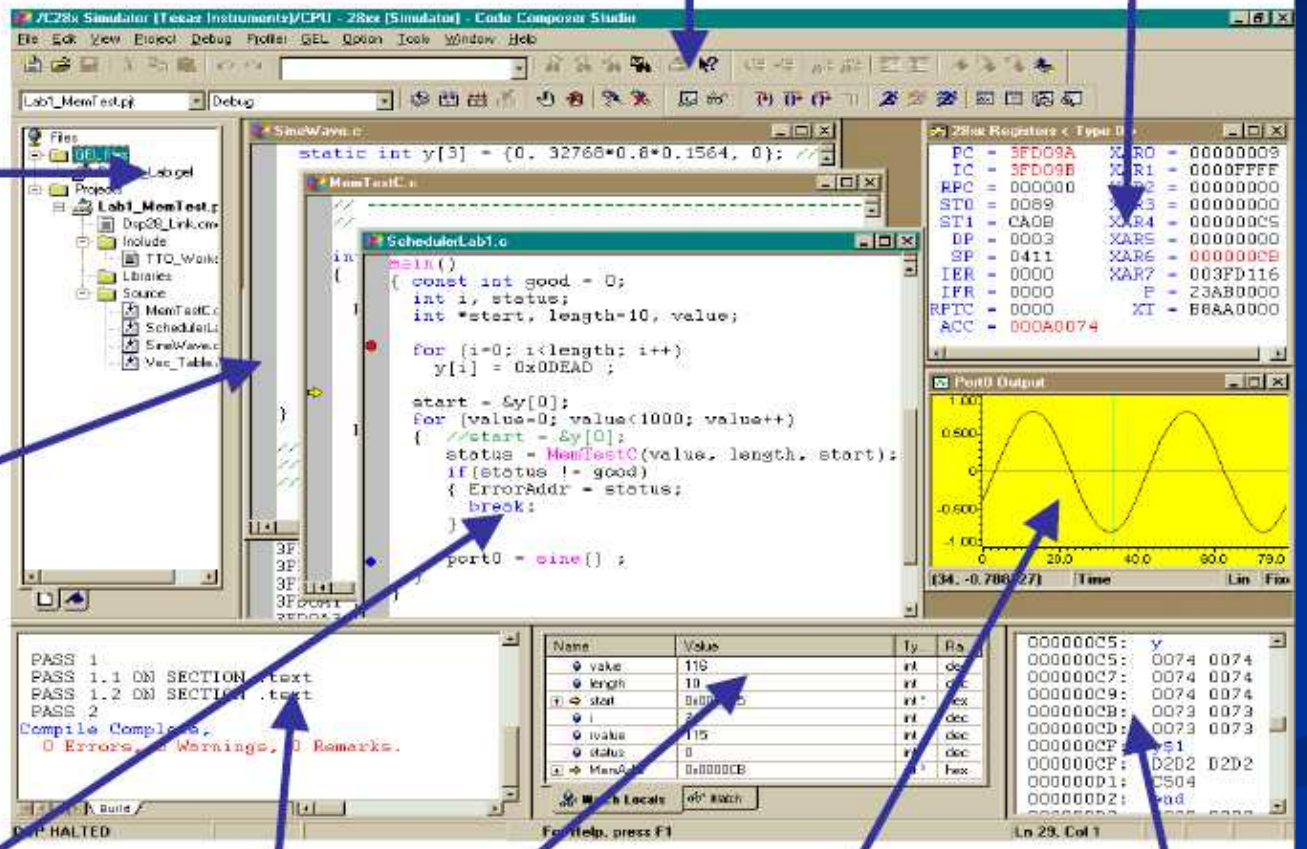
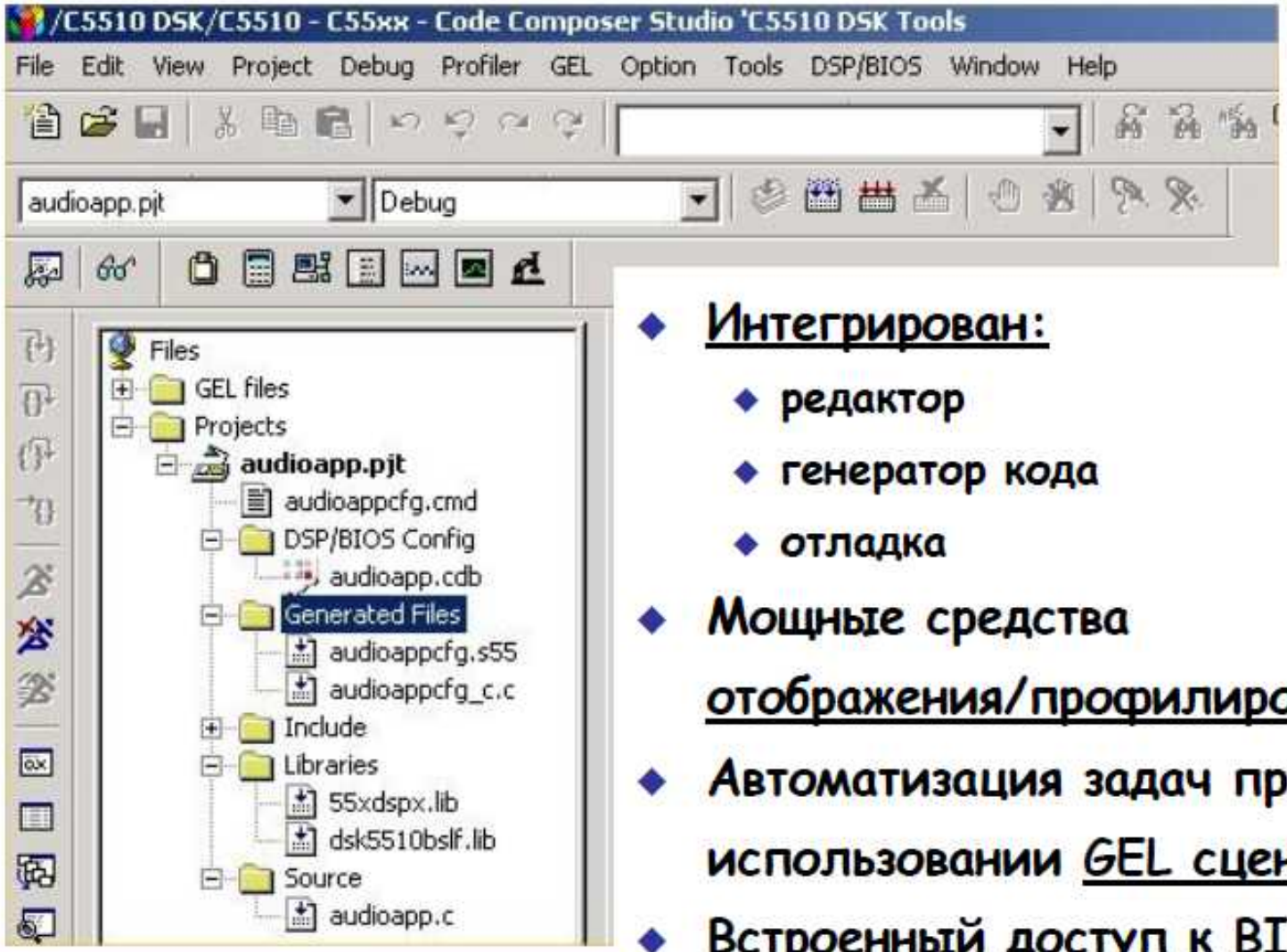
Интегрированная среда программирования Code Composer Studio

Управление проектом:
- Исходные и объектные файлы
- Файл зависимостей

Полная C/C++ и ассемблер отладка:
- Смешанный режим
- Дизасемблер
- Установка точек останова и пробных точек



Интегрированная среда программирования Code Composer Studio



- **Интегрирован:**
  - редактор
  - генератор кода
  - отладка
- Мощные средства отображения/профилирования
- Автоматизация задач при использовании GEL сценариев
- Встроенный доступ к BIOS функциям

Интегрированная среда программирования Code Composer Studio

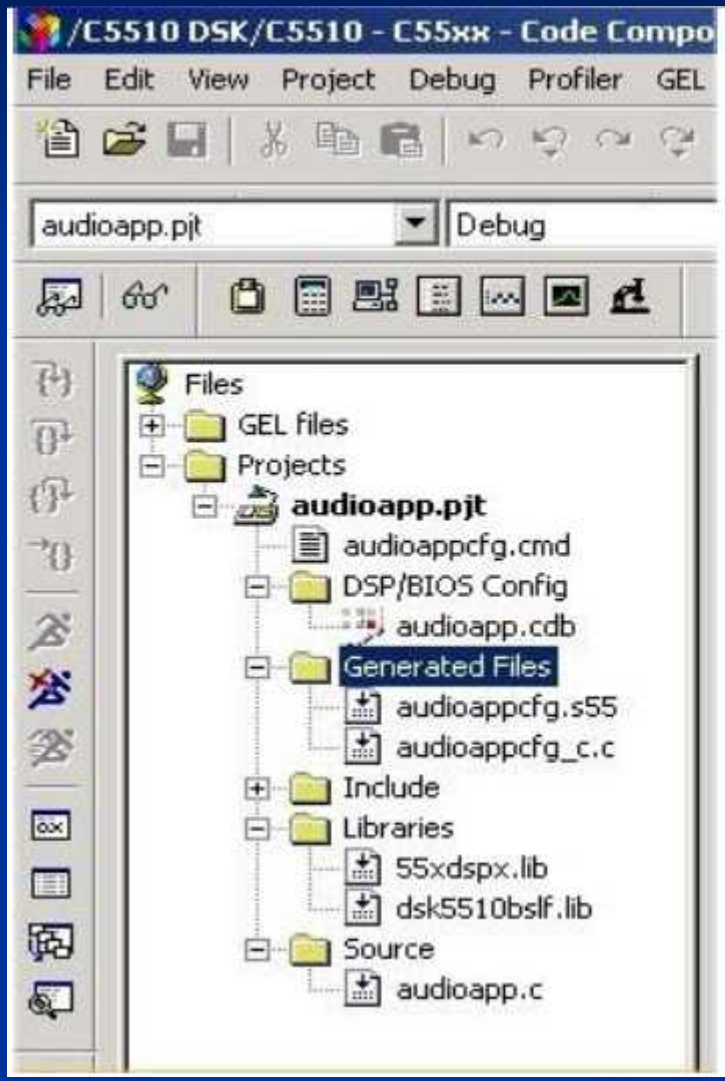
- *Отладка*
  - *Загрузка исполняемого кода*
  - *Запуск в реальном времени или шаг за шагом*
  - *Точки останова*
  - *Профилирование*
  - *Сохранение дампа памяти в файл*
  - *Формирование сигнала по содержимому памяти*

Интегрированная среда программирования Code Composer Studio

- Для построения проекта необходимы:
- Исходные файлы (C,C++ и/или ассемблер) (*.c, *.asm)
  - Текстовые файлы с проверкой синтаксиса, в основном для ассемблера *.asm
- Командный файл компоновки(*.cmd)
- Файл проекта(*.pjt)
  - Содержит исполняемый файл для CCS
  - Определяет все исходные файлы и командный файл компоновки, необходимые для компиляции исполняемого файла

Интегрированная среда программирования Code Composer Studio

- **Проект в CCS. Файл проекта *.pjt содержит:**



Исходные файлы:

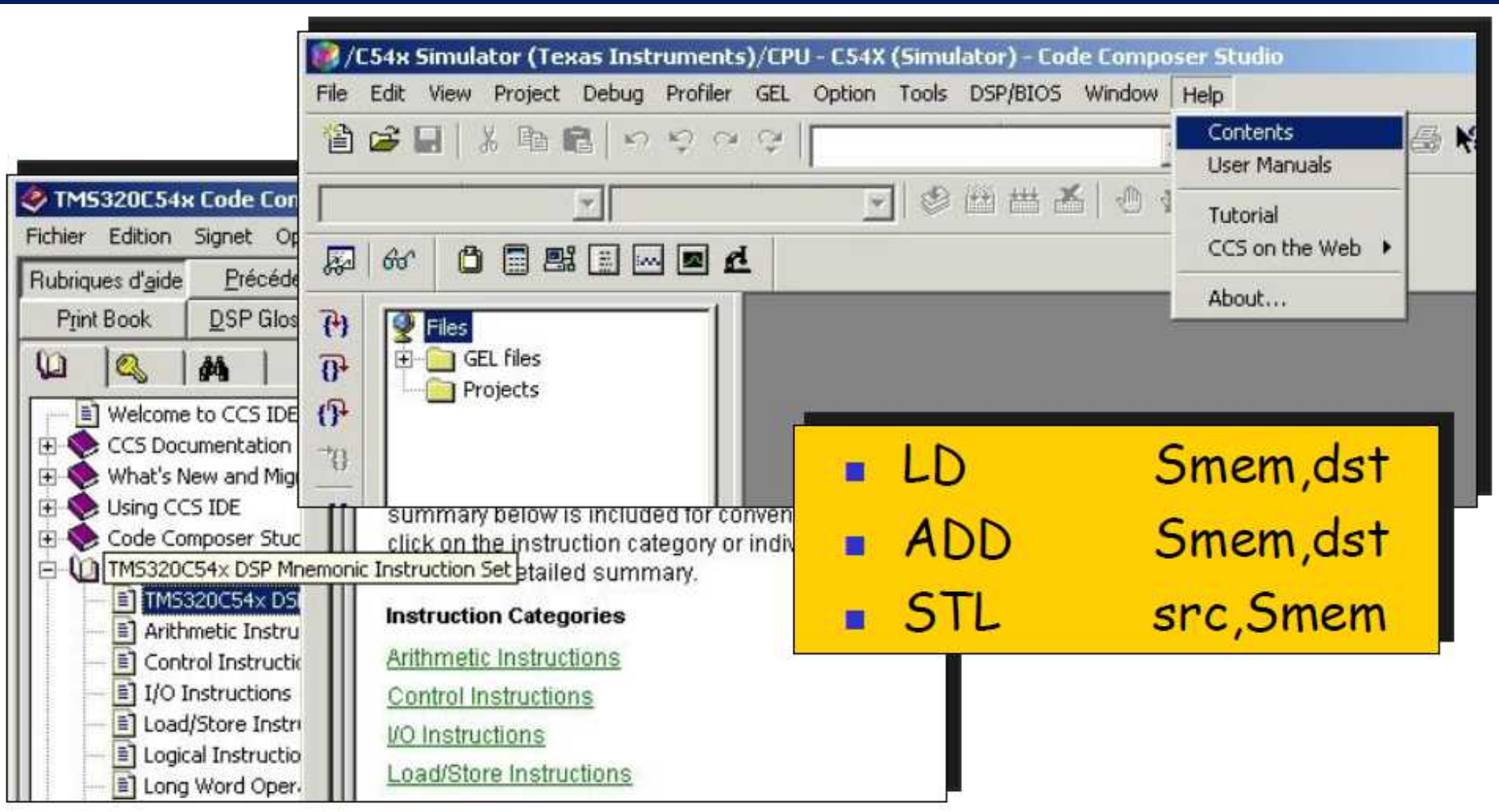
- Исходные файлы (C, ASM)
- Библиотеки
- DSP/BIOS конфигурация
- Компоновщик и др.

