

Подготовка к работе

Глава "Подготовка к работе" написана в предположении, что читатель впервые решил изучить GNAT GPL 2010 for Windows. Если это не так, то можно пропустить весь или частично материал главы 2.

Установка в системе среды разработки и библиотек

Для установки комплекта среды разработки Вам понадобится минимум три файла со всего дистрибутива GNAT GPL 2010 for Windows, собственно библиотека OEM и файл архива примеров для этой книги – файл для установки собственно среды разработки (gnat-gpl-2010-i686-pc-mingw32-bin.exe 68514 Kbyte 08.06.2010 18:31), файл для установки библиотеки (win32ada-gpl-2010.exe 2606 Kbyte 19.01.2010 18:01), файл утилиты MAKE (gnumake-3.79.1-pentium-mingw32msv.exe 122 Kbyte 09.01.2006 19:22), файл архива библиотеки OEM (oem2010.zip 4815 Kbyte 15.07.2010 11:14), файл утилит и тестов (oemtools2010.zip 17764 Kbyte 15.07.2010 11:14) и файл архива примеров (oemtutorials-gui2010.zip 8796 Kbyte 15.07.2010 11:15). Естественно здесь указаны самые старые версии файлов. Весь полный дистрибутив GNAT GPL 2010 for Windows, имеет размер более 210 Mbyte. Дистрибутивы можно получить и по интернету с сайтов <http://libre.adacore.com> и <http://www.mediascan.by>, соответственно GNAT GPL 2010 for Windows и библиотеки OEM.

В первую очередь устанавливается дистрибутив GNAT GPL 2010 for Windows. Для этого просто запускается на выполнение файл "gnat-gpl-2010-i686-pc-mingw32-bin.exe" и далее следовать инструкции "мастера" установки. Необходимо установить среду разработки на диск "C:\". В результате на диске появится каталог GNAT с подкаталогом 2010 ("C:\GNAT\2010\"). Этот подкаталог является корнем для всех остальных.

Скопируйте файл "gnumake-3.79.1-pentium-mingw32msv.exe" в подкаталог "C:\GNAT\2010\bin\" под именем "make.exe".

У Вас должны быть права локального администратора. Сейчас желательно перезагрузить ОС, в этом случае гарантированно войдут в действия пути поиска программ назначенные при установке дистрибутива GNAT GPL 2010 for Windows.

Следующим шагом можно устанавливать библиотеки и другие ресурсы дистрибутива.

После установки GNAT GPL 2010 for Windows необходимо установить библиотеку "win32ada". Для этого запускается на выполнение файл "win32ada-gpl-2010.exe" и далее следовать инструкции "мастера" установки.

Теперь всё готово для установки библиотеки OEM. Выберите диск и каталог для установки. Например: "C:\Root305\ProjKA\2010\". Разархивируйте в него файл oem2010.zip, oemtools2010.zip и файл oemtutorials-gui2010.zip. Должны появиться три подкаталога: "C:\Root305\ProjKA\2010\oem2010\", "C:\Root305\ProjKA\2010\oemtools2010\" и "C:\Root305\ProjKA\2010\oemtutorials-gui2010\". Запустите скрипт "install.cmd" на выполнение из каталога oem\. После выполнения скрипта библиотека OEM установлена.

Первый запуск графической интегрированной среды разработки GPS

Зарегистрируйтесь в ОС под именем под которым Вы будете работать в GPS. Ваше имя должно быть обязательно на латинице. Это необходимо для формирования файлов состояния GPS в Вашем профиле ОС. GPS не распознает русскую кодировку в именах файлов и каталогов (это касается текущей версии GPS).

Запустите на выполнение файл GPS.EXE через меню или через иконку на рабочем столе графической консоли (рисунки приведены для версии GPS входящей в состав GNAT GPL 2009 for Windows, в данном случае это не влияет на понимание материала).

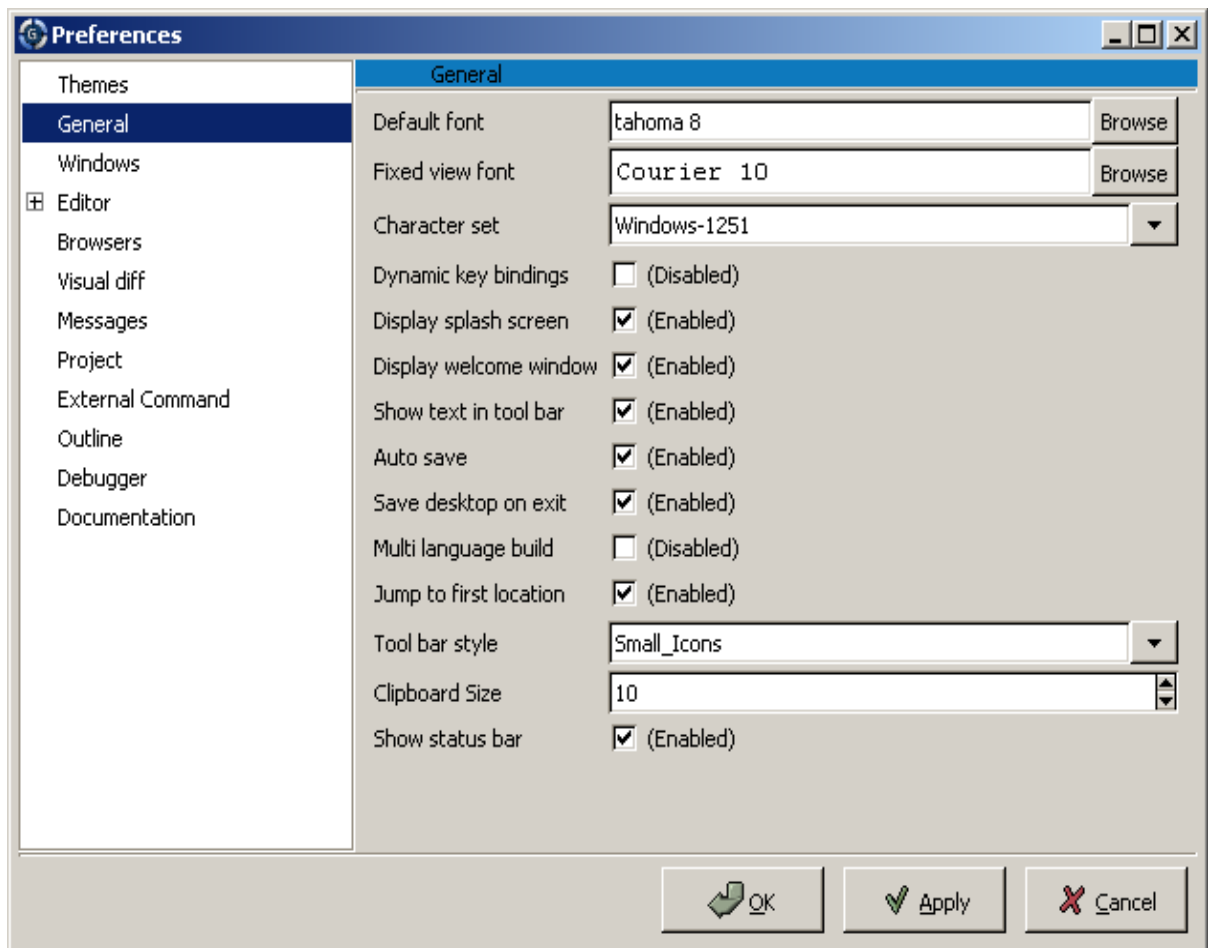


Рис. 3 Наберите вручную Windows-1251 в поле ввода Character set.

Наберите вручную Windows-1251 в поле ввода **Character set** при первом запуске GPS как показано на Рис 3 и выйдите из программы. После этого можно применять эту русскоязычную кодировку в комментариях и в строках.

Создаем первый проект

Вначале сразу желательно сразу определиться: где на диске вы будете создавать проект. Автор предпочитает для этой цели использовать отдельный подкаталог, который создается заранее и имеет смысловое значение определяющее время его создания по годам. Например: C:\Root305\ProjKA\2010\ – проекты созданные в 2010 году. Ещё раз хочется обратить внимание на то, что все имена файлов и каталогов должны быть на латинице. Создадим каталог проекта и подкаталог для промежуточных файлов трансляции Рис. 4.

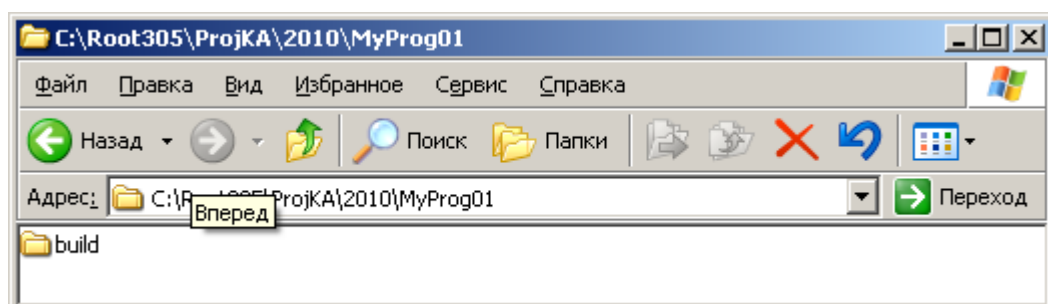


Рис. 4 Создадим каталог проекта

Запустим на выполнение GPS, должно появиться окно диалога Рис.5.

