

Интеграция программ OrCAD и MATLAB/Simulink

при помощи интерфейса SLPS

Данная статья посвящена возможностям интерфейса SLPS, позволяющего объединить схемотехническое моделирование в программе OrCAD с функциональным моделированием в программе MATLAB/Simulink.

Иосиф Златин

zlatin@pochta.ru

Интерфейс PSpice SLPS (далее будет использоваться аббревиатура SLPS) является инструментом связи между моделирующим устройством программы MATLAB/Simulink фирмы MathWorks и схемотехническим моделирующим устройством PSpice A/D (далее будет использоваться аббревиатура PSpice), который был разработан совместно компаниями Cybernet Systems и Cadence Design Systems. Используя SLPS, можно вставить электрическую схему PSpice в функциональную схему Simulink и связать функциональное и схемотехническое моделирование схемы. Эта система объединяет преимущества каждого из моделирующих устройств.

Размещение блока SLPS в функциональной схеме Simulink дает возможность Simulink использовать механизм анализа PSpice. Для блоков SLPS можно использовать схемы PSpice, созданные при помощи редактора схем Capture.

Поскольку SLPS является инструментом связи между MATLAB/Simulink и PSpice, эта статья предполагает, что читатель умеет пользоваться программами MATLAB/Simulink и OrCAD. Самую последнюю версию SLPS можно загрузить по адресу: www.cybernet.co.jp/slps/download.

Перед установкой SLPS на компьютере сначала должны быть установлены следующие программы: Matlab 6.5 и OrCAD 10.0 с SP2 (или более новые версии этих программ).

Для выполнения моделирования с помощью SLPS необходимо иметь лицензии для вышеупомянутых программ MATLAB/Simulink и OrCAD. Дополнительно файл лицензии Cadence должен включать лицензию SLPS.

Перед установкой SLPS необходимо закрыть все открытые приложения. Установка SLPS выполняется запуском установочного файла SLPS. Необходимо установить следующие два пути к папке SLPS в MATLAB:

- [SLPS]/SLPS
- [SLPS]/SLPSdemos

Замечание: [SLPS] — папка, куда установлен SLPS.

После этого SLPS готов к использованию. Для того чтобы деинсталлировать SLPS, в панели управ-

ления Windows необходимо дважды щелкнуть по **Add/Remove Programs**, выбрать SLPS из списка и затем нажать на кнопку **Add/Remove**. Когда полностью удаляется SLPS, стираются оба пути к папке SLPS, установленные в MATLAB. При установке новых версий программы OrCAD SLPS необходимо установить повторно. Перед повторной установкой SLPS требуется сначала деинсталлировать его. Установка пути в MATLAB не требуется, если этот пакет не модернизировался.

Порядок осуществления анализа

Для использования схем PSpice в качестве блоков SLPS необходим файл CIR (файл с расширением cir). Файл этого типа содержит информацию установок анализа PSpice и список используемых цепей. Файл CIR создается при выполнении моделирования в PSpice A/D. Для использования схемы, созданной в PSpice для блока SLPS, необходимо выполнить анализ переходных процессов в PSpice A/D. Анализ выполняется только для создания файла CIR. Когда изменяется схема, для того чтобы изменения были отражены в файле CIR, всегда повторяется анализ в PSpice. После окончания моделирования необходимо закрыть окно PSpice.

Вызовем библиотеку SLPS в командном окне MATLAB с помощью команды **slpslib** (рис. 1). Интерфейс SLPS используется для размещения блока SLPS на функциональной схеме Simulink.

Дважды щелкнем по блоку SLPS, чтобы открыть окно SLPS Settings (рис. 2).

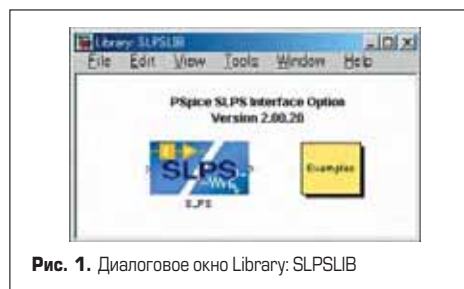


Рис. 1. Диалоговое окно Library: SLPSLIB

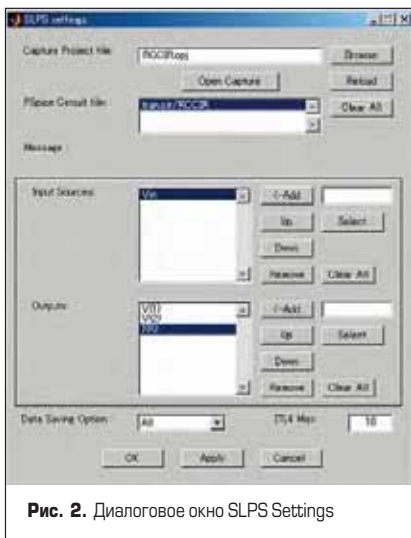


Рис. 2. Диалоговое окно SLPS Settings

В диалоговом окне SLPS Settings необходимо выполнить следующие шаги:

1. В поле **Capture Project file** обозначим файл проекта Capture (*.orj), содержащий назначенную схему PSpice.
2. Закроем редактор схем Capture, если он до этого был открыт. Откроем проект (*.orj), для этого нажмем кнопку Open Capture в диалоговом окне SLPS Settings (рис. 2).
3. Если схема была изменена, можно обновить информацию, нажав кнопку **Reload**. Для очищения позиций в окне SLPS Settings нажмем кнопку **Clear All**.
4. Все файлы CIR из проекта, выбранные в поле **Capture Project file**, будут перечислены в поле **PSpice Circuit file**.
5. Сообщения об ошибках и состоянии моделирования отображаются в поле **Message**.
6. Для того чтобы вставить входные данные на схеме (из Simulink в блок SLPS) в списке **Input Sources**, определим источник напряжения (V) и источник тока (I). Если выбран источник напряжения, входной сигнал будет подаваться на схему как значение напряжения, а если выбран источник тока, оно будет подаваться как значение тока. В этом списке перечислены входные сигналы к блоку SLPS. Должен быть определен не менее чем один входной источник.
7. Нажмем кнопку **Input Select**, и все источники, содержащиеся на схеме, на которую ссылается выбранный файл CIR, будут перечислены в диалоговом окне **IN/OUT select**. Щелкнем по источнику, который необходимо использовать, и добавим его в список **Input Sources**.
8. Используем кнопки **Input add/move/delete** для добавления и изменения последовательности или удаления позиций из списка в **Input Sources**.
9. В списке **Outputs** определим данные, которые выводятся из блока SLPS в Simulink.
10. Нажмем кнопку **Outputs Select**, и все выходные переменные в файлах CIR будут перечислены в диалоговом окне **IN/OUT select**. Щелкнем на переменных, которые используются, и добавим их в список **Outputs**.
11. Используем кнопки **Outputs add/move/delete** для добавления и изменения последователь-

ности или удаления позиций из списка **Outputs**.

12. В поле **Data Saving Option** определим, какие данные сохраняются в течение моделирования в PSpice. Они сохраняются в файле данных (*.dat) PSpice. В выпадающем списке можно выбрать следующие опции:
 - **ALL** — сохраняются все данные. В этом случае PSpice будет долго анализировать сохраняемую информацию, и размер файла будет большой;
 - **Selection Only** — сохраняются только данные, определенные в списке **Outputs**;
 - **None** — данные анализа не сохраняются, что ускоряет анализ.
13. В поле **ITL4 Max** выберем значение ITL4. Опция ITL4 устанавливает верхний предел числа повторных вычислений в шаге при выполнении анализа переходных процессов. Увеличивая ее значение, можно увеличить максимальное число вычислений для получения результата моделирования. Увеличение этого значения не оказывает влияния на точность результата. Если возникнут проблемы со сходимостью с SLPS, анализ автоматически выполняется снова после увеличения значения ITL4 в 10 раз. Если значение, увеличенное в 10 раз, превышает значение, определенное с ITL4 Max, анализ использует значение ITL4 Max, и если снова возникнут проблемы со сходимостью, анализ будет прерван. Если проблемы со сходимостью происходят часто, увеличим значение ITL4 Max, например, до 1000. Однако если определено очень большое значение для ITL4 Max, приложение будет занято вычислениями в течение долгого времени. Значения, которые могут быть установлены в качестве выходных данных SLPS:
 - узловое напряжение;
 - ток, проходящий через устройство;
 - рассеиваемая мощность устройства.
 Формат выходных данных SLPS соответствует PSpice:
 - **Node voltage** (узловое напряжение). Синтаксис: V ([NODENAME]). NODENAME — название узла на схеме.
 - **Current passing through a device** (протекающий через устройство ток). Синтаксис: I ([DEVICENAME]). DEVICENAME — позиционное название устройства. Направление тока, проходящего через устройство, соответствует PSpice. За положительное направление тока принимается направление от вывода 1 устройства к выводу 2.
 - **Pin inflow current** (Ток вывода). Синтаксис: I [PINNAME] ([DEVICENAME]). PINNAME — название вывода элемента. За положительное направление тока принимается направление тока, к выводу.
 - **Power dissipation of device** (Мощность рассеяния устройства). Синтаксис: W ([DEVICENAME]). Мощность рассеяния вычисляется как $I \cdot V$, но берется ее абсолютная величина, так что направление тока не имеет значения. Параметры моделирования Simulink устанавливаются в диалоговом окне **Simulation Parameters** (рис. 3). Для этого в меню **Simulation** окна Simulink выберем команду **Simulation Parameters**.

