



第11章

SOPC系统开发技术

11.1 32位NiosII嵌入式核

11.1.1 Nios II结构

表 11-1 Nios II 处理器系列型号表

特性	Nios II/f (快速)	Nios II/s (标准)	Nios II/e (经济)
流水线	6 级	5 级	无
乘法器	1 周期	3 周期	软件方式实现
支路预测	动态	静态	无
指令缓冲	可设置	可设置	无
数据缓冲	可设置	无	无
可定制指令	256	256	256
说明	最佳性能优化	体积小, 速度快	占用最少的逻辑资源

11.1 32位NiosII嵌入式核

11.1.1 Nios II结构

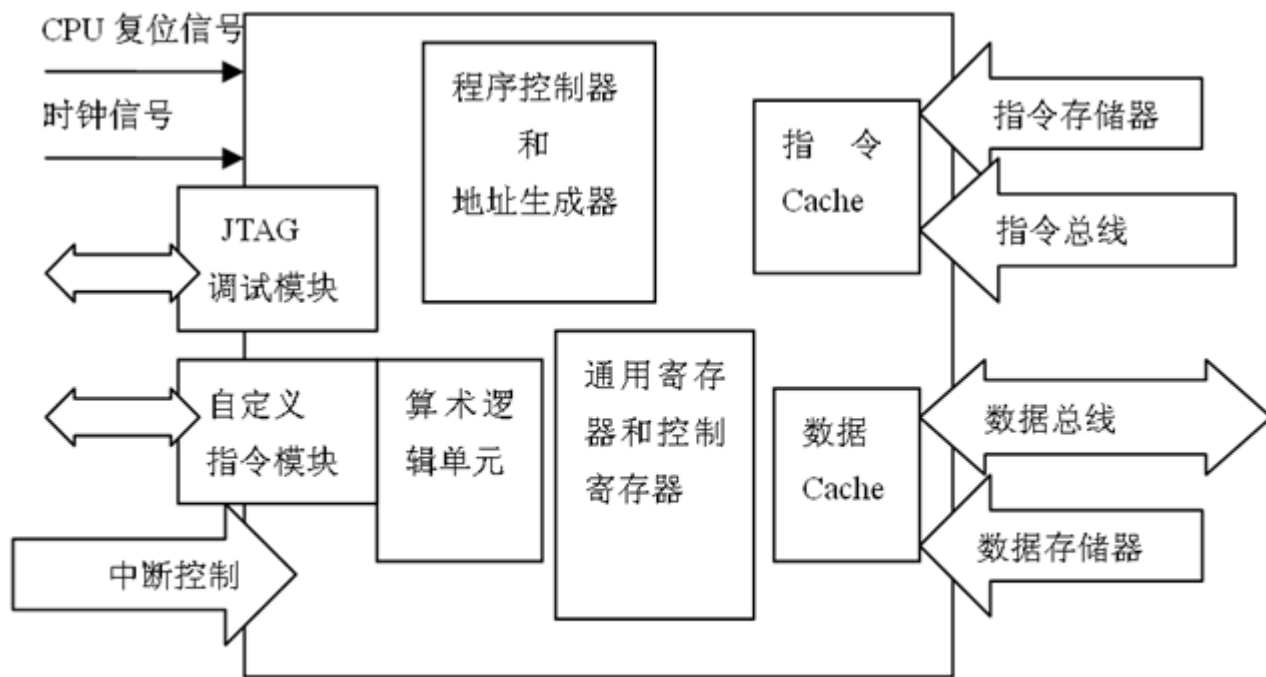


图 11-1 Nios II CPU 内部结构示意图



11.1 32位NiosII嵌入式核

11.1.2 Nios II系统的优势

1. 根据需要可实现不同硬件性能组合配置
2. 良好的性能指标
3. 降低系统成本

11.2 SOPC系统设计流程

11.2.1 Nios II系统设计流程

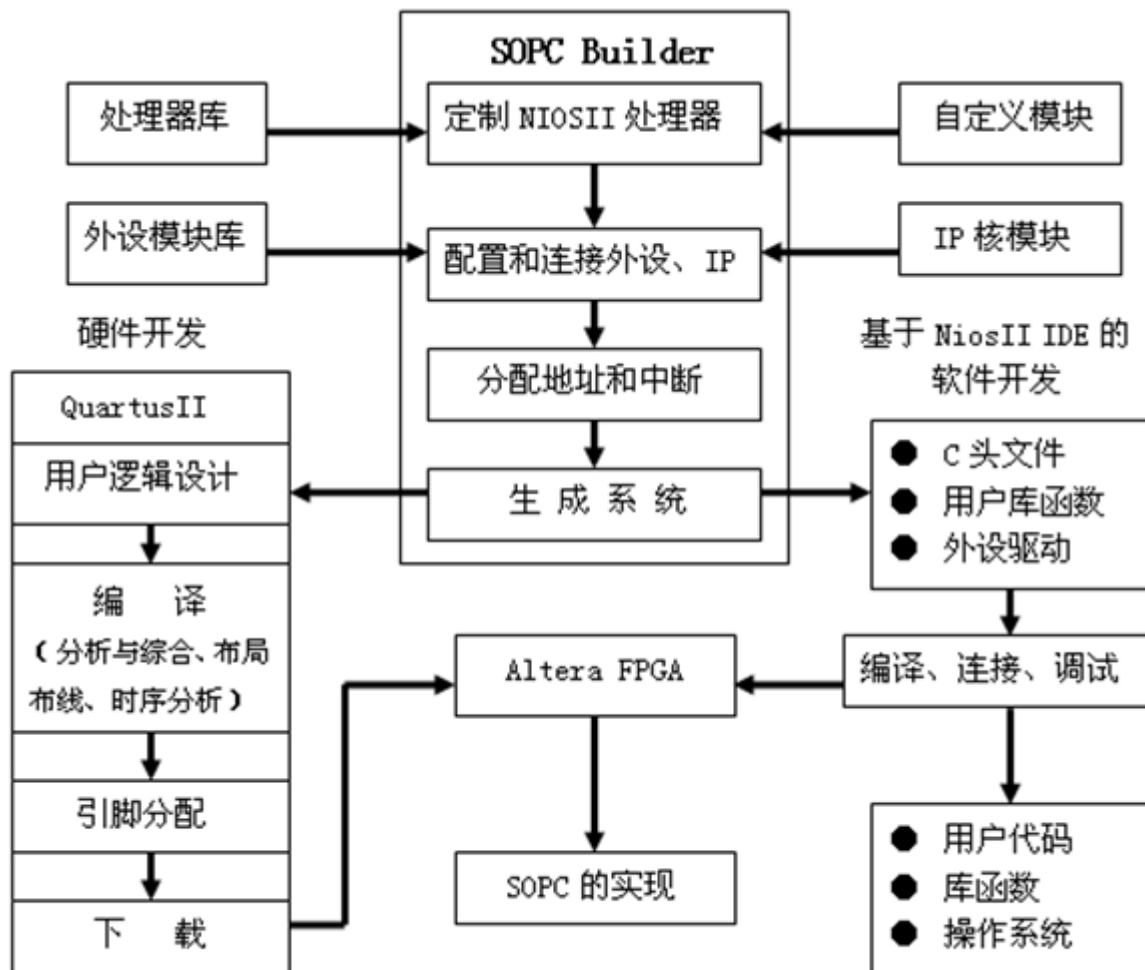


图 11-2 Nios II 软硬件开发流程图

11.2 SOPC系统设计流程

11.2.2 Avalon总线外设

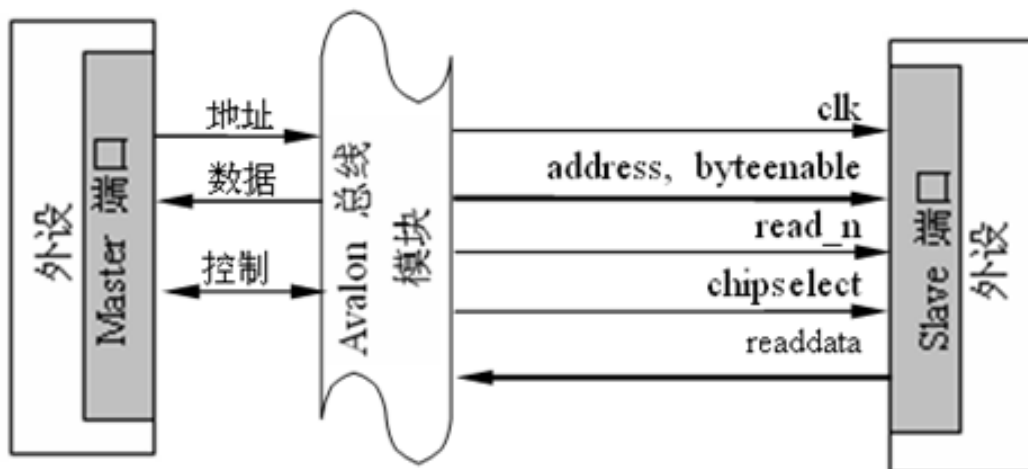


图 11-3 Avalon 从读传输

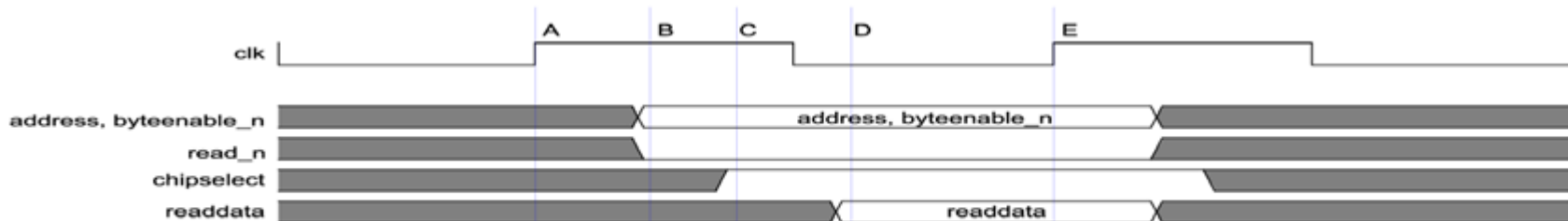


图 11-4 Avalon 从读传输时序

11.2 SOPC系统设计流程

11.2.2 Avalon总线外设

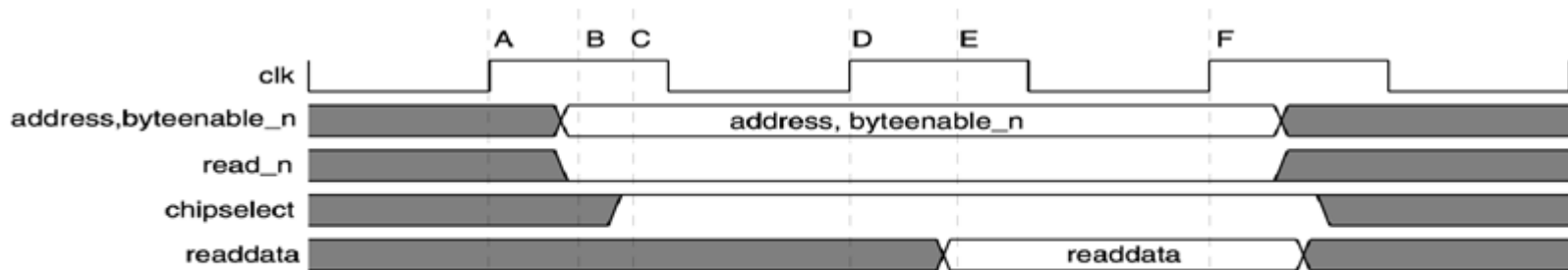


图 11-5 Avalon 带延迟从读传输时序

11.2 SOPC系统设计流程

11.2.2 Avalon总线外设

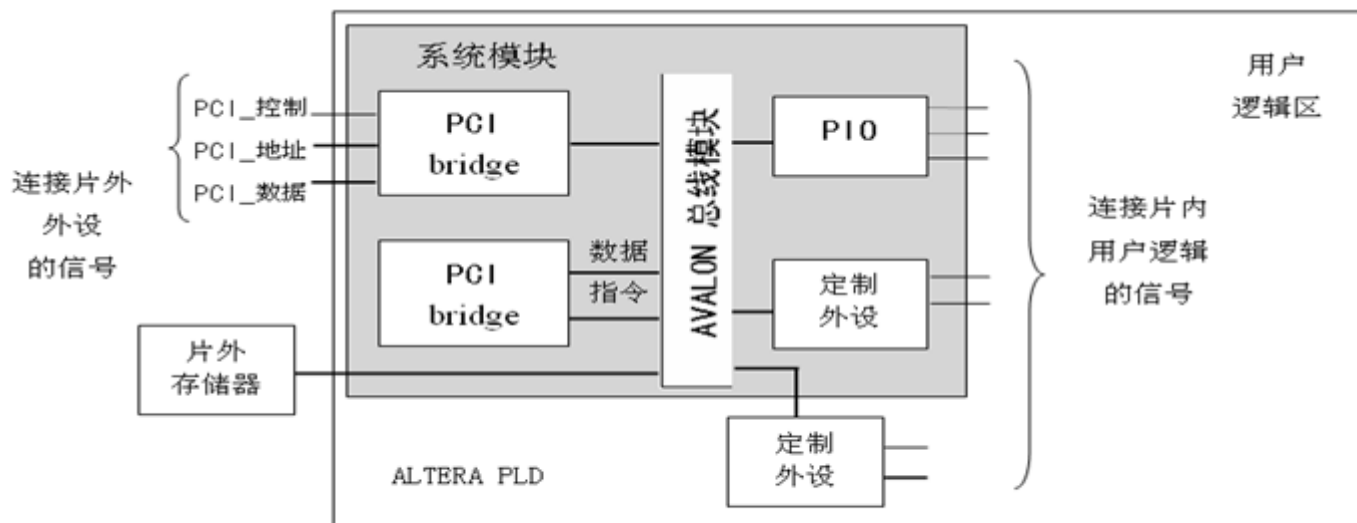


图 11-6 为 NiosII 定制外设示意图

11.2

SOPC系统设计流程

11.2.3 DMA结构