Настройки проекта:

- Опции компиляции (компилятор и asm)
- Конфигурации компилятора
- DSP/BIOS
- Компоновщик

Интегрированная среда программирования Code Composer Studio

- Применяемые инструкции и помощь online:



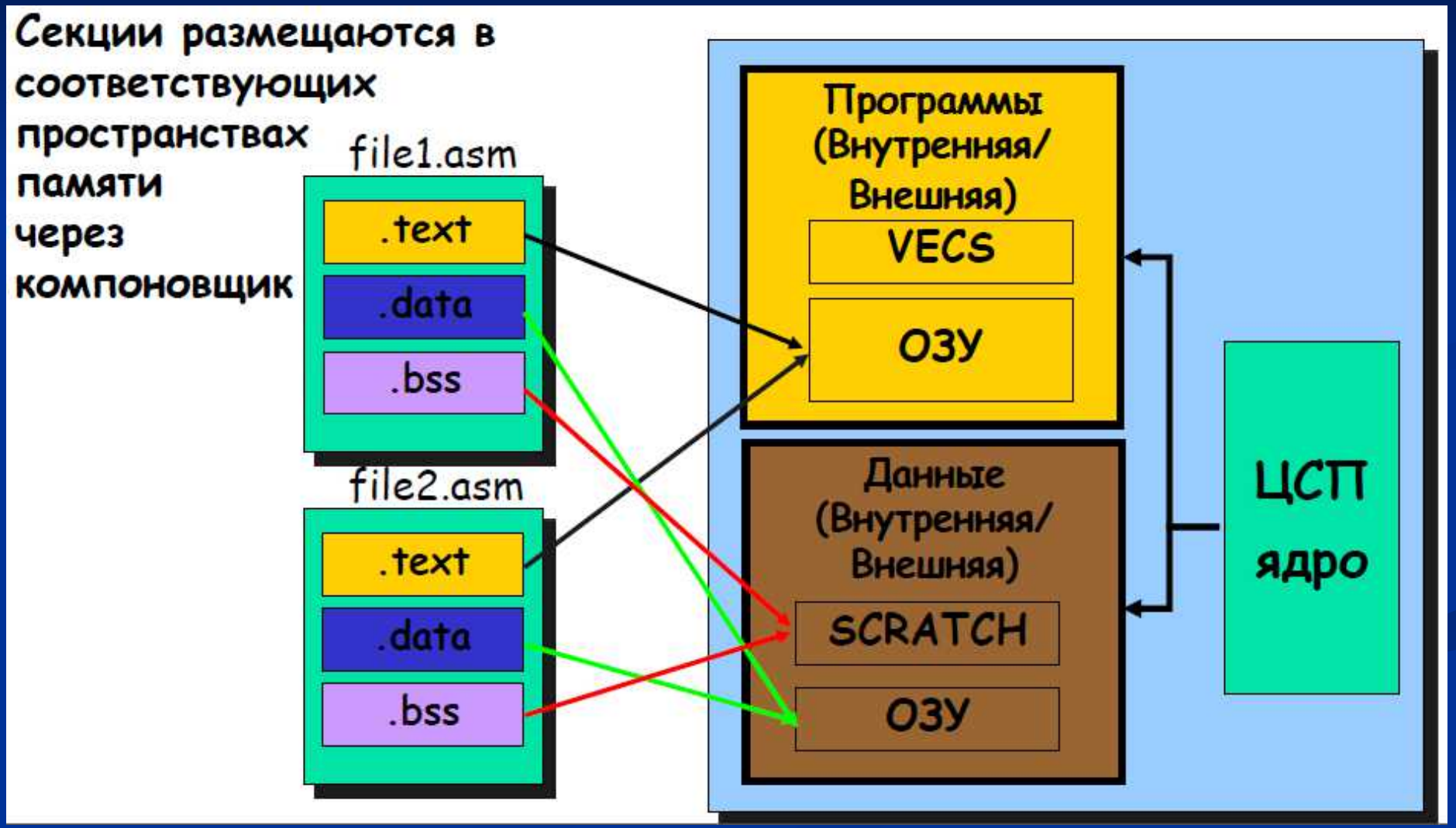
Интегрированная среда программирования Code Composer Studio

- **Командный файл компоновки:**

```
MEMORY
{
        PAGE 0: VECS:    origin = 0080h, length = 0080h
                                          /* Internal Program RAM */
                PRAM:    origin = 100h, length = 1f00h
                                          /* Internal Program RAM */
        PAGE 1: SCRATCH:origin = 0060h, length = 0020h
                                          /* Scratch Pad Data RAM */
                INRAM:   origin = 2000h, length = 1fffh
}                                         /* Internal Data RAM    */
SECTIONS
{   .text     >     PRAM  PAGE 0
    .data     >     INRAM PAGE 1
    .bss      >     SCRATCH PAGE 1
}
```

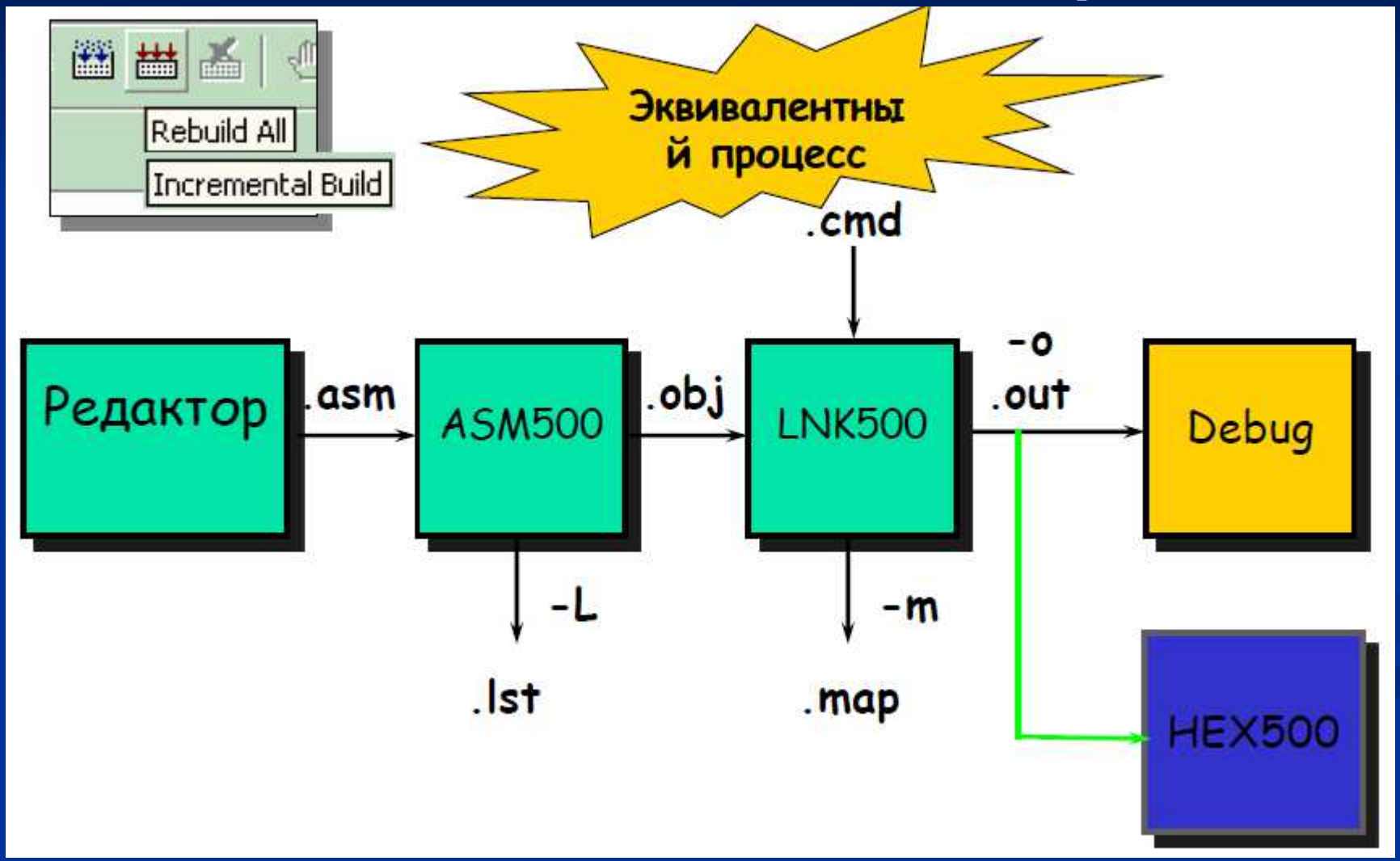
Интегрированная среда программирования Code Composer Studio

- **Пространства памяти и программные секции:**



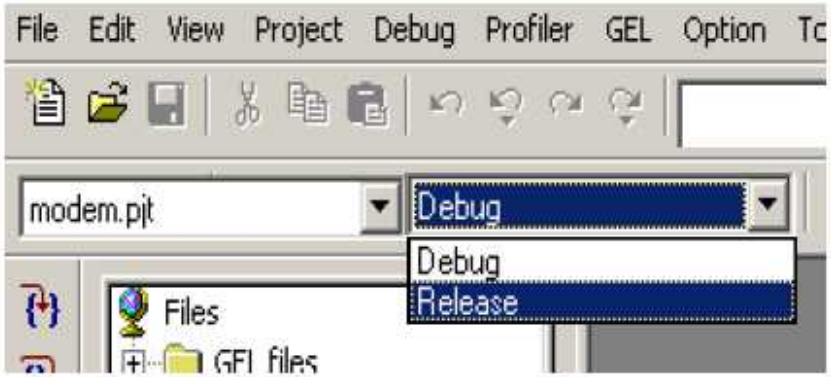
Интегрированная среда программирования Code Composer Studio