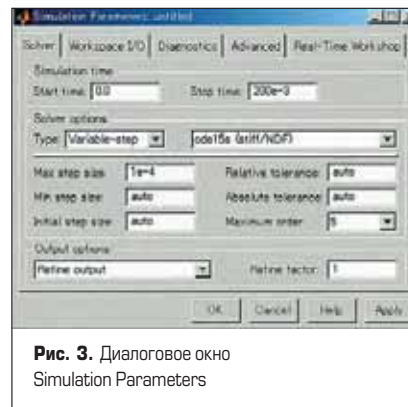


Рис. 3. Диалоговое окно Simulation Parameters

SLPS может использовать опции в поле Solver Options вкладки Solver (Решатель): **fixed step** (фиксированный шаг), **variable step** (переменный шаг).

Для анализа функциональной схемы Simulink содержащий блок SLPS в меню **Simulation** выберем команду **Start**.

Если возникнут проблемы со сходимостью в PSpice в процессе анализа с Simulink, необходимо увеличить параметр ITL4 в PSpice до значения, определенного в **ITL4 Max** в установках SLPS, и повторить анализ. Однако если PSpice не сможет выполнить вычисления, будет открыто окно, показанное на рис. 4.



Рис. 4. Диалоговое окно Convergence Error

Здесь: **Abort** — прекращение анализа; **Interact** — повторное вычисление при измененных опциях анализа PSpice.

Если выбрать **Interact**, откроется окно PSpice Option Parameter Setting (рис. 5).

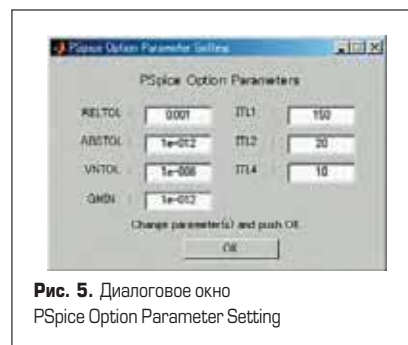


Рис. 5. Диалоговое окно PSpice Option Parameter Setting

После изменения этих параметров нажмем кнопку **OK**. Программа PSpice продолжит вычисление, используя измененные опции анализа. Описания опций анализа приведены в таблице.

Обычно если проблемы со сходимостью возникают в процессе анализа, устанавливаются значение ITL4 больше 1000. При этом можно улучшить сходимость, не изменяя точности анализа.

Необходимо проверять выходные данные SLPS в Simulink, используя блок Scope (осциллограф) (рис. 6).

Таблица. Опции анализа PSpice

Параметр	Описание
ITL1	максимальное число итераций в течение расчета режима цепи по постоянному току в рабочей точке (bias point)
ITL2	максимальное число итераций в течение анализа DC
ITL4	максимальное число итераций при переходе к следующему моменту времени в режиме TRAN
VNTOL	абсолютная точность расчета напряжений в режиме TRAN
ABSTOL	абсолютная точность расчета токов в режиме TRAN
RELTOL	относительная точность расчета напряжения и тока в режиме TRAN
GMIN	минимальная проводимость ветви цепи (проводимость ветви, меньшая GMIN, считается равной нулю)

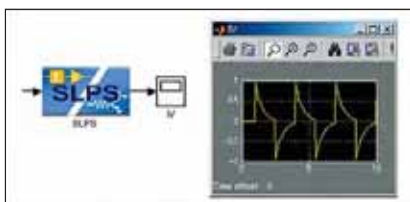


Рис. 6. Проверка выходных данных при помощи блока Scope (осциллограф) в Simulink

Когда анализ выполняется процессором PSpice, вызываемым из SLPS, так же как при моделировании в PSpice A/D, создаются файл данных (*.dat) и выходной файл (*.out). Однако если в окне SLPS Settings выбрано **None** в **Data Saving Option**, будет создан пустой файл данных PSpice. Файлы данных со следующими названиями создаются в той же папке, где находится файл CIR:

- файл данных PSpice: SLPS_[название файла CIR].dat;
- выходной файл PSpice: SLPS_[название файла CIR].out.

Можно проверить результаты, полученные в SLPS, открыв файл данных PSpice в PSpice A/D:

1. Запустим PSpice A/D — для этого выберем **Start/Program/OrCAD10.../PSpiceAD**.
2. Откроем файл данных PSpice, для этого в меню **File** выберем команду **Open**. Выберем файл данных, созданный с SLPS. При этом сигналы отобразятся в PSpice A/D (рис. 7).