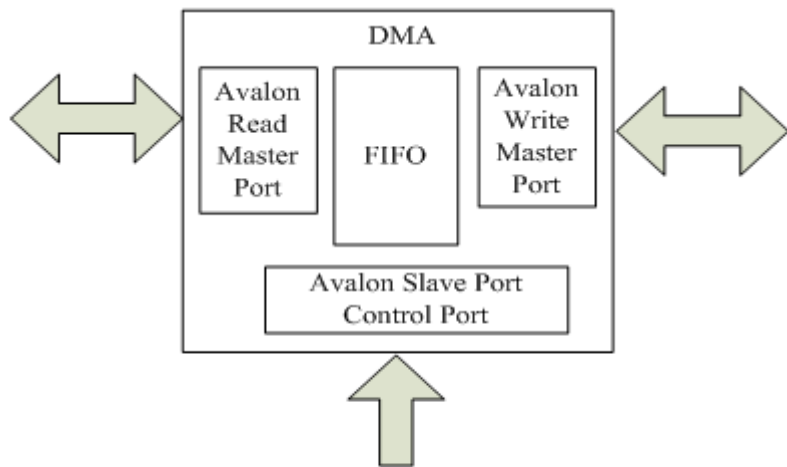


图 11-7 DMA 外设

11.2 SOPC系统设计流程

11.2.4 定制指令

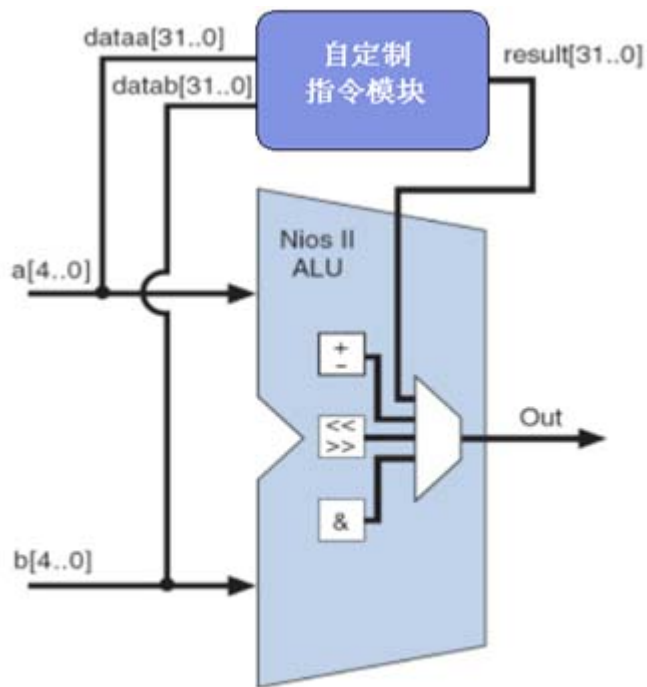


图 11-8 定制指令逻辑模块

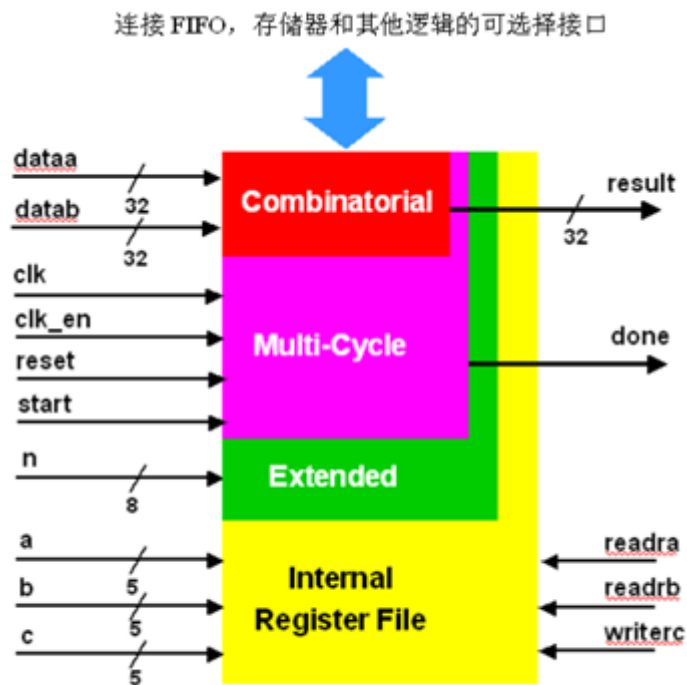


图 11-9 定制指令逻辑模块接口

11.3 SOPC系统设计示例

11.3.1 Nios II 硬件系统设计流程



图 11-10 设定名称



图 11-11 设定芯片型号及时钟

11.3 SOPC系统设计示例

11.3.1 Nios II硬件系统设计流程



图 11-12 选择 cpu 类型

11.3 SOPC系统设计示例

11.3.1 Nios II 硬件系统设计流程

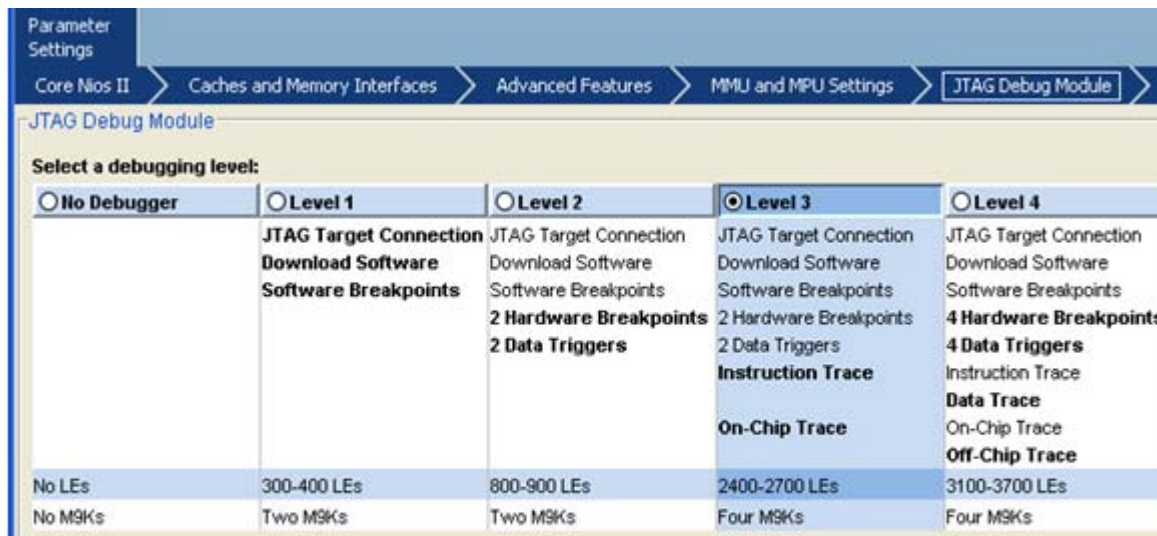


图 11-13 指定 Level 3

11.3 SOPC系统设计示例

11.3.1 Nios II 硬件系统设计流程



图 11-14 组件 JTAG UART 设置

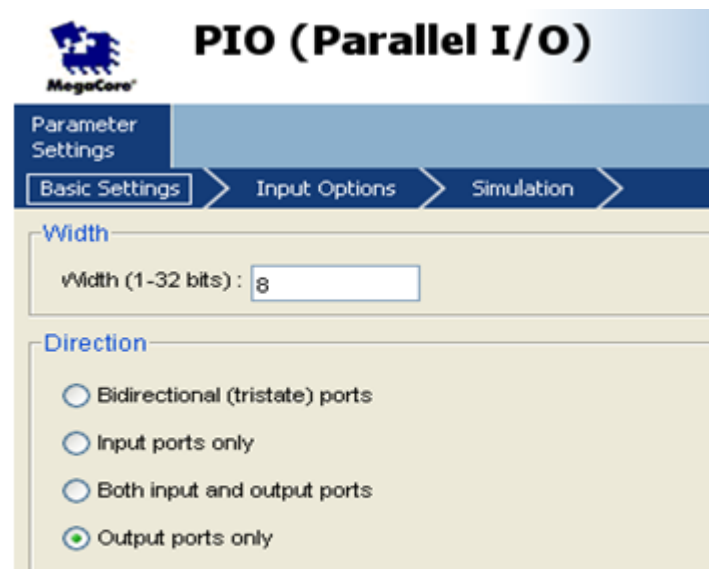


图 11-15 组件 led 输出设置窗

11.3 SOPC系统设计示例

11.3.1 Nios II 硬件系统设计流程

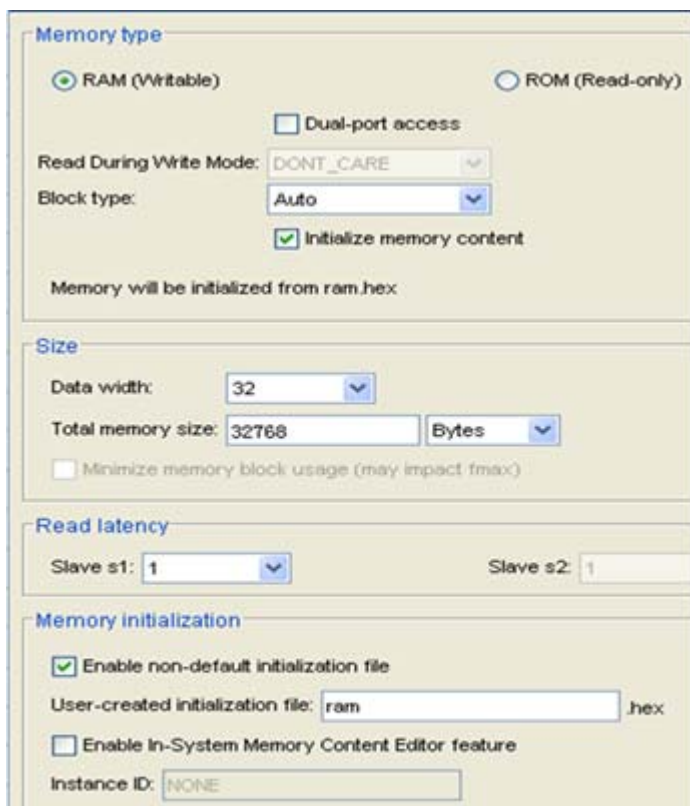


图 11-16 组件 RAM 设置

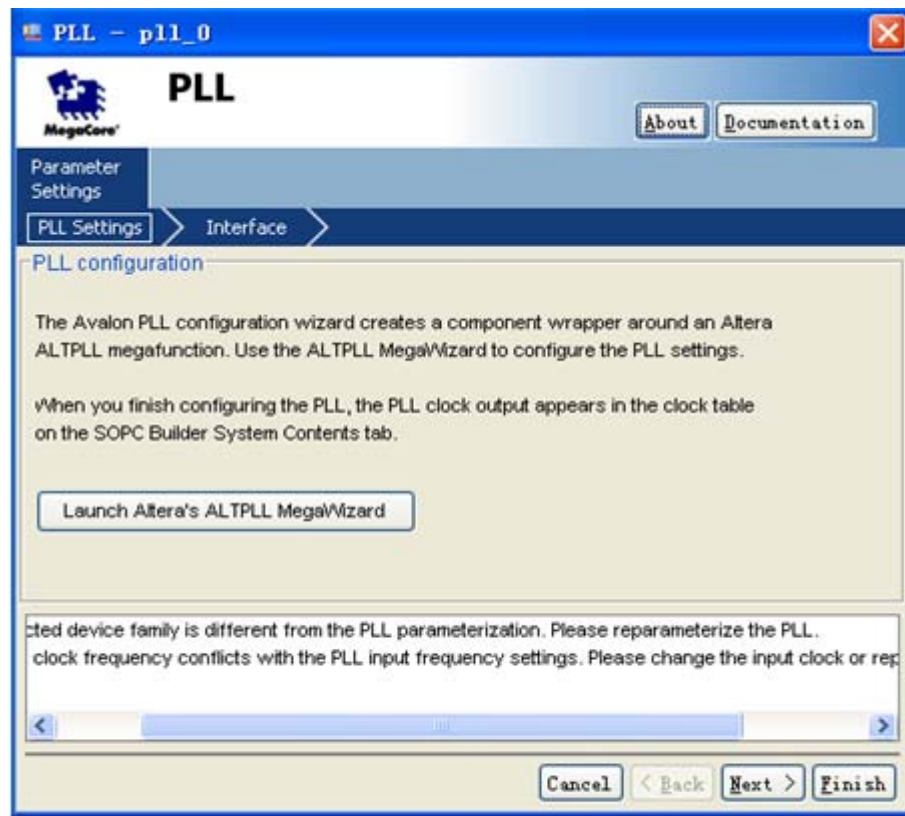


图 11-17 组件 PLL 设置

11.3 SOPC系统设计示例

11.3.1 Nios II 硬件系统设计流程

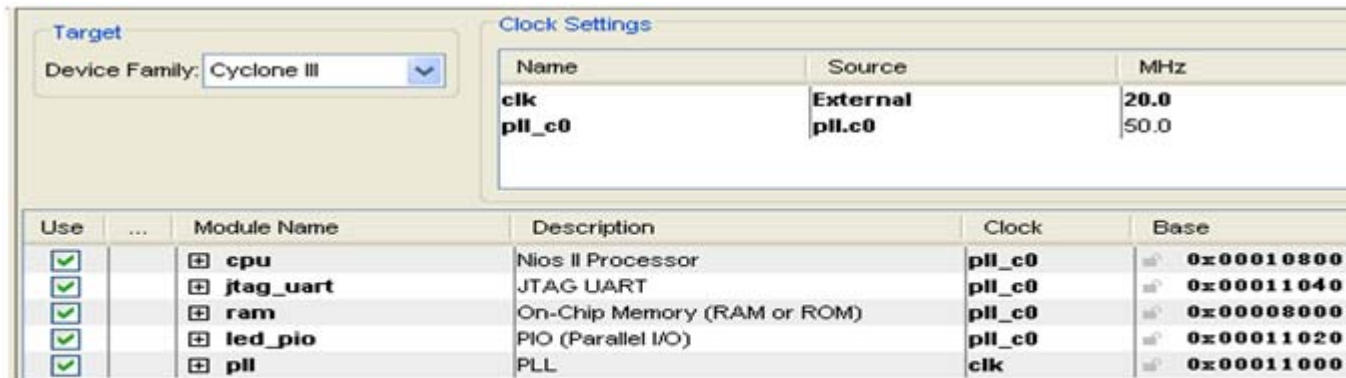


图 11-18 clock settings 窗口



图 11-19 设置系统运行空间

11.3 SOPC系统设计示例

11.3.1 Nios II 硬件系统设计流程

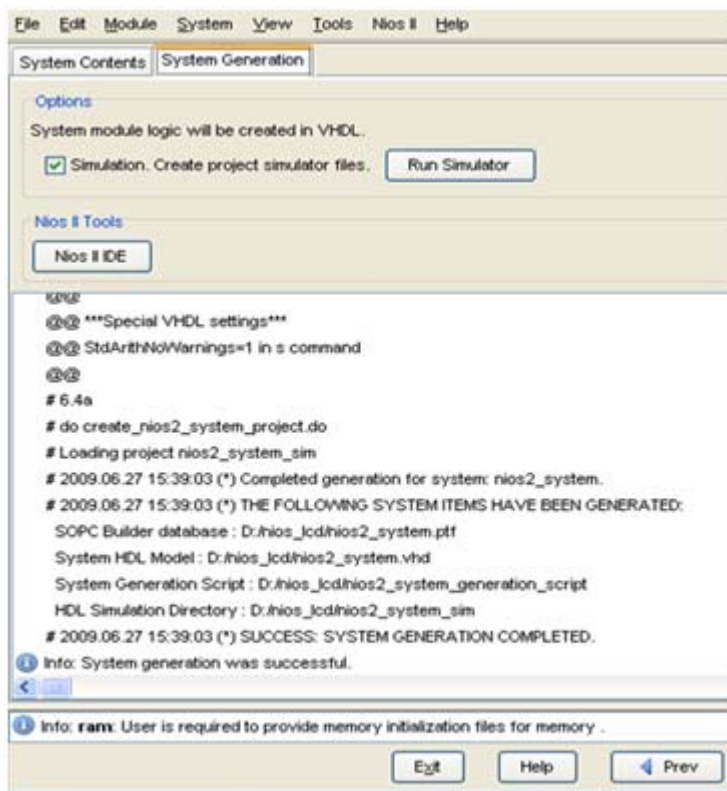


图 11-20 生成 CPU