- **Компиляция исполняемого файла:**

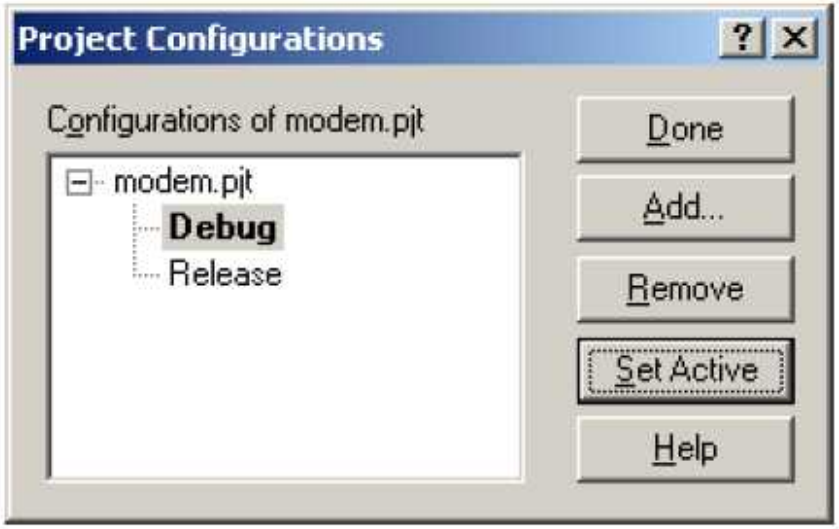


Интегрированная среда программирования Code Composer Studio

- **Конфигурации компиляции по умолчанию:**



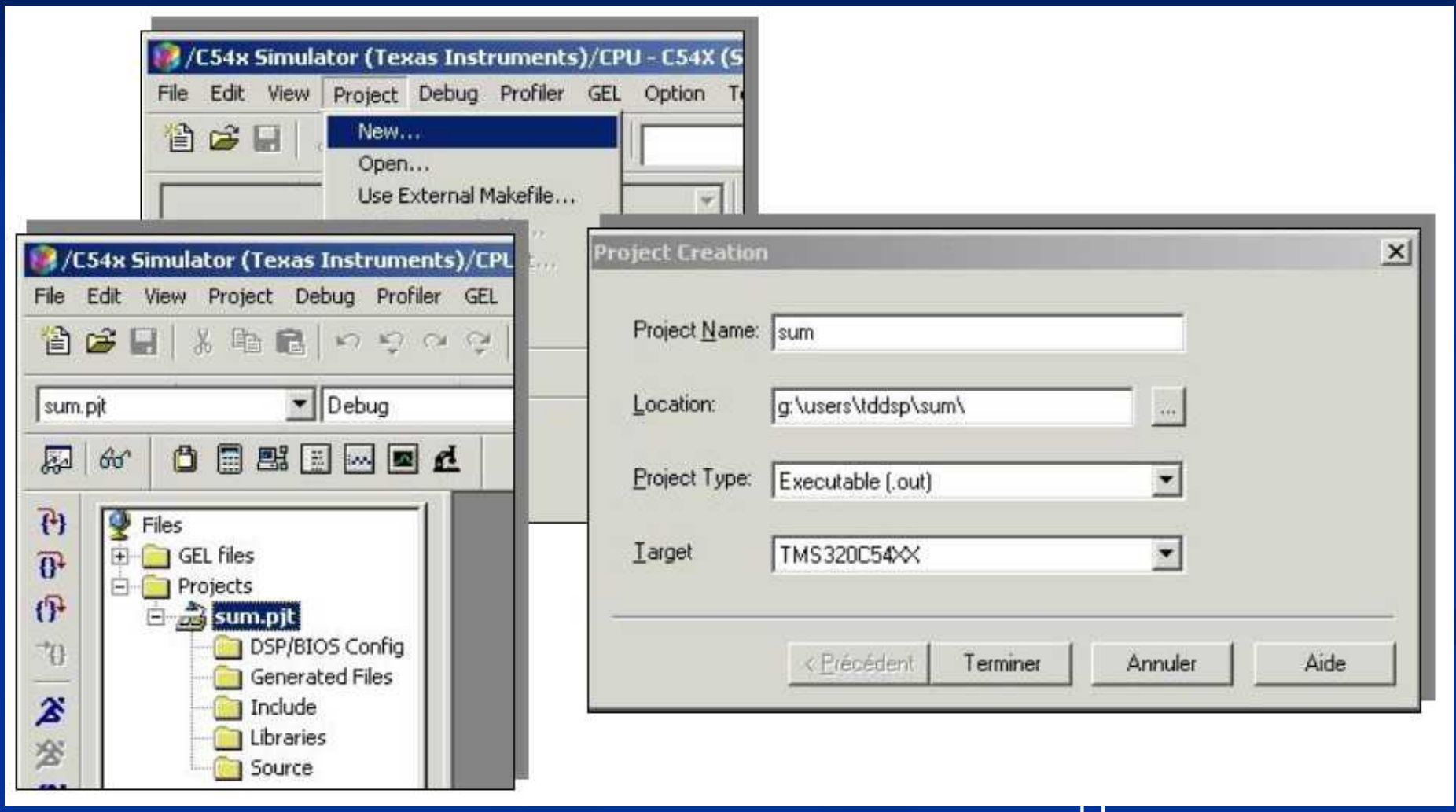
- Для нового проекта, CCS автоматически создает две конфигурации:
  - Debug (без оптимизации)
  - Release (оптимизированная)



- Добавление/удаление конфигураций компиляции через *Project Configurations*
- Редактирование конфигураций:
  1. Выбор активной
  2. Изменение опций
  3. Сохранение проекта

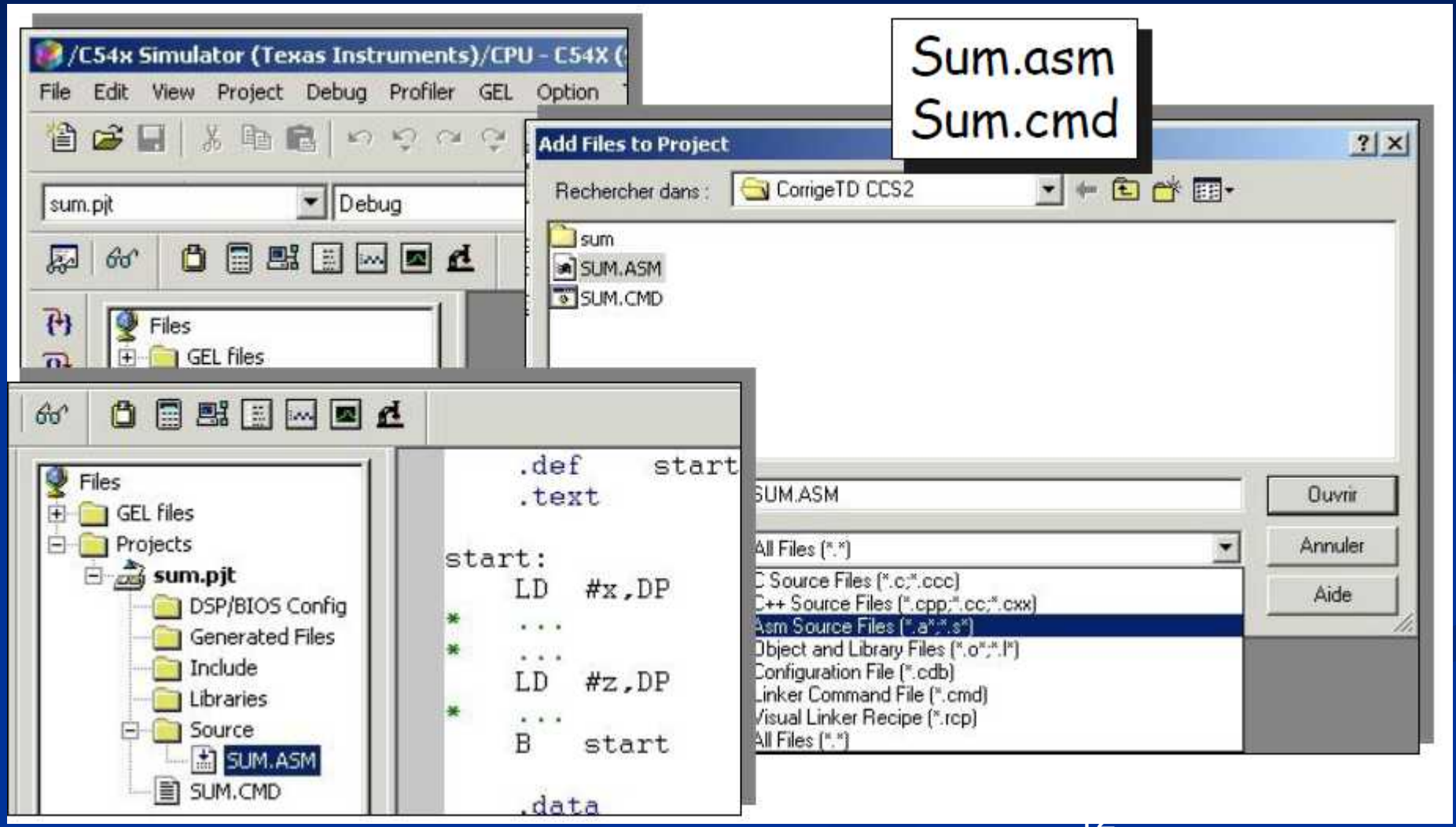
Интегрированная среда программирования Code Composer Studio

- **Создание проекта:**



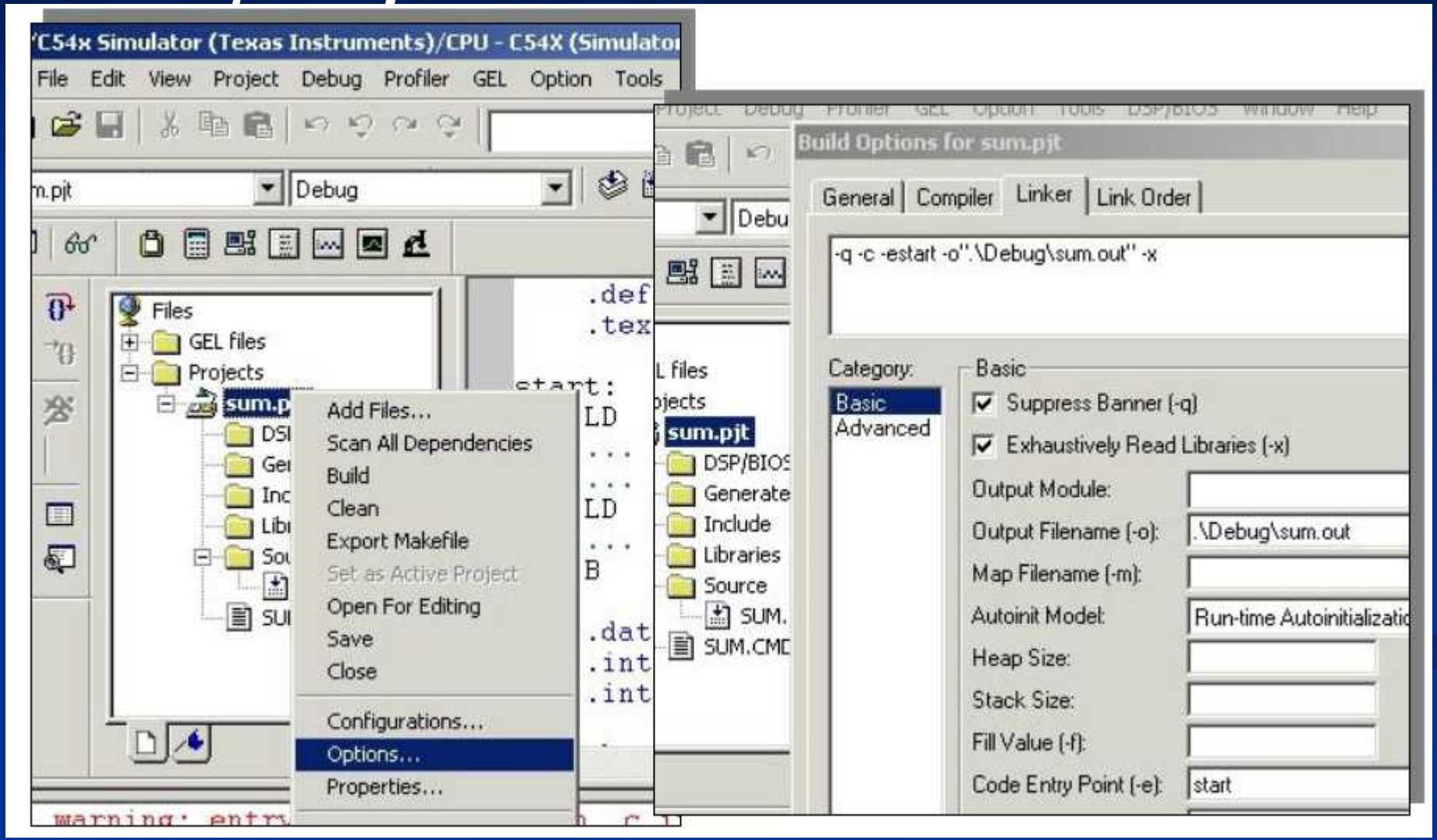
Интегрированная среда программирования Code Composer Studio

- **Создание проекта:**



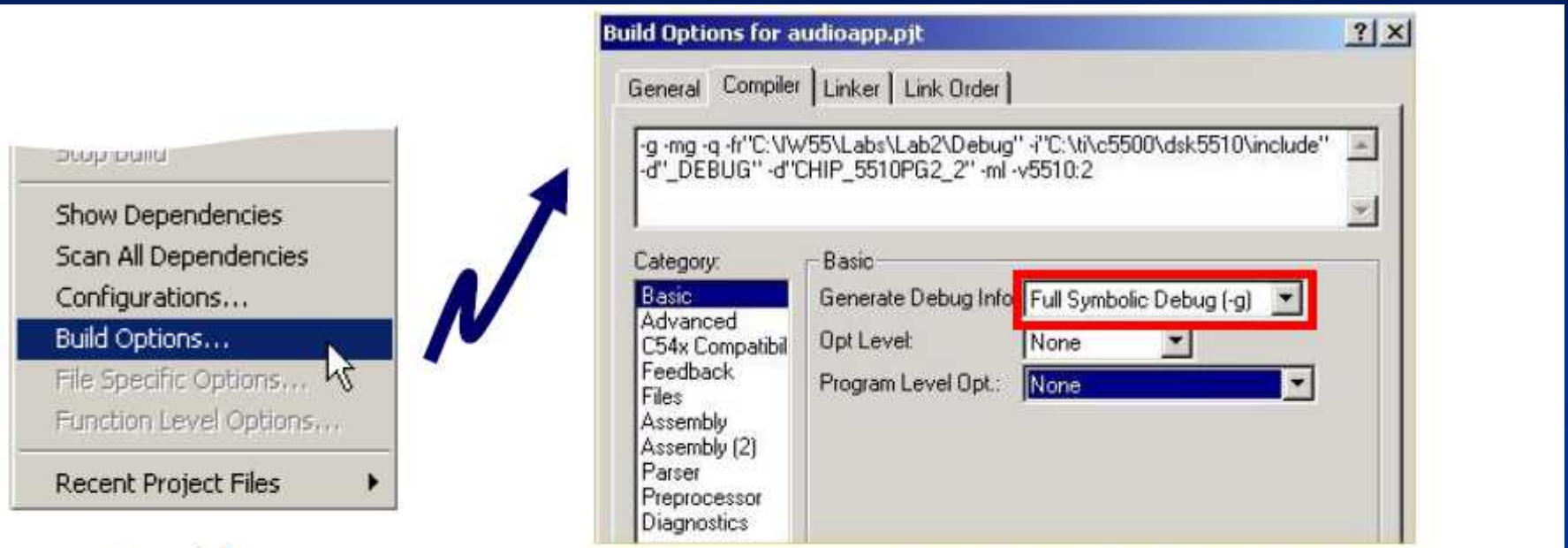
Интегрированная среда программирования Code Composer Studio