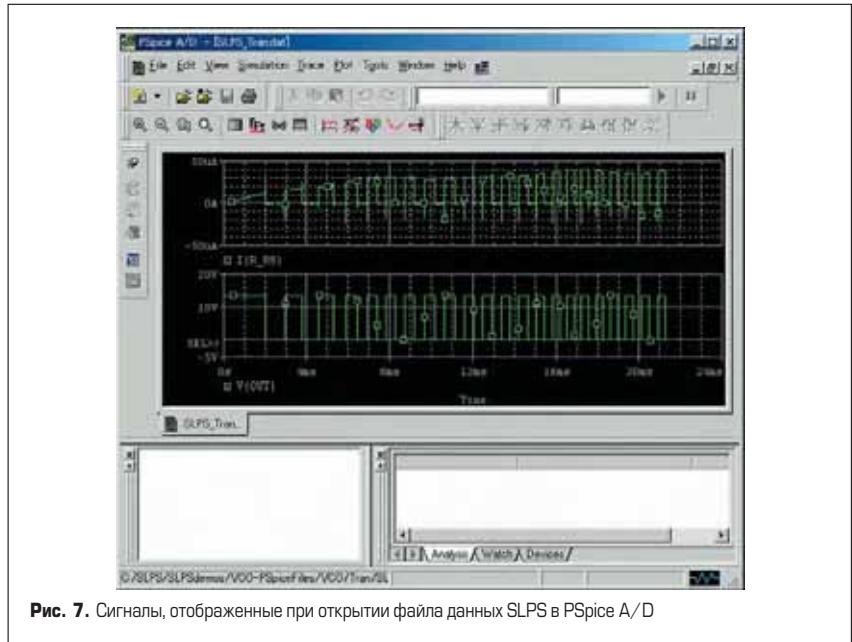


Рис. 7. Сигналы, отображенные при открытии файла данных SLPS в PSpice A/D

Пример анализа

Для выполнения анализа:

1. Запустим Capture — для этого в меню **Start** выберем **Start/All Programs/OrCAD10.../Capture (CIS)**.
2. Создадим проект с помощью Capture, для этого откроем диалоговое окно **New Project** выберем **New/Project...** в меню **File** редактора схем Capture (рис. 8).
3. В поле **Name** введем название проекта **MOSCKT**.
4. В поле **Create a New Project Using** определим тип нового проекта **Analog or Mixed A/D**.
5. Для сохранения файлов в поле **Location** определим местоположение рабочей папки **C:/WORK**.
6. Щелкнем **OK**.
7. В открывшемся диалоговом окне **Create PSpice Project** (рис. 9) выберем **Create a blank project** и щелкнем **OK**.
8. Откроется окно проекта и отобразится страница схемы (рис. 10).
9. Создадим схему проекта (рис. 11). Схема состоит из транзистора MOSFET (полевой транзистор МОП со структурой металл-оксид-полупроводник), резисторов и источников



Рис. 8. Диалоговое окно New Project

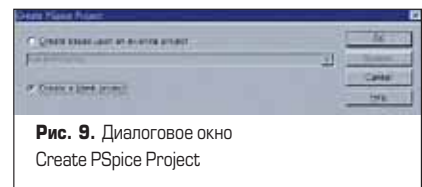


Рис. 9. Диалоговое окно Create PSpice Project

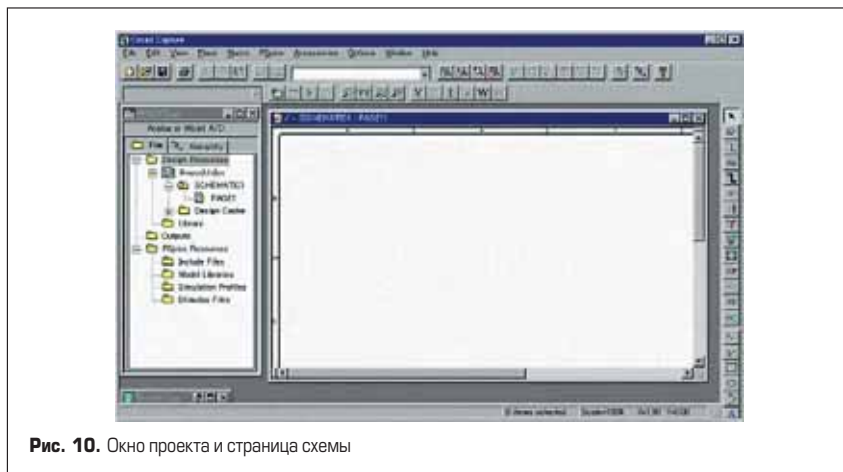


Рис. 10. Окно проекта и страница схемы

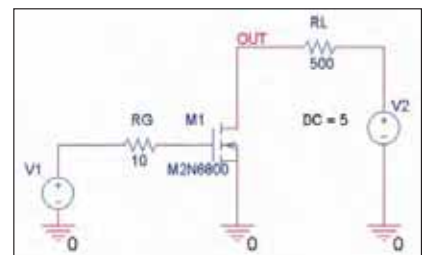


Рис. 11. Схема проекта

на проводе, присоединенном к стоку полевого транзистора.