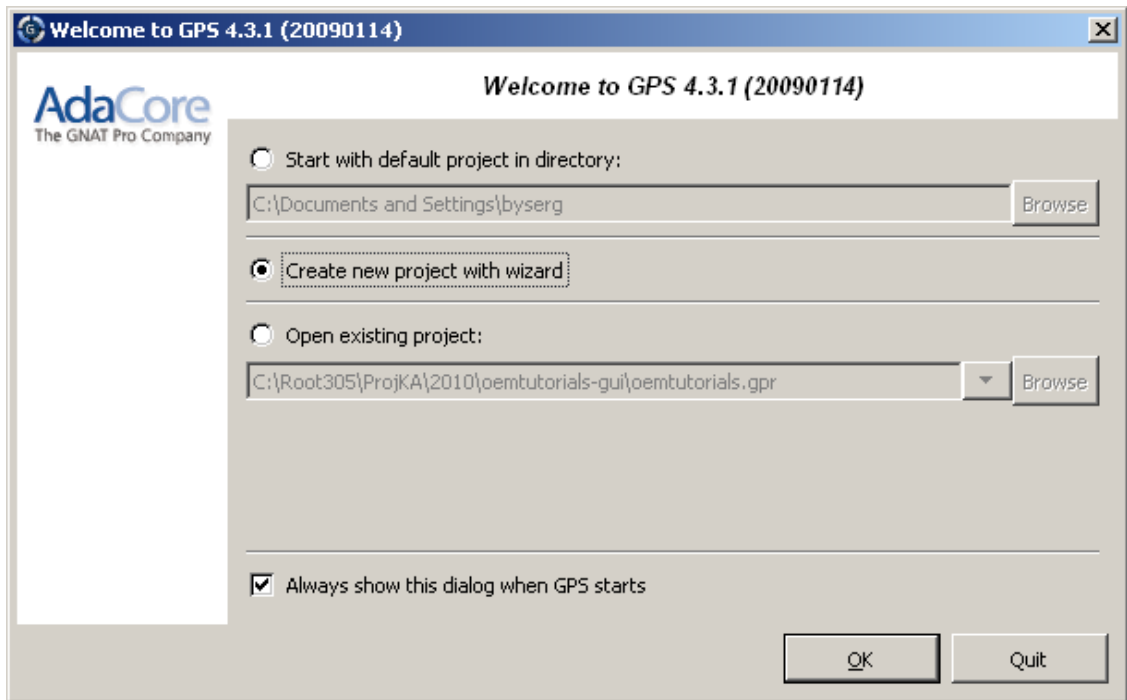


Рис. 5. Окно приветствия программы GPS

Выберем **Create new project with wizard** и нажмем кнопку **OK**. В следующем диалоге оставим всё не изменяя и нажмем кнопку **Forward** Рис. 6.

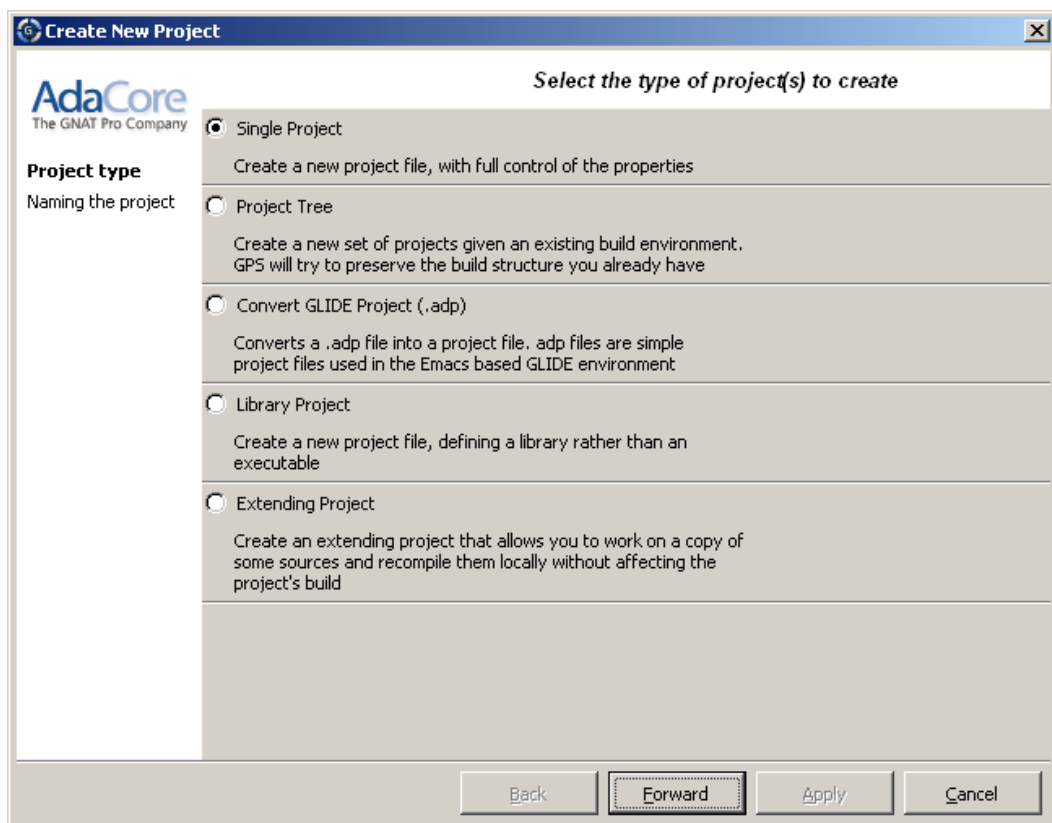


Рис. 6. Создадим Single Project

В следующем диалоговом окне определим имя проекта (имя проекта должно как правило совпадать с именем главной процедуры и файла где эта процедура будет описана), место расположения файла проекта Рис. 7.

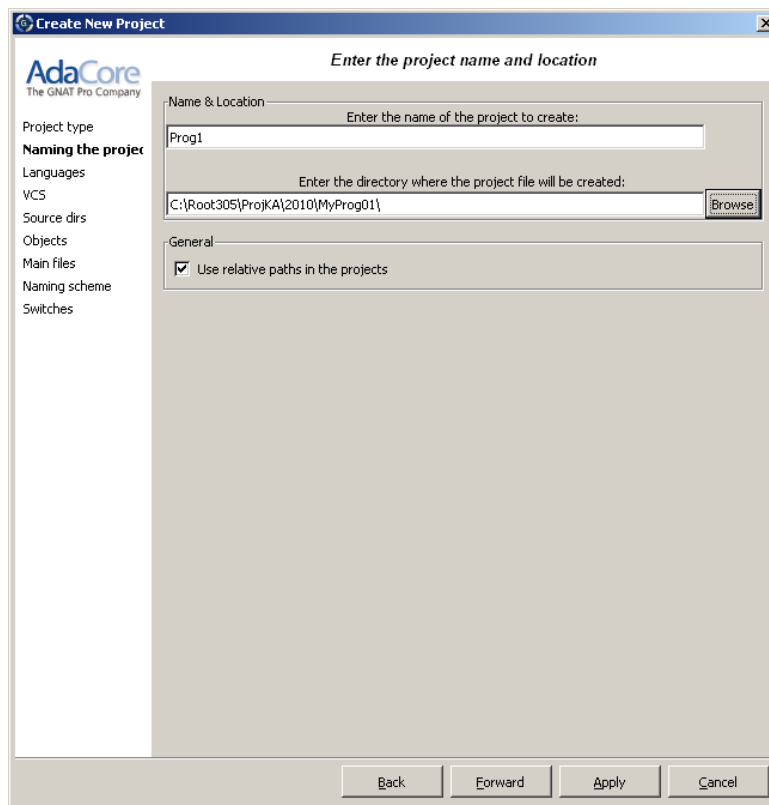


Рис. 7. Диалог выбора имени проекта и место его расположения

Кнопка **Browse** позволяет выбрать директорий проекта через диалоговое окно Рис. 8.