- **Опции проекта:**



Интегрированная среда программирования Code Composer Studio

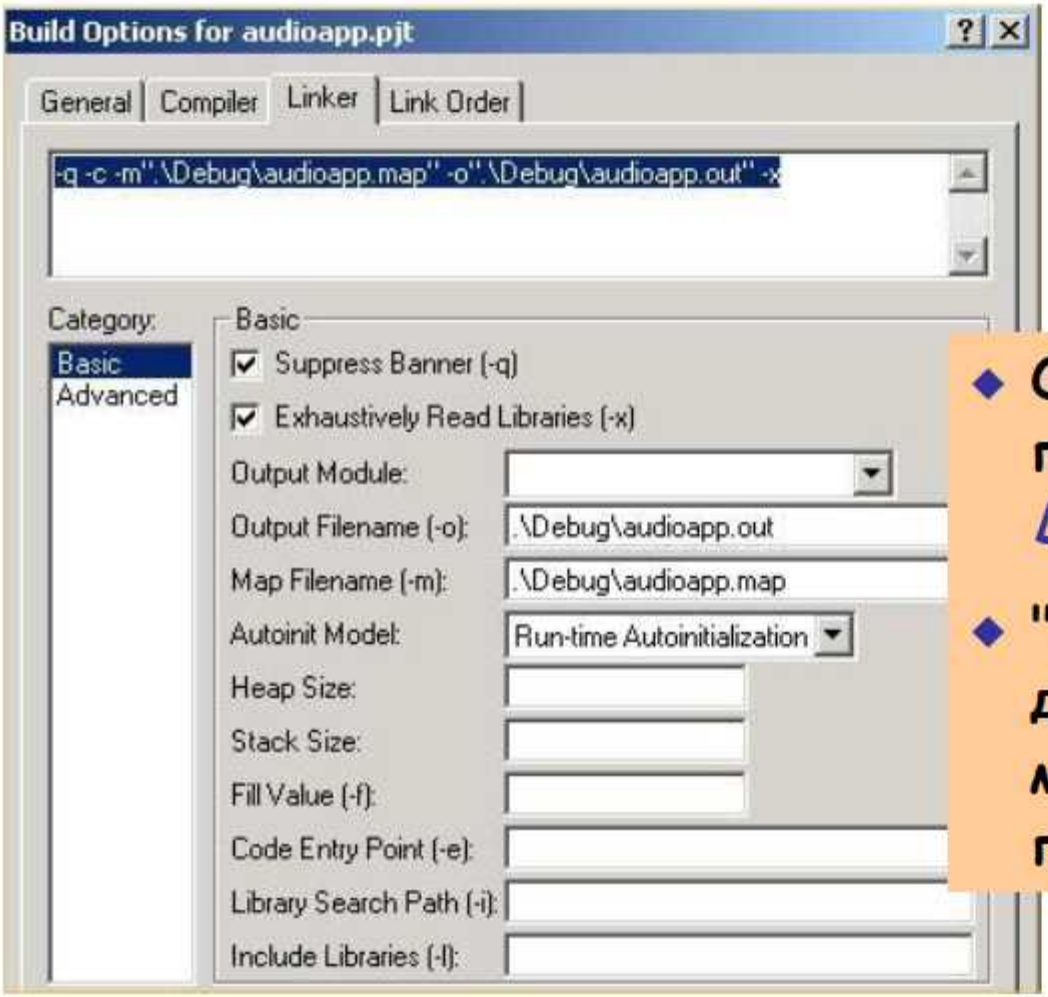
- **Опции компилятора:**



- Управление многими аспектами компиляции:
  - Уровни оптимизации
  - Выбранное устройство
  - Опции компилятора/компоновщика
  - Другие

Интегрированная среда программирования Code Composer Studio

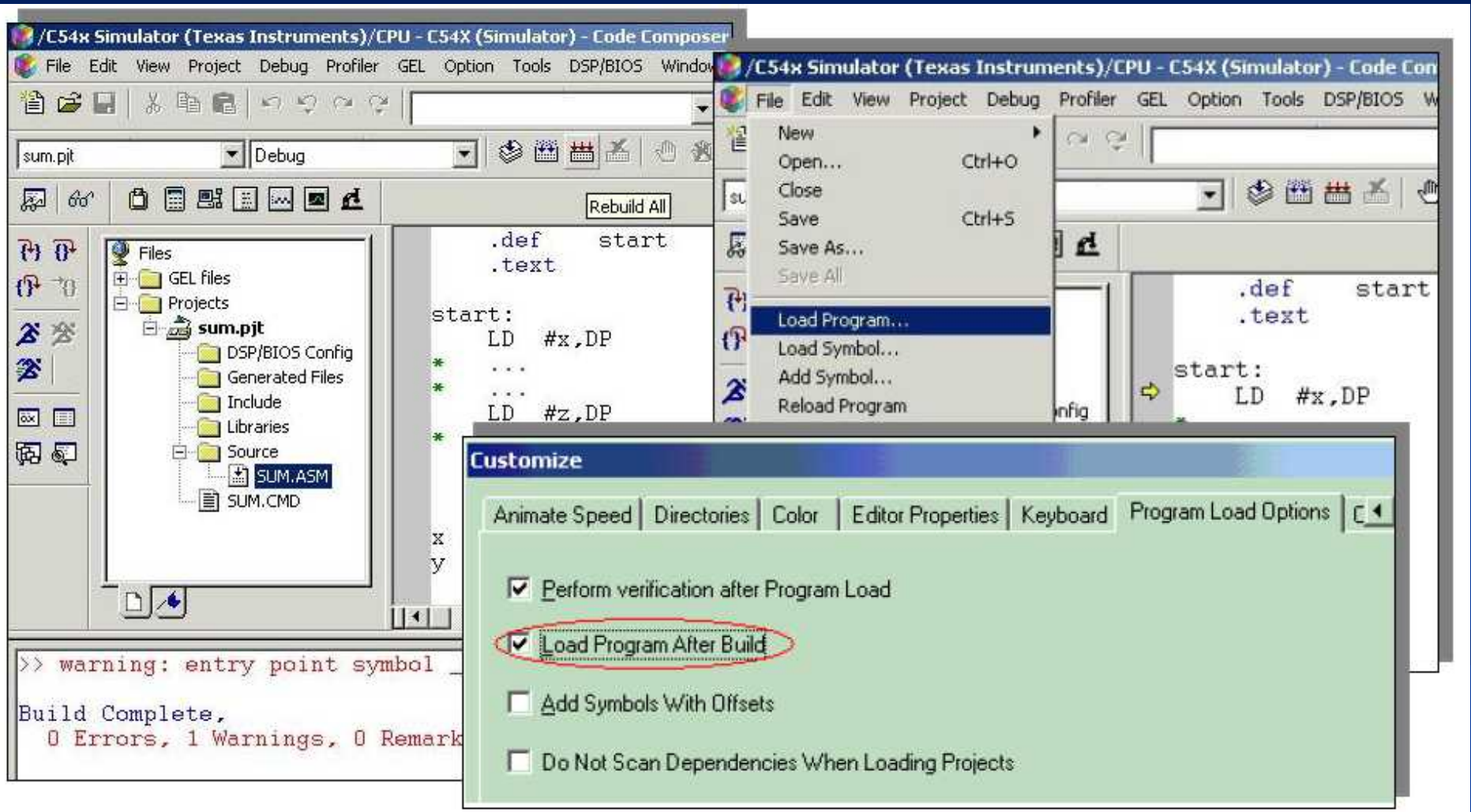
- **Опции компоновщика:**



- Опции компоновщика показаны в конфигурации *Debug* для проекта **.pjt*
- ".\Debug\" указывает на директорий уровнем ниже места расположения проекта (.pjt)

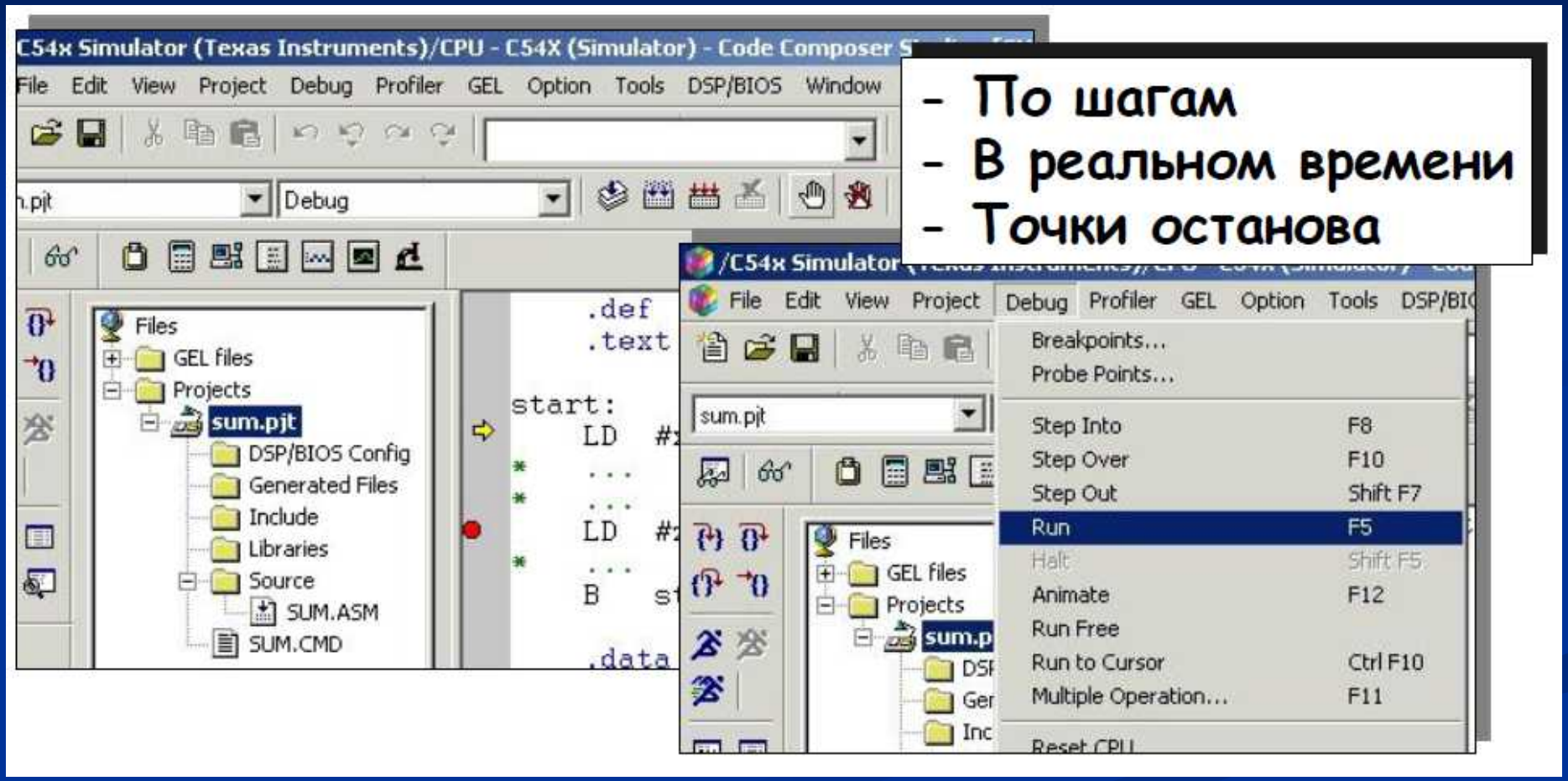
Интегрированная среда программирования Code Composer Studio

- **Построение проекта:**



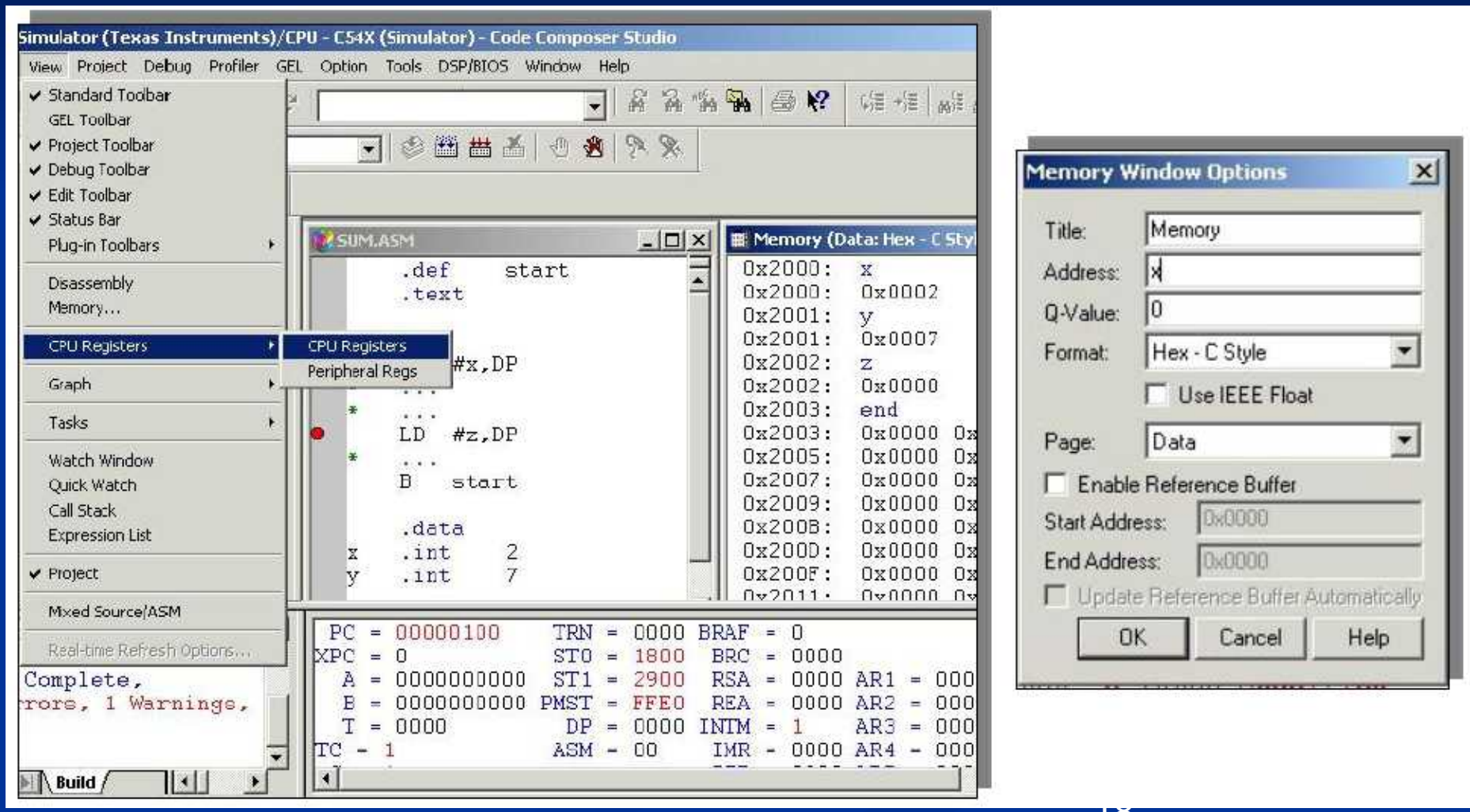
Интегрированная среда программирования Code Composer Studio

