

MicroDixi

К у р с м о л о д о г о б о й ц а

Установка под Windows

•
Работа с Dixi в среде MS-DOS

•
**Главные принципы работы в
MicroDixi**

Примечание. Документация неофициальная, используйте её
на свой страх и риск.

Версия 1.1

15.04.2007

Алексей Тепляков

*Реальная смерть приходит с косой,
а виртуальная с Dixi.*

*Если программа полезна, ее нужно доработать,
а если - бесполезна, то следует разработать к
ней документацию.*

Dixi бояться - в IT не ходить.

1. УСТАНОВКА ПОД WINDOWS

Работать с **Dixi** мы будем в **Windows** (если вы сдаёте лабораторные в **ТТУ**, то учтите, что вам всё равно их придётся делать в среде **Solaris** на компьютерах **Sun**, так что **Dixi** у вас на компе будет установлена в целях изучения и практики).

Для **MicroDixi** естественной средой обитания является **DOS**. С одной стороны, можно запускать эту программу как обычно из **Windows**, но это не является наилучшим решением, т.к. создаёт ряд проблем:

- Нет гарантии, что ваша видеокарта поддерживает режим, необходимый для правильной работы **Dixi**. В общем случае запуск программы вызывает «синий экран», что, понимаете, не совсем хорошо сказывается на здоровье вашей машины;
- Нет возможности работать в оконном режиме;
- Нет возможности делать скриншоты (например, для извлечения схем).

Список, по идее, можно продолжать. Иными словами, требуется альтернативный подход. И он, конечно, существует, хотя и достаточно заумный. Итак, приступим, помня, что наша цель – запуск **Dixi** в полноценной **DOS**-среде.

Обратите внимание, что данный метод требует достаточно мощное и производительное железо и разрешающая способность экрана должна быть как минимум **1024x768**.

ШАГ 1 (необходимый софт):

Идея заключается в создании **виртуальной рабочей станции**. Для этого нам понадобится следующее средство от **Microsoft**, носящее название **Virtual PC 2007**:

<http://www.microsoft.com/windows/products/winfamily/virtualpc/default.mspx>

На рабочую станцию нам необходимо установить **MS-DOS**. Причём нам понадобится образ установочного диска. Почему, будет описано позже. Я использовал версию **7.1**, её можно скачать тут:

<http://www.starspirals.net/ttu/materjalid/Arvutid/Dixi/dos71cd.iso>

Ну и нам, конечно, понадобится установочный диск **Dixi**. Он тоже доступен для скачивания (вместе с **Dixi** на образе есть ещё **список компонентов и юзермануал**):

<http://www.starspirals.net/ttu/materjalid/Arvutid/Dixi/dixi.iso>

Ну и опционально могу предложить **виртуальный привод FDD**:

<http://www.starspirals.net//ttu/materjalid/Arvutid/Dixi/vfd21-050404.zip>

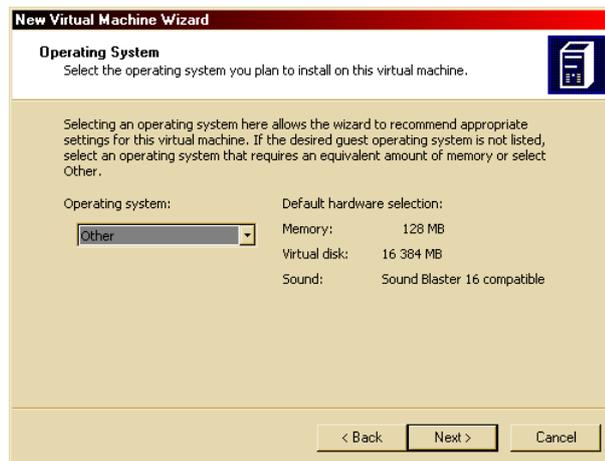
Он пригодится для переноса данных между виртуальной рабочей станцией и системой-хозяином.

Теперь, когда у нас есть весь необходимый софт, перейдём непосредственно к настройке системы.

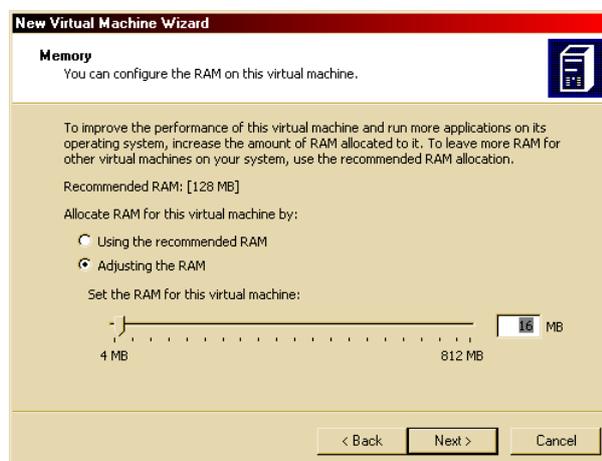
ШАГ 2 (установка Virtual PC):

Здесь ничего особенно сложного нет. Просто устанавливаем программу (если у вас **Windows XP Home** игнорируем предупреждение о несовместимости системы).

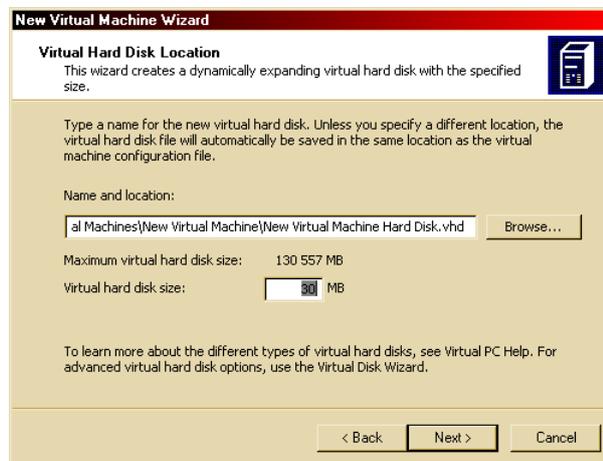
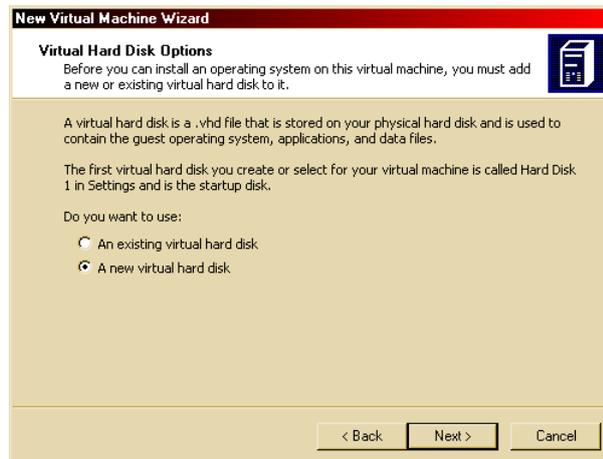
При первом запуске задаём параметры новой виртуальной машины. Выбирайте тип операционной системы «**Other**». Приведу общие рекомендации.



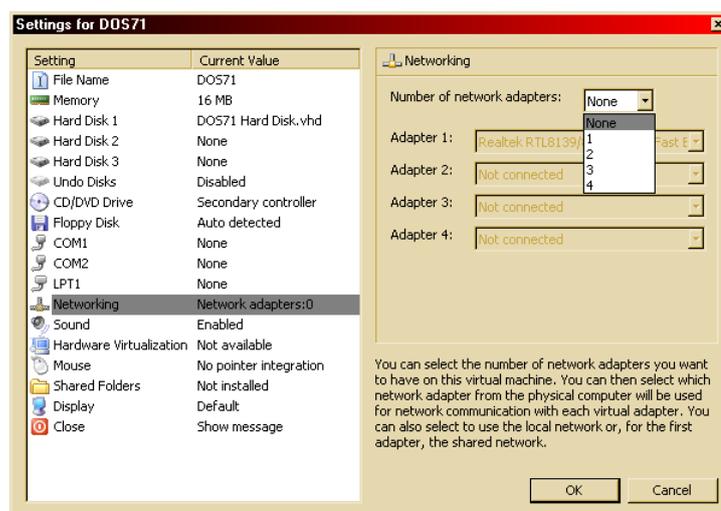
- 1) Отведите достаточно (но не слишком много) памяти для виртуальной системы. Я ставил от **8 до 16 Мб**, для **DOS**-а очень даже подойдёт:



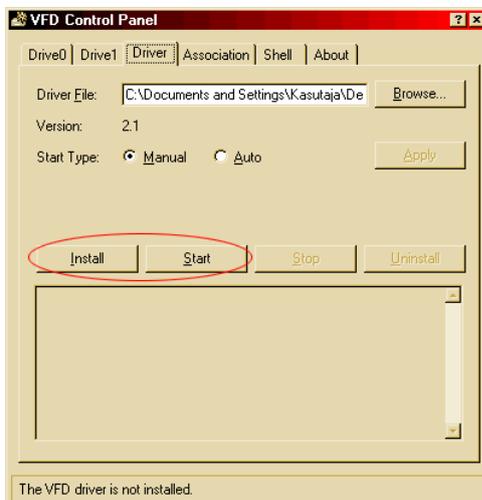
- 2) Создайте также виртуальный жёсткий диск, размером где-то **20-30 Мб**, больше вряд ли понадобится;



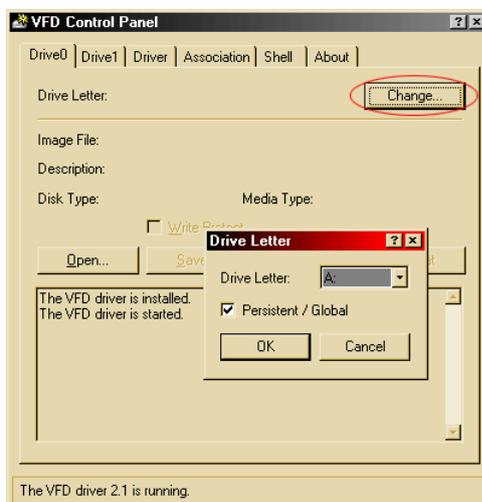
- 3) После создания системы рекомендуется отключить в ней сетевую поддержку, т.к. она нам также не нужна.