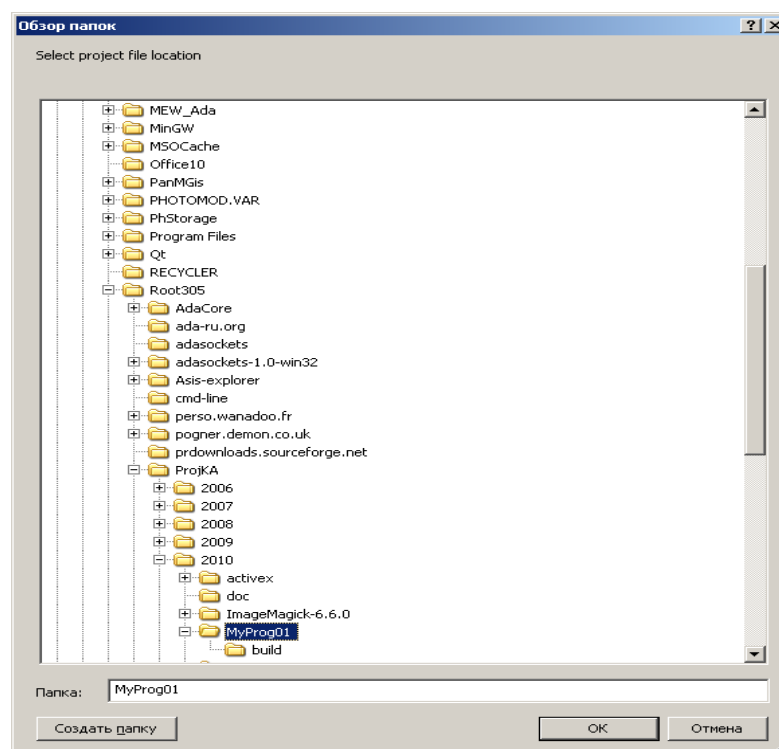


Рис. 8. Диалог выбора или создания директория

Нажмем кнопку **Forward** Рис. 7. В следующем диалоговом окне оставим всё без изменения и нажмем кнопку **Forward** Рис. 8.

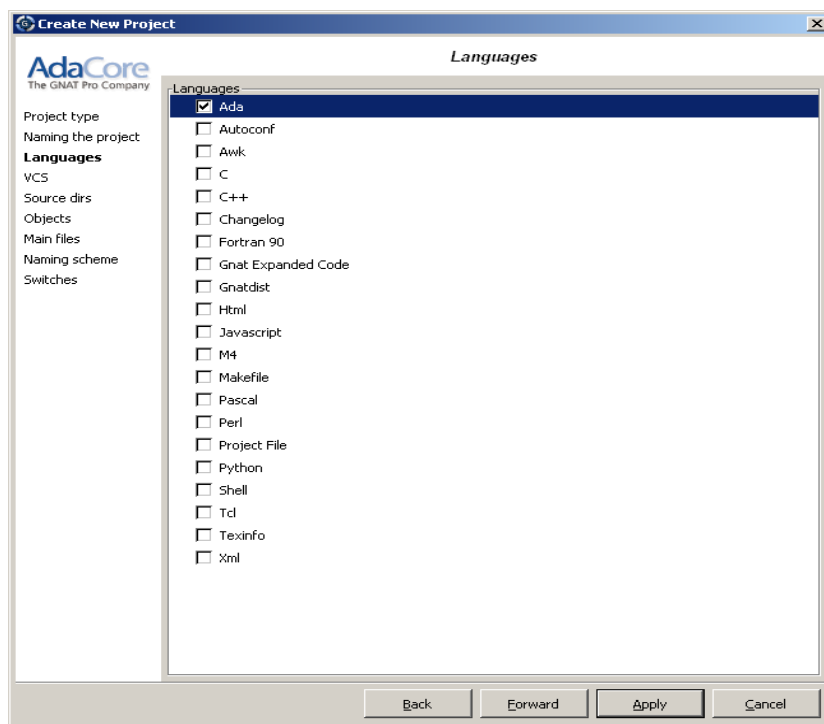


Рис. 9. Диалог выбора языков программирования в проекте

Следующих два окна диалогов также оставляем без изменения, последовательно нажимая кнопку **Forward**. Необходимо только убедиться в правильности выбора каталога размещения файлов программы (C:\Root305\ProjKA\2010\MyProg01) Рис. 10.

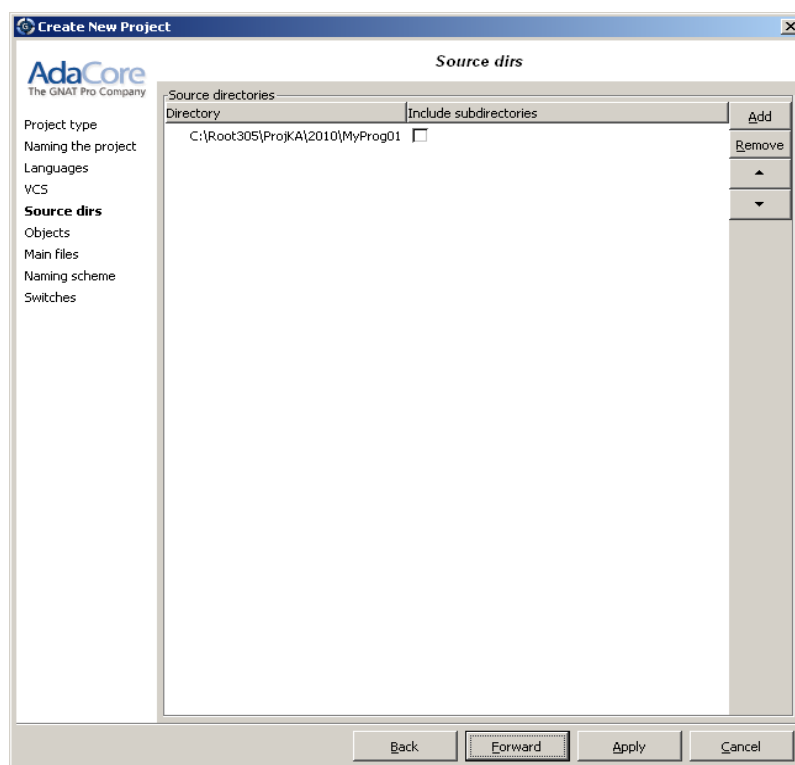


Рис. 10. Диалог выбора каталогов файлов программ.

На следующем шаге необходимо определить каталог для файлов трансляции и каталог для исполнения программы. Для этого, используя кнопку **Browse**, лучше переопределить каталоги по умолчанию Рис. 11,

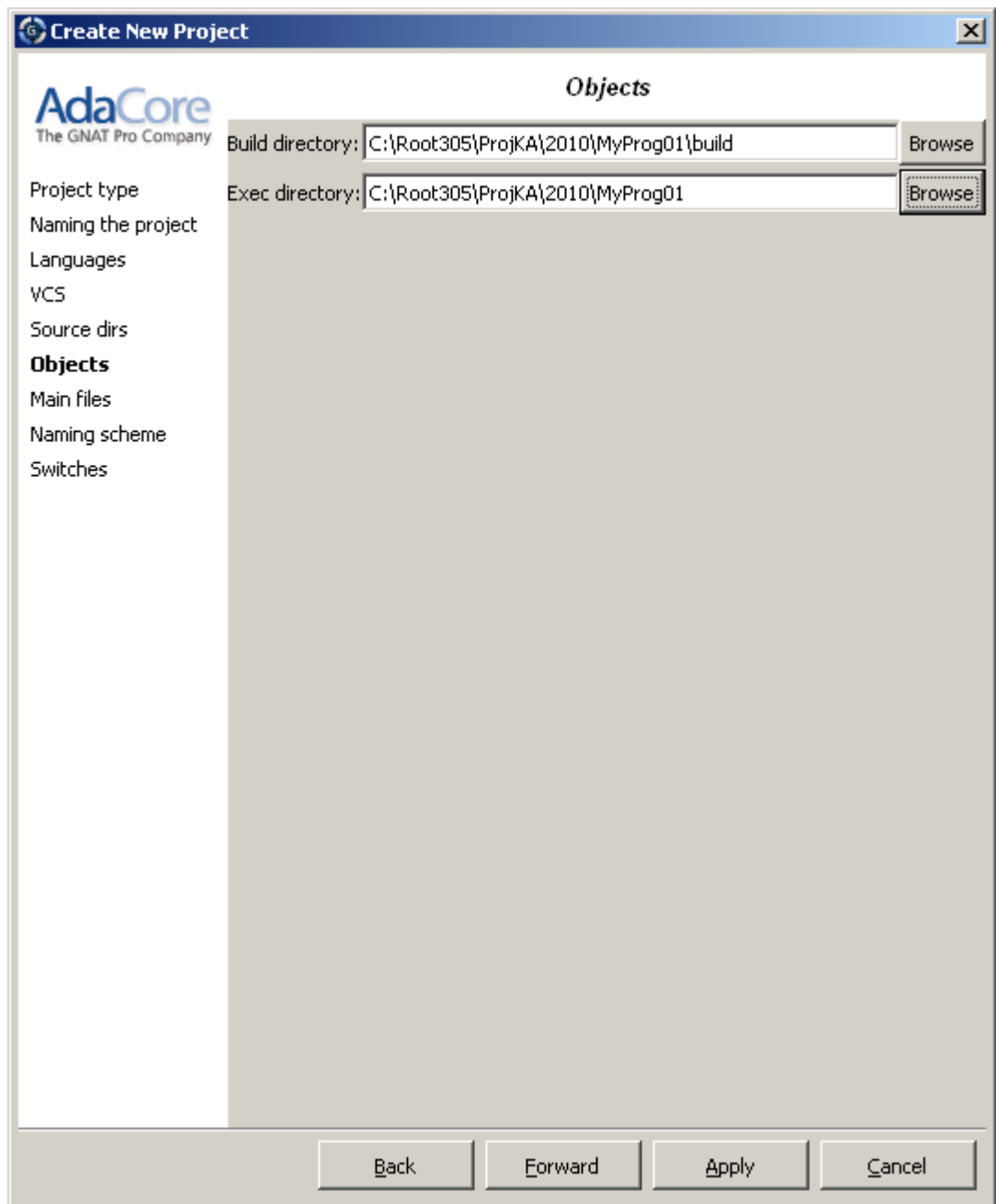


Рис. 11. Определим каталог Build и Exec

В следующем диалоговом окне определяем главный файл проекта. Для этого в диалоге **Main Files** нажимаем кнопку **Add**. В появившемся диалоге **Select a File** в поле **Имя файла** вводим с клавиатуры `Prog1.adb`. Рис. 12. Нажимаем кнопку **открыть** – должны получить результат как на Рис. 13. Необходимо понимать, что на данном этапе мы прописали только имя файла в проекте. Физически сам файл `Prog1.adb` не создан и в нашем случае не существует.

