**Лабораторная работа 3**

**Программы с использованием арифметических и логических инструкций, инструкций загрузки и сохранения**

**Цель работы**: изучить основные режимы адресации ПЦОС TMS320C40: регистровый, прямой, косвенный, непосредственный. Научиться использовать инструкции загрузки и сохранения, а также логические и арифметические инструкций. Научиться писать и выполнять программы с использованием изученных классов инструкций.

В данной лабораторной работе была создана программа, вычисляющая значение следующего выражения F(x,y,z)=ax^2-by^2+az. Для этого написал следующий код:

\_myvect .sect"myvect";Созданем секцию для векторов прерываний

reset .word \_c\_int00 ;RESET вектор

.space 1;Резервируем место для NMI прерывания

;Создаем таблицу векторпов программных прерываний

\_mytrap .sect"mytrap";Секция для векторов программных прерываний

;Создаем стек

\_mystack .usect"mystack",100;Резервируем 100 слова под стек

.text

stacka .word \_mystack;адрес mystrack секции

ivta .word \_myvect;адрес myvect секции

tvta .word \_mytrap;адрес mytrap секции

ieval .word 1;значения регистра разрешения прерываний

gctrl .word 1EF78000h;значение регистра управления памятью

lctrl .word 1EF78000h;значение регистра управления памятью

mctrla .word 100000h;адрес регистра управления глобольной памятью

\_c\_int00: ;точка входа в программу, при reset начинаем отсюда

; Инициализация регистра DP

ldp stacka;

;Устанавливаем указатель на таблицу прерываний

ldi@ivta, AR0

ldpe AR0, IVTP;

;Устанавливаем укзатель на таблицу программных прерываний

ldi@tvta, AR0

ldpe AR0,TVTP

;Иницилизирум трегистр управления глобольной паматью

ldi@mctrla, AR0

ldi @gctrl,R0

sti R0, \*AR0

;Инициализируем регистр управления локальной памятью

ldi@lctrl, R0

sti R0, \*+AR0(4)

;Инициализируем указатель стека

ldi@stacka, SP

;Инициализируем регистр разрешения прерываний

;Запись 1 в IIE разрешает прерывание таймера

ldi @ieval,IIE

;Глобально разрешаем прерывания, очищаем и размещаем кэш

or 3800h, ST

Br begin;переход к началу приложения

nop

nop

nop

;Вычисление значение F(x,y,z)=ax^2-by^2+az

;Размещаем данные

.data

a .word 2

b .word 3

z .word 1

x .word 5

y .word 4

;Код программы снова в секции text

.text

begin;Начало программы

;Устанавливаем DP

ldp a

ldi 0,R0;R0 использум для хранения результата

;вычисляем a\*x\*x

ldi @a,AR0 ;AR0=a

mpyi @x,AR0 ;AR0=a\*x

mpyi @x,AR0 ;AR0=a\*x\*x

;вычисляем b\*x\*x

ldi @b,AR1; Ar1=b

mpyi @y,AR1 ;AR1=b\*y

mpyi @y,AR1 ;AR1=b\*y\*y

;вычисляем a\*z

ldi @a,AR2 ; ar2=a

mpyi @z,AR2 ;AR2=a\*z

;сейчас ar0=ax^2 ar1=by^2 ar2=az

;сложим эти значения

subi AR1,AR0 ;ar0=ar0-ar1

addi AR2,AR0 ;ar0=ar0+ar2

;ar0 содержит результат вычисленного выражения

;сохраним его по адресу 80001000h

ldhi 8000h,AR3

or 1000h,AR3 ;AR3=80001000h

sti AR0,\*AR3

;<<вставте сюда код вашего приложения>>

stop br stop;

nop

nop

nop

.end

Далее проверил правильность кода, закомпилировав проект. На рисунке 1 можно увидеть, что проект был написан без ошибок.

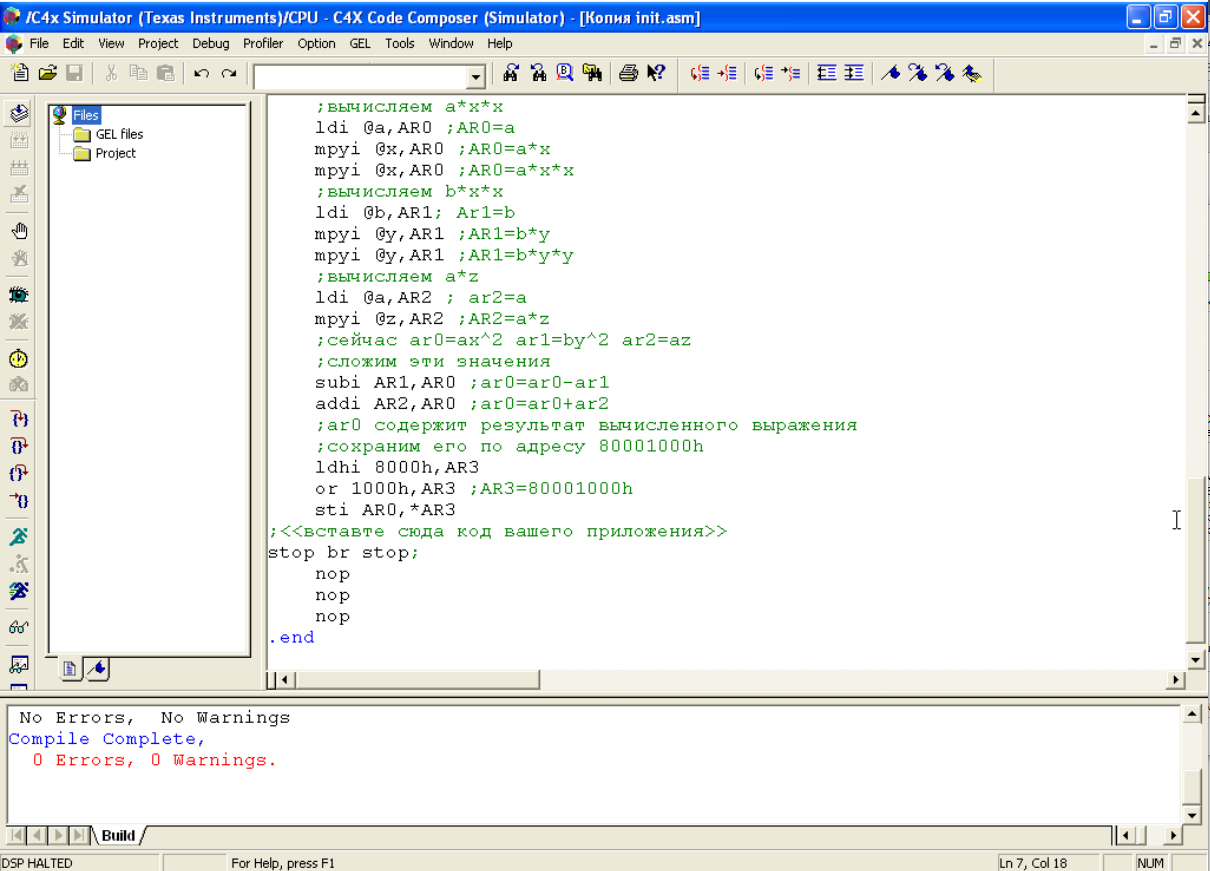


Рисунок 1 – Компиляция проекта