При желании установите **виртуальный FDD** (не забывайте «вытаскивать дискету» каждый раз, когда запускаете виртуальную систему, иначе она просто не запустится без вмешательства в её **BIOS**). Для этого распакуйте архив в какую-нибудь папку (рекомендуется эту папку потом никуда не перемещать) и запустите **vfdwin.exe** – контрольную панель. В открывшемся окне (вкладка **Driver**) выбираем **Install** и затем **Start**:

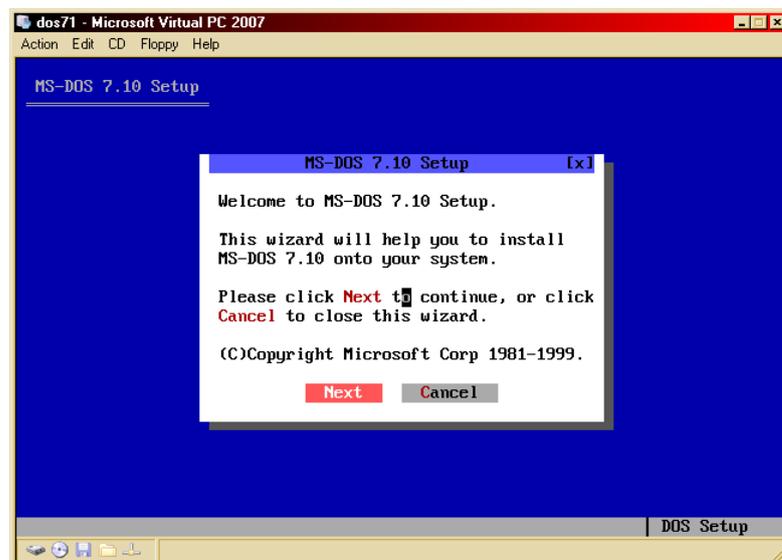
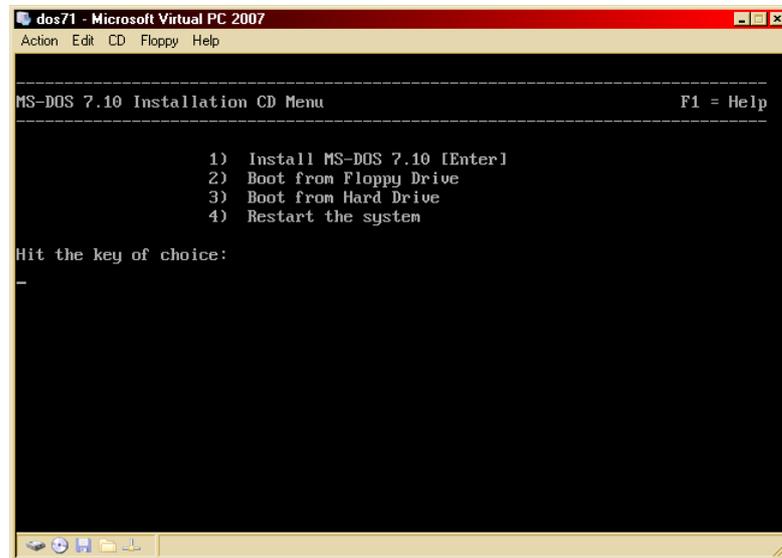


Теперь во вкладке **Drive0** поменяйте букву привода на удобную вам, и здесь же вы получаете возможность управлять вашими виртуальными дискетами (используя директиву **Open**, дальше, я думаю, разберётесь):



ШАГ 3 (установка MS-DOS):

Теперь пришла пора ставить **MS-DOS 7.1**. Для этого смонтируем на виртуальный привод **Virtual PC 2007** скачанный образ установочного диска **DOS**-а и перезагружаем систему. Устанавливаем **MS-DOS** (я, к примеру, ставил без адд-онов – гораздо проще и быстрее), отвечая на вопросы программки **Setup**.



ШАГ 4 (установка MicroDixi):

После установки **MS-DOS**-а и перезагрузки системы вы окажетесь в старом добром **command-prompt**-е, где все команды набираются вручную. Можно, конечно, воспользоваться директивой **dosshell** (здесь и далее все указания командной строки будут оформляться **таким шрифтом**), однако я буду рассказывать, как всё это делается без неё. Предполагается, что вы уже знаете, как работать в **DOS**.

***Примечание.** Пользуйтесь **правым ALT**-ом чтобы выводить курсор за рамки окна виртуальной системы.*

Первым делом монтируем образ диска Dixi в Virtual PC. Далее, набираем в командной строке:

```
D:\install d c
```

Произойдёт установка MicroDixi. Дальше, если мы находимся в корневой директории C:\ то для перехода в директорию с Dixi пишем

```
cd caesys
```

Здесь выводим следующее указание:

```
setpath c
```

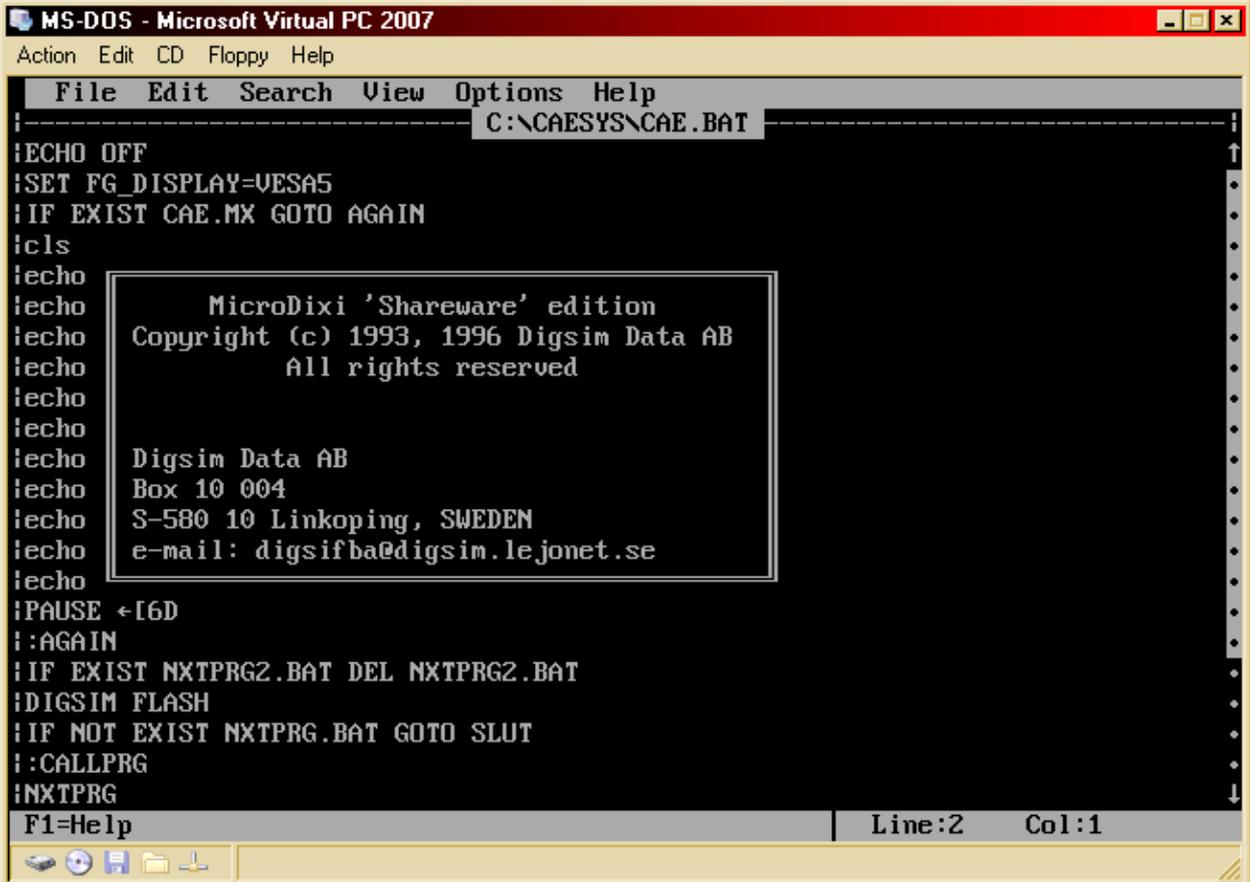
Теперь практически всё готово. Осталось малость – отредактировать файл cae.bat. Для этого пишем

```
edit cae.bat
```

И вставляем строчку

```
SET FG_DISPLAY = VESA5
```

между первой и второй, вот так:

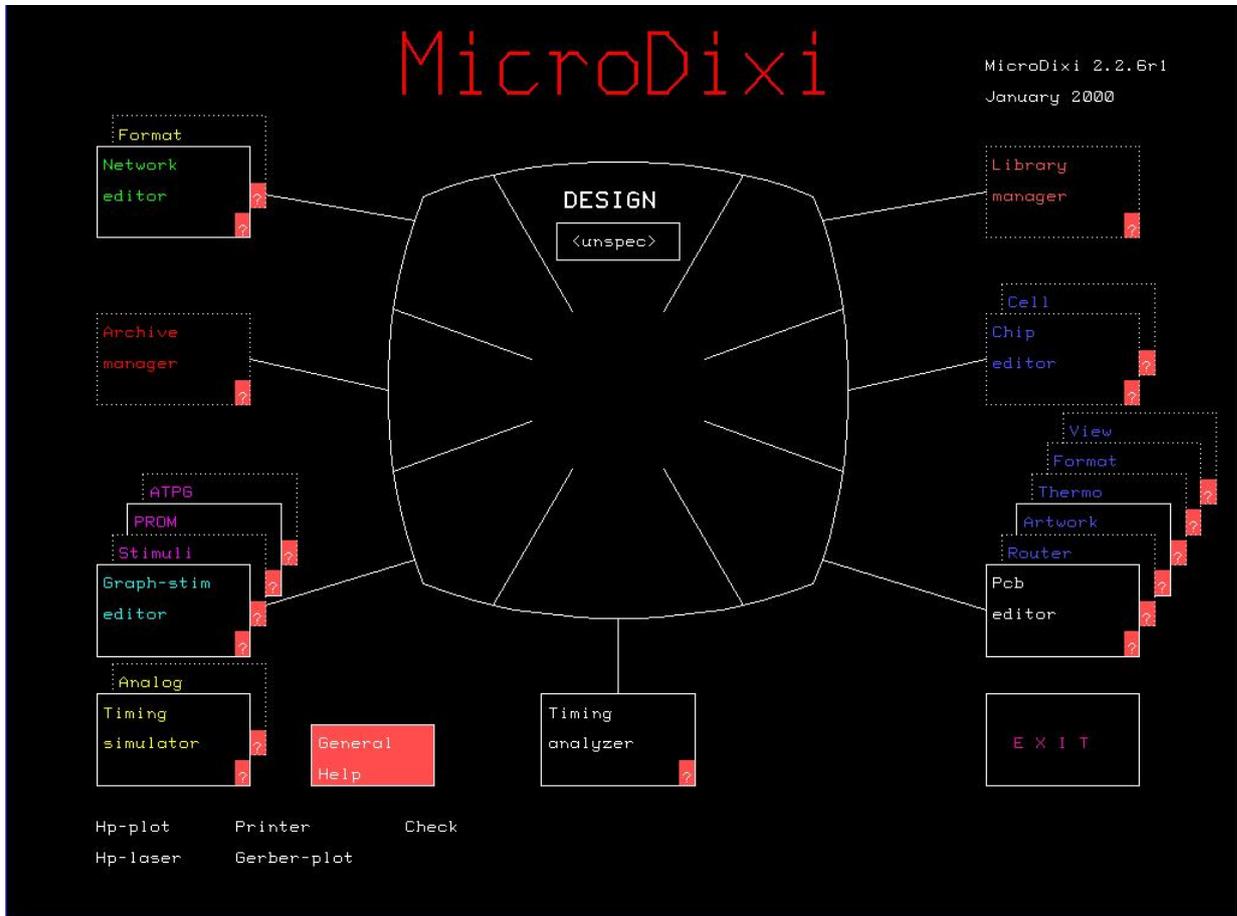


```
MS-DOS - Microsoft Virtual PC 2007
Action Edit CD Floppy Help
File Edit Search View Options Help
----- C:\CAESYS\CAE.BAT -----
ECHO OFF
SET FG_DISPLAY=VESA5
IF EXIST CAE.MX GOTO AGAIN
cls
echo
echo      MicroDixi 'Shareware' edition
echo      Copyright (c) 1993, 1996 Digsim Data AB
echo      All rights reserved
echo
echo      Digsim Data AB
echo      Box 10 004
echo      S-580 10 Linkoping, SWEDEN
echo      e-mail: digsifba@digsim.lejonet.se
echo
:PAUSE <[6D
:AGAIN
IF EXIST NXTPRG2.BAT DEL NXTPRG2.BAT
DIGSIM FLASH
IF NOT EXIST NXTPRG.BAT GOTO SLUT
:CALLPRG
:NXTPRG
F1=Help | Line:2 Col:1
```

После чего сохраняем файл. Всё, **Dixi** готова к работе. Перекрестившись, вводим

cae

и получаем в ответ такое окошко:



Поздравляю! Вы установили **Dixi** на виртуальную машину и теперь сможете получить все преимущества такого её использования.

Во втором разделе я вкратце расскажу как можно манипулировать **Dixi** под **DOS**-ом, а в третьем разделе познакомлю с базовыми принципами работы внутри программы.

2. РАБОТА С DIXI В СРЕДЕ MS-DOS

2.1. ЗАПУСК

После запуска виртуальной машины и загрузки MS-DOS-а, пишем

```
cd caesys  
cae
```

Надеюсь, понятно, что **CAESYS** – это директория **Dixi**. Все проекты будут сохранены именно в эту папку.

2.2. КОПИРОВАНИЕ ПРОЕКТОВ ИЗ ПАПКИ DIXI НА СМЕННЫЙ НОСИТЕЛЬ

Находясь в директории **CAESYS**, вводим, например, такое указание:

```
copy proj_nm.* a:
```

где **proj_nm** – название проекта,
a: - привод, на который осуществляется копирование. Если вы установили виртуальный привод, то данные будут скопированы на него. Только буква может быть другая.

Звёздочка означает, что мы копируем все существующие файлы с разными расширениями и указанным именем.

После закрытия виртуальной системы, виртуальный привод станет доступен реальной операционной системе и вы сможете получить скопированные файлы.

2.3. КОПИРОВАНИЕ ПРОЕКТОВ СО СМЕННОГО НОСИТЕЛЯ В ПАПКУ DIXI

Делается аналогично. Положите файлы, которые необходимо пренести на виртуальную машину, в корневую **A:** и введите, находясь в любом месте на виртуальной системе следующее:

```
copy a:\proj_nm.* c:\caesys\proj_nm.*
```

После чего можете запускать **Dixi**, набирать в ней название проекта и пользоваться файлами.

2.4. УДАЛЕНИЕ ПРОЕКТОВ

Находясь в директории **CAESYS**, введите:

```
del proj_nm.*
```

и все файлы проекта будут удалены.

Теперь, когда мы разобрались с базовыми операциями с файлами, перейдём, наконец, к работе внутри **MicroDixi**.

3. ГЛАВНЫЕ ПРИНЦИПЫ РАБОТЫ В MICRODIXI

Общие замечания

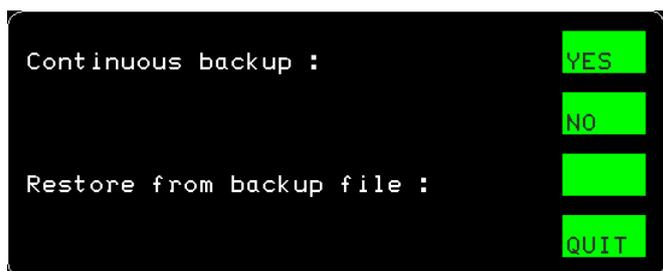
- Вместо привычного курсора в **Dixi** используется прицел, состоящий из горизонтальной и вертикальной полос на весь экран. С первого взгляда трудно сообразить, зачем это сделано, однако в итоге это бывает даже удобно;
- **Dixi** не будет правильно работать в среде **Solaris** с включённым **NumLock**;
- При создании проекта все файлы, которые должны относиться к данному проекту, нужно называть одинаково;
- Вводимые данные не чувствительны к регистру.

Приступая к проекту, вводим в центральное окошко в главном меню под надпись **DESIGN** название нашего проекта. В дальнейшем используйте в рамках этого проекта только это название.

3.1. NETWORK EDITOR

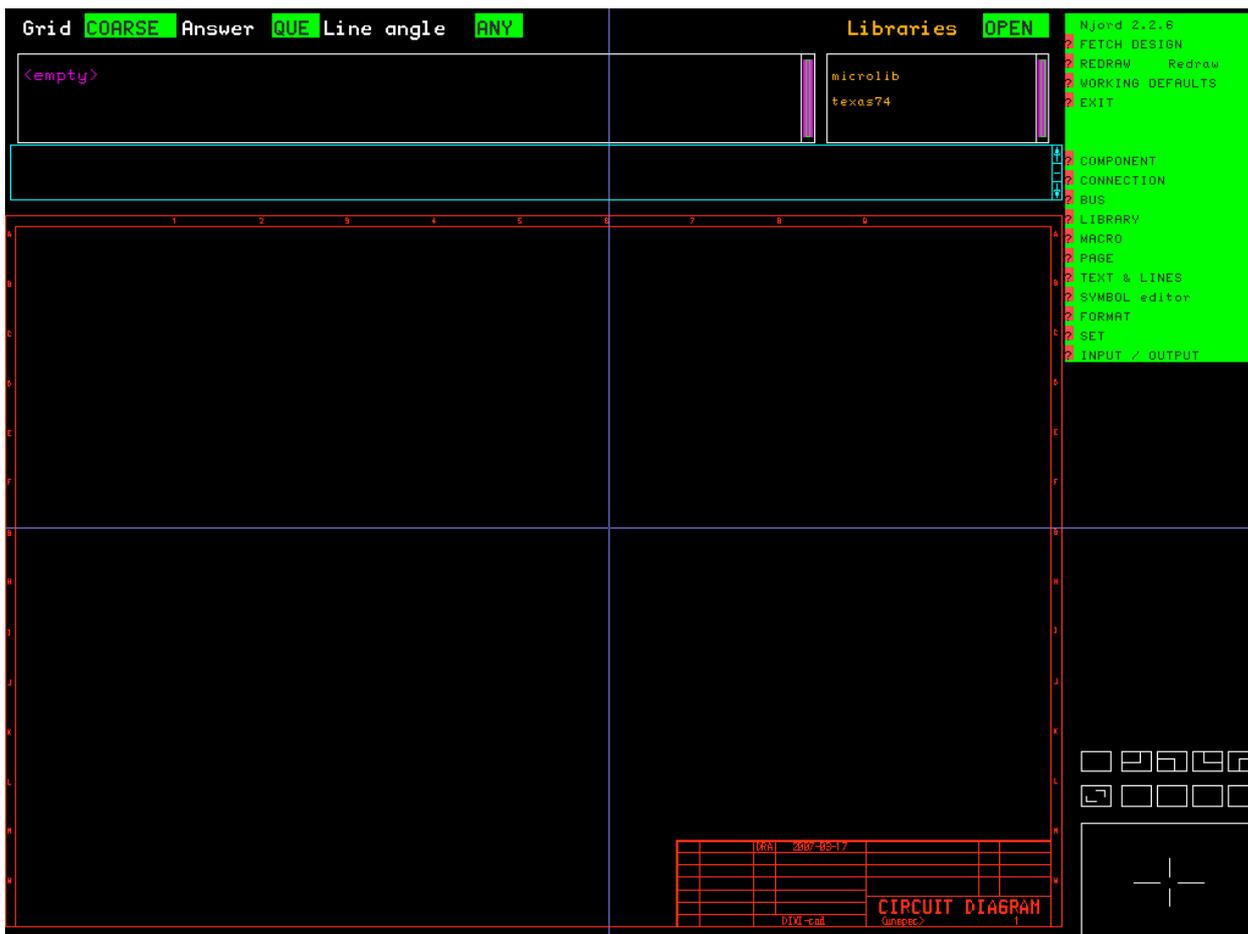
Как правило, любой проект начинается с построения схемы. Для этого в **MicroDixi** существует приложение **Njord**.