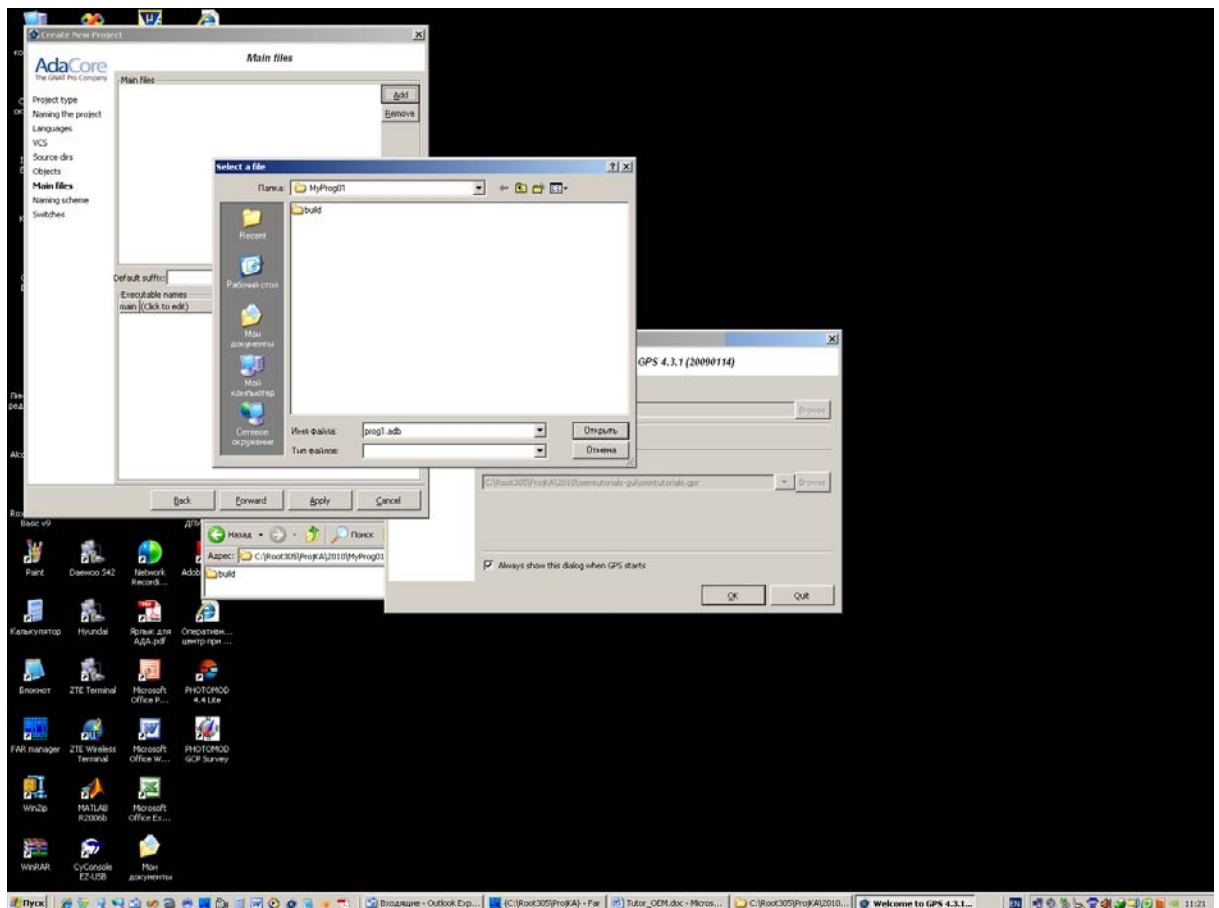


Рис. 12. Определение имени главного файла программы в проекте

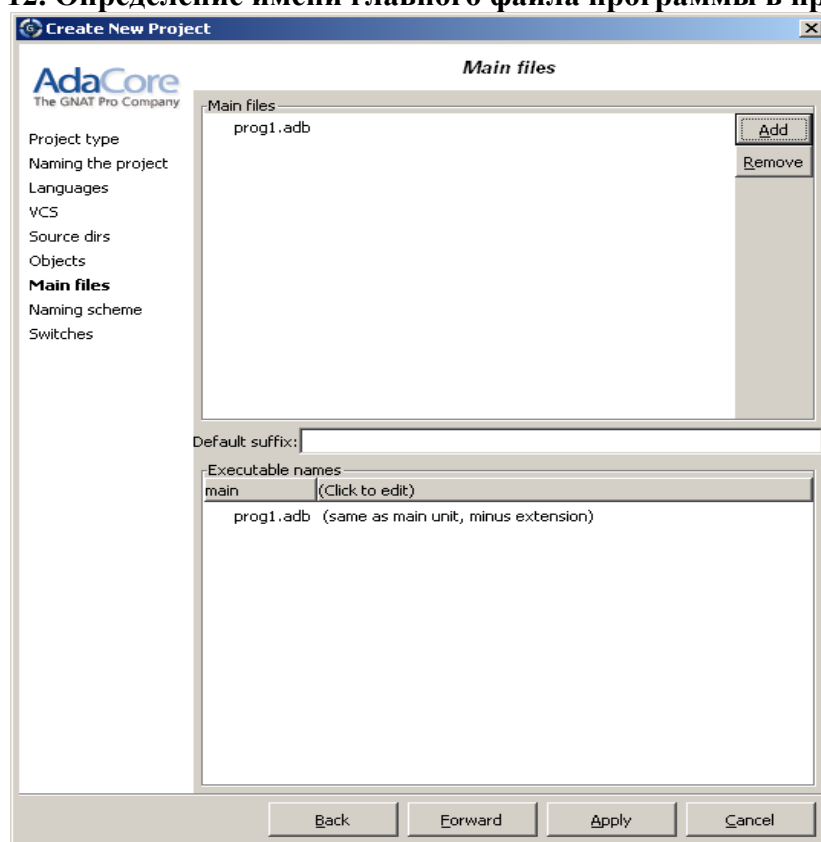


Рис. 13. Диалог определения главного файла программы в проекте

Следующее диалоговое окно определяет соглашения о именовании файлов программы для выбранного языка программирования и пакетов которые в неё входят. Оставляем всё по

умолчанию и переходим к следующему завершающему диалогу. На данном этапе можно тоже оставить всё по умолчанию, поэтому нажимаем кнопку **Apply**.

Проект создан и загружен в GPS. В окне Message одно предупреждение об отсутствии файла программы на языке Ada (красным цветом) Рис.14.