- **Запуск проекта:**



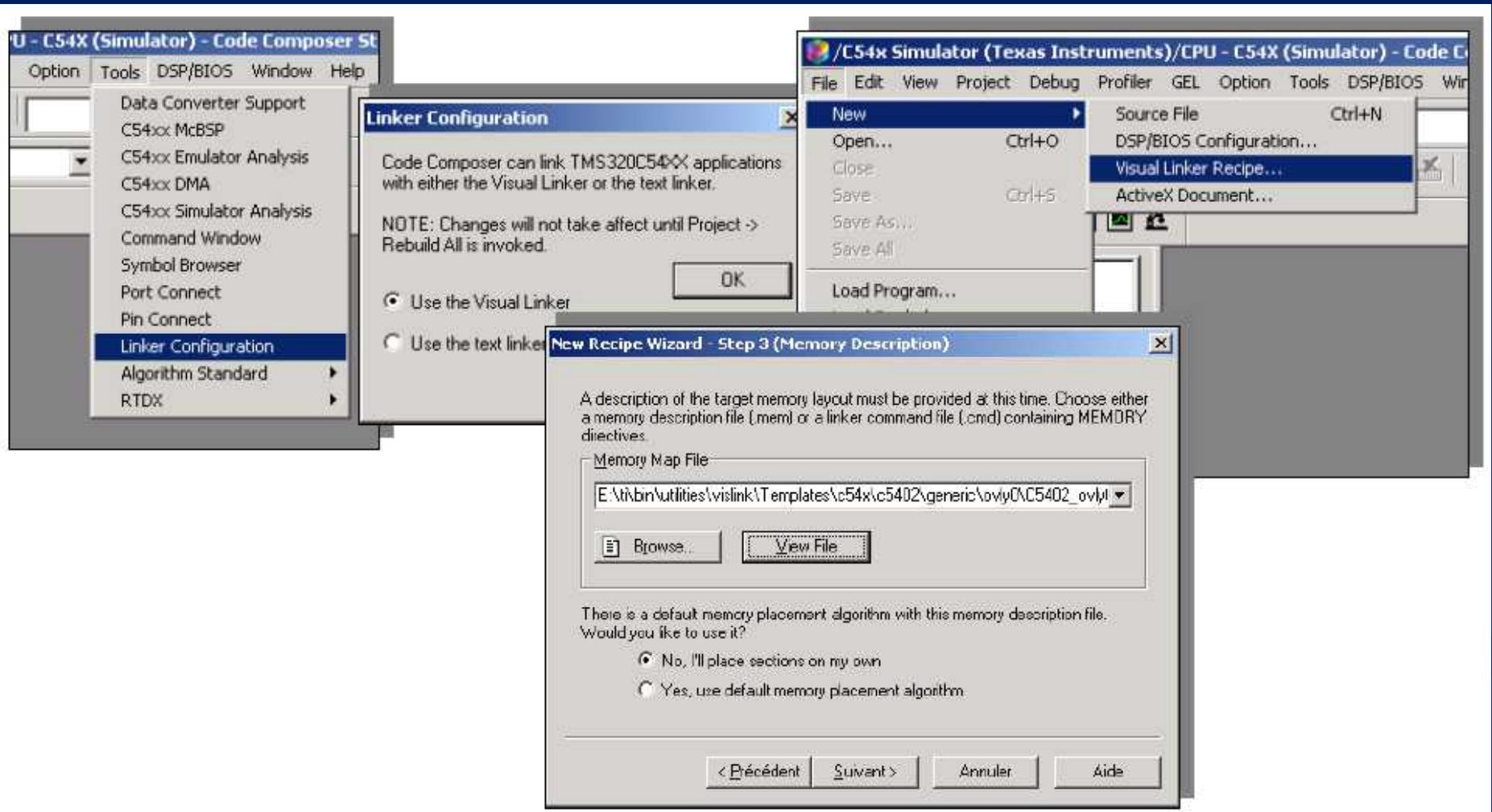
Интегрированная среда программирования Code Composer Studio

- **Просмотр памяти и регистров ЦП:**



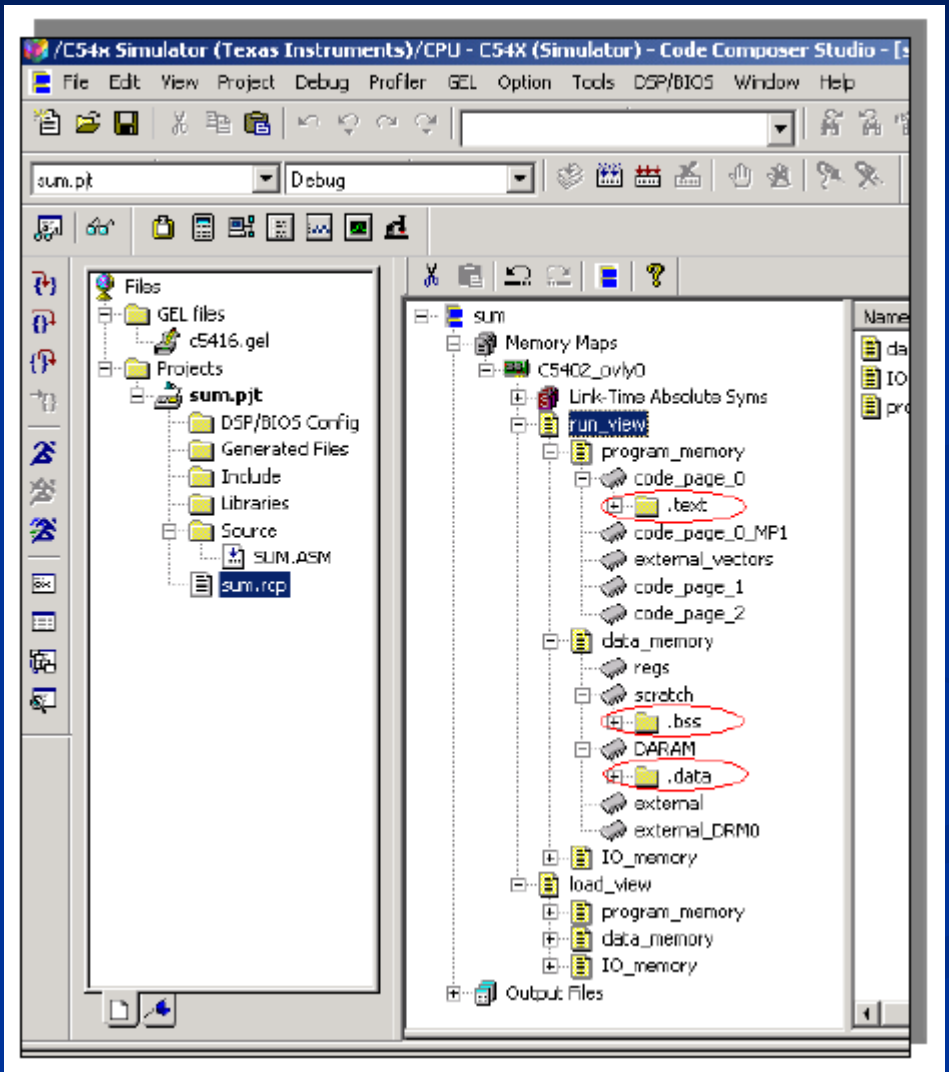
Интегрированная среда программирования Code Composer Studio

- **Визуальный компоновщик:**



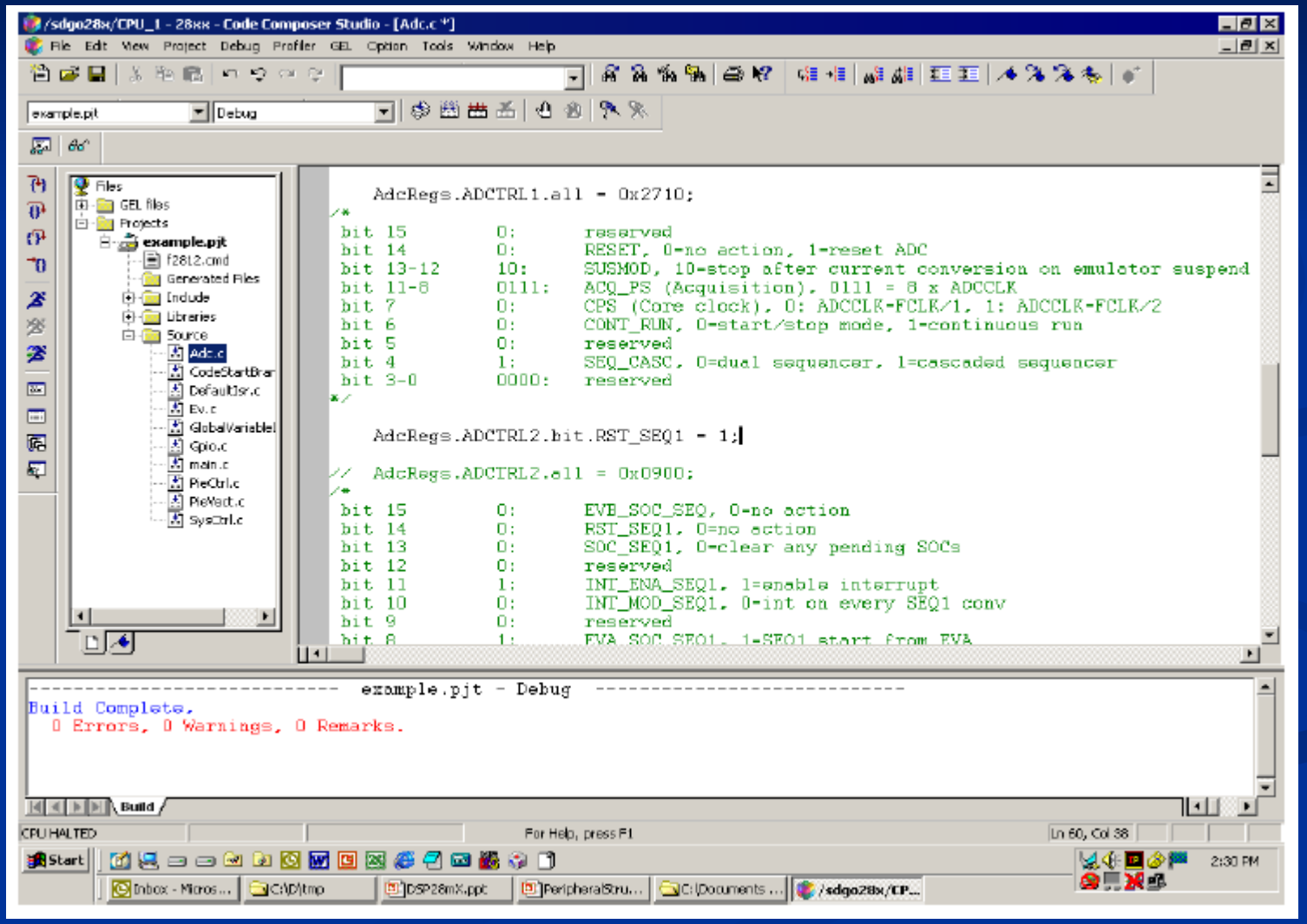
Интегрированная среда программирования Code Composer Studio