Из главного меню выбираем **Network Editor** и видим такое сообщение (оно одинаково для нескольких приложений программного пакета **Dixi**):



Прицеливаемся в **YES** и открываем огонь.

Дальше вылезает экран, с которым мы сейчас будем разбираться:



- Белое окошко сверху слева – это список используемых в схеме элементов. Справа от него располагается список библиотек элементов;
- Голубое окошко ниже – это информационное окно, в котором помимо сообщений возникают иногда подтверждения тех или иных действий; кроме того, при создании входов схемы мы тащим провода именно в это окошко. В нём же появляются выбранные нами элементы схемы;
- Красное большое окно – это непосредственно рабочая область;
- Надписи, выделенные зелёным цветом, есть ни что иное, как пункты и подпункты меню. Пункты верхнего уровня **НАПИСАНЫ В ВЕРХНЕМ РЕГИСТРЕ**, а вторичные – как обычно.
- Справа снизу видно селектор области зума.

Когда первый шок от всего этого пройдёт, приступим к работе. Дальше я привожу типичную схему работы в **Njord-e**.

Если вам нужно отредактировать ранее созданный дизайн, то воспользуйтесь функцией **FETCH DESIGN**. Если начинаете с нуля, то необходимо проделать следующие операции.

1) СОСТАВИТЬ СПИСОК ЭЛЕМЕНТОВ, НЕОБХОДИМЫХ ДЛЯ РЕАЛИЗАЦИИ СХЕМЫ, И ЗАДАТЬ ИХ В DIXI

Первым делом, составим список необходимых элементов. Воспользуемся таблицей:

<http://www.pld.ttu.ee/java/dixi/>

Она имеет такой вид:

Part number	N	L	LS	ALS	S	H	Function	Comment
00	*	*	*		*	*	4x2NAND	
00A				*			4x2NAND	
00F			*		*		4x2NAND	
01	*		*	*		*	4x2NAND	OC
02	*	*	*	*	*		4x2NOR	
02F					*		4x2NOR	
03	*	*	*		*		4x2NAND	OC
03A				*			4x2NAND	OC
04	*	*	*		*	*	6xINV	

Задавать полный номер элемента надо так. Поскольку мы рассматриваем серию **Texas Instruments 74**, то начинаться номер любого компонента будет с числа **74**. После этого числа без пробела записываем ту комбинацию букв, что для данного элемента отмечена звёздочкой на зелёном фоне (букву **N** записывать не надо). Далее идёт номер детали слева. Например, **7400** или **74LS00** даст нам **4x2NAND**.

Положим, нам необходим **2x4AND**. Смотрим по таблице:

19			*				6xINV	Schmidt trigger
20	*	*	*		*	*	2x4NAND	
20A				*			2x4NAND	
20F					*		2x4NAND	
21			*	*		*	2x4AND	
22	*		*	*	*	*	2x4NAND	OC, (133/155 - 30mA)
26	*		*				4x2NAND	OC, 15V
27	*		*	*			3x3NOR	

Составляем:

74 (как обычно) + **LS** (отмечено звёздочкой; **N** не отмечено, так что запиши мы **7421**, **Dixi** бы заявила, что нет такого элемента у неё в библиотеке) + **21** (номер элемента).

Окончательно, чтобы получить в **Dixi** элемент **2x4AND** нужно указать номер детали:

74LS21

Составив список необходимых элементов, будем добавлять их в схему в **Dixi**.

Прицеливаемся в **COMPONENT** и стреляем. Снизу тут же выскакивает подменю, в котором выбираем уже **Define**. Теперь стреляем в голубое окно по надписи **Fetch component**, которое должно уже там стоять. Появляется **Type number**, куда мы вводим необходимый номер элемента. Нажимаем Enter, и после подтверждения элемент появляется в голубом окошке. Вытаскивайте сколько нужно этих элементов на схему. Если нужно больше, чем даётся, надо повторно стрелять необходимому номеру в окошке со списком элементов.

Повторяйте данную операцию, пока все необходимые элементы не будут вытащены в рабочее пространство.

Если вы сохраните дизайн, а в списке элементов будут прописаны такие, которые не реализованы в схеме, при следующей загрузке дизайна они из этого списка будут удалены.

Чтобы удалить элемент на схеме, пользуйтесь **Erase**, чтобы переместить – **Move**.

2) СОЗДАТЬ МАКРОСЫ (ОПЦИОНАЛЬНО)

Часто схемы бывают большими. Чтобы сэкономить место, можно воспользоваться макросами. Другой вариант такой, что необходимого элемента нет в библиотеке, а вам он позарез нужен (скажем, 4-х входовой **NOR**). Разберём как раз такой пример.

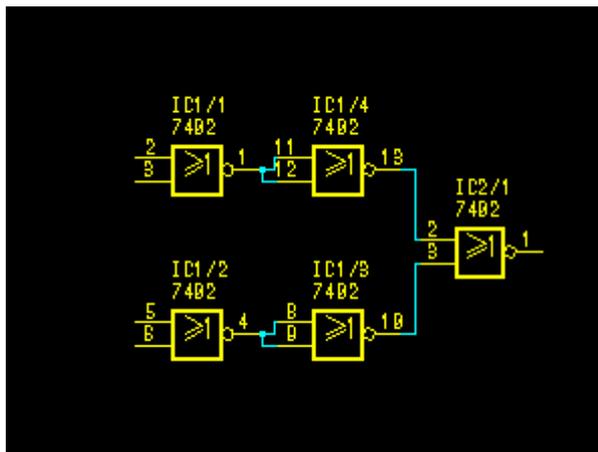
Итак, нам нужен 4-х входовой **NOR**. Сделаем следующее.

Выбираем в большом меню **MACRO** и ниже **Create New**. Задаём уникальное имя новому элементу, например, **4NOR**.

Дальше жмём **Inspect** в том же подменю, и выбираем макрос в окошке со списком элементов схемы. В дальнейшем я просто буду называть эту операцию **инспектировать** макрос. Справа от **Inspect** есть **Top**. Нажав на эту штуку мы вернёмся к верхнему уровню, т.е. к большой схеме. Но сейчас делать этого не будем.

Теперь нажимаем **COMPONENT**, **Define** и так дальше – собираем макрос. 4-х входовой **NOR** соберём из двухвходовых. Номер элемента – **7402**. Когда необходимые элементы вытащены, создаём необходимые соединения (**CONNECTION**, **Make**) внутри макроса (внешних соединений делать не надо). Всё, что связано с соединениями, будет изложено далее.

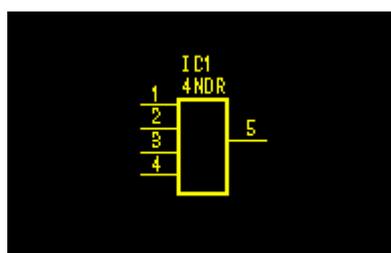
Должна получиться такая схема:



Когда соединения готовы, жмём **MACRO, Top, COMPONENT, Define** и вытаскиваем элемент **4NOR** – т.е. наш свежесозданный макрос. Он пока выглядит несимпатично, просто как квадрат:



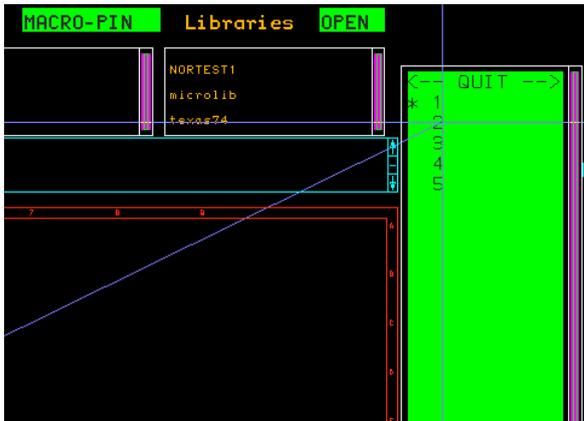
Подарим нашему макросу **пины**. Для входов создаём соединения (**CONNECTION, Make**), щёлкая по левой стороне квадрата, а для выходов – по правой. Щёлкнув, ведём в голубое окошко. На вопрос **Identity** отвечаем отрицательно, на предложение **Identity** ввести вручную просто жмём Enter. Таким образом сделаем 4 пина слева и 1 пин справа. Должно получиться следующее:



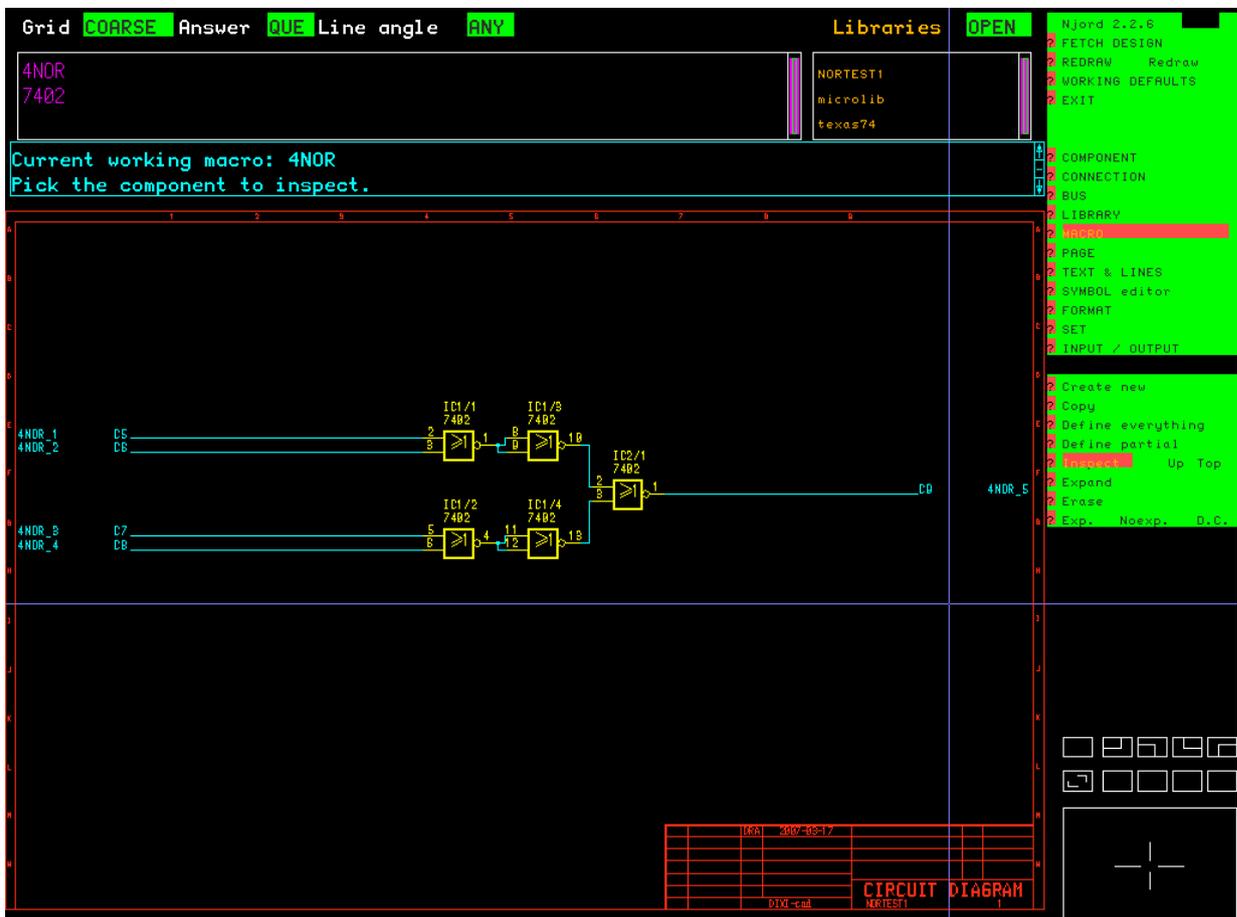
Осталось соединить внутренние выходы и входы макроса с пинами. Инспектируем макрос и вновь идём в **CONNECTION, Make**. Обращаем внимание, что на самом верху экрана появляется при этом **MACRO PIN**.



Берём по очереди каждый вход и выход и ведём в **MACRO PIN**. Справа появляется меню, в котором следует выбрать интересующий нас пин макроса. Стреляем по номеру пина нашим соединением и потом дотаскиваем соединение в голубое окошко.

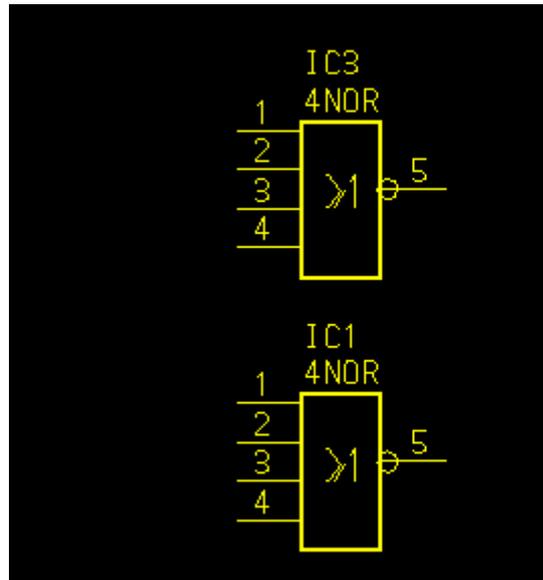


Когда мы проделали всю эту работу, на экране появляется что-то такое:



Теперь жмём **Top** и всё, пользуемся макросом как обычным элементом. Если вам нужен больше, чем один **4NOR**, то просто вытаскивайте его из списка элементов. Он будет с готовыми пинами и соединениями.

Единственное, можно ещё сделать безжизненный прямоугольник с пинами более похожим на **4NOR**. Это делается в **SYMBOL editor**. Однако я не буду рассказывать сейчас как, разберитесь сами ;-)



3) СОЗДАТЬ СОЕДИНЕНИЯ

В меню **Network Editor**-а за соединения отвечает пункт **CONNECTION**. Подпункты делают следующее:

Make – создать

Rename – переименовать

Disconnect – удалить провод

Route-ALL – автоматически перестроить провода

Соединяем так:

- 1) С помощью **Make** выбираем начальную точку соединения (пин какого-либо элемента или точка на уже существующем проводе).
- 2) Ведём провода к нужным пинам или точкам провода (каждый раз прицельно щёлкая по ним).
- 3) Когда все нужные точки будут соединены, тащим провод в голубое окошко. Отвечаем на вопрос **Identity**, и провода появляются на схеме.

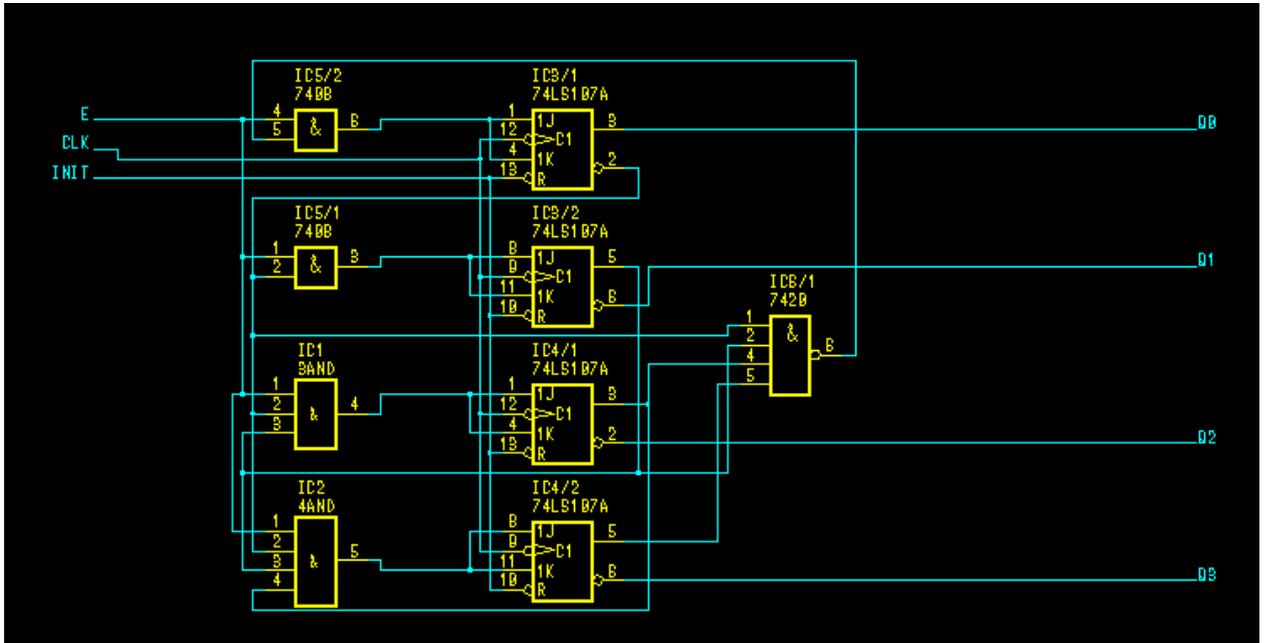
Если надо сделать **вход** или **выход** схемы, то надо просто выбрать входной/выходной пин (или провод) и тянуть его в голубое окошко.

Совет. Ускорить работу можно, выставив автоматический ответ на вопросы (требования подтверждения), включая вопрос об **Identity**. Для этого надо щёлкнуть в самом верху экрана поле **Answer**, чтобы там высветилось слово **YES** вместо **QUE**.

Кроме того, вместо того, чтобы вести конечный провод в голубое окошко можно просто **дважды щёлкнуть** левой клавишей мышки.

Когда мы создали все необходимые соединения, то пришло время их переименовать удобным для нас образом. Для этого используем функцию **CONNECTION**, **Rename**. Стреляем по входному или выходному проводу и пишем новое название.