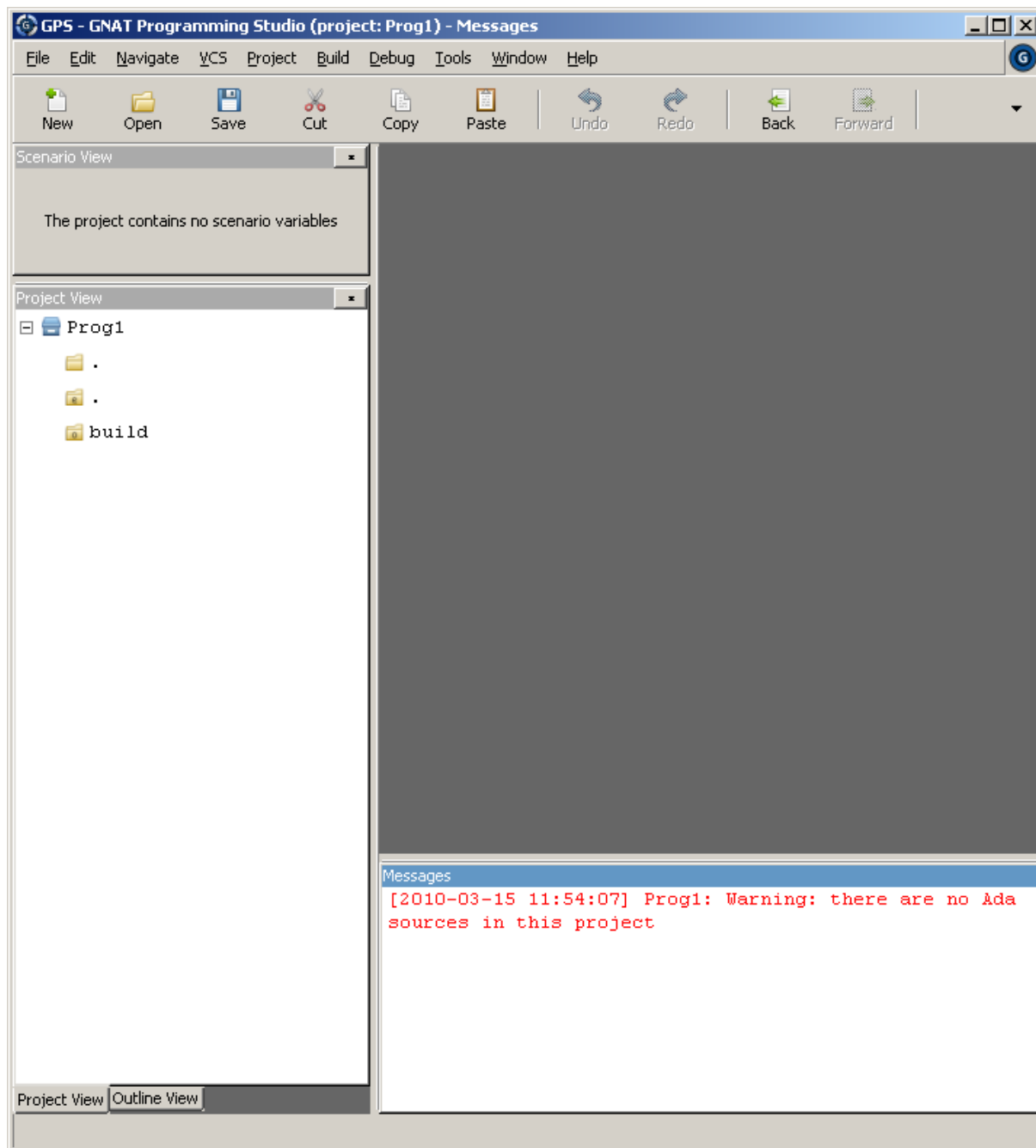


Рис. 14. Окно GNAT Programming Studio с предупреждением об отсутствии файла исходного текста программы в каталоге назначенном в проекте

Создадим файл программы, для этого нажмем последовательно кнопку New и Save. В поле **Имя файла** набираем Prog1.adb и нажимаем кнопку **Сохранить**, Рис 15.

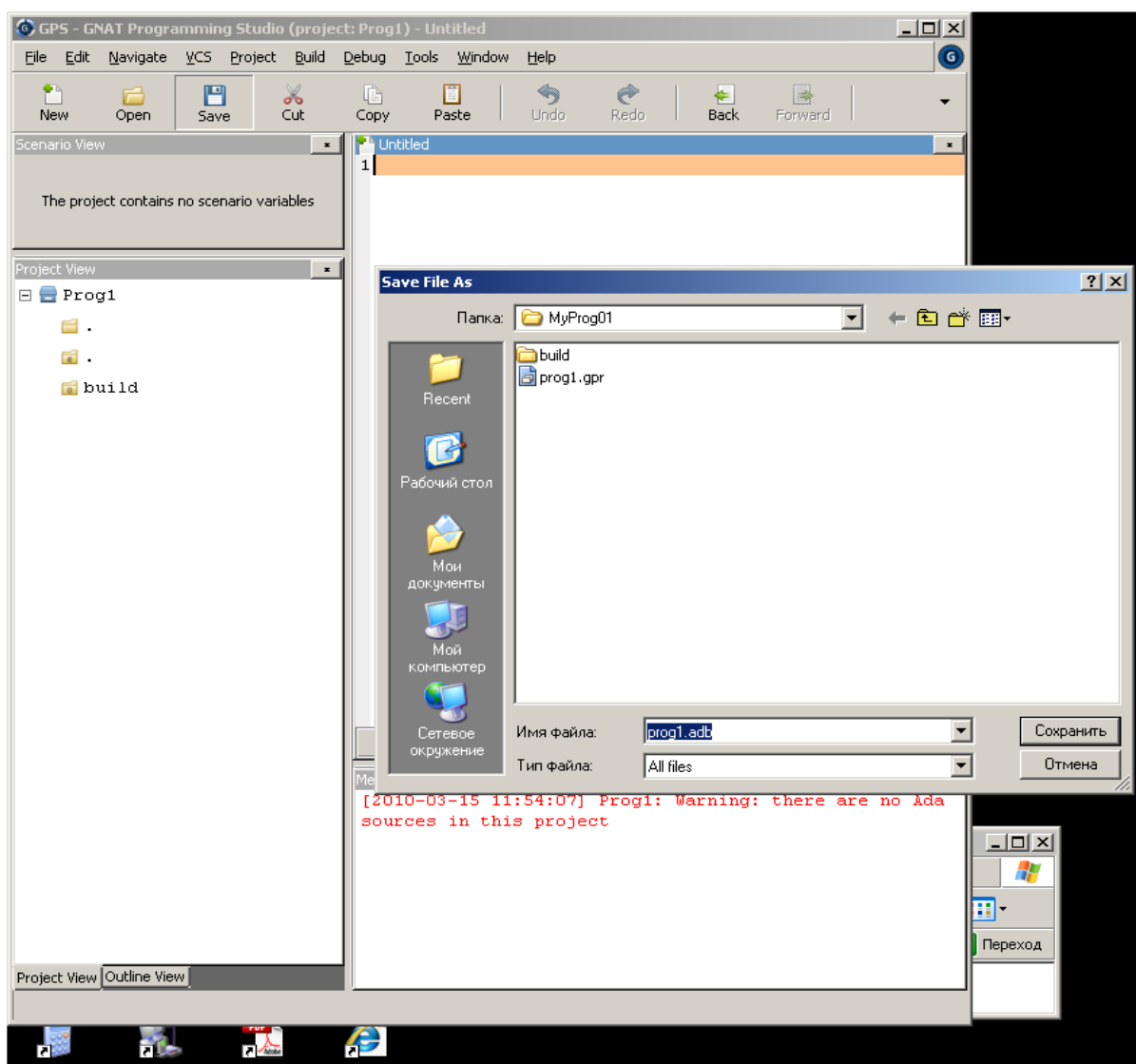


Рис. 15. Создание нового файла в проекте

Наберем текст программы в соответствующем окне.
 With Ada.Text_IO;

```

procedure prog1 is
begin
  Ada.Text_IO.Put_Line("My first program");
end prog1;
```

Нажмем кнопку **Build Main**. В результате мы получим готовую программу к запуску Prog1.exe Рис. 16. и Рис. 17.