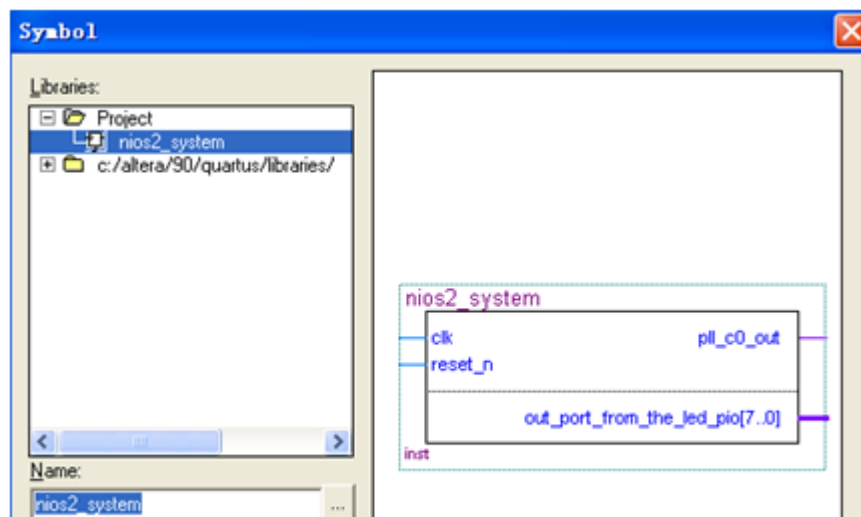


图 11-21 加入系统顶层文件

11.3 SOPC系统设计示例

11.3.1 Nios II 硬件系统设计流程

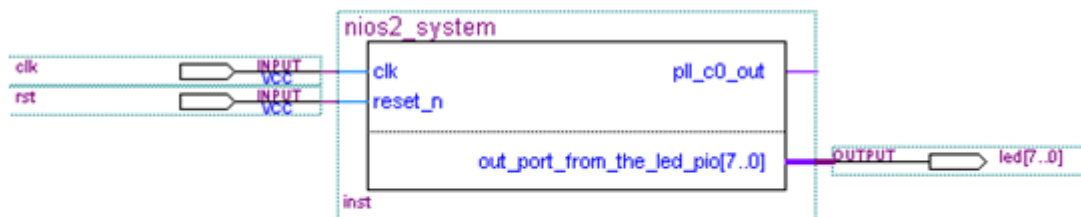


图 11-22 加上引脚的的原理图模块

	To	Location	Enabled
1	clk	PIN_22	Yes
2	led[0]	PIN_11	Yes
3	led[1]	PIN_10	Yes
4	led[2]	PIN_7	Yes
5	led[3]	PIN_4	Yes
6	led[4]	PIN_3	Yes
7	led[5]	PIN_2	Yes
8	led[6]	PIN_1	Yes
9	led[7]	PIN_144	Yes
10	rst	PIN_69	Yes

图 11-23 对引脚进行锁定

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程

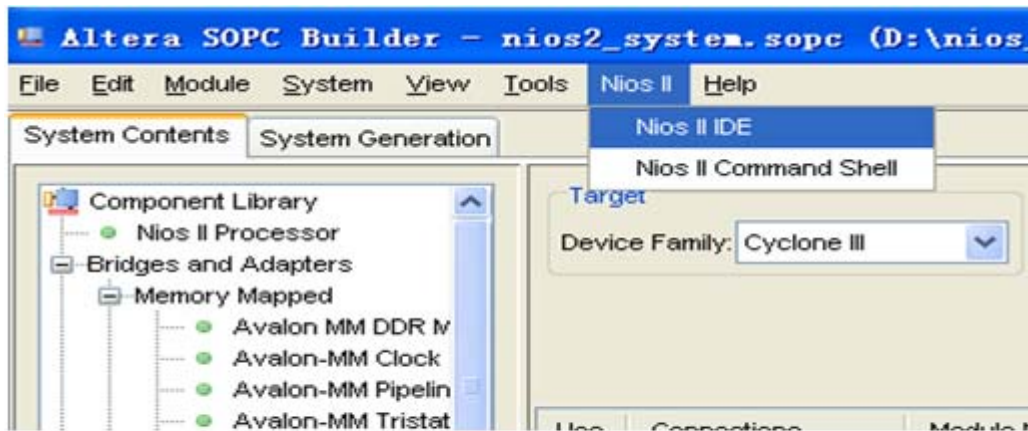


图 11-24 点击 Run Nios II IDE，进入软件开发环境

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程



图 11-25 选择进入 IDE 软件设计/调试平台

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程

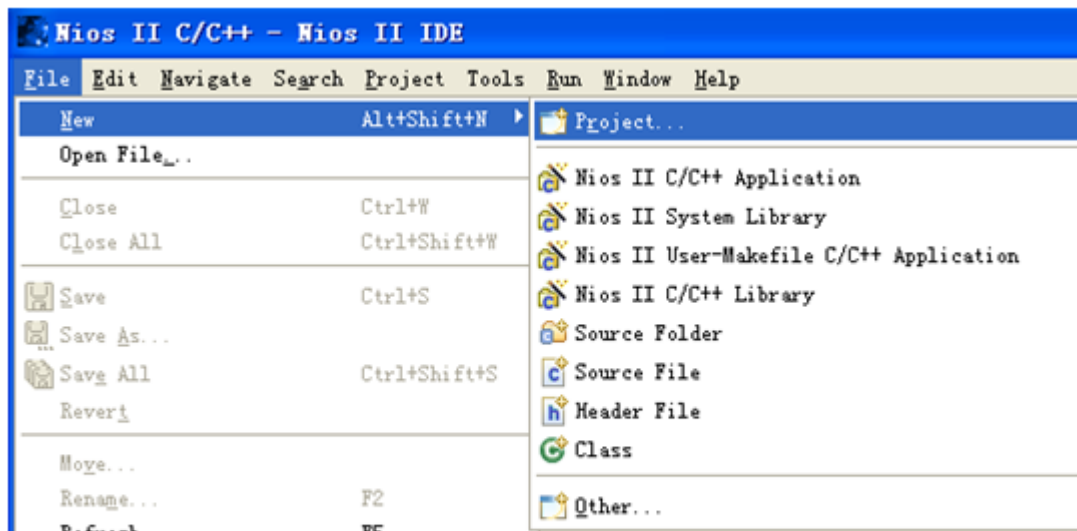


图 11-26 建立一个软件实例工程

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程

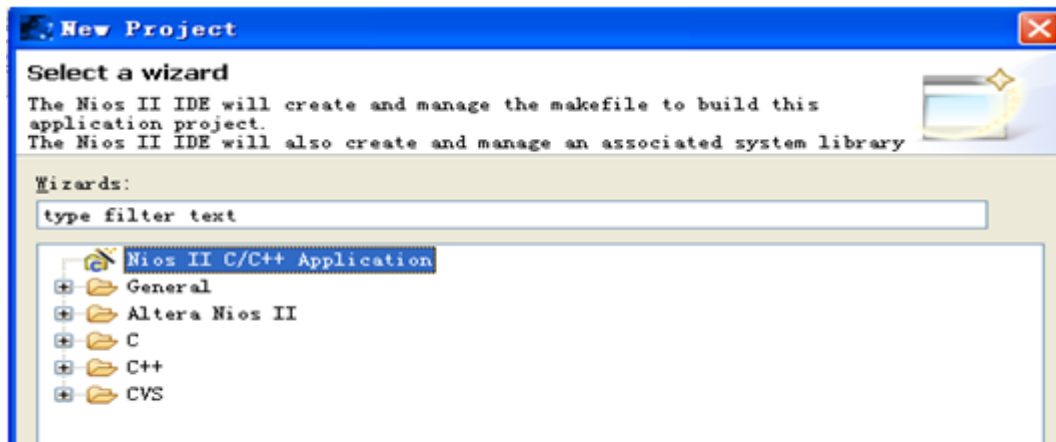


图 11-27 选择 C/C++应用

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程

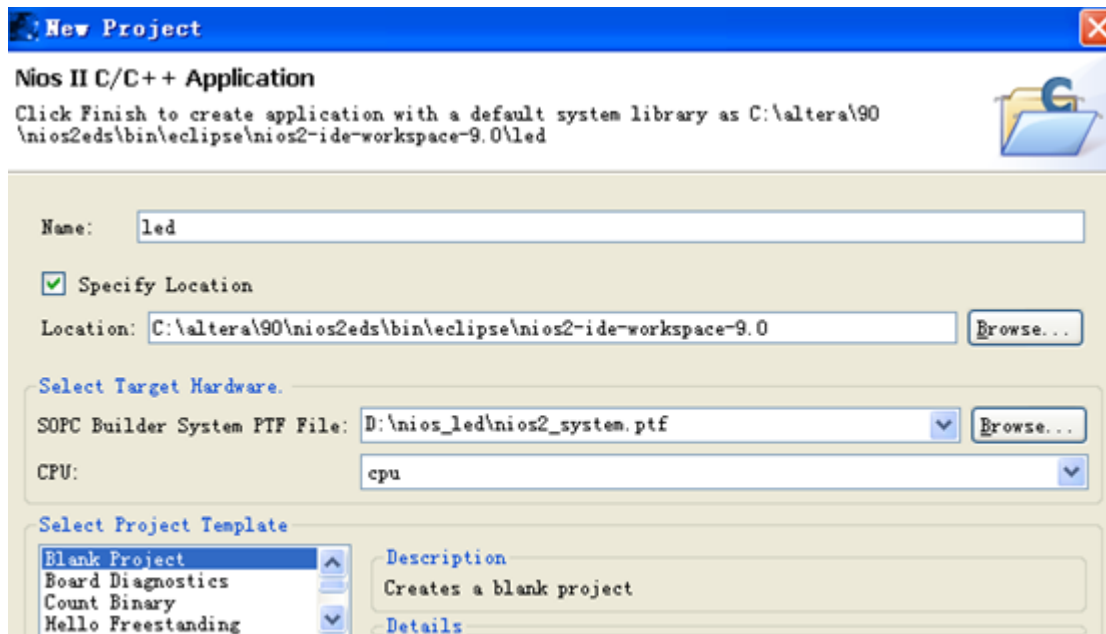
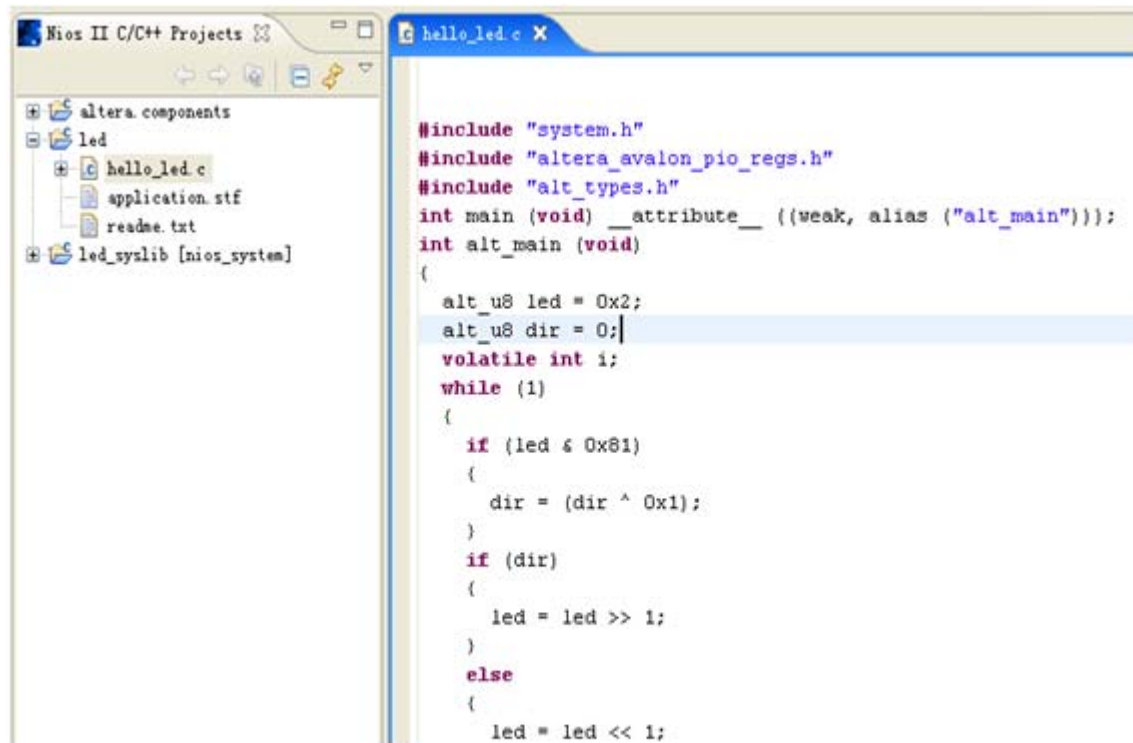