При составлении схемы используем компоненты из следующих библиотек:

- M2N6800 (из POWERMOS.olb);
- R (из ANALOG.olb);
- VSRC (из SOURCE.olb).

10. После окончания редактирования схемы сохраним проект, выбрав для этого **Save** в меню **File**.

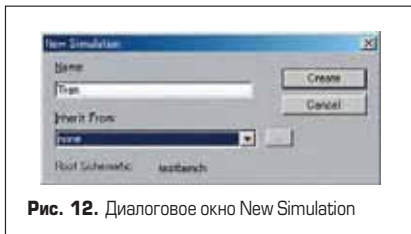


Рис. 12. Диалоговое окно New Simulation

11. Для установки анализа в меню **PSpice** выберем **New Simulation Profile**. В открывшемся диалоговом окне **New Simulation** (рис. 12) в текстовом поле **Name** напечатать **Tran**, выберем **none** в выпадающем списке **Inherit From** и нажмем кнопку **Create**.

12. В открывшемся диалоговом окне **Simulation Settings** (рис. 13), определим следующее:
 а. В выпадающем списке **Analysis type** выберем тип анализа **Time Domain (Transient)**;
 б. В поле **Run to time** определим конечное время моделирования — **3u** (3 мкс).

13. Нажмем кнопку **Применить**, а затем **ОК**.

14. Выберем в меню **PSpice** команду **Run**. Программа **PSpice A/D** откроется и запущется моделирование. Когда моделирование закончится, откроется пустое окно графика.

15. Для отображения сигнала поместим маркер на схеме. В меню **PSpice** выберем **Markers/Voltage Level**. Поместим маркер на провод, присоединенный к стоку полевого транзистора MOSFET (рис. 14).

Напряжения сигнала на стоке полевого транзистора в окне **PSpice** отображено на рис. 15.

Для создания функциональной схемы:

1. Запустим **MATLAB** в меню **Start**, для этого выберем **Start/All Programs/MATLAB.../MATLAB...**

2. Сделаем текущей папкой **MATLAB** рабочую папку, где сохранены файлы **PSpice: cd c:\work**.

3. Создадим новую модель в командном окне **MATLAB**, для этого в меню **File** выберем **New/Model**.

4. Вызовем библиотеку **SLPS** в командном окне **MATLAB** с помощью директивы **slpslib**.

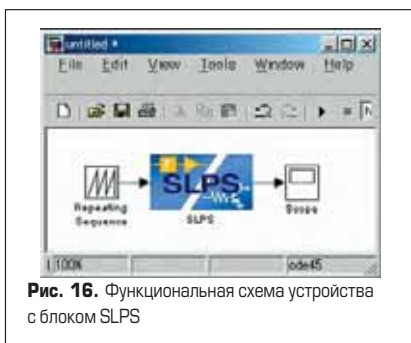


Рис. 16. Функциональная схема устройства с блоком SLPS

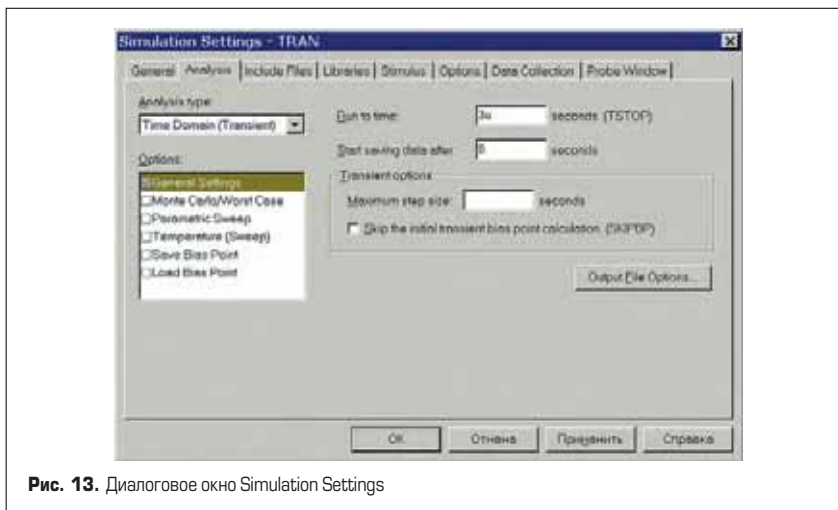


Рис. 13. Диалоговое окно Simulation Settings



Рис. 14. Маркер на стоке транзистора

5. Создадим функциональную схему устройства, используя блок **SLPS**. Используем блоки **Repeating Sequence** из библиотеки **Sources** и **Score** из библиотеки **Sinks** (рис. 16). Параметры блока **Repeating Sequence** приведены на рис. 17.