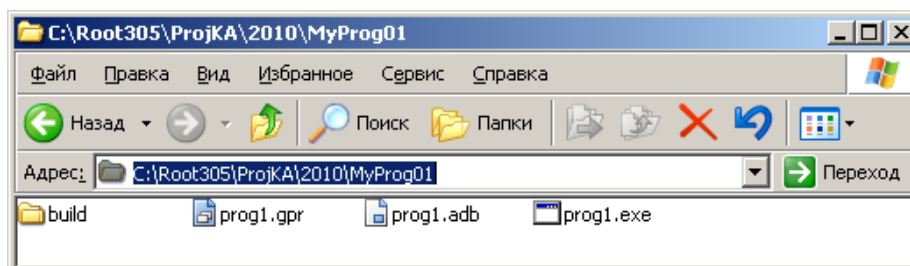


Рис. 16. Каталог проекта после трансляции

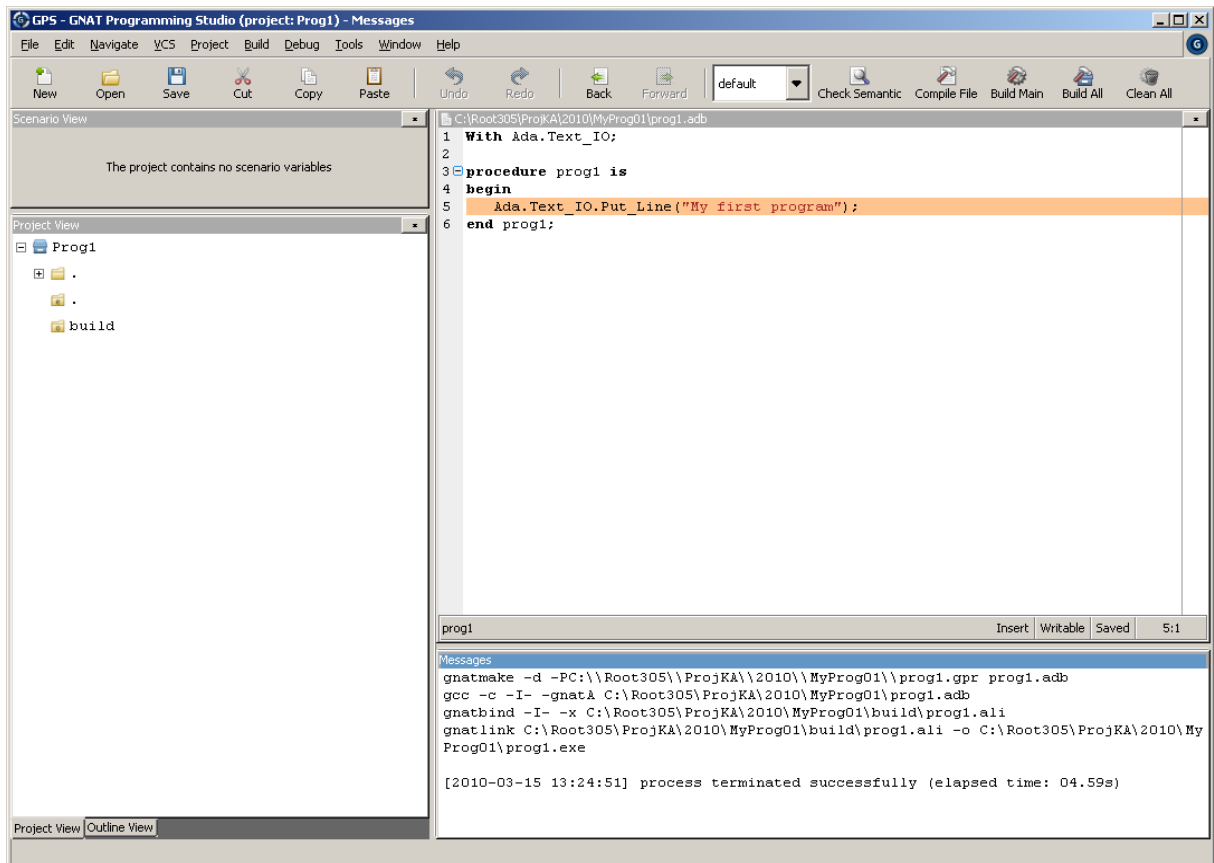


Рис. 17. Окно GPS после успешной трансляции программы

Запустим стандартную консоль CMD.EXE, перейдем в каталог проекта и запускаем программу Prog1.exe. Результат выполнения программы показан на Рис. 18.

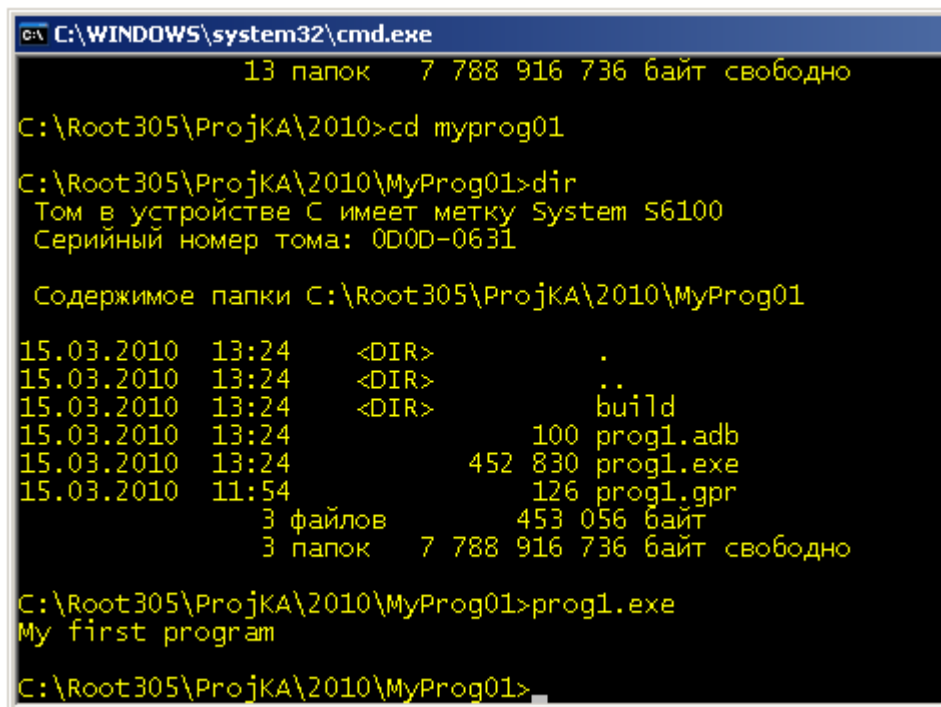


Рис. 18. Программа печатает на стандартной консоли "My first program"

Подключение библиотеки OEM

К существующему проекту подключить установленную библиотеку OEM достаточно просто. Откроем проект в режиме редактирования текста Рис. 19 и Рис.20.

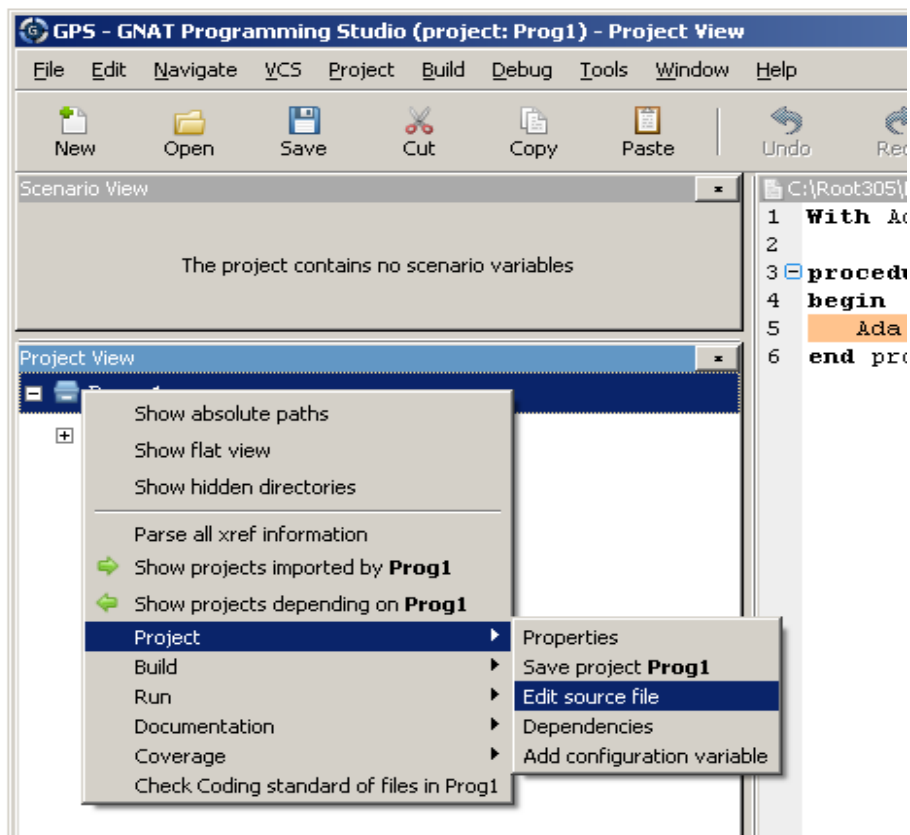


Рис. 19. Открытие файла проекта в режиме редактирования текста

Контекстное меню проекта вызывается путем нажатия правой клавиши мыши. Наберем в первой строке: *with "OEM";* .

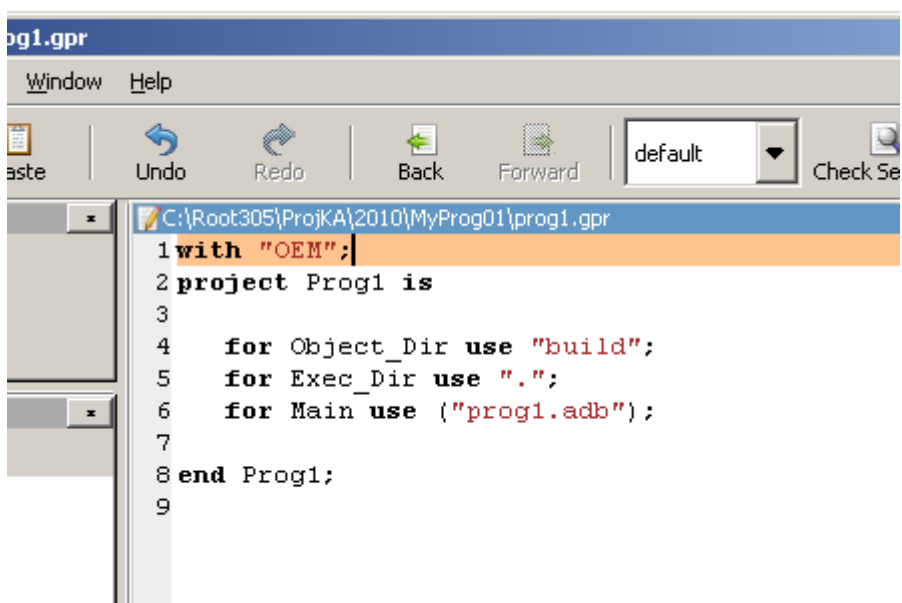


Рис. 20. Подключение в проект библиотеки OEM

Затем сохраняем файл проекта и перезагружаем его Рис. 21.