图 11-28 在示例库中选择一个 C 程序实例

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程



```
#include "system.h"
#include "altera_avalon_pio_regs.h"
#include "alt_types.h"
int main (void) __attribute__ ((weak, alias ("alt_main")));
int alt_main (void)
{
    alt_u8 led = 0x2;
    alt_u8 dir = 0;
    volatile int i;
    while (1)
    {
        if (led & 0x01)
        {
            dir = (dir ^ 0x1);
        }
        if (dir)
        {
            led = led >> 1;
        }
        else
        {
            led = led << 1;
        }
    }
}
```

图 11-29 进入 Nios II IDE 窗口

【例 11-1】led.c

```
#include "system.h"
#include "altera_avalon_pio_regs.h"
#include "alt_types.h"
int main (void) __attribute__ ((weak, alias ("alt_main")));
int alt_main (void)
{
    alt_u8 led = 0x2;
    alt_u8 dir = 0;
    volatile int i;
    while (1)
    {
        if (led & 0x81)
        {
            dir = (dir ^ 0x1);
        }
        if (dir)
        {
            led = led >> 1;
        }
        else
        {
            led = led << 1;
        }
        IOWR_ALTERA_AVALON_PIO_DATA(LED_PIO_BASE, led);
        i = 0;
        while (i<400000)
            i++;
    }
    return 0; }

```

计示例

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程

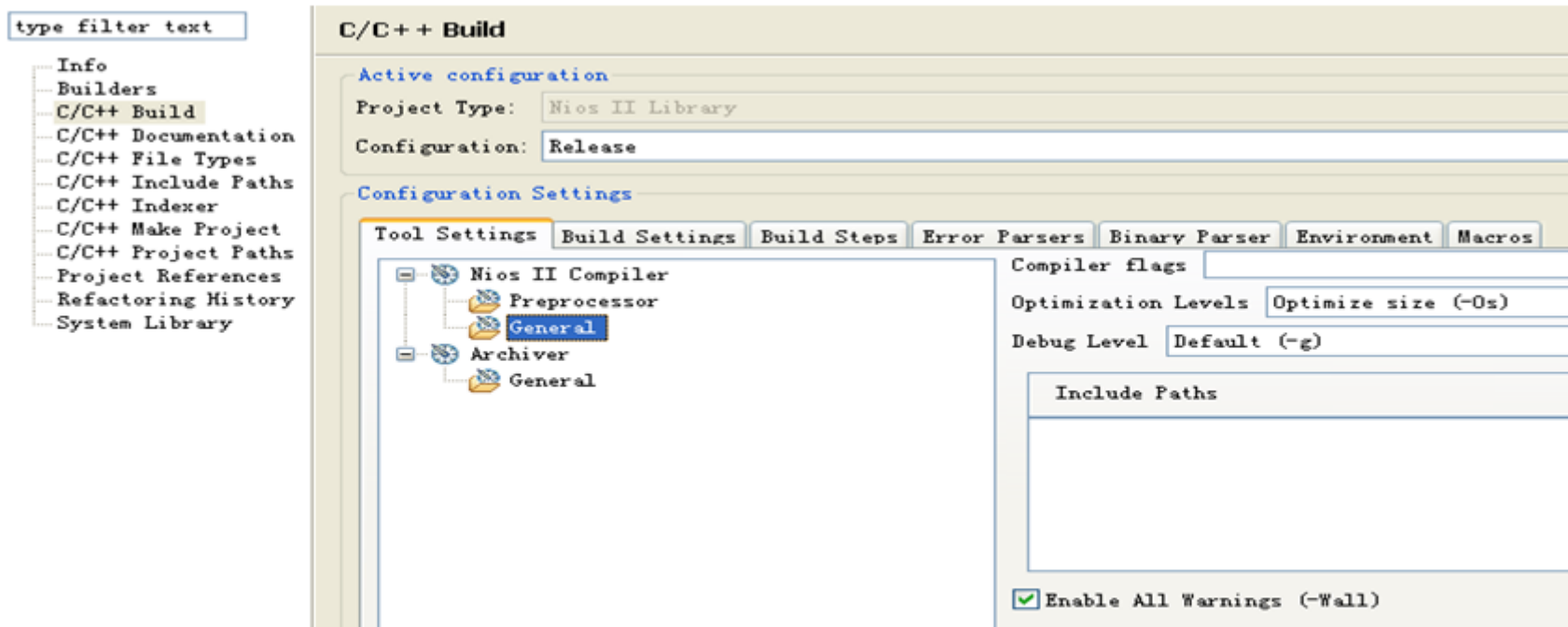


图 11-30 设置库文件优化级别

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程

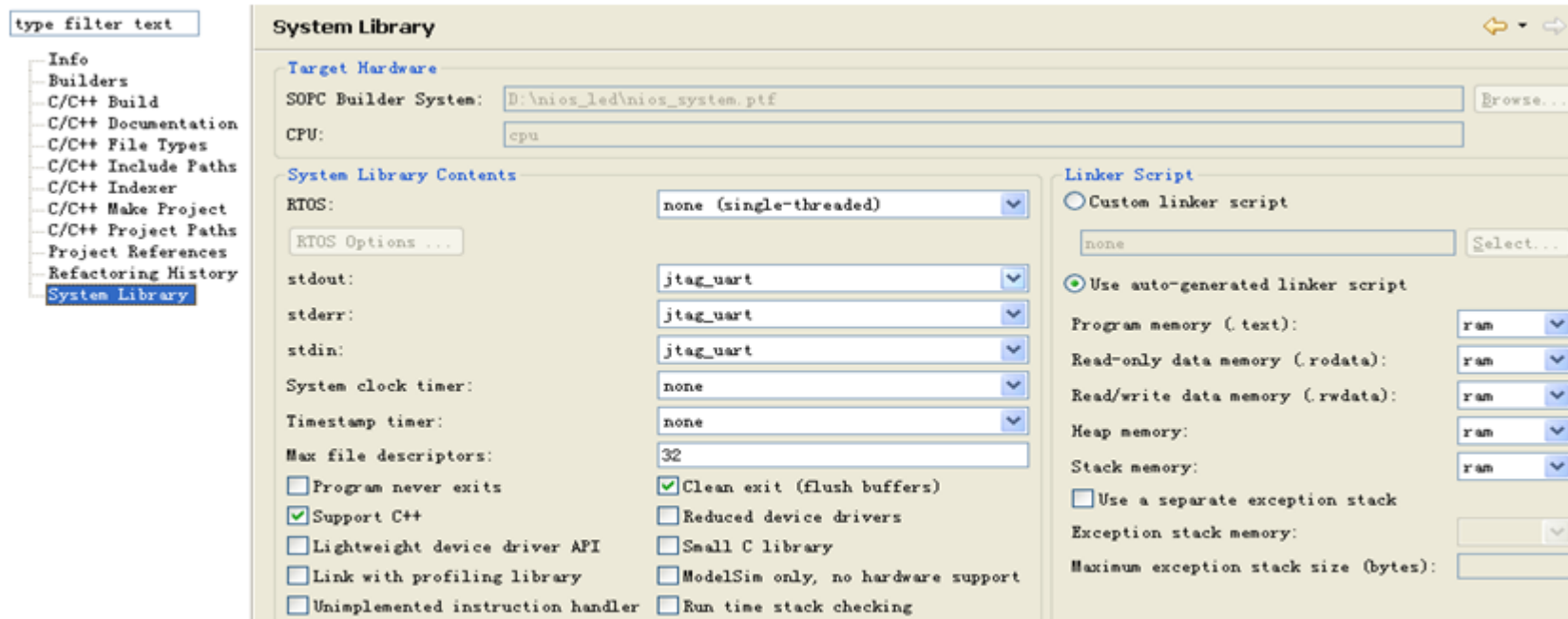


图 11-31 优化设置库

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程

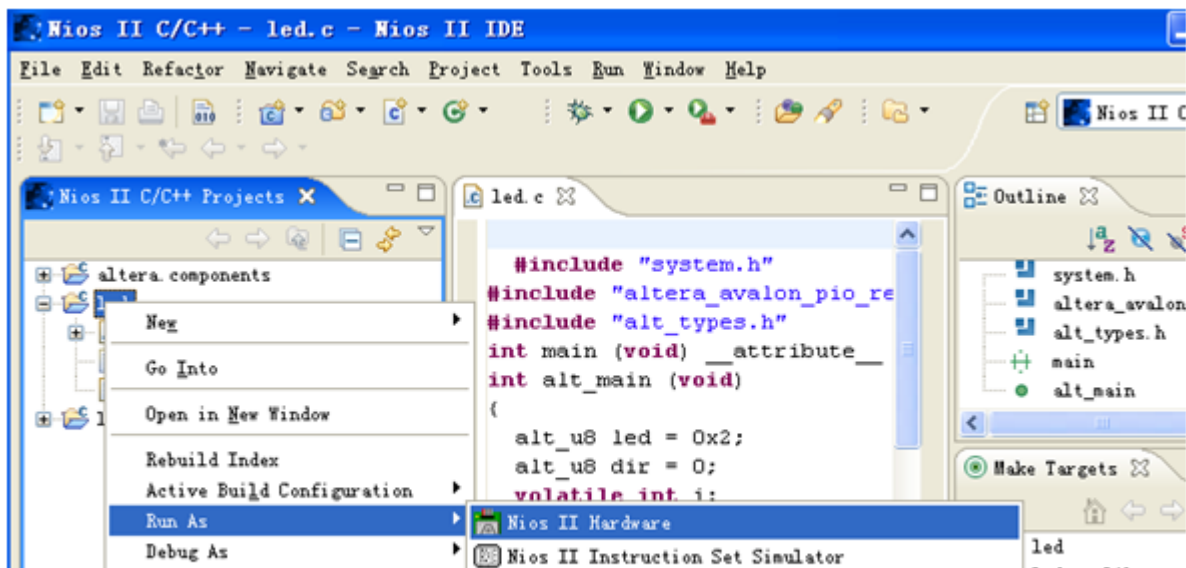
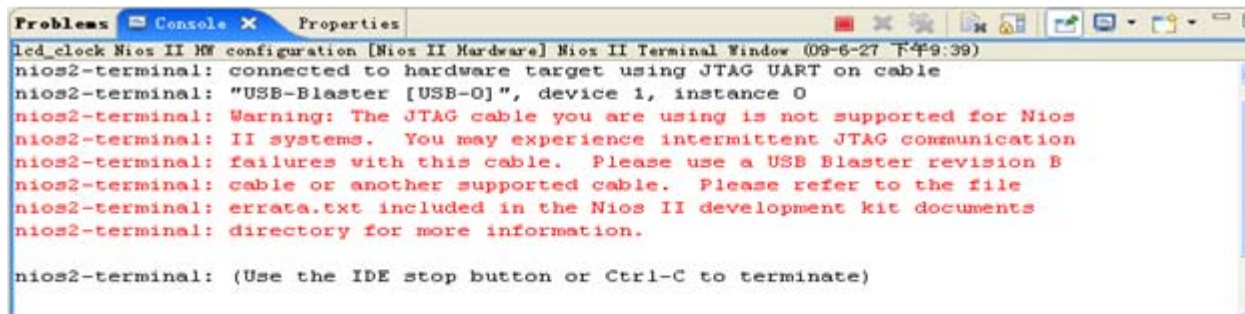