6. Когда модель будет закончена, сохраним ее, для этого в меню **File** выберем **Save**.

7. Сохраним ее как **mosckt.mdl** в текущей папке **MATLAB**.

Замечание: если модель не сохранена в текущей папке, нельзя установить приведенные ниже параметры блока **SLPS**.

Для наладки параметров блока **SLPS**:

1. Откроем диалоговое окно **SLPS settings**, дважды щелкнув по блоку **SLPS** (рис. 18). В поле **Capture Project file** обозначим созданный файл проекта **mosckt.opj**. Файлы **cir**, включенные в проект (созданные для каждой установки анализа **PSpice**), перечисляются в поле **PSpice Circuit File**. Здесь указан только файл **tran.cir**, поэтому он выбирается автоматически.

2. Обозначим вход и выход блока **SLPS**. Нажмем кнопку **Select** в установочном окне **SLPS Settings** справа от **Input Sources**.

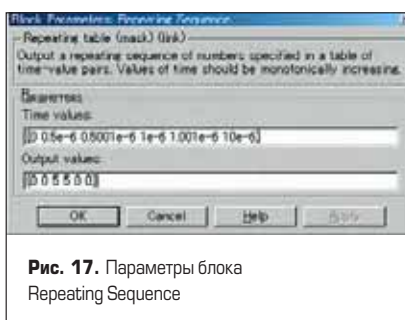


Рис. 17. Параметры блока Repeating Sequence

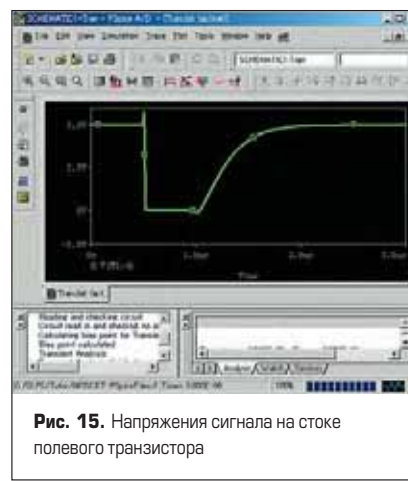


Рис. 15. Напряжения сигнала на стоке полевого транзистора

3. В открывшемся диалоговом окне **IN/OUT select** (рис. 19) выберем **V1** как входной источник и нажмем кнопку **Close**. Источник **V1** появится в списке **Input Sources** окна **SLPS Settings**.

4. Для обозначения выхода нажмем кнопку **Select** справа от **Output** в установочном окне **SLPS Settings** и в открывшемся диалоговом окне **IN/OUT select** выберем из списка **V(OUT)**. Затем нажмем кнопку **Close**. **V(OUT)** появится в списке **Output** окна **SLPS Settings**.

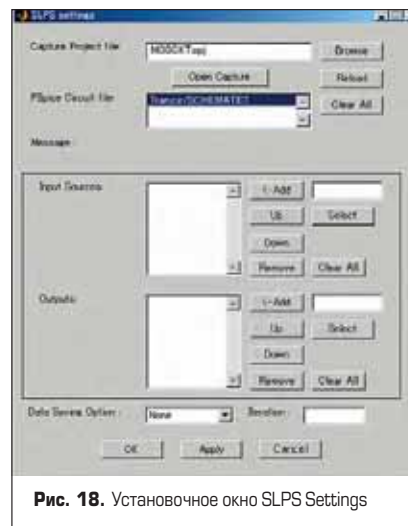


Рис. 18. Установочное окно SLPS Settings



Рис. 19. Диалоговое окно IN/OUT select

5. Закроем окно SLPS Settings, нажав кнопку **ОК** в нижней части окна.

Параметры моделирования Simulink устанавливаются в диалоговом окне **Simulation parameters**. Для установки параметров моделирования Simulink:

1. В меню **Simulation** окна Simulink выберем команду **Simulation Parameters**.



Рис. 20. Диалоговое окно Simulation Parameters

2. В открывшемся диалоговом окне **Simulation Parameters** (рис. 20) в поле **Simulation time** установим время окончания анализа **Stop time** такое же, как и в PSpice — $3e-6$ (3 мкс). В поле **Solver options** в выпадающих списках **Type**: выберем фиксированный шаг **Fixed step, Discrete**. В поле **Fixed step size** (размер фиксированного шага) напечатаем $1e-9$. После окончания установки параметров сохраним их, выбрав для этого **Save** в меню **File**.
3. Запустим моделирование, выбрав **Start** в меню **Simulation**.
4. После окончания анализа для отображения сигнала дважды щелкнем по блоку **Scope** (рис. 21). При этом отобразится тот же са-

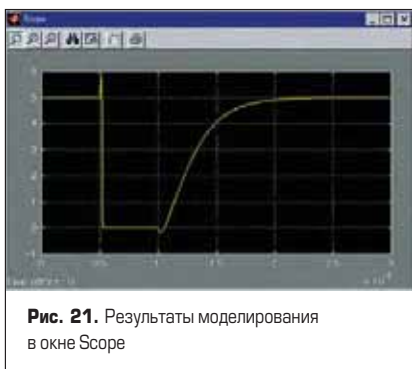


Рис. 21. Результаты моделирования в окне Scope

мый сигнал, что получился в результате моделирования в PSpice (рис. 15).