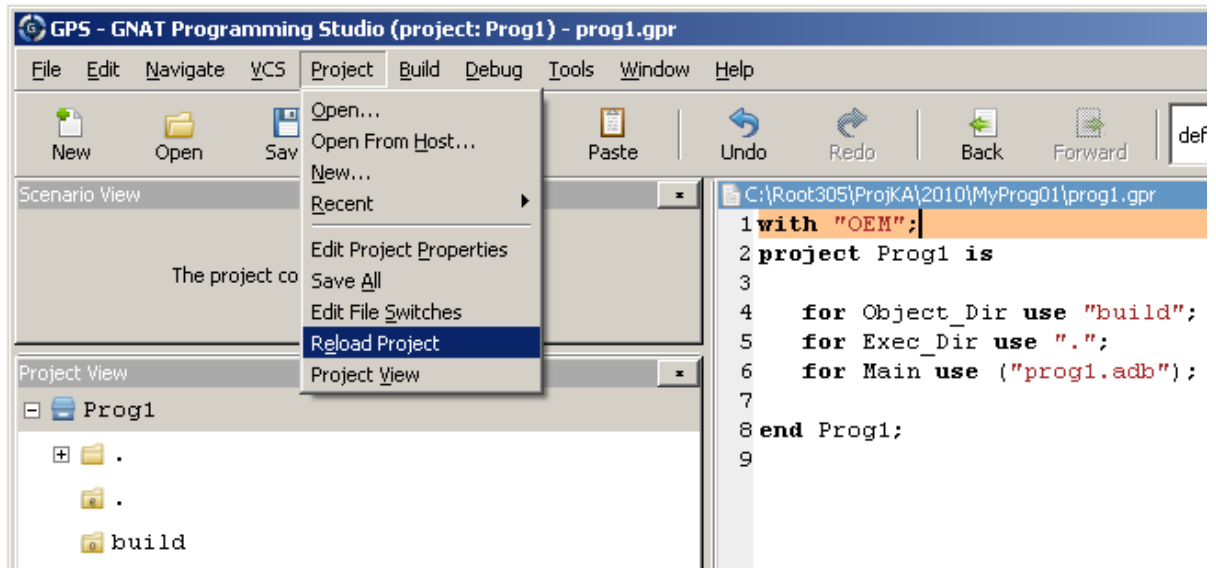


Рис. 21. Перегрузка файла проекта через меню GPS

Файл проекта в редакторе GPS можно закрыть. Теперь в нашей программе можно использовать все пакеты библиотеки OEM. Рис. 22. показывает дерево проектов во вкладке Project View программы GPS.

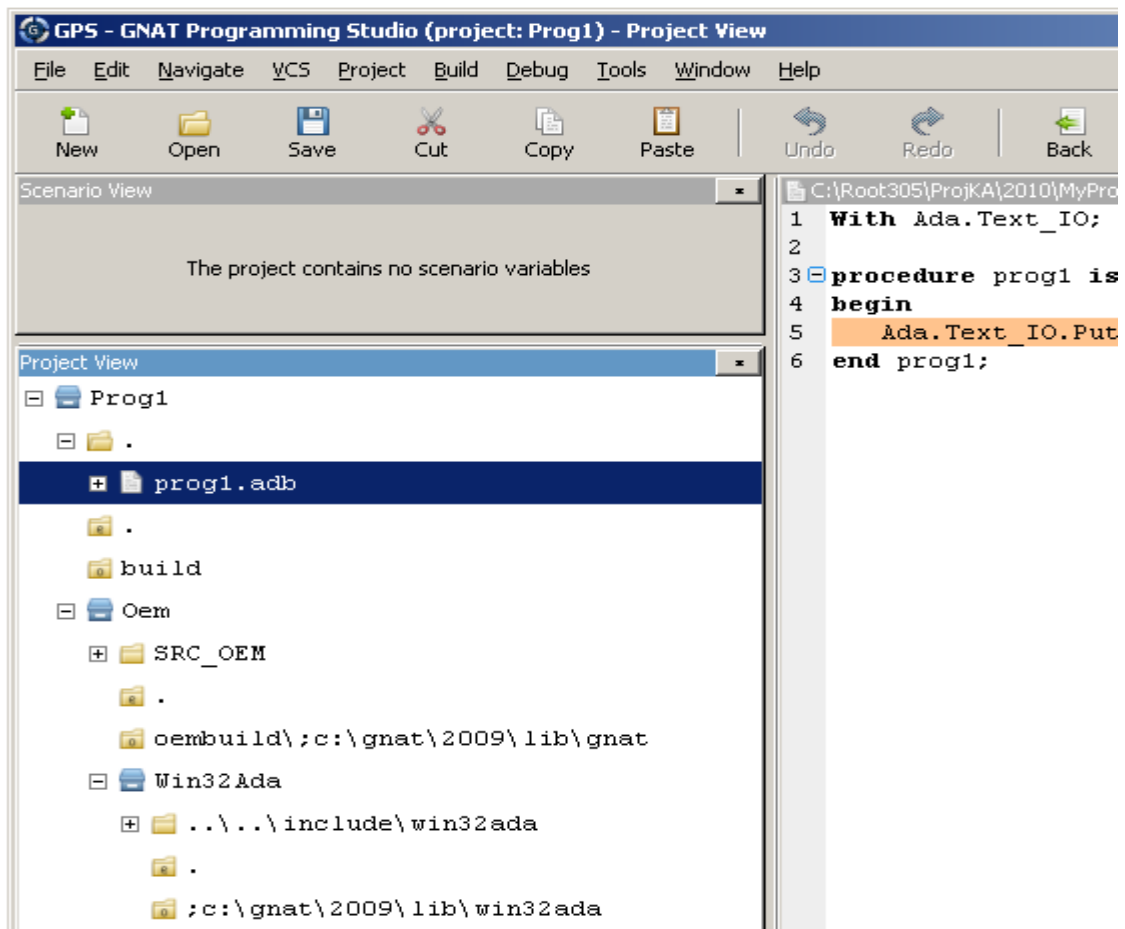


Рис. 22. Дерево проекта Prog1 с подключенной библиотекой OEM

Выводим сообщения на русском языке на консоль.

Необходимо убедиться что в GPS установлена нужная кодировка. Для этого вызываем через меню диалог **Preferences** Рис. 23. и убеждаемся что кодировка Windows-1251 установлена Рис. 3.

