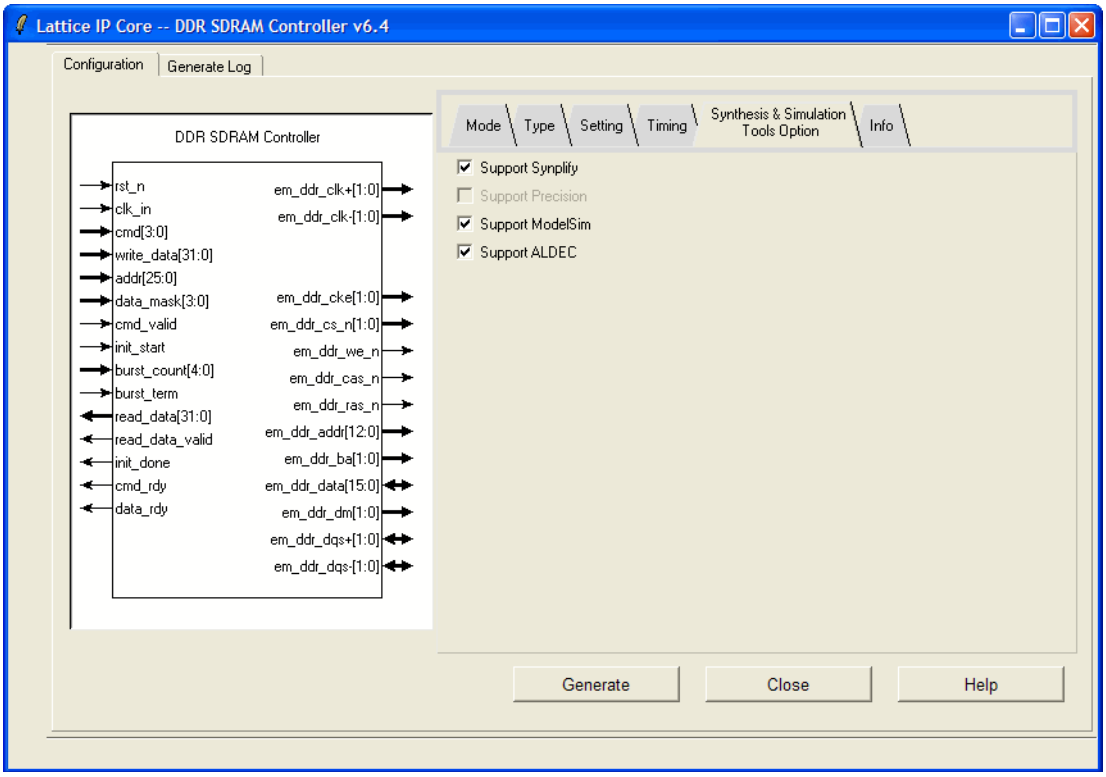
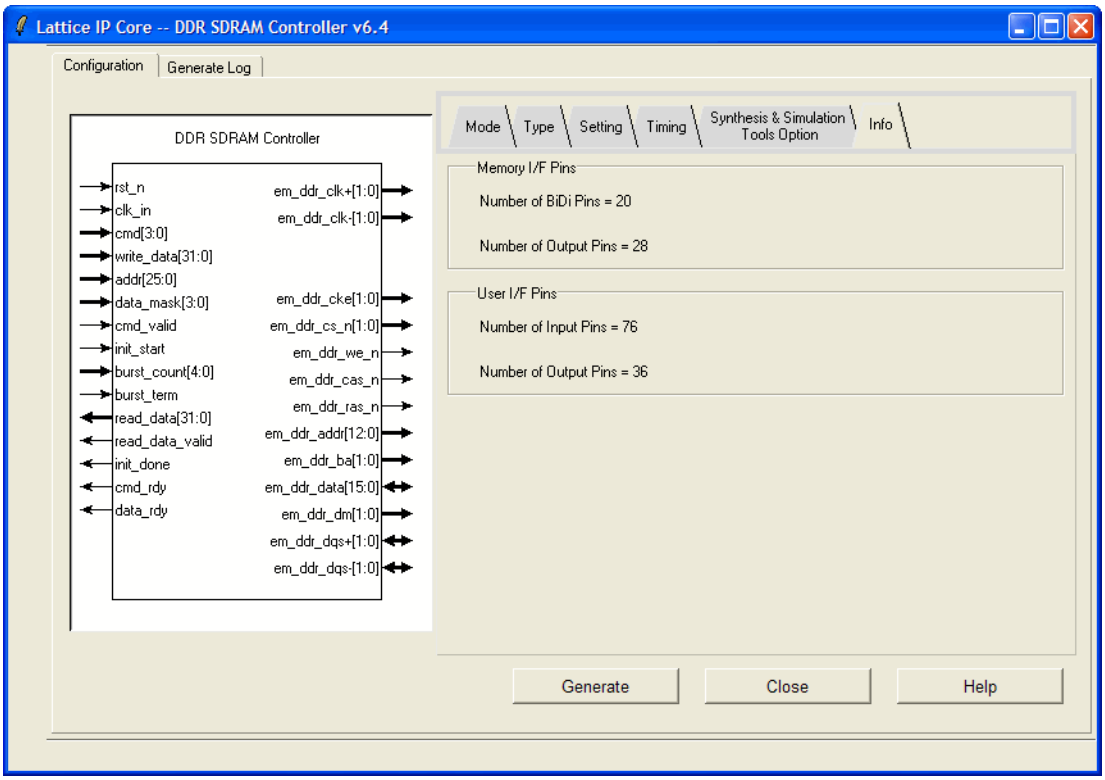


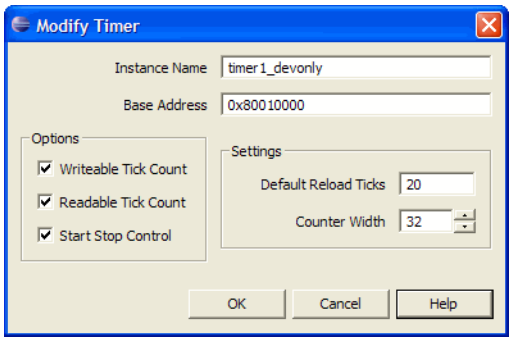
Figure 16: Configuration of DDR SDRAM Controller v6.4 (Instance Name: ddr_sdram) in Info Tab of IPexpress



Timer

Figure 17 shows the component configuration for a timer called time2_devonly.

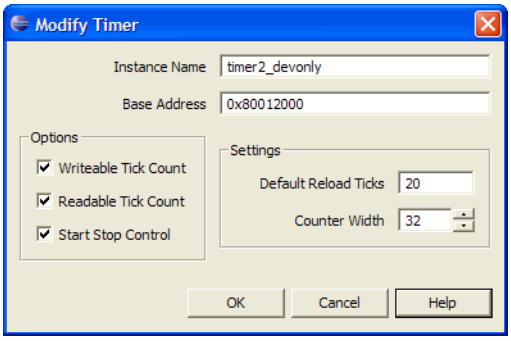
Figure 17: configuration of Timer (Instance Name: timer1_devonly) in MSB



Timer

Figure 18 shows the component configuration for a timer called timer2_devonly.

Figure 18: Configuration of Timer (Instance Name: timer2_devonly) in MSB



Index