图 11-32 编译、下载并在 Nios II CPU 中运行

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程



```
lcd_clock Nios II HW configuration [Nios II Hardware] Nios II Terminal Window (09-6-27 下午9:39)
nios2-terminal: connected to hardware target using JTAG UART on cable
nios2-terminal: "USB-Blaster [USB-0]", device 1, instance 0
nios2-terminal: Warning: The JTAG cable you are using is not supported for Nios
nios2-terminal: II systems. You may experience intermittent JTAG communication
nios2-terminal: failures with this cable. Please use a USB Blaster revision B
nios2-terminal: cable or another supported cable. Please refer to the file
nios2-terminal: errata.txt included in the Nios II development kit documents
nios2-terminal: directory for more information.

nios2-terminal: (Use the IDE stop button or Ctrl-C to terminate)
```

图 11-33 C 程序下载成功，启动运行

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程

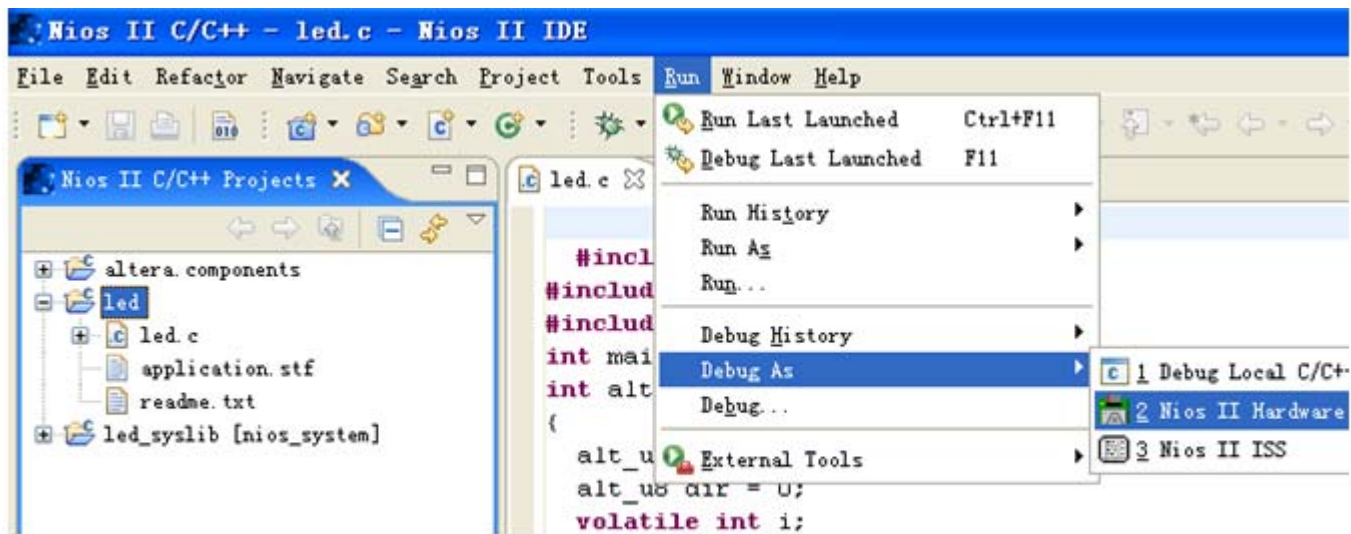


图 11-34 选择单步/跟踪调试模式运 C 程序

11.3 SOPC系统设计示例

11.3.2 Nios II 软件设计流程

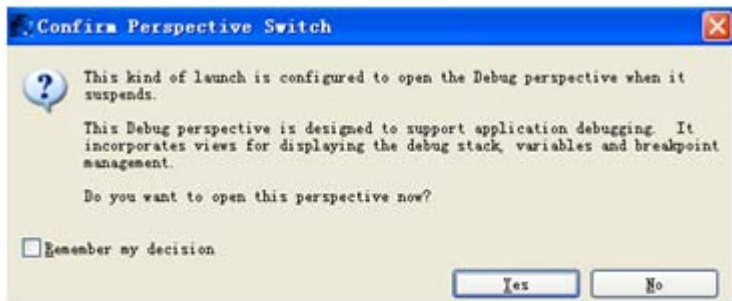


图 11-35 选择单步/跟踪调试模式运 C 程序

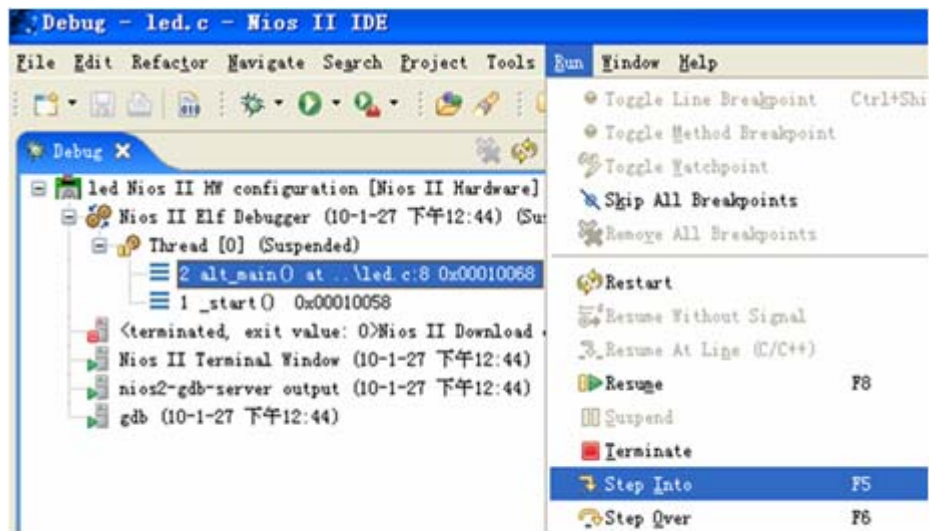


图 11-36 单步/跟踪调试窗

11.4 SOPC系统接口设计

11.4.1 用户自定义组件设计

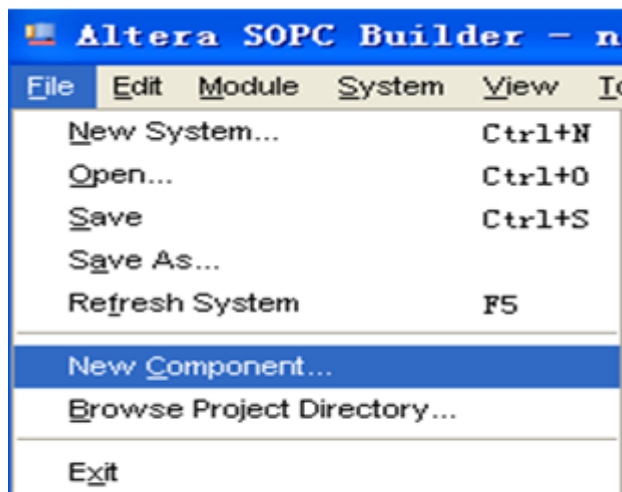
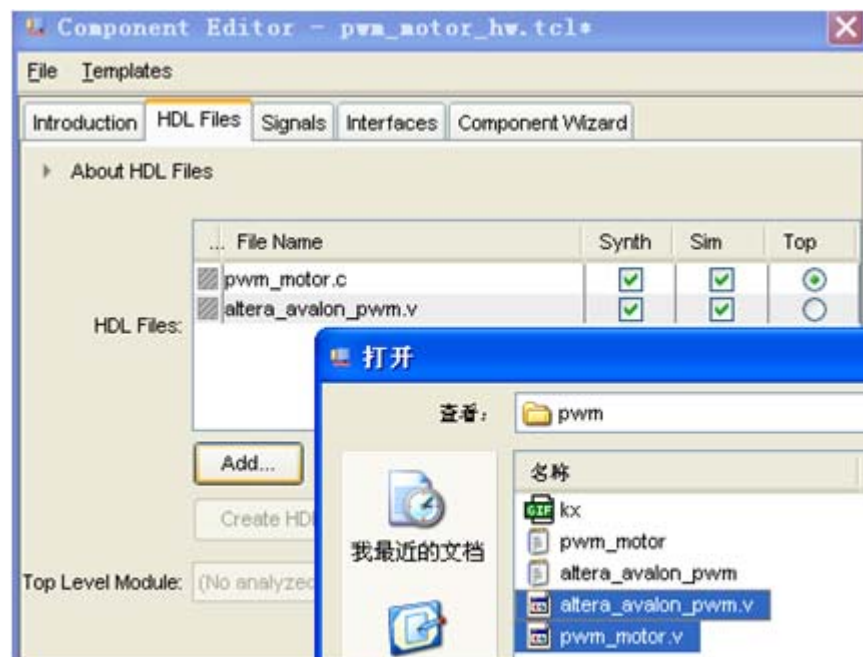


图 11-37 选择自定义菜单图



11-38 将逻辑模块加入进 Nios II 中

11.4 SOPC系统接口设计

11.4.1 用户自定义组件设计

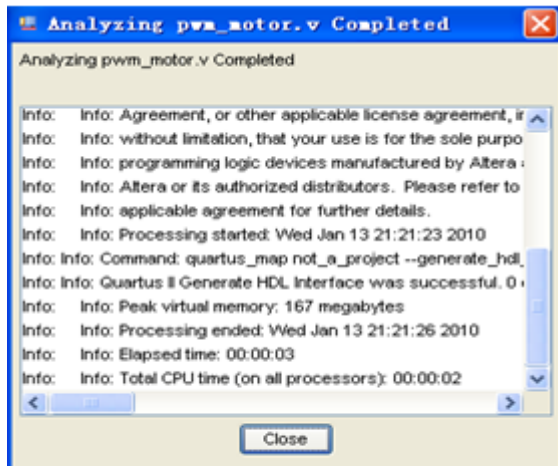


图 11-39 分析窗口

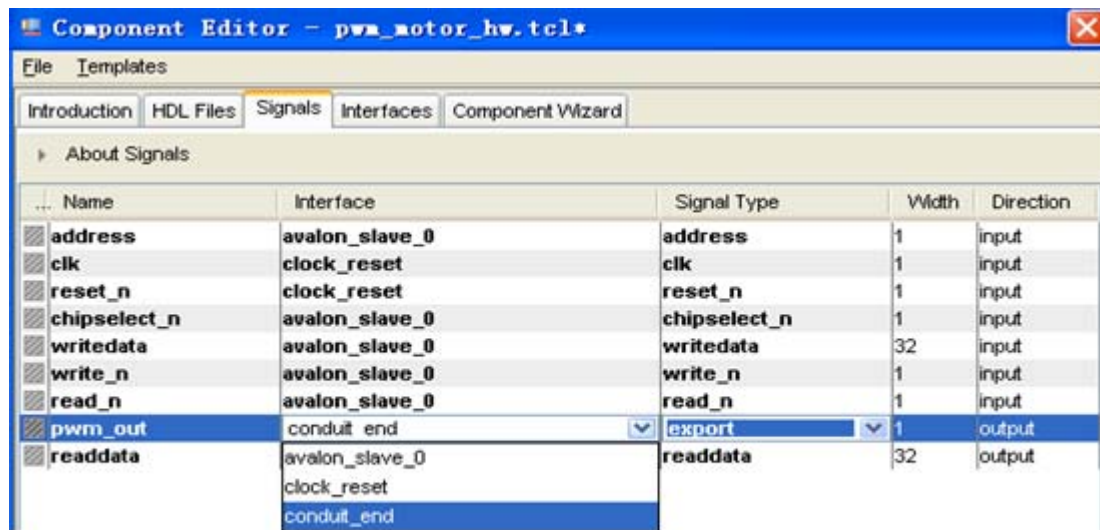


图 11-40 定义程序的端口信号

11.4 SOPC系统接口设计

11.4.1 用户自定义组件设计

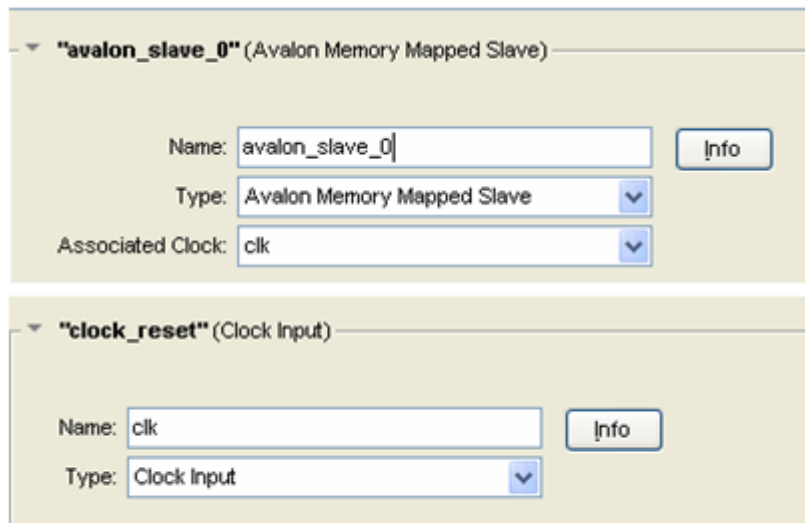


图 11-41 修改端口信号

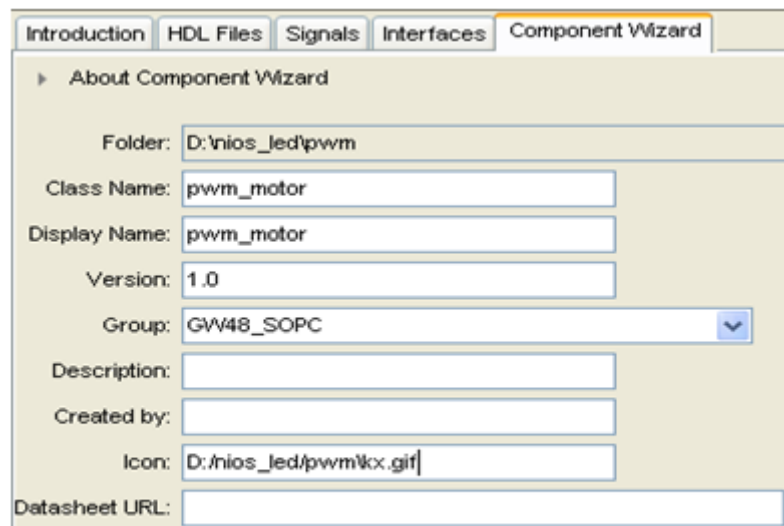


图 11-42 指定元件的名称和目录



11.4 SOPC系统接口设计

11.4.2 用户自定义指令设计

【例 11-2】mult.V

```
module mult (dataa,datab,result,reset,start,clk_en,clk);  
    input[31:0] dataa, datab;    output[31:0] result;  
    input reset, start, clk_en, clk;  
    wire[31:0] result;    wire[31:0] rt;  
    assign rt=dataa[15:0] * datab[15:0];  
    assign result=rt;  
endmodule
```