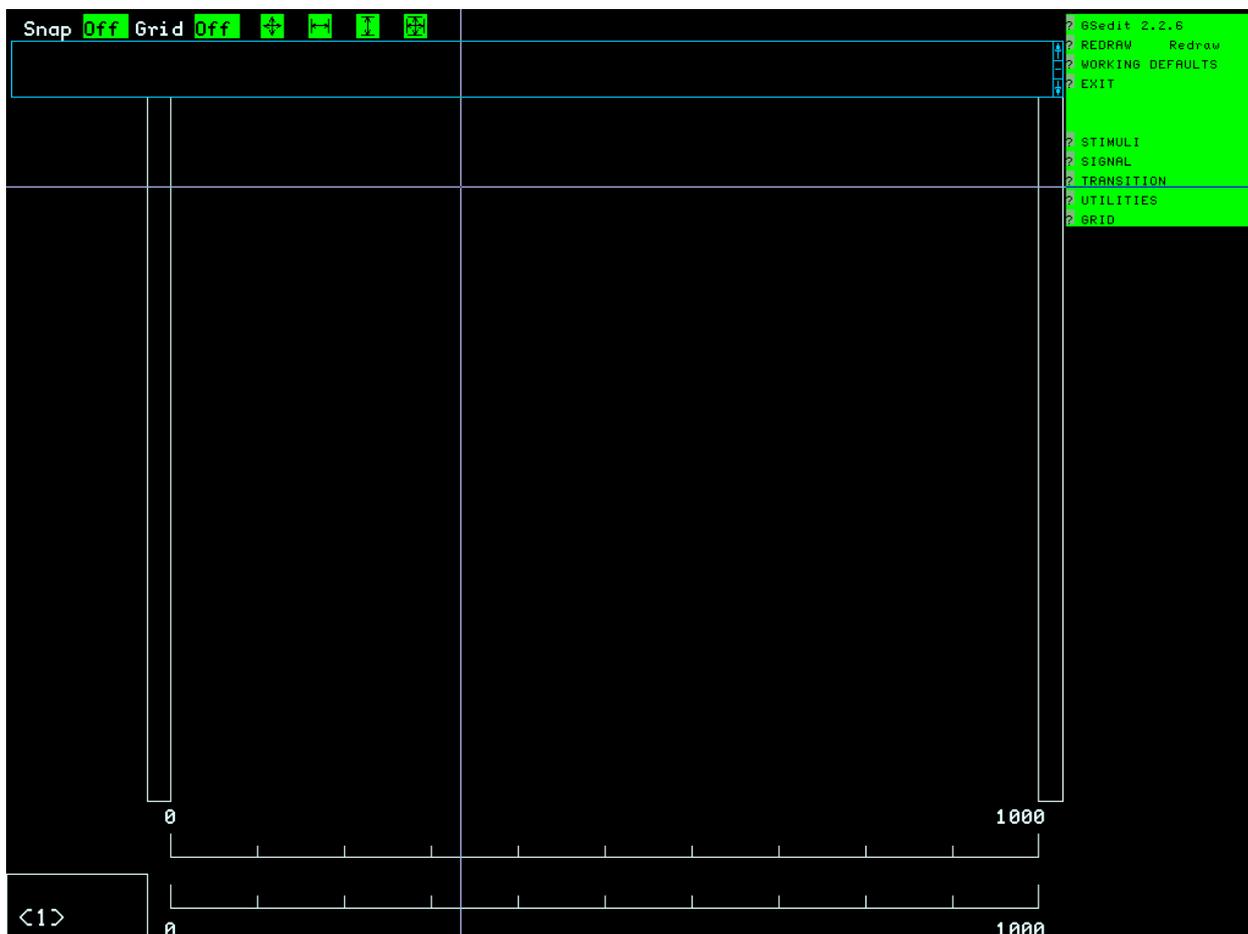
В конечном итоге получаем готовую схему. Например, такую:



Когда схема готова, нужно протестировать её работу и проанализировать результат. Об этом читайте дальше. Жмём **EXIT** и на вопрос о сохранении дизайна отвечаем утвердительно (кстати, другого способа сохранения в природе **Dixi** не заложено).

3.2. GRAPH-STIM EDITOR

Здесь задаются необходимые входовые стимулы (сигналы переменного низкого/высокого уровня).



Иерархия меню здесь аналогична **Njord**-у.

Меню **STIMULI** позволяет в числе прочего загружать стимулы для их редактирования (если они уже были созданы). Для этого воспользуйтесь следующими командами: **STIMULI, Load**, введите имя проекта с сохранёнными стимулами, **Select**, и выбрать из появившегося списка нужные стимулы.

Меню **SIGNAL** позволяет создать сигналы. На этом мы остановимся подробнее.

Жмём **SIGNAL, Add**. Стреляем в голубое окошко. Теперь **без запятой (через пробел)** вводим названия сигналов, которые служат входными для ранее отредактированной схемы (они должны совпадать с названиями проводов). Например:

```
e reset clk
```

Получаем такую картину:



Теперь нужно, чтобы наши сигналы со временем менялись. Для этого существует меню **TRANSITION**. Сигналы прорисовываются справа от их названий и могут быть либо заданными (неопределённость), либо иметь низкий уровень, либо иметь высокий уровень.

Щёлкнем по **TRANSITION**. Теперь видим кучу разного барахла, из которого мы рассмотрим несколько пунктов:

Toggle – попеременно менять высокий/низкий уровень;

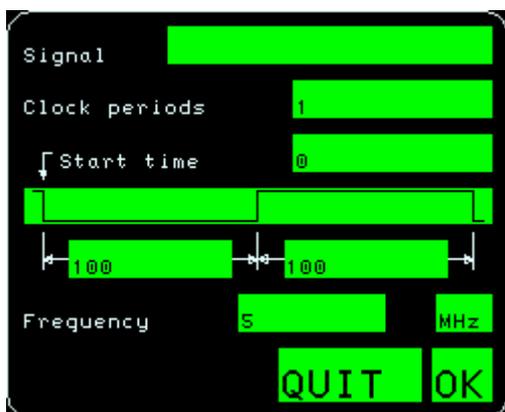
High – выставлять высокий уровень;

Low – выставлять низкий уровень.

Если вы будете выставлять уровни вручную, то поймёте преимущества прицела. Для изменения уровней просто щёлкайте по пространству справа от названий сигналов.

Кроме того, полезно знать ещё как минимум две функции меню **TRANSITION**. Это часы (**Clock**) и счётчик (**Counter**).

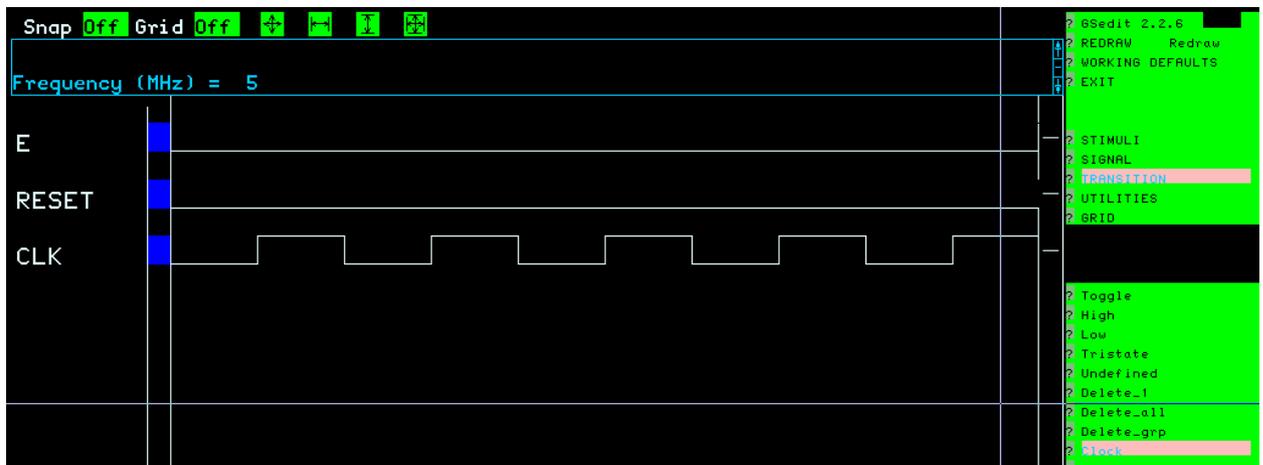
Clock



Щёлкаем по зелёному полю рядом с надписью **Signal** и вводим название сигнала, который необходимо сделать носителем тактовой частоты (это чаще всего необходимо для синхронизации).

Затем вводим число периодов часов (я, например, ввёл **100**). Можете также изменить частоту, но в принципе для ряда тестов **5 MHz** – это то, что надо.

Нажимайте **OK** и получите такое:



Counter

Предположим, наша схема имеет 4 входа, которые являлись бы битами определённого счёта и считали бы от 0 до 15. Для реализации этого воспользуемся **счётчиком**.

Добавим необходимые сигналы, назвав их **x1**, **x2**, **x3** и **x4**. Введём их именно в такой последовательности, т.к. **x1** должен быть самым младшим битом, а **x4** – самым старшим.