- **Визуальный компоновщик:**



Интегрированная среда программирования Code Composer Studio

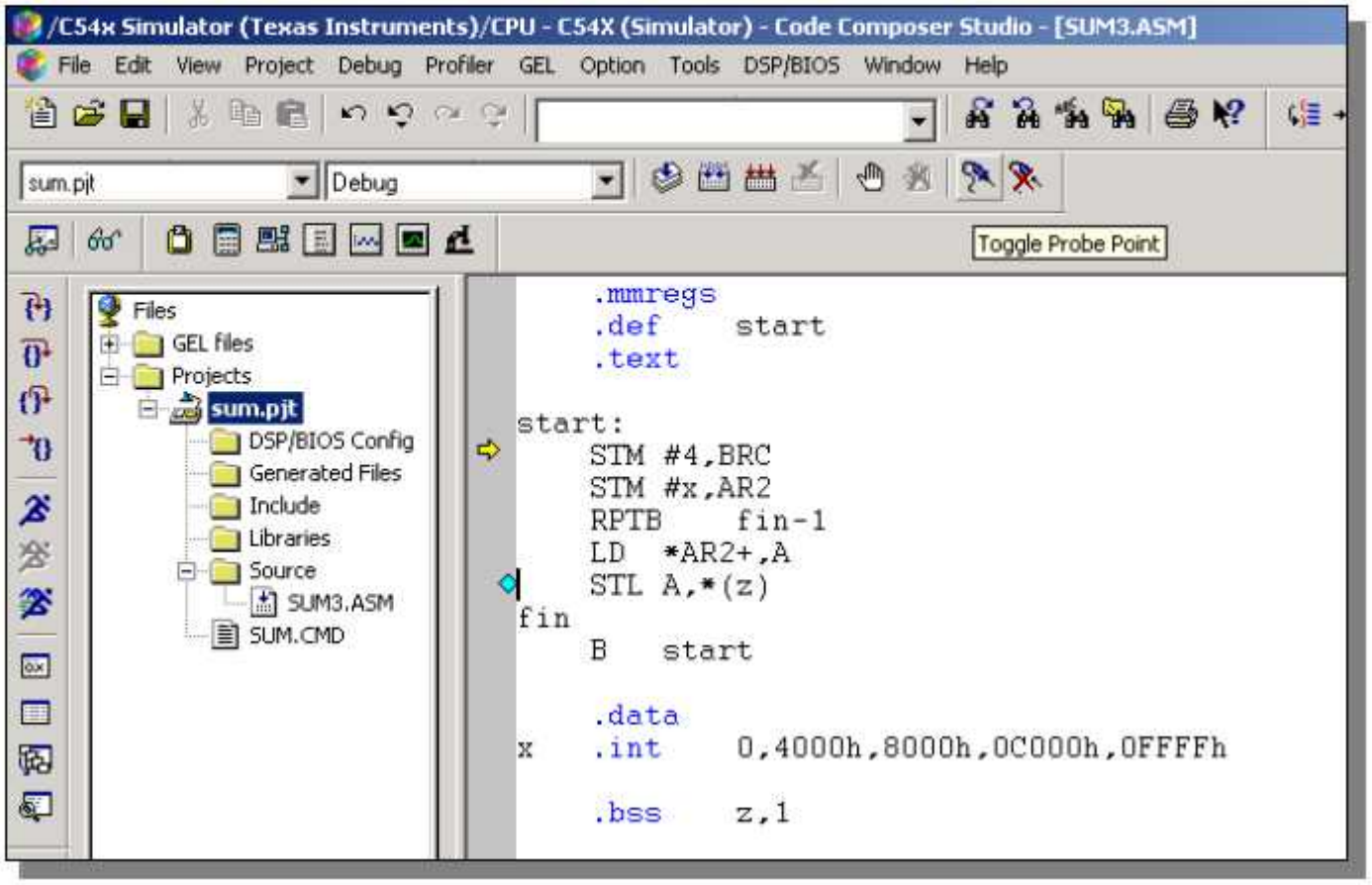
- **Code Maestro:**



Интегрированная среда программирования Code Composer Studio

- **Динамическое отображение графика:**

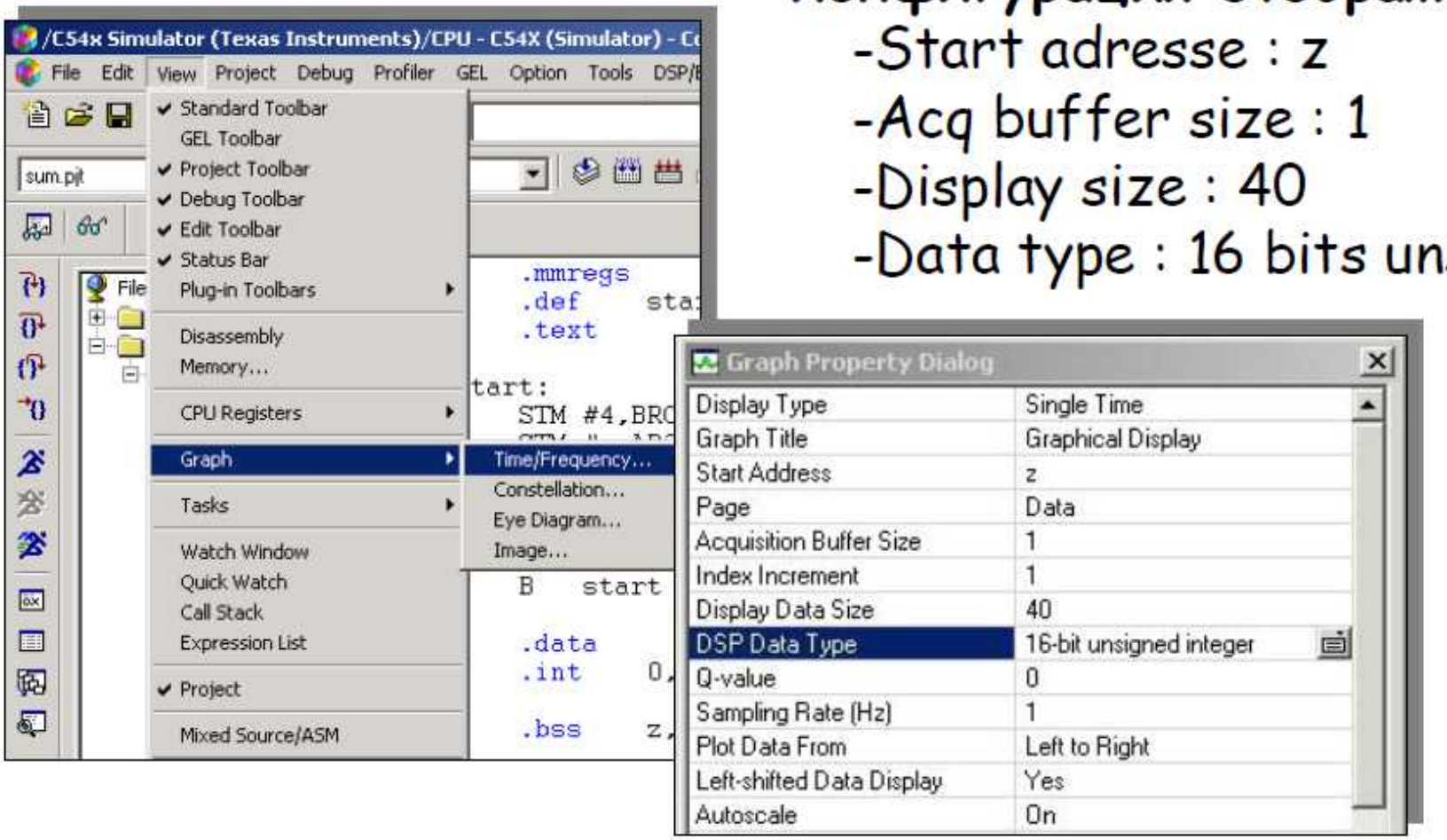
- Встраивание Probe point



Интегрированная среда программирования Code Composer Studio

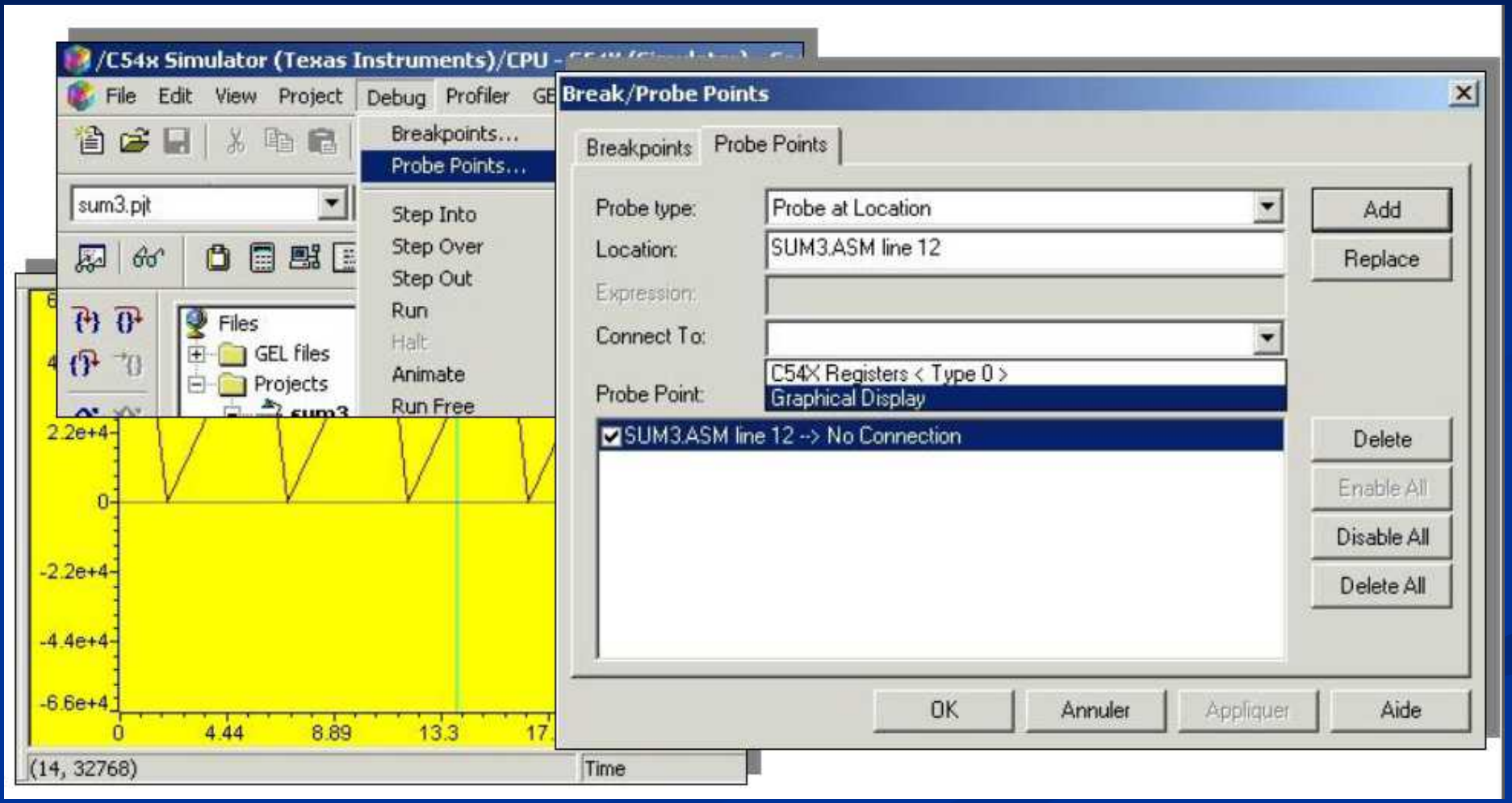
- **Динамическое отображение графика:**

-Конфигурация отображения
-Start adresse : z
-Acq buffer size : 1
-Display size : 40
-Data type : 16 bits unsigned