11.4 SOPC系统接口设计

11.4.2 用户自定义指令设计

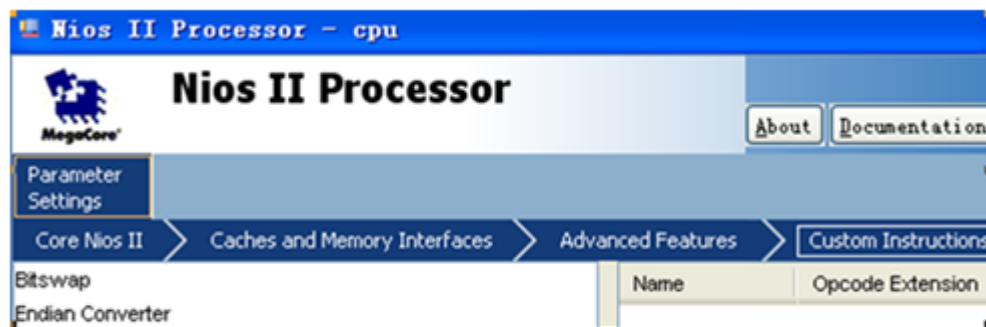


图 11-43 打开 CPU 的自定义指令对话框

11.4 SOPC系统接口设计

11.4.2 用户自定义指令设计

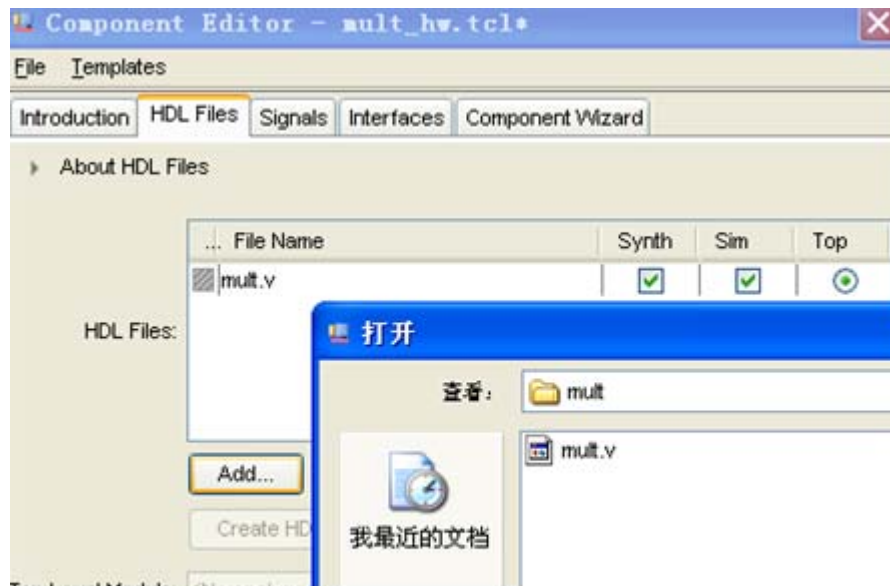


图 11-44 加入已设计好的乘法指令 Verilog 程序

11.4 SOPC系统接口设计

11.4.2 用户自定义指令设计

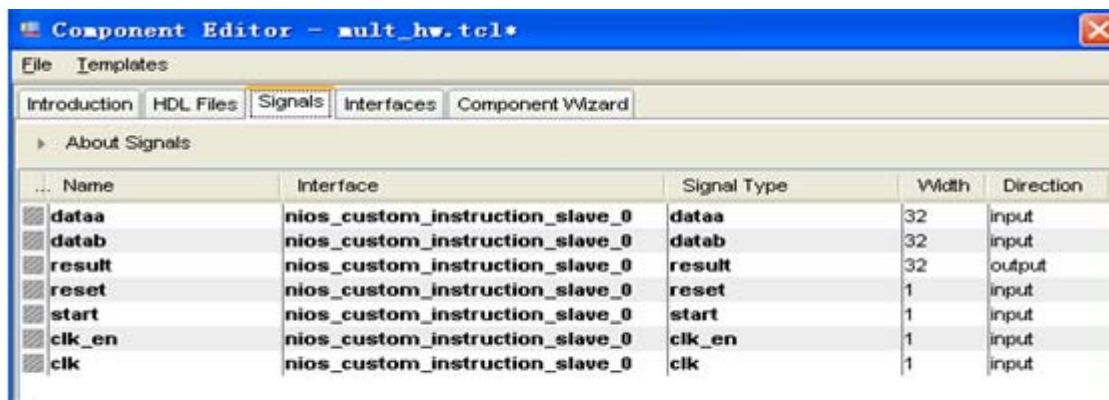


图 11-45 定义程序的端口信号

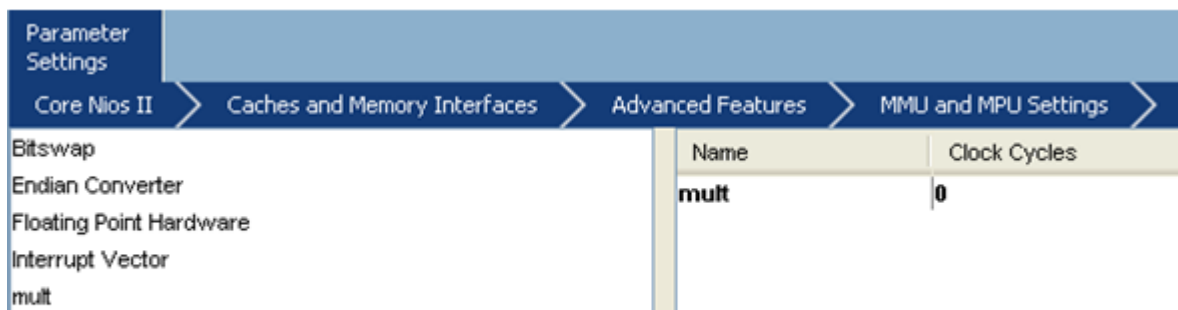


图 11-46 已加入一条名为 mult 的乘法指令

11.4 SOPC系统接口设计

11.4.3 IDE Flash编程下载



图 11-47 EPCS Serial Flash Controller 组件

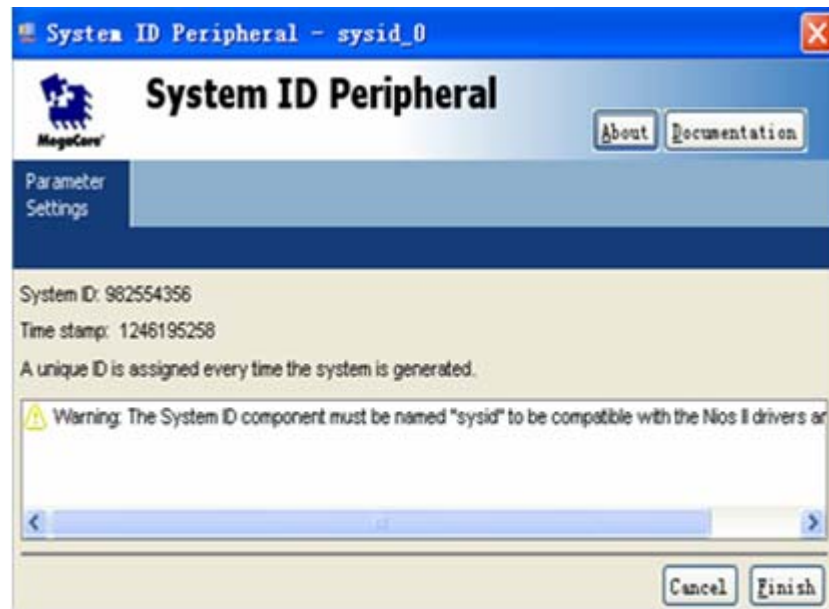


图 11-48 加入系统 ID 组件

11.4 SOPC系统接口设计

11.4.3 IDE Flash编程下载

Reset Vector:	Memory:	epcs_flash_controller	▼	Offset:	0x0	0x00000000
Exception Vector:	Memory:	ram	▼	Offset:	0x20	0x00020020

图 11-49 设置系统运行空间

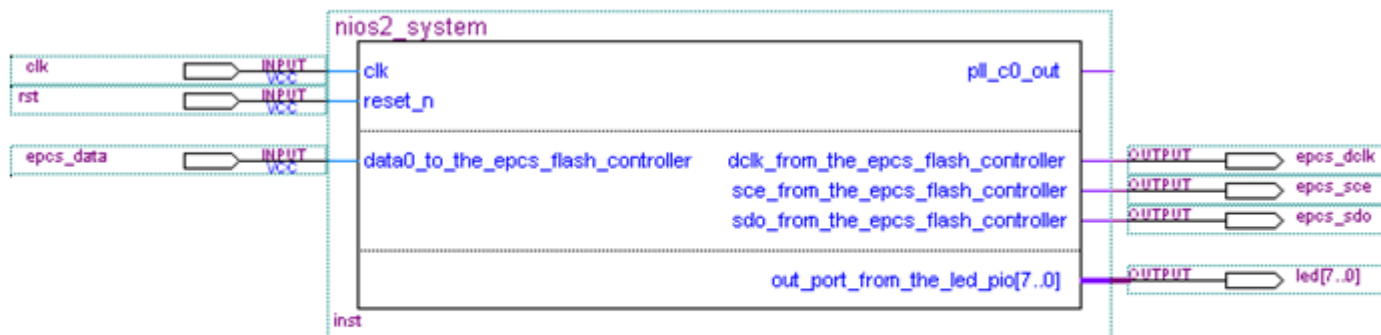


图 11-50 加上 EPCS 的的原理图

11.4 SOPC系统接口设计

11.4.3 IDE Flash编程下载

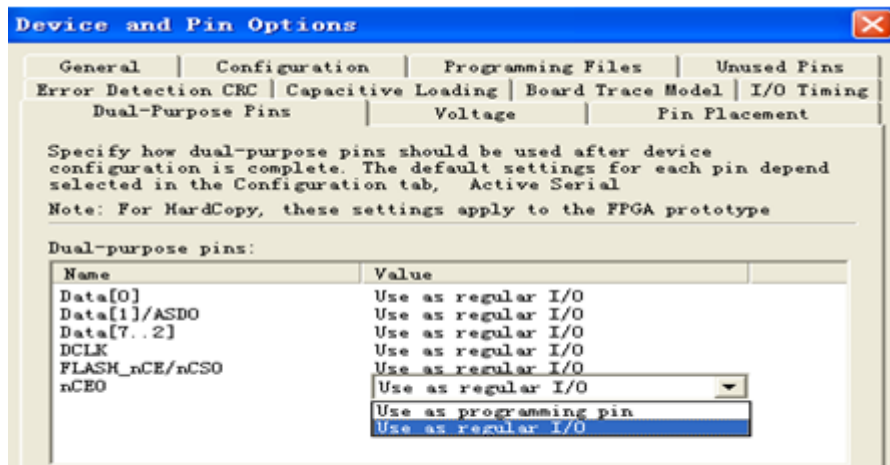


图 11-51 更改为普通 I/O 口

clk	PIN_22
epcs_data	PIN_13
epcs_dclk	PIN_12
epcs_sce	PIN_8
epcs_sdo	PIN_6
led[0]	PIN_144
led[1]	PIN_1
led[2]	PIN_2
led[3]	PIN_3
led[4]	PIN_4
led[5]	PIN_7
led[6]	PIN_10
led[7]	PIN_11
rst	PIN_69