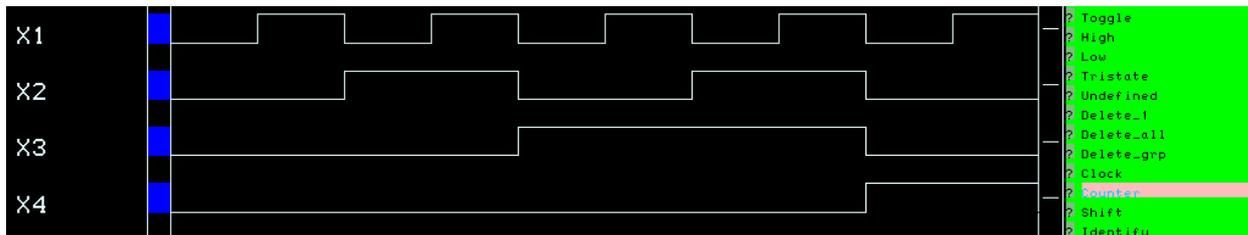
Дальше выстрелим по **TRANSITION, Counter**. Видим следующее:



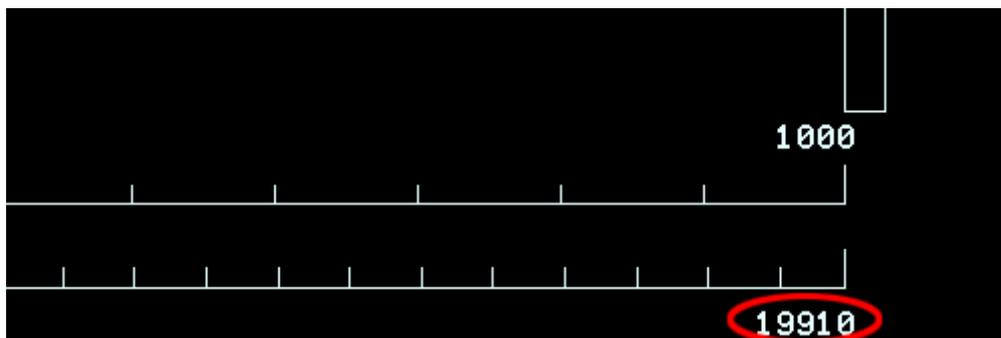
Нажав на **Signals**, ползём прицелом к названиям сигналов и выделяем их (это практически аналогично способу в **Windows**). В нашем случае, это сигналы **x1 ... x4**. Нетрудно догадаться, что сигналы, которые необходимо включить в счётчик, должны быть соседями.

После выделения в зелёной области меню должно появиться число 4, говорящее о количестве участвующих в счётчике сигналов.

Дальше, выставьте в **Time increment** значение **100**, и в **Steps** аналогично. **Count** оставьте **BINARY**. Потом нажимаем **OK** и получаем соответственно этому счётчик, который считает так:



Таким образом, выставьте все необходимые вам сигналы. После чего найдите число в нижней части экрана на правом конце самой второй шкалы:



Запомните это число. Оно понадобится вам в симуляторе.

Вот теперь жмём **EXIT**, подтверждаем сохранение стимулов (не забудьте задать для них общее имя проекта).

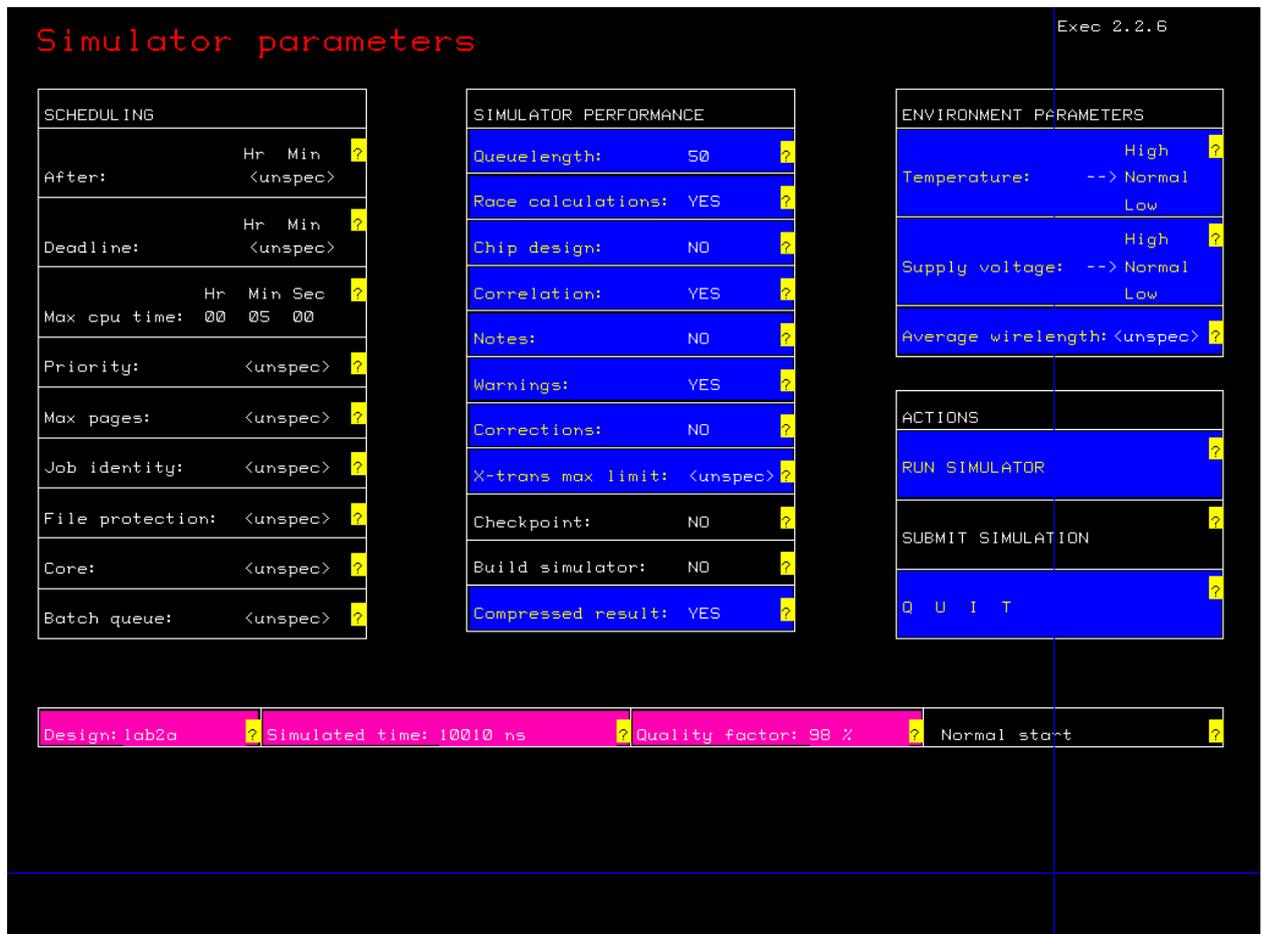
Внимание. Если вы работаете с **Dixi** на **Solaris-е**, то обязательно откройте в директории, в которой хранятся файлы-исходники **Dixi**, терминал и в нём пропишите:

fixdixi

Эта директива проверит файлы стимулов и исправит их. Иначе вы не сможете правильно запустить **симулятор**, о котором будет рассказано далее.

3.3. TIMING SIMULATOR

Заходим в него из главного меню. Видим что-то такое:



Здесь вводим в фиолетовые поля внизу:

Design – название дизайна. Если оно уже выставлено, то вводить его повторно не обязательно.

Внимание. Если вы работаете на **Sun-ax**, то ввести имя дизайна в это поле **обязательно нужно**, даже если оно там уже проставлено, иначе **симулятор не работает**.

Simulated time – вводим то самое число, что мы запомнили из **Graph-Stim Editor-a**.

На всё остальное пока не обращаем внимания.

Теперь можно спокойно стрелять в расположенную справа большую кнопку **RUN SIMULATOR**. Если вы работаете на **Sun-ax**, то после некоторого времени появятся фиолетовые часики, которые малость потикают и пропадут; ну а если вы под DOS-ом, то вылезет чёрный экран на котором будет очень быстро бежать время, по истечении которого мы попадём обратно в главное меню.

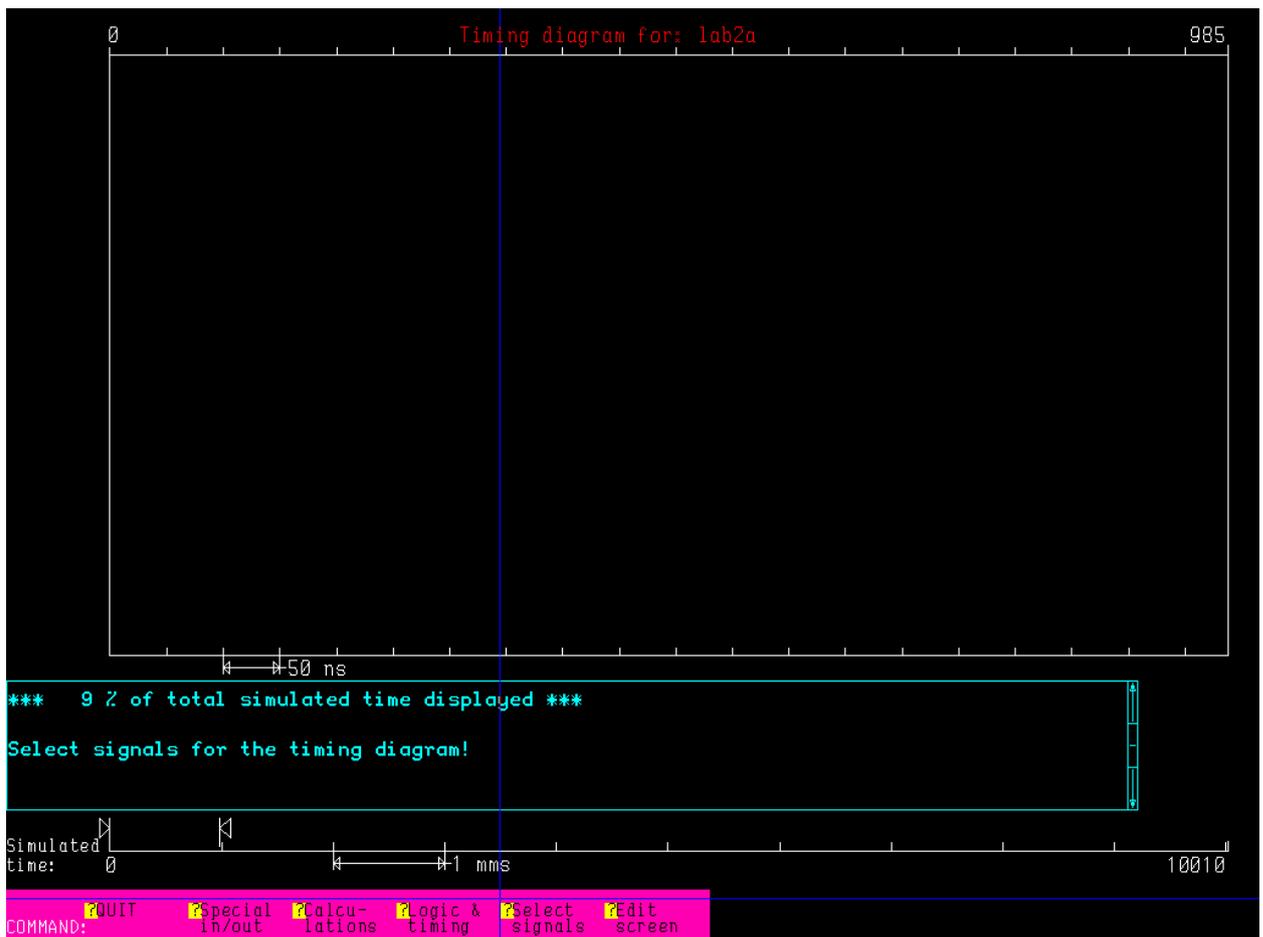
Итак, дело за малым – теперь нужно исследовать результаты симуляции.