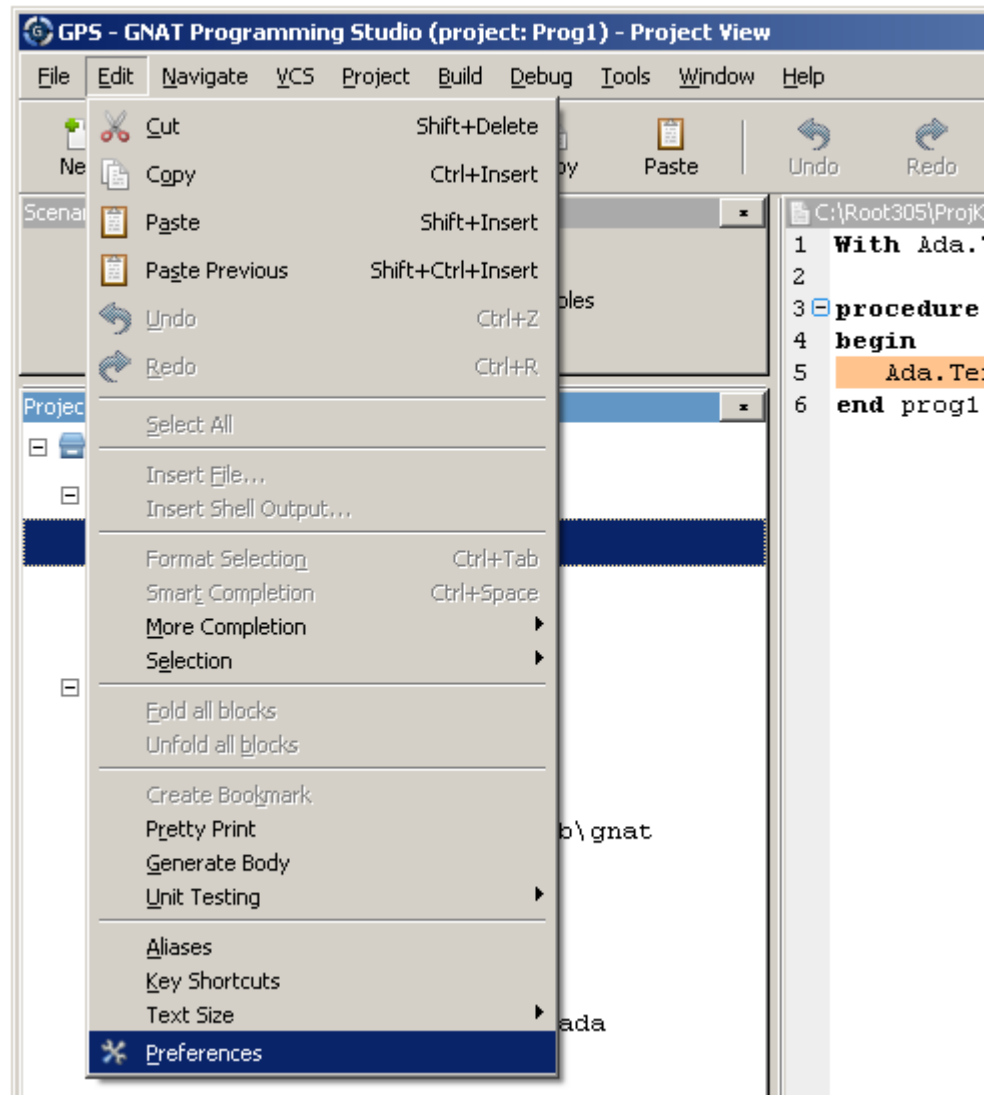


Рис. 23. Вызов из меню диалога Preferences

Если кодировка установлена, то можно смело переключаться на русскую раскладку и модифицировать текст программы. Нажимаем кнопку Build Main и запускаем её в консоли на выполнение Рис. 24., но прочитать сообщение невозможно.

```
With Ada.Text_IO;  
  
procedure prog1 is  
begin  
  Ada.Text_IO.Put_Line("My first program - Это моя первая программа");  
end prog1;
```

```

C:\WINDOWS\system32\cmd.exe

C:\Root305\ProjKA\2010\MyProg01>dir
Том в устройстве C имеет метку System S6100
Серийный номер тома: 0D0D-0631

Содержимое папки C:\Root305\ProjKA\2010\MyProg01

15.03.2010  13:24    <DIR>          .
15.03.2010  13:24    <DIR>          ..
15.03.2010  13:24    <DIR>          build
15.03.2010  13:24                100 prog1.adb
15.03.2010  13:24           452 830 prog1.exe
15.03.2010  11:54           126 prog1.gpr
                3 файлов           453 056 байт
                3 папок           7 788 916 736 байт свободно

C:\Root305\ProjKA\2010\MyProg01>prog1.exe
My first program

C:\Root305\ProjKA\2010\MyProg01>prog1.exe
My first program - !Ю ьр яхЕтр яЁюЕрььр

C:\Root305\ProjKA\2010\MyProg01>

```

Рис. 24. Видно что прочитать сообщение на консоле невозможно

Дело в том что консоль работает в режиме совместимости с DOS программами и использует другую - OEM кодировку (иногда её ещё называют "ДОСовской") в отличии от "Графической КОНСОЛИ" оконной системы MS Windows.

Для того чтобы всё таки вывести на консоль текст на русском языке доработаем программу:

```

With Ada.Text_IO;
With OEM;

procedure prog1 is

  function SO(s : String) return String is
    sl : String := s;
  begin
    OEM.To_OEM(sl);
    return sl;
  end SO;

begin
  Ada.Text_IO.Put_Line(SO("My first program - Это первая программа"));
end prog1;

```

Теперь русский текст выводится правильно на консоль Рис. 25.

Примечание:

Для изучения синтаксиса языка программирования Ada можно почитать книгу Гавва А. Е. "Адское" программирование. Ada-95. Компилятор GNAT: [Электрон. ресурс]. – <http://www.ada-ru.org>. [1].

```

C:\WINDOWS\system32\cmd.exe

C:\Root305\ProjKA\2010\MyProg01>prog1.exe
My first program - Это первая программа

C:\Root305\ProjKA\2010\MyProg01>

```

Рис. 25. Доработанная программы выводит текст на русском языке в читаемом виде

Совместное использование консоли и графических окон

Небольшая доработка программы позволяет выдавать сообщения в графическом окне:

```
with Ada.Text_IO;
with OEM.GWindows.GStrings;
with OEM.GWindows.Message_Boxes;

procedure prog1 is
  use OEM.GWindows.Message_Boxes;

  function SO(s : String) return String is
    sl : String := s;
  begin
    OEM.To_OEM(sl);
    return sl;
  end SO;

begin
  Ada.Text_IO.Put_Line(SO("My first program - Это первая программа"));
  Message_Box (Title =>
    OEM.GWindows.GStrings.To_GString_From_String
      ("Tutorial1 - учебный пример №1"),
    Text => OEM.GWindows.GStrings.To_GString_From_String
      ("My first program - Это первая программа"),
    Icon => Exclamation_Icon);

end prog1;
```

На Рис. 26 показан результат работы программы. Более подробно пакет GWindows библиотеки OEM будет рассмотрен в главе 3.

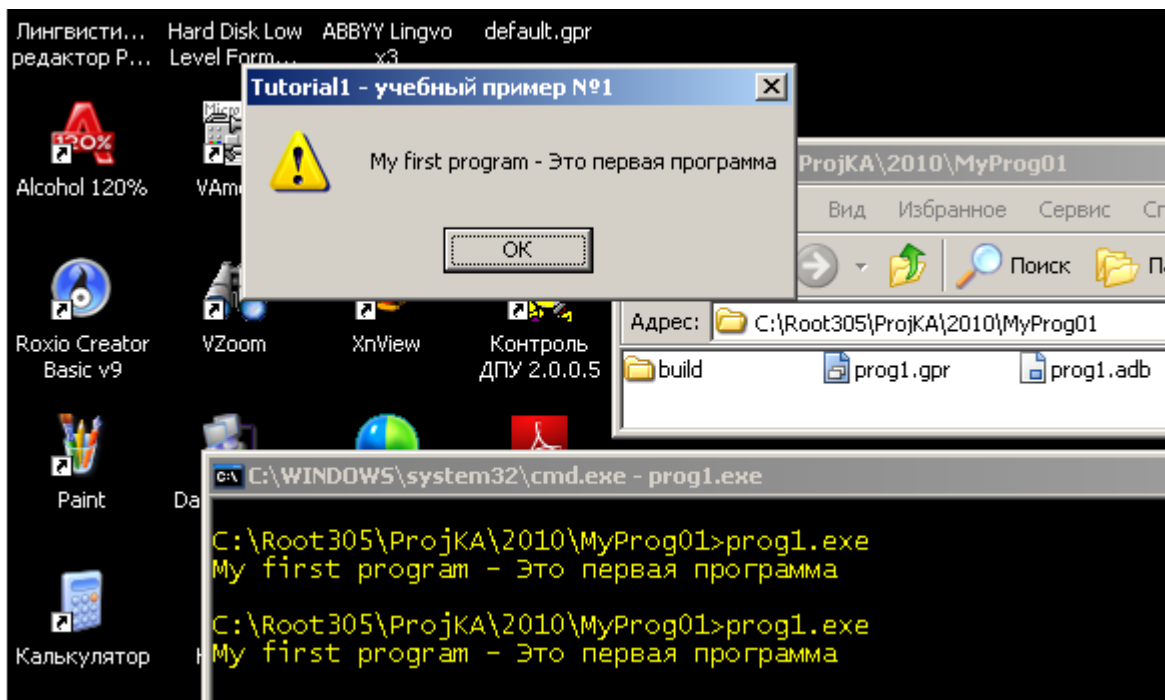


Рис. 26. Производим вывод сообщений на консоль и в окно диалога

ЛИТЕРАТУРА

1. Гавва А. Е. “Адское” программирование. Ada-95. Компилятор GNAT: [Электрон. ресурс]. – <http://www.ada-ru.org>.
2. Рыбин С. И., Годунко В. GNAT Pro – Ада – технология для серьезных проектов. Семинар, выставка PTS-2009, Минск. [Электрон. ресурс]. – <http://www.mediascan.by/index.files/GNAT-AdaCore.pdf>.
3. Мищенко В. О., Гахов А.В. Обучение проектированию систем и основам параллельных вычислений на базе языка Ада в Харьковском национальном университете. Семинар, выставка PTS-2009, Минск. [Электрон. ресурс]. – <http://www.mediascan.by/index.files/adaedu.pdf>.
4. Рыбин С. И., Годунко В. Ада – перспективы использования в индустрии и образовании. Семинар, выставка PTS-2009, Минск. [Электрон. ресурс]. – <http://www.mediascan.by/index.files/Ada-AdaCore.pdf>.
5. Киркоров С. И., Киркорова Л. С. Параллельные алгоритмы математических моделей: исследование локальности и применение языка Ada. Ж-л: ВІСНИК Харківського національного університету імені В.Н. Каразіна, №863 Серія: Математичне моделювання. Інформаційні технології. Автоматизовані системи управління, Випуск 12, стр. 129-142, Харків, 2009.
6. The GNAT Academic Program [Электрон. ресурс]. – <http://www.adacore.com/home/academia/>
7. Библиотека OEM [Электрон. ресурс]. – http://www.mediascan.by/index.files/CD_Ada-2009_OEM.zip