图 11-52 引脚锁定

11.4 SOPC系统接口设计

11.4.3 IDE Flash编程下载



图 11-53 打开 Flash 编程对话框

11.4 SOPC系统接口设计

11.4.3 IDE Flash编程下载

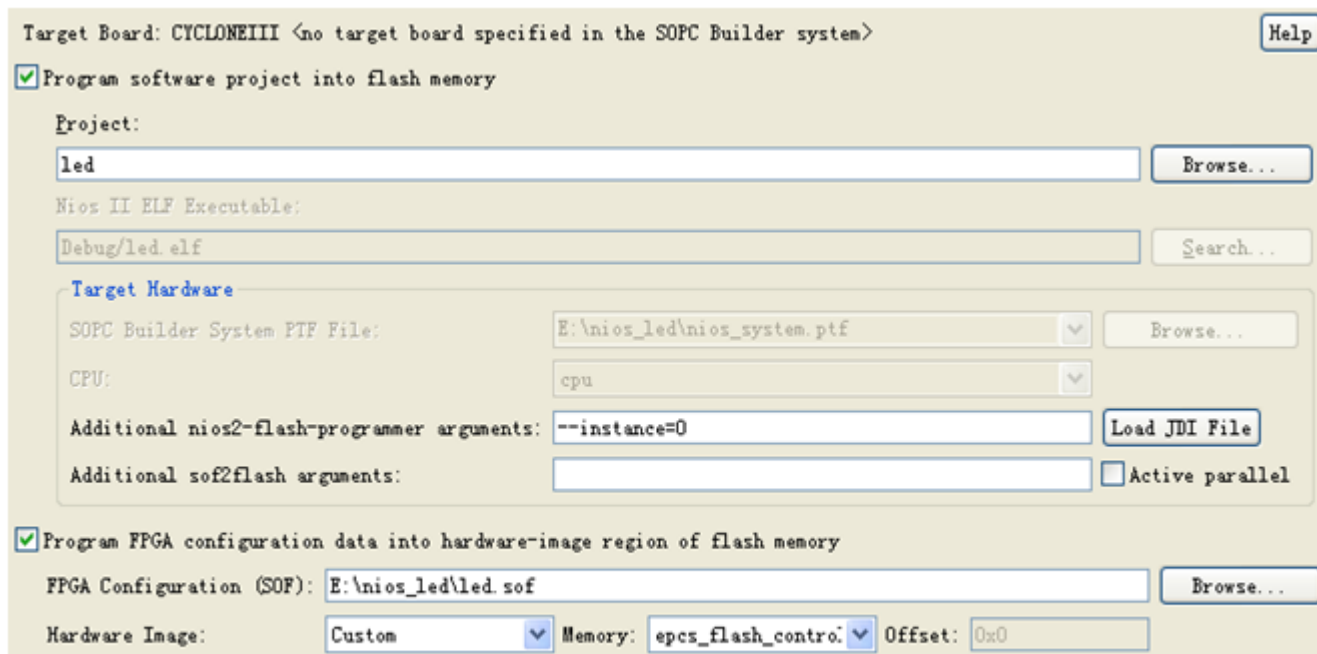


图 12-54 在 Flash 编程对话框中完成必要设置

11.5 SOPC系统综合设计

11.5.1 综合硬件设计

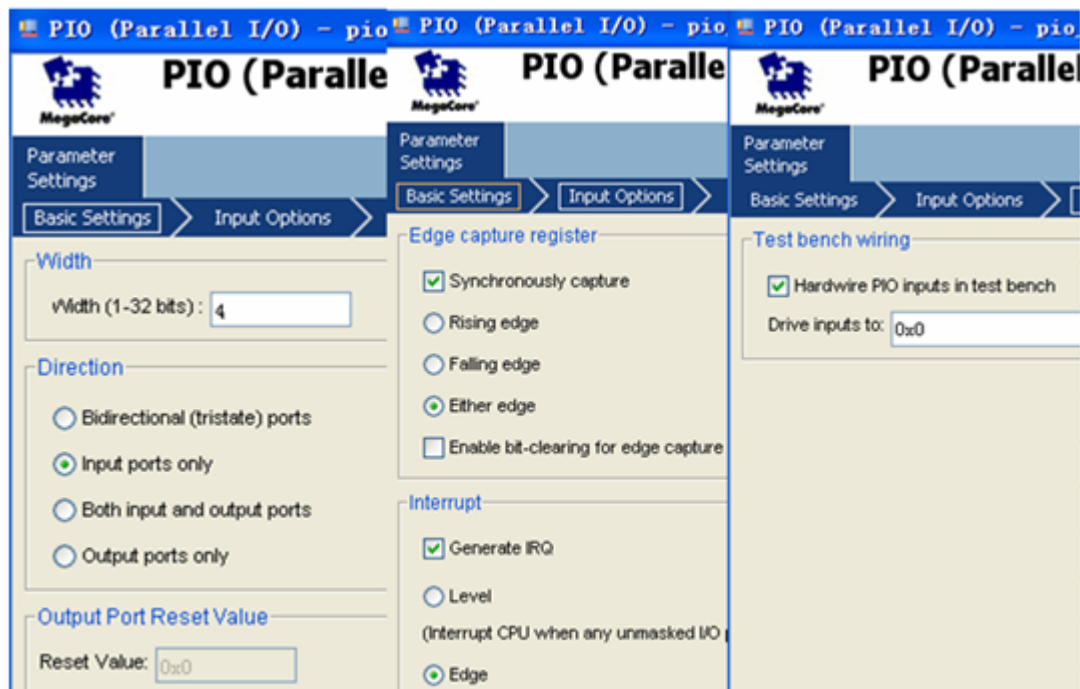


图 11-55 组件 PIO 输入口设置窗

11.5 SOPC系统综合设计

11.5.1 综合硬件设计

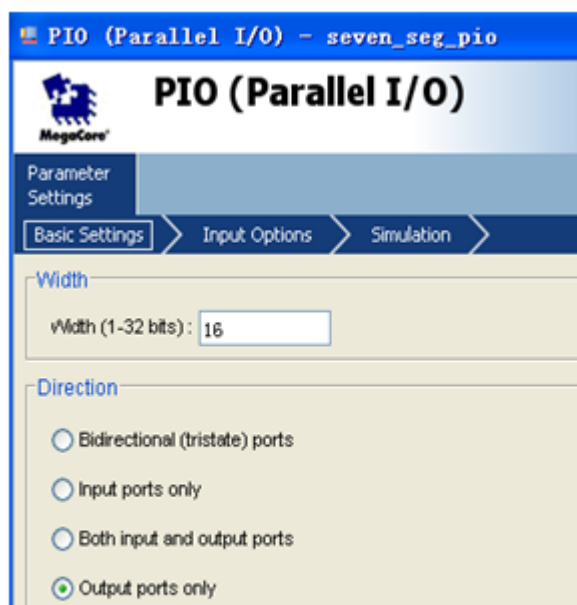


图 12-56 组件 PIO 输入口设置窗

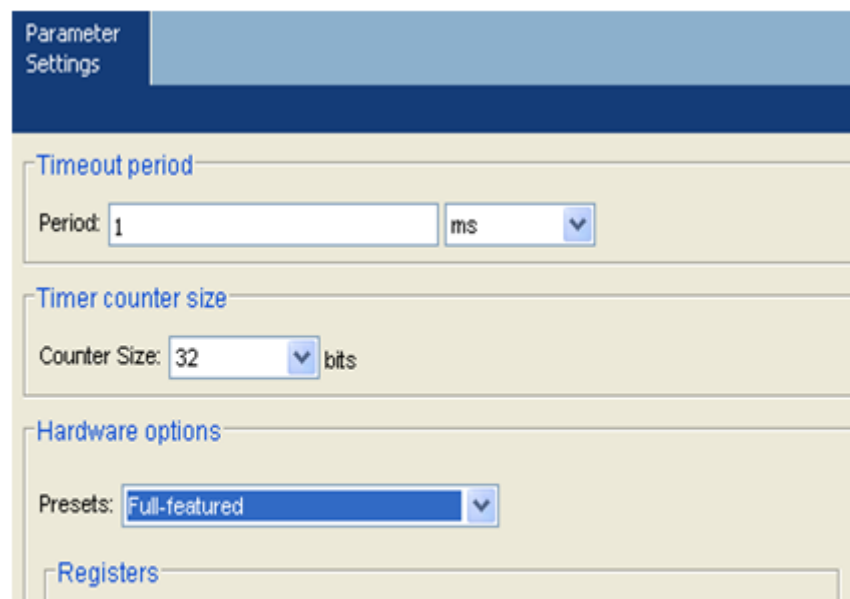


图 12-57 组件 Timer 设置窗

11.5 SOPC系统综合设计

11.5.1 综合硬件设计

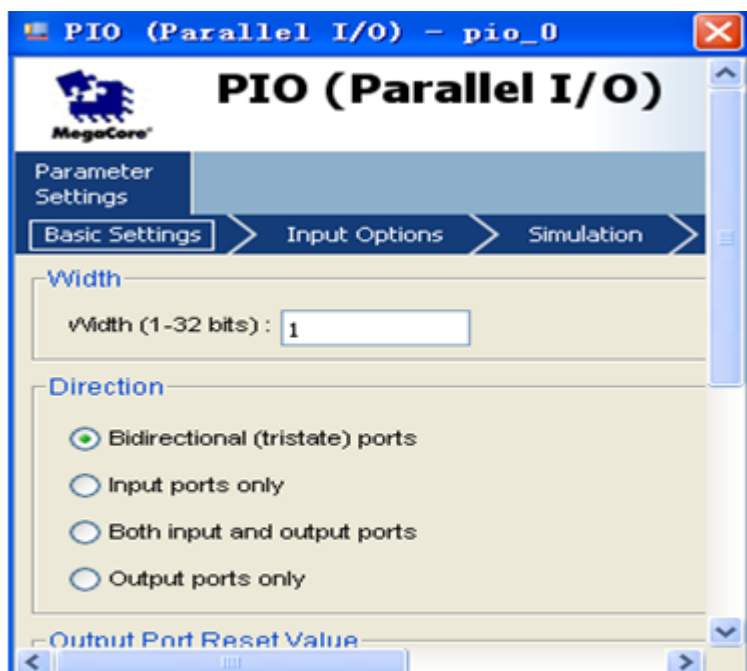


图 11-58 并行输入/输出设置窗

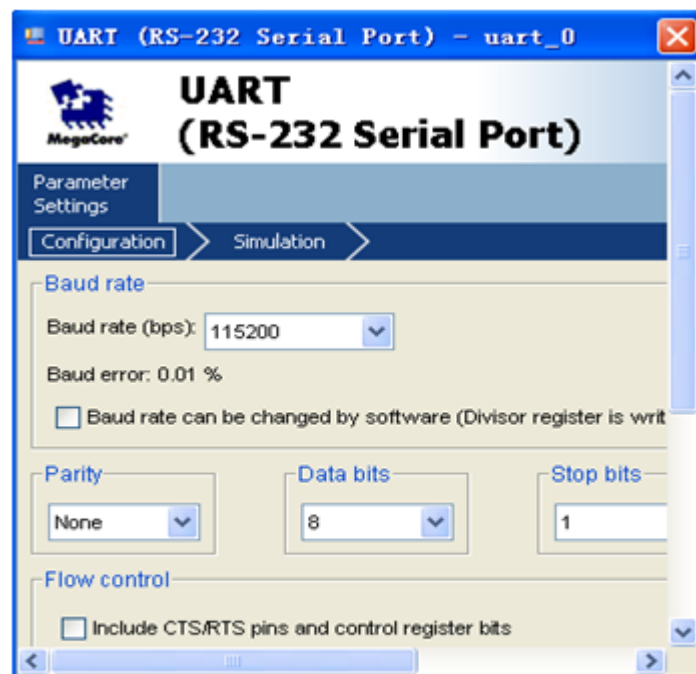


图 11-59 串行 UART 设置窗

11.5 SOPC系统综合设计

11.5.1 综合硬件设计

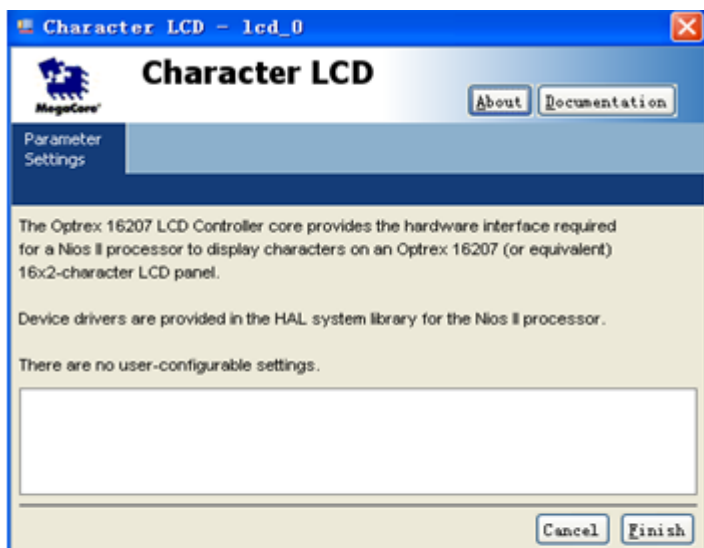


图 11-60 字符液晶组件设置

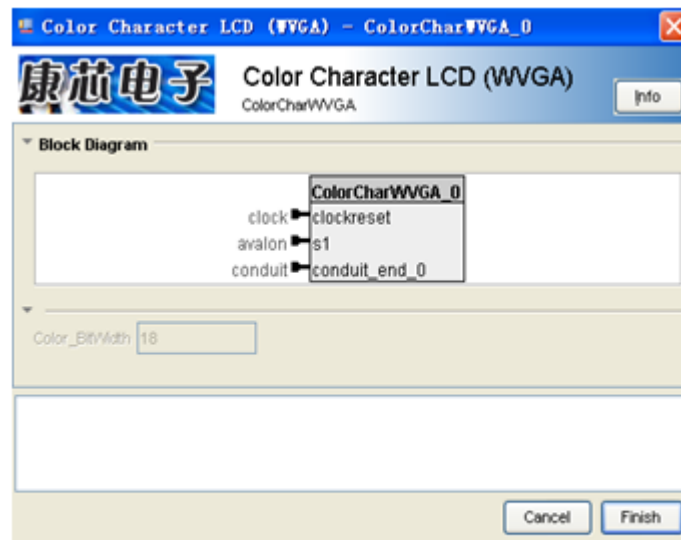


图 11-61 彩色液晶组件设置

11.5 SOPC系统综合设计

11.5.1 综合硬件设计



图 11-62 直流电机组件设置

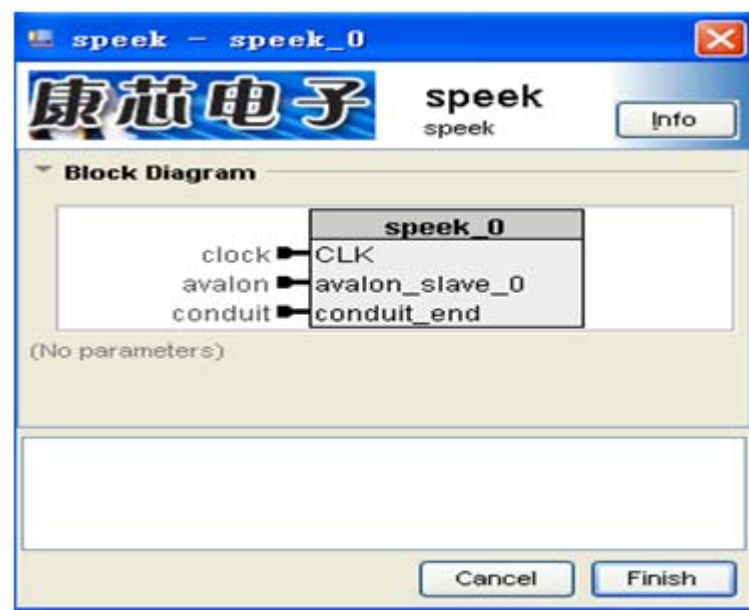


图 11-63 PWM 组件设置

11.5 SOPC系统综合设计

11.5.1 综合硬件设计

Target

Device Family: Cyclone III

Clock Settings

Name	Source	MHz
clk	External	20.0
pll_c0	pll_c0	50.0

Use	...	Module Name	Description	Clock	Base	End
<input checked="" type="checkbox"/>		cpu	Nios II Processor	pll_c0	0x00001000	0x000017ff
<input checked="" type="checkbox"/>		jtag_uart	JTAG UART	pll_c0	0x00001c08	0x00001caf
<input checked="" type="checkbox"/>		ram	On-Chip Memory (RAM or ROM)	pll_c0	0x00020000	0x0002c7ff
<input checked="" type="checkbox"/>		led_pio	PIO (Parallel I/O)	pll_c0	0x00001c80	0x00001c8f
<input checked="" type="checkbox"/>		pll	PLL	clk	0x00001c40	0x00001c5f
<input checked="" type="checkbox"/>		button_pio	PIO (Parallel I/O)	pll_c0	0x00001c70	0x00001c7f
<input checked="" type="checkbox"/>		seven_seg_pio	PIO (Parallel I/O)	pll_c0	0x00001ca0	0x00001caf
<input checked="" type="checkbox"/>		sys_clk_timer	Interval Timer	pll_c0	0x00001c00	0x00001c1f
<input checked="" type="checkbox"/>		scl	PIO (Parallel I/O)	pll_c0	0x00001cb0	0x00001cbf