3.4. TIMING ANALYZER

Наступает самый интересный момент. Анализ входных и выходных сигналов.

Нажав на **Timing analyzer**, в главном меню попадаем в голубое окно на пол экрана, в котором увидим слово **Detective** и предложение ввести название дизайна. Если вы в самом начале работы с проектом вводили в центральное окно **Dixi** общее название проекта и потом его не меняли, то можете просто нажать Enter. Если нет – то вводите название дизайна и опять-таки жмите Enter. Возникнет кое-какая статистика, от которой вас избавит очередное нажатие Enter.

Следующий экран пугает своей пустотой.

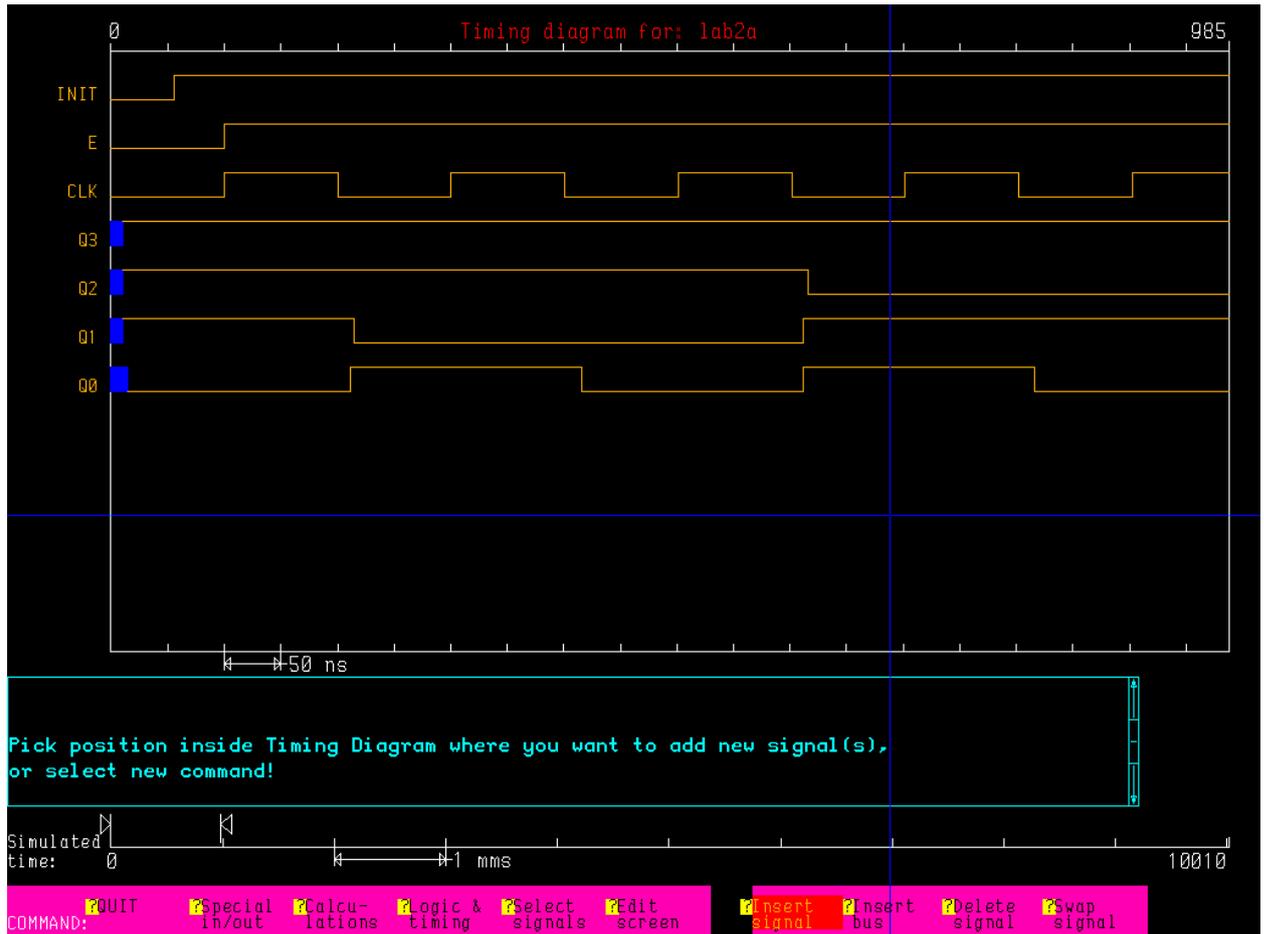


А где же наши сигналы? Ну не беда, просто дяди из **Digsim**-а позаботились о нас и позволили вводить все сигналы вручную.

Ищем в фиолетовом меню пункт **Select signals** и стреляем в него. Справа появляется ещё одно меню, жмём там на **Insert Signal**.

Теперь **через запятую** вводим все требуемые названия сигналов (как входные, так и выходные).

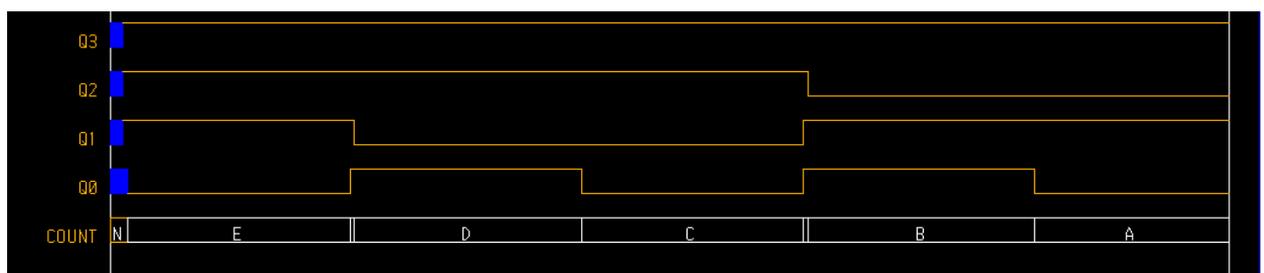
В моём случае возникает такое:



Кстати, сигналы Q0 ... Q3 здесь – это биты счёта, поэтому целесообразно вводить их, начиная с самого старшего – так будет удобнее анализировать их изменения. Для ещё большего удобства сейчас добавим сюда **шину**. **Шина** – это всего лишь интерпретация нескольких сигналов в счётной форме. Т.е. если у нас комбинация сигналов Q3, Q2, Q1, Q0 будет, например, **1110**, то шина покажет «E».

Добавляем шину так. Жмём на **Insert bus** рядом с **Insert signal**. Стреляем в голубое окошко. Теперь вводим имя шины, затем количество участвующих битов (не трудно догадаться, что их в этом случае **4**) и затем названия сигналов-битов, начиная с самого старшего (т.е. **Q3, Q2, Q1 и Q0**). Потом жмём Enter ещё раз, чтобы выйти из режима ввода шины.

Рисуется такая штука (если вы работаете на **Sun**-ах, то надо ещё и обновить экран, это делается в **Edit screen, Redraw screen**):



Теперь приступаем к анализу результатов.

Чтобы видеть другую область диаграммы входных-выходных сигналов, используем **Edit Screen, Zoom**. Когда активирована команда **Zoom**, то можно редактировать область зума. Для этого сначала щёлкаем на нижней линейке начальную точку обзора, а затем конечную (сначала слева, потом справа):



После проведения анализов поведения сигналов, либо восторженно кричим «Да! У меня получилось!», либо озадаченно чешем затылок и возвращаемся к схеме и перепроверяем её.

Естественно, после каждой проверки и модификации схемы следует по-новой запускать симулятор...

ЗАКЛЮЧЕНИЕ

Ну вот и подошёл к концу наш краткий обзор основных принципов работы с **Dixi**. Конечно, очень многое осталось за кадром, но надеюсь, вы получили достаточно информации для самостоятельного дальнейшего изучения этой удивительной программы.

И если у вас ещё что-то не получается, не отчаивайтесь. Помните, дорогу осилит идущий. Успехов!

15 апреля, 2007
Алексей Тепляков

P.S.

Сообщения о неточностях и ошибках в этом пособии, а также ваши предложения можете высылать мне на емейл:

alex@starspirals.net