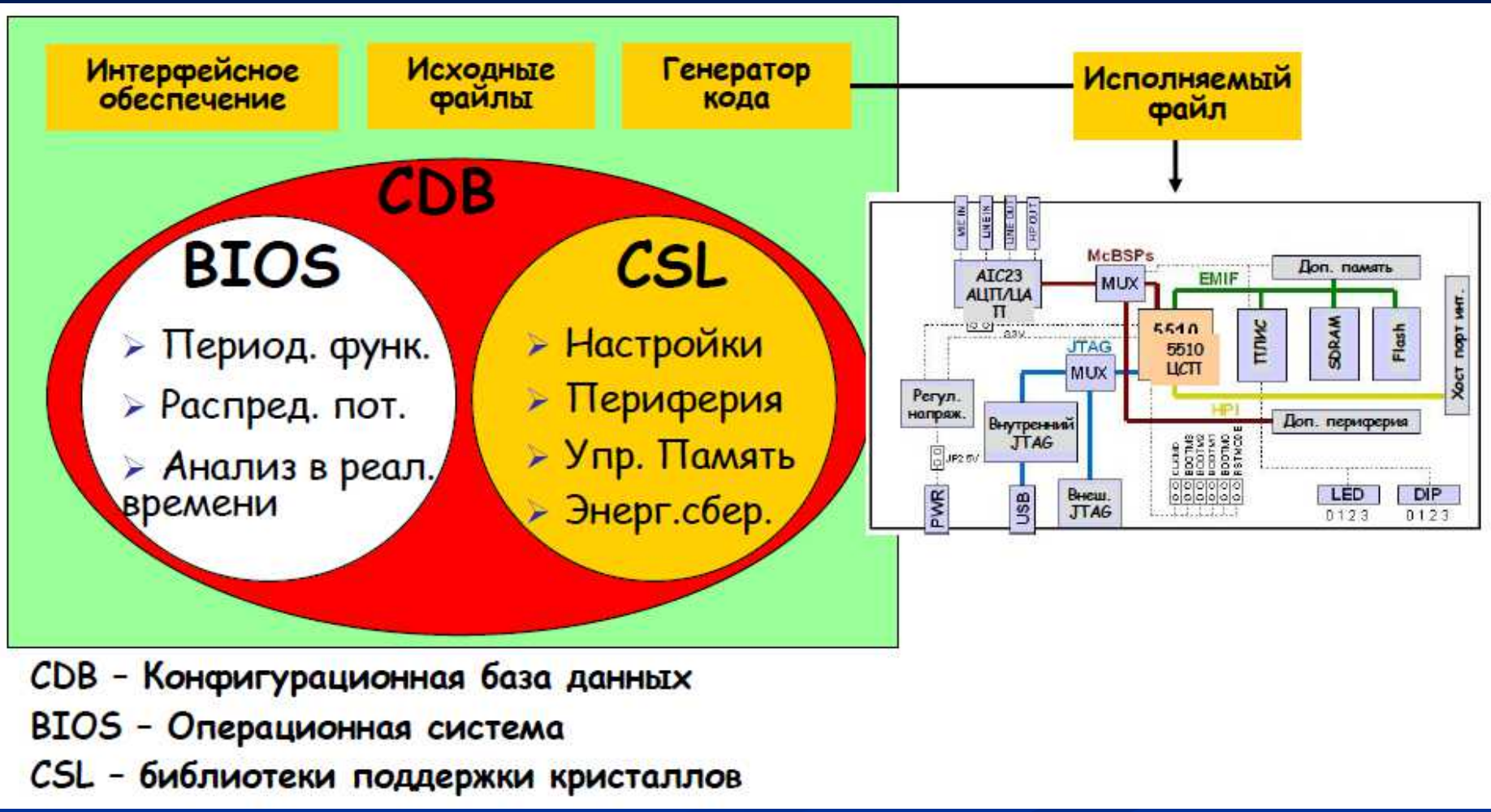


Интегрированная среда программирования Code Composer Studio

- **Динамическое отображение графика:**



Интегрированная среда программирования Code Composer Studio



CDB – Конфигурационная база данных

BIOS – Операционная система

CSL – библиотеки поддержки кристаллов

Интегрированная среда программирования Code Composer Studio

- **Условные обозначения в Ассемблере:**

- Любой печатный ASCII текст разрешен
- Использовать расширение .asm для файлов
- Команды и директивы не могут начинаться с начала строки
- Комментарии к любой строке после точки с запятой

Интегрированная среда программирования Code Composer Studio

- **Построение проекта:**

Пример для обработки : z=x+y

коде

get x
add y
store z

loop

константы

x = 2
y = 7

переменные

z

```
        .text
start:  LD    #X,DP
        LD    @x,A
        ADD   @y,A
        STL   A,*(z)
        B     start
```

```
        .data
x       .int  2
y       .int  7
```

```
        .bss  z,1
```

Интегрированная среда программирования Code Composer Studio

- **Директивы ассемблера и типы данных:**

Основные директивы

.sect	инициализируемая секция для кода и данных
.usect	не инициализируемая секция create для данных
.byte	8-битная константа выравнивание по словам
.int(.word)	16-битная константа
.long	32-битная константа
.ref/.def	используется для определения символов
.global	совместно с.ref и .def combined

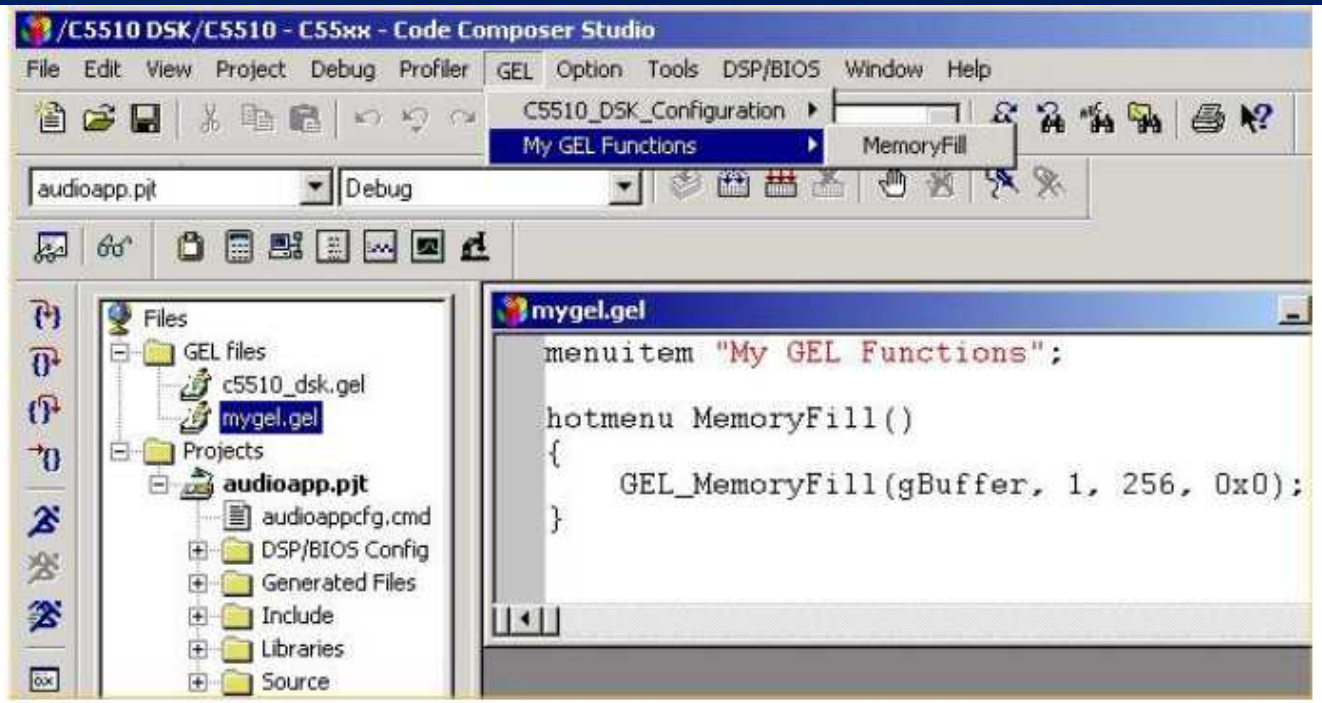
Типы данных

10	десятичный (default)
0Ah, 0xA	шестнадцатеричный
1010b, 1010B	бинарный

.set/.equ	равенство значению или символу
.asg	назначение ассемблерных констант

Интегрированная среда программирования Code Composer Studio

- **Автоматизация при использовании GEL сценариев:**

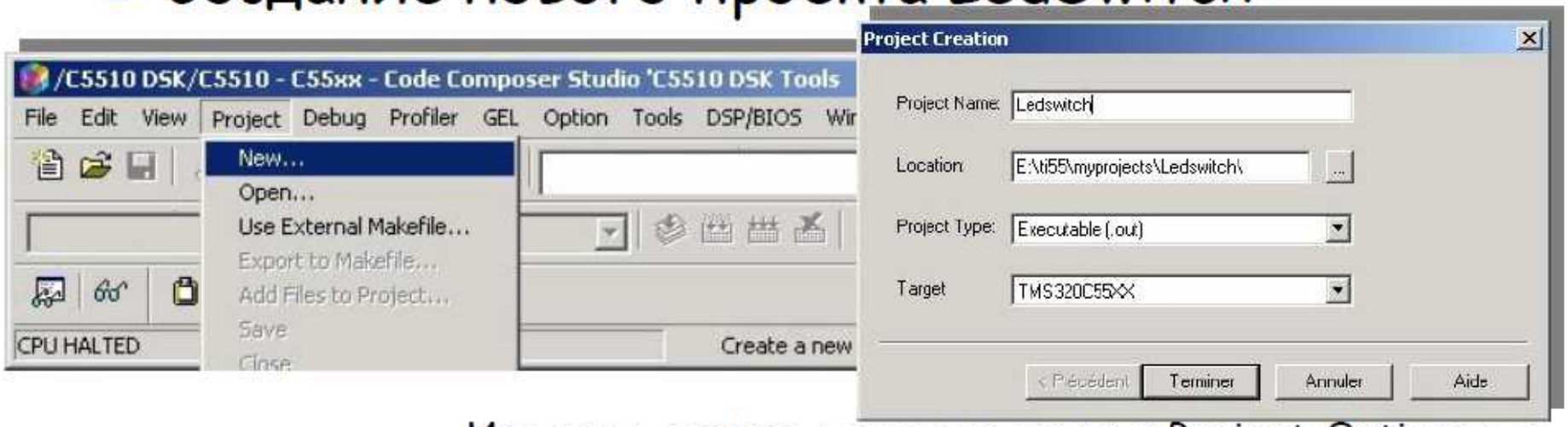


- GEL: General Extension Language
- Синтаксис языка C

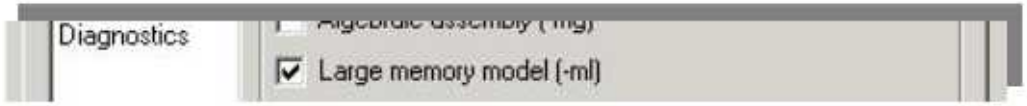
- Большое число команд отладки в виде GEL функций
- Написание собственных функций
- Создание элементов меню GEL

Code Composer Studio. Пример создания проекта

- Создание нового проекта LedSwitch

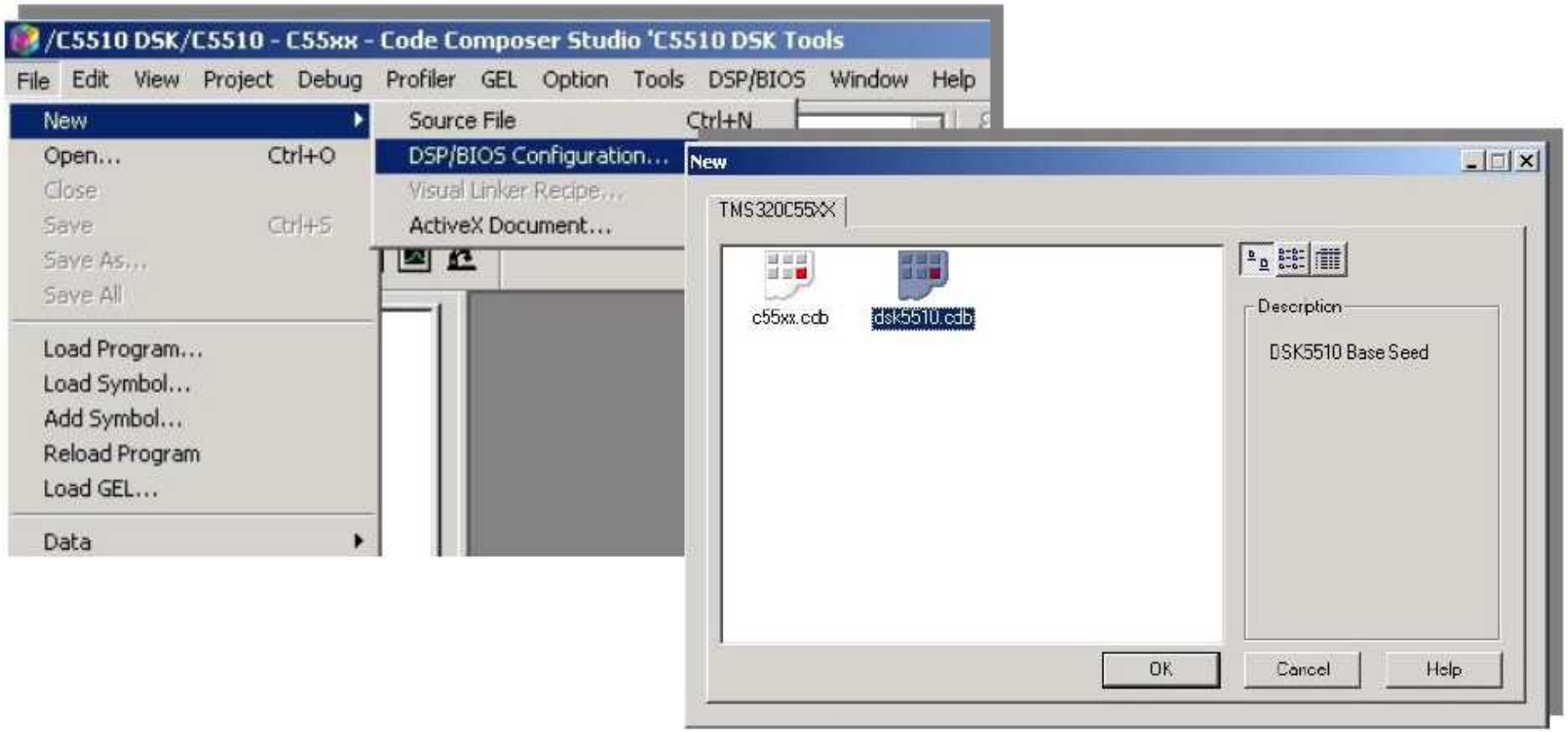


- Изменить модель памяти в меню « Project>Options » и выбрать « large memory model »