Примеры анализа

Пример 1. Схема RC

Вспользуемся простой схемой RC, показанной на рис. 22 (файл проекта RCCIR.opj).

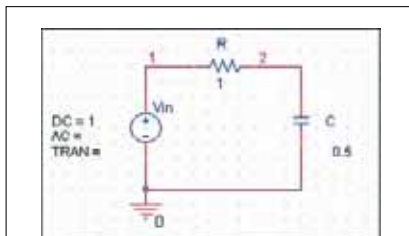


Рис. 22. Простая RC-схема

Функциональная схема Simulink (rcmdl.m) показана на рис. 23.



Рис. 23. Функциональная схема проверки блока SLPS

Параметры блока SLPS установлены, как показано на рис. 2:

- Capture Project file: **RCCIR.opj**;
- PSpice Circuit file: **tran.cir**;
- Input Sources: **Vin** (входным для Simulink является значение напряжения источника напряжения Vin);
- Outputs: **V (1) / V (2) / I (R)** (для Simulink значения напряжения в узлах 1 и 2 на схеме и значение тока, протекающего через резистор R, являются выходными данными);
- Data Saving Option: **None** (для ускорения анализа выбрано None, чтобы не сохранять проанализированные данные в SLPS);
- ITL4 Max: **10** (проблемы со сходимостью в таких схемах бывают редко, потому что это линейная схема, таким образом, оставляем значение ITL4 Max по умолчанию 10). Результаты моделирования (рис. 24) могут быть получены при выполнении **rcplt.m** в командном окне MATLAB или щелчком на значке **START** на функциональной схеме.

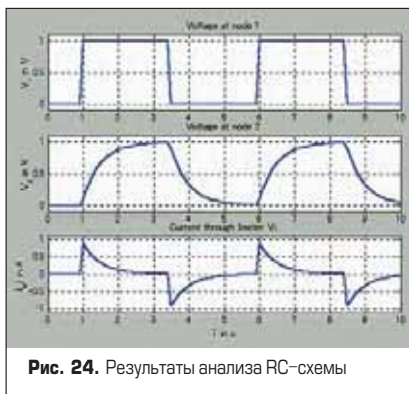


Рис. 24. Результаты анализа RC-схемы

Пример 2. Схема управления скоростью двигателя постоянного тока.

Схемы силовой электроники, такие как преобразователи электропитания и устройства управления двигателем, могут за короткое время переключать большой ток и высокое напряжение. Если перед вами стоит задача смоделировать полную схему такого типа, используя моделирующее устройство, подобное PSpice, необходимо описать все элементы устройства с помощью принципиальных схем. В этом случае полная схема становится сложной, так что в начальной стадии проекта придется потратить много времени. По этой же причине могут возникнуть проблемы со сходимостью, и может потребоваться более длительное время для анализа данных, либо он может прерваться.

Использование на первой стадии проектирования точных моделей приводит к очень медленному моделированию. Поэтому рекомендуется применять упрощенные модели для начальных стадий проектирования. На рис. 25 показана схема управления скоростью двигателя постоянного тока.

На этой схеме имеется обмотка статора (возбуждения), которая создает постоянный магнитный поток возбуждения. Эта обмотка моделируется индуктивностью L_S и сопротивлением R_S . Когда напряжение V_S прикладывается к обмотке статора, протекает ток I_S , создавая поток ψ . Обмотка ротора смоделирована индуктивностью L_R и резистором R_R . Когда напряжение V_R прикладывается к обмотке ротора, протекает ток I_R . V_R является выходным сигналом контроллера, который управляет дифференциальным входным напряжением V_D . Запишем уравнение цепи для обмотки статора:

$$V_S = R_S I_S + \frac{d\psi}{dt}$$

Вообще, магнитный поток ψ — нелинейная функция тока статора I_S . Для простоты предположим, что ψ — линейная функция I_S :

$$\psi = L_S I_S$$

Тогда получим:

$$V_S = R_S I_S + L_S \frac{dI_S}{dt} \quad \text{или} \quad \frac{dI_S}{dt} = \frac{V_S - R_S I_S}{L_S}$$