<input checked="" type="checkbox"/>		sda	PIO (Parallel I/O)	pll_c0	0x00001cc0	0x00001ccf
<input checked="" type="checkbox"/>		uart	UART (RS-232 Serial Port)	pll_c0	0x00001c20	0x00001c3f
<input checked="" type="checkbox"/>		lcd_display	Character LCD	pll_c0	0x00001c90	0x00001c9f
<input checked="" type="checkbox"/>		xlcd_wvga	Char/WVGA	pll_c0	0x00001800	0x00001bff
<input checked="" type="checkbox"/>		motor_pwm	pwm_motor	pll_c0	0x00001cf0	0x00001cf7
<input checked="" type="checkbox"/>		spk	speek	pll_c0	0x00001cd0	0x00001cdf
<input checked="" type="checkbox"/>		epcs_flash_controller	EPCS Serial Flash Controller	pll_c0	0x00000000	0x000007ff
<input checked="" type="checkbox"/>		sysid	System ID Peripheral	pll_c0	0x00001ca0	0x00001ca7

图 11-64 Nios II 完整组件窗

11.5 SOPC系统综合设计

11.5.1 综合硬件设计



图 11-65 选择 Tcl 文件

实验与设计

11-1 多功能数字钟



图 11-66 加入液晶控制 C 程序

实验与设计

11-1 多功能数字钟

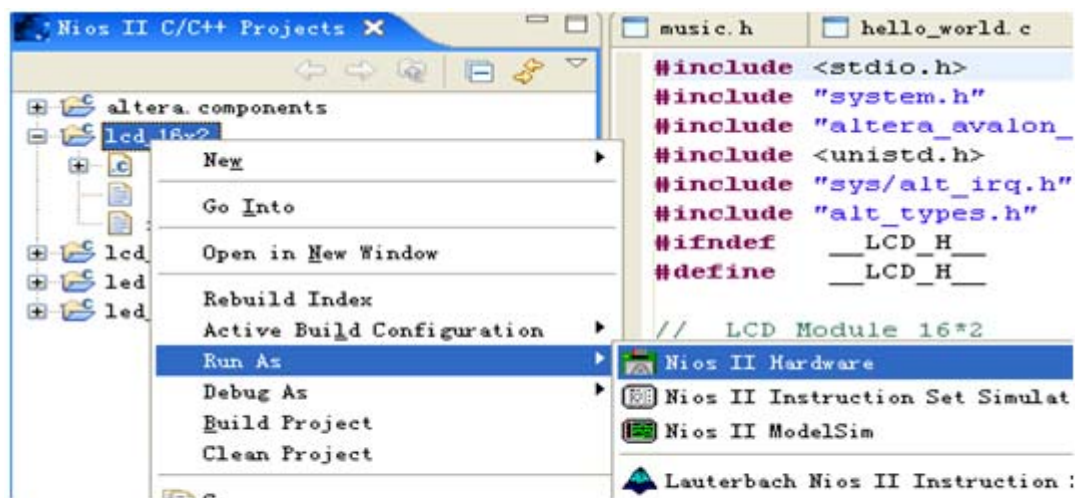


图 11-67 运行软件程序

实验与设计

11-2 彩色液晶显示控制电路设计



图 11-68 新建彩色液晶软件工程

实验与设计

11-3 基于Nios II的直流电机控制

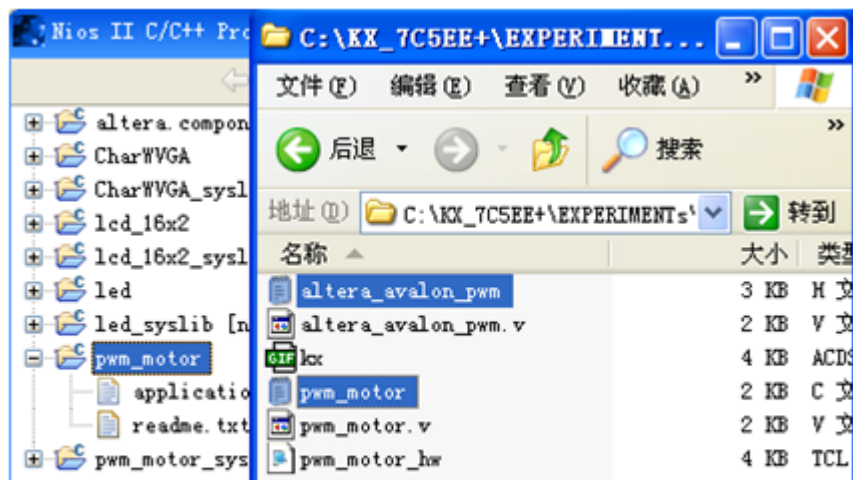


图 11-69 pwm_motor 软件工程

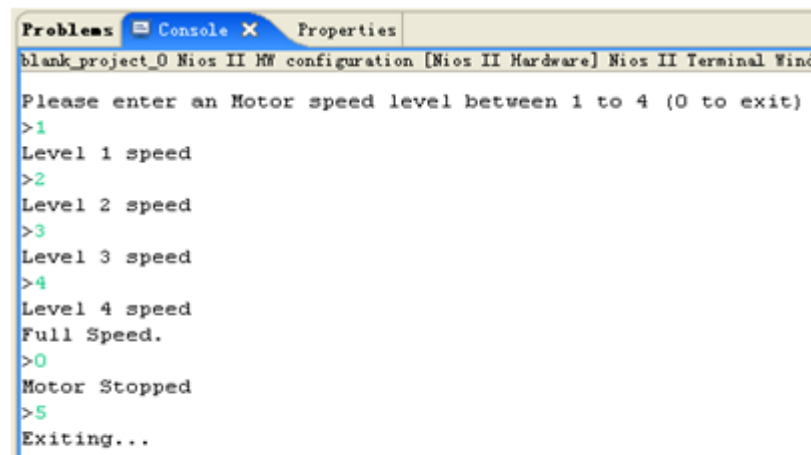


图 11-70 直流电机控制台



实验与设计

11-4 自定义硬件乘法器

【例 11-3】 citest.c

```
#include <stdio.h>
#include "system.h"
int main (void)
{
    int x,y,z ;
    int xr,yr;
    xr=569;
    yr=1923;
    x=xr;
    y=yr;
    printf("\n%08d %08d \n",x,y);
    z=ALT_CI_MULT(x,y);    //使用 nm_mult()调用了乘法自定义指令
    printf("\n%08d %08d %08d\n",x,y,z);
}
```

实验与设计

11-4 自定义硬件乘法器



图 11-71 自定义指令 C 工程: mult

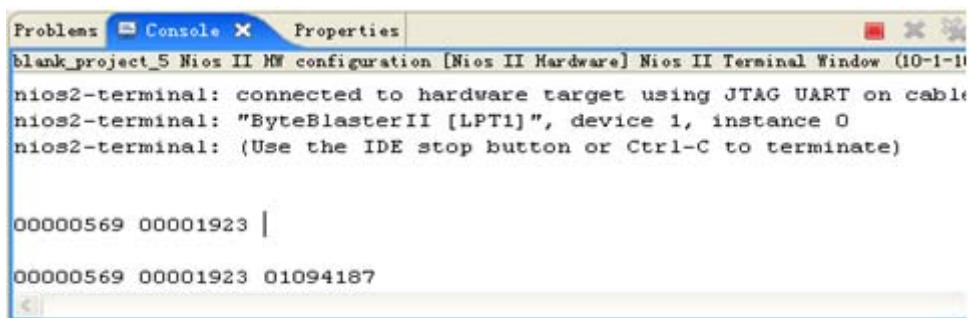


图 11-72 自定义指令 mult 执行结果

实验与设计

11-5 乐曲演播控制

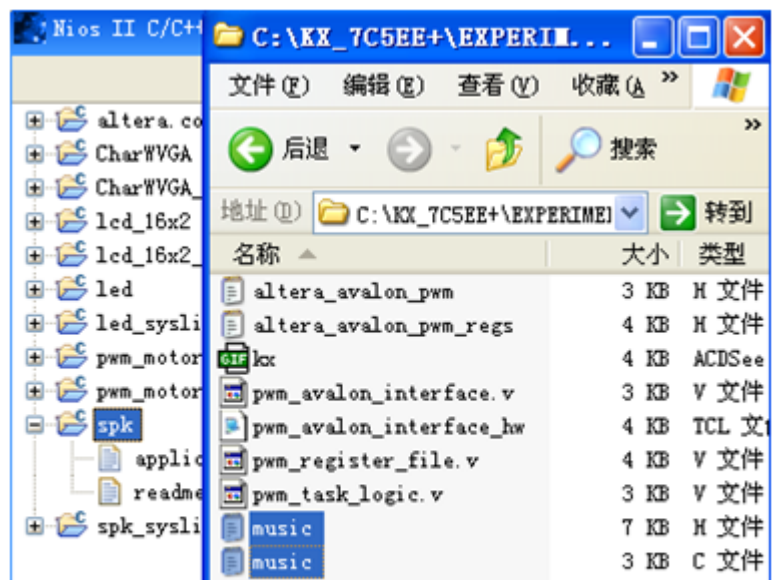


图 11-73 加入音乐软件程序

实验与设计

11-6 基于UART的I2C总线传输



图 11-74 I2C 软件工程

实验与设计

11-6 基于UART的I2C总线传输



图 11-75 打开串口窗口

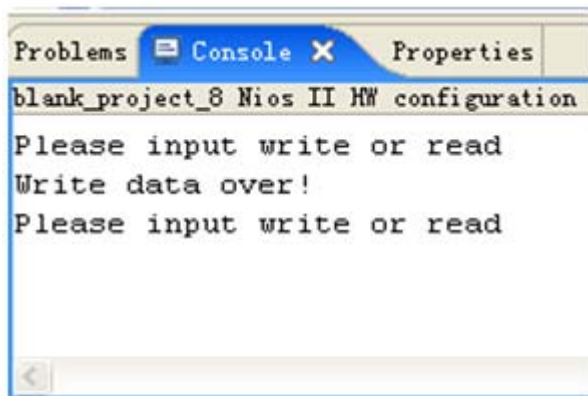


图 11-76 显示写操作完成

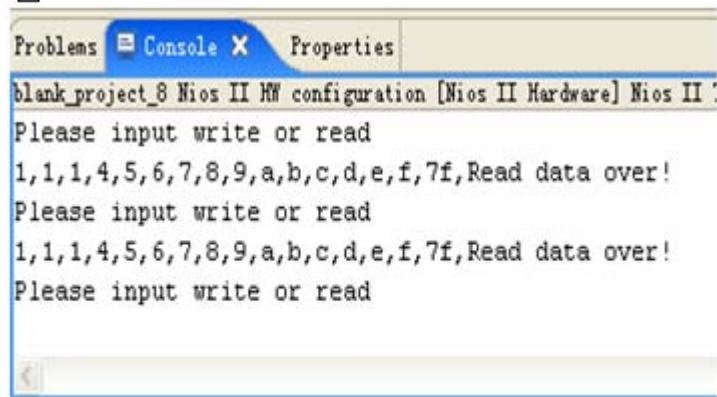


图 11-77 显示读操作完成