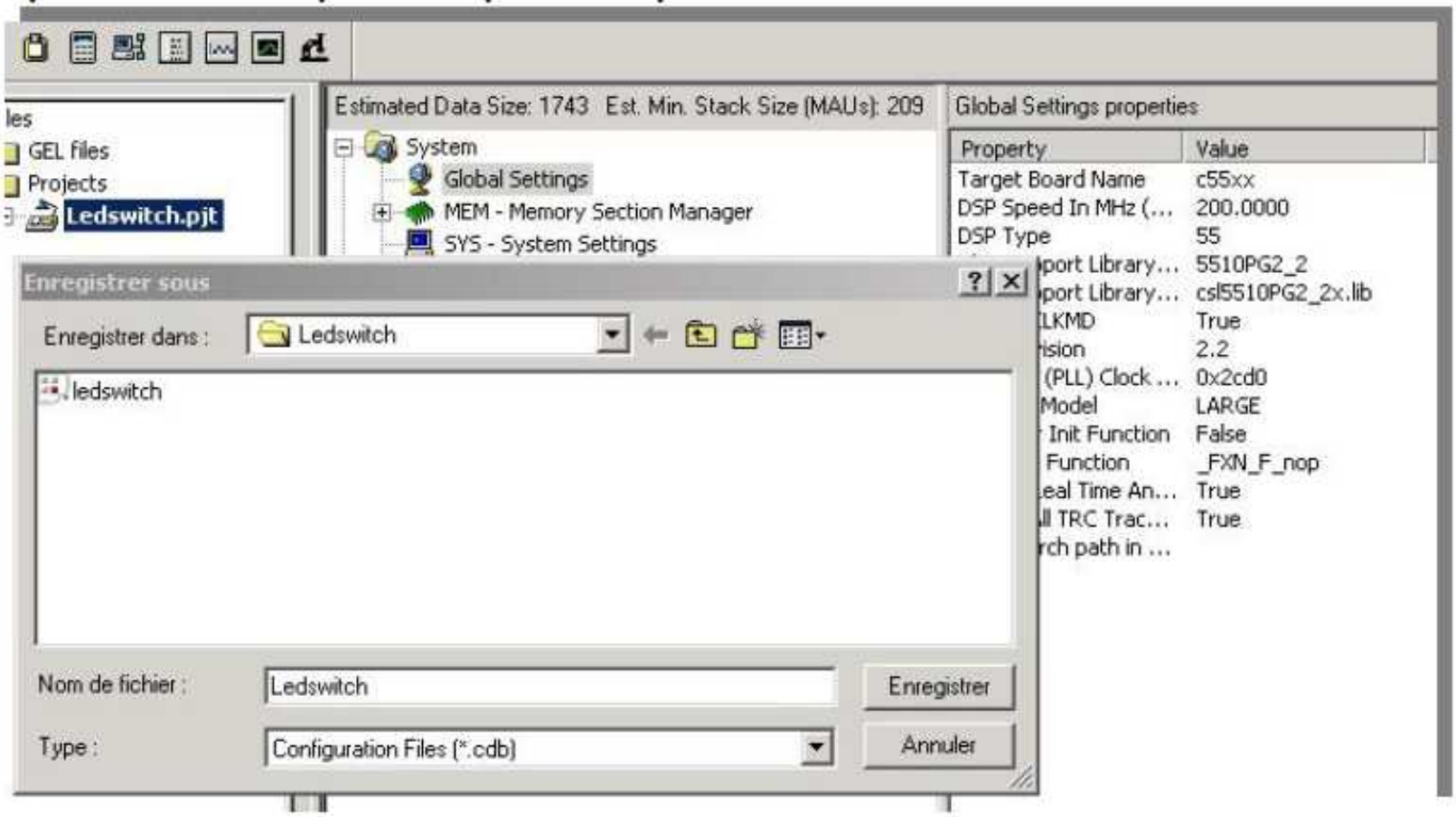
Code Composer Studio. Пример создания проекта

- Выбрать новый конфигурационный файл (*.cdb)



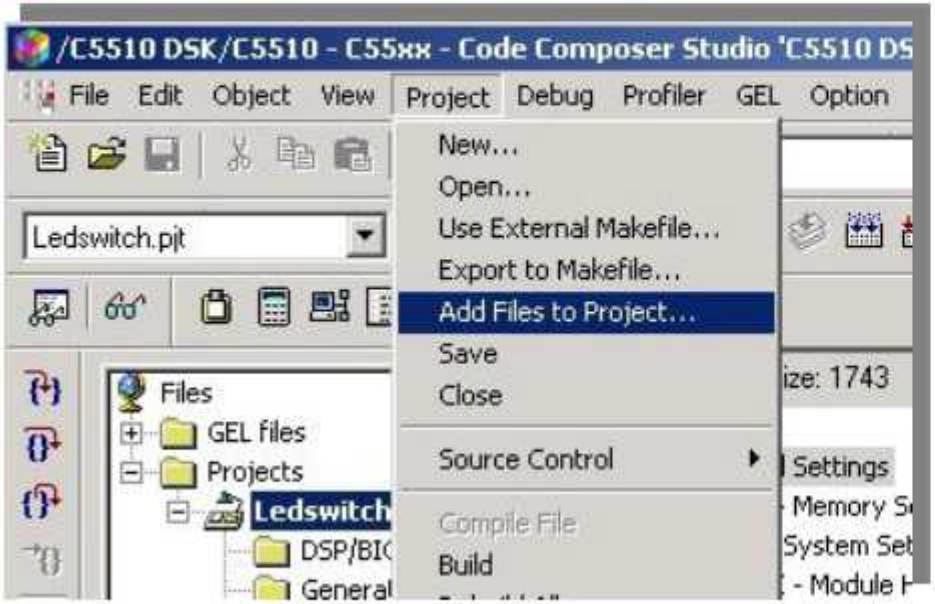
Code Composer Studio. Пример создания проекта

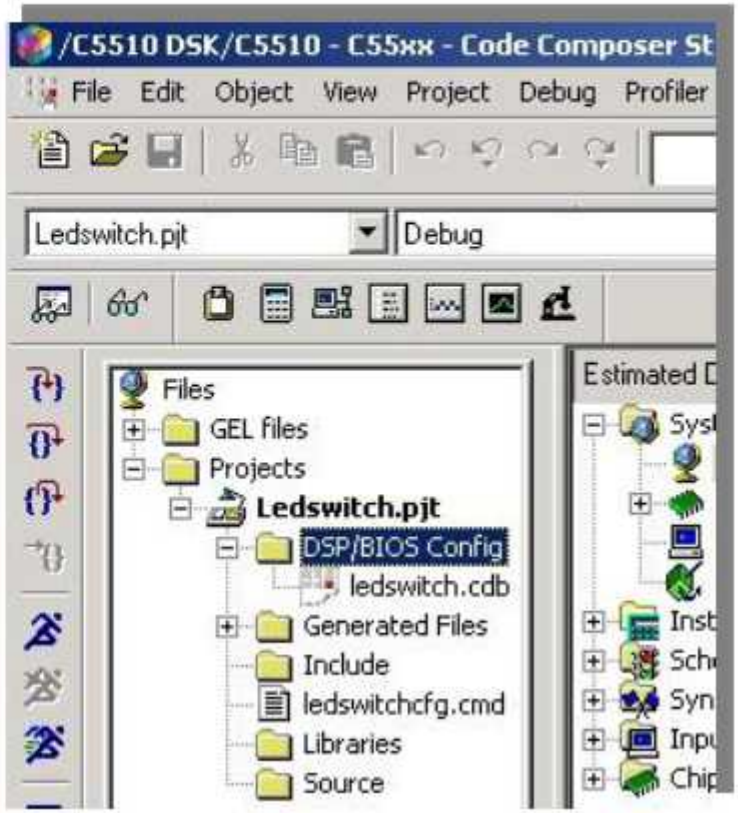
- Сохранить (File>Save) новый конфигурационный файл в директорий проекта



Code Composer Studio. Пример создания проекта

- Добавить в проект два сгенерированных файла: конфигурационный файл (*.cdb) и командный файл компоновки (*.cmd)

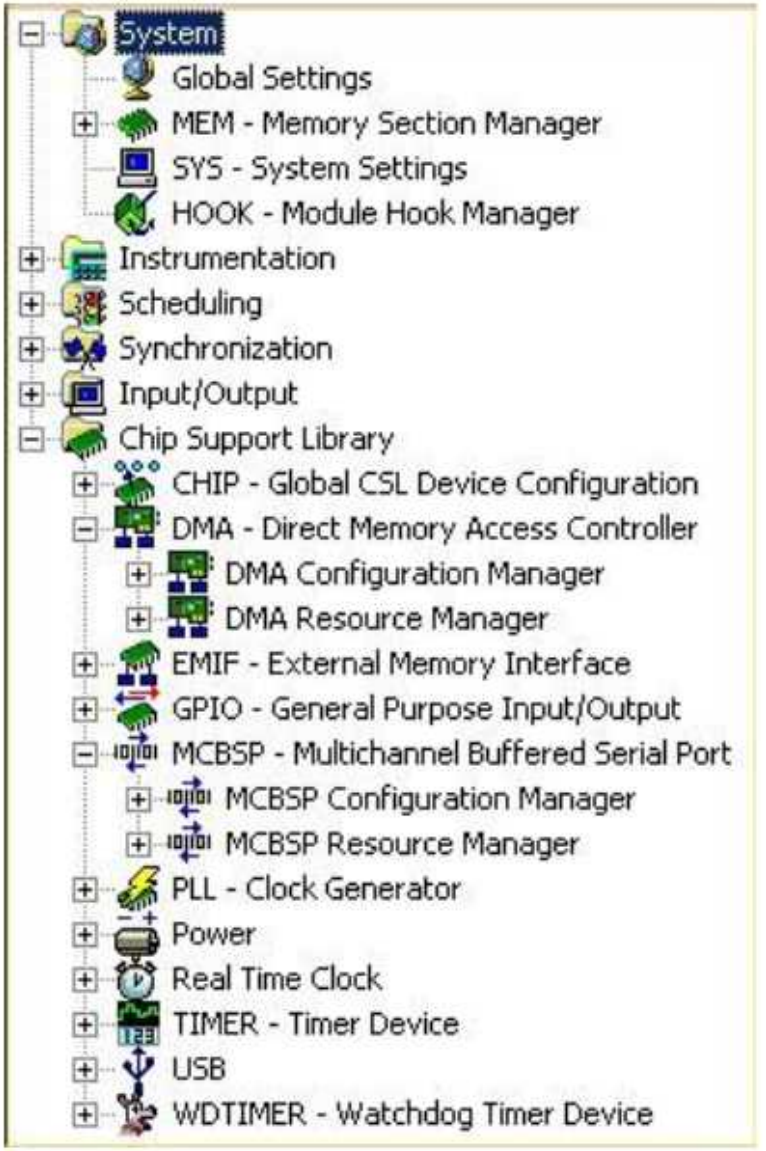




Code Composer Studio. Пример создания проекта

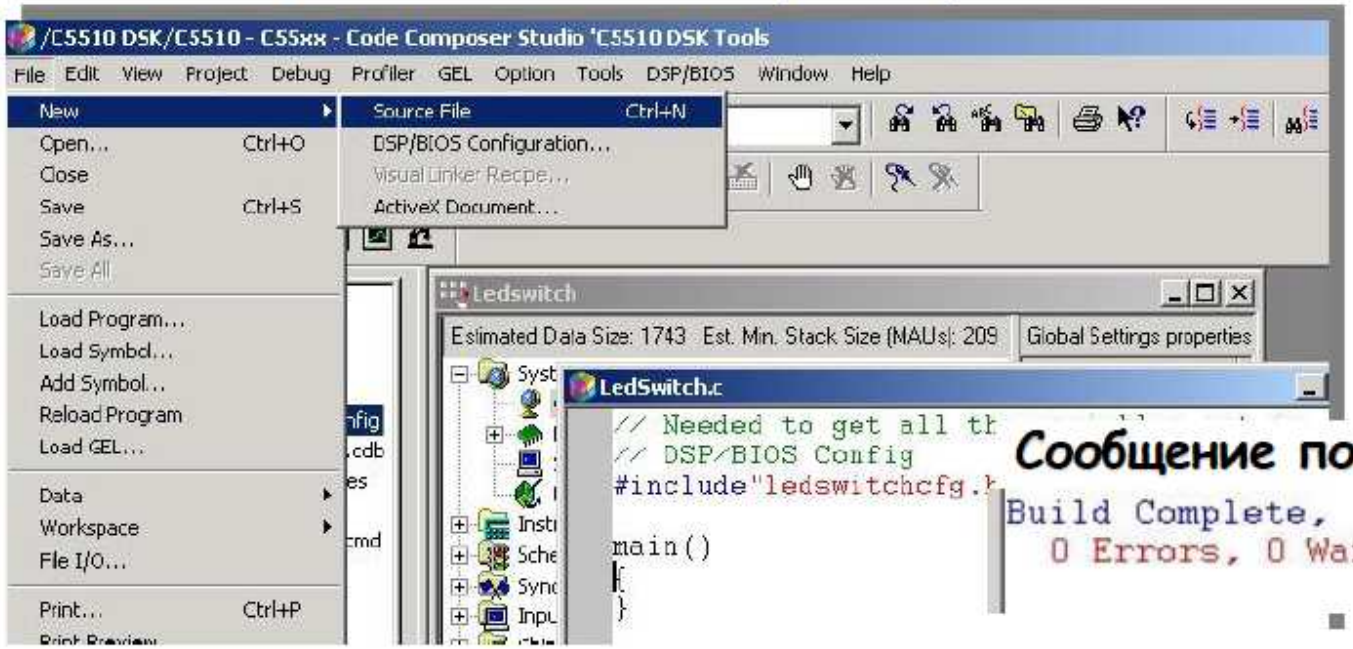
- **CDB – конфигурационная база данных:**

- Настройка диспетчеров BIOS, RTA и других функций BIOS
- Автоматическое управление: библиотеками реального времени, векторами прерываний, системным сбросом и т.д.
- Управление конфигурацией памяти системы (создание CMD файла)
- CSL упрощает настройку периферии



Code Composer Studio. Пример создания проекта

- Создать основной файл: LedSwitch.c, который должен включать сгенерированные заголовочные файлы
- Добавить его к проекту



Сообщение после компиляции …

```
Build Complete,
  0 Errors, 0 Warnings, 0 Remarks.
```

Code Composer Studio. Пример создания проекта