Для уравнения цепи обмотки ротора получим:

$$V_R = R_R + L_R + L_R \frac{dI_R}{dt} + V_I$$

или

$$\frac{dI_R}{dt} = \frac{V_R - R_R I_R - V_I}{L_R}$$

Напряжение V_I индуцируется в обмотке ротора, когда он вращается в поле возбуждения. Напряжение V_I может быть записано как произведение постоянной C_1 , потока ψ и угловой скорости ω :

$$V_I = C_1 \psi \omega$$

Для моментов на оси ротора можно записать уравнение равновесия:

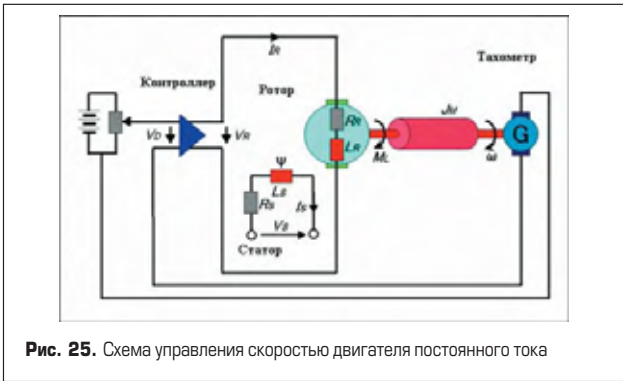


Рис. 25. Схема управления скоростью двигателя постоянного тока

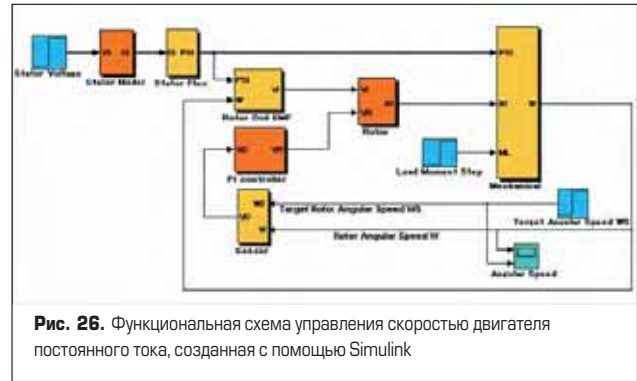


Рис. 26. Функциональная схема управления скоростью двигателя постоянного тока, созданная с помощью Simulink

$$J_M \frac{d\omega}{dt} = M_R - M_L - M_F,$$

где $M_R = C_2 \psi I_R$, $M_F = C_3 \omega$.

- M_R — вращающий момент ротора, который является линейной функцией потока возбуждения ψ и тока ротора, умноженного на константу C_2 .
- M_L — заданный момент нагрузки.
- M_F — момент внутреннего трения двигателя, который принимается линейной функцией угловой скорости, умноженной на константу C_3 .
- J_M — момент инерции ротора.

Контроллер управляется разностью напряжений V_D между напряжением на потенциометре скорости, который устанавливает номинальную угловую скорость, и напряжением на тахометре, которое пропорционально угловой скорости. Можно записать V_D как разность между угловыми скоростями, умноженную на константу C_4 .

$$V_D = C_4(\omega_S - \omega).$$

Для контроллера PI (пропорционально-интегрального регулирования) запишем напряжение, прикладываемое к обмотке ротора:

$$V_R = C_5 \int V_D dt + C_6 V_D.$$

Для моделирования сначала воспользуемся функциональной схемой системы, созданной с помощью Simulink (рис. 26).

Параметры системы:

- $C1 = 0.50$ $C2 = 0.50$;
- $C3 = 0.10$; % $N \cdot m \cdot s$ $C4 = 0.01$; % $V \cdot s$;
- $C5 = 606.0$; % $1/s$ $C6 = 100.0$;
- $LR = 0.10$; % $H = V \cdot s / A$ $LS = 10.0$; % $H = V \cdot s / A$;
- $RR = 0.50$; % $Ohm = V / A$ $RS = 500.0$; % $Ohm = V / A$;
- $JM = 1.0$; % $kg \cdot m^2$.

Результаты моделирования данной схемы Simulink (графики угловых скоростей) приведены на рис. 27.

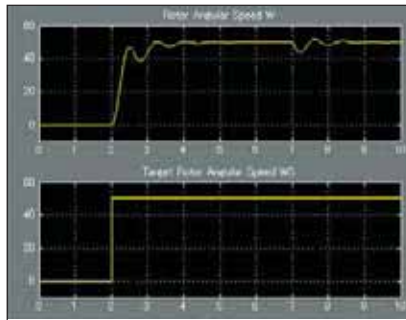


Рис. 27. Графики угловых скоростей

Напряжение статора V_S включается за время $t = 1$ с, а через 2 с ротор начинает вращаться. Приблизительно через 6 с угловая скорость достигает установившегося значения 50 рад/с. Когда $t = 7$ с, момент нагрузки уменьшается до 43,6 рад/с. Поэтому контроллер увеличивает

V_R , и новое устойчивое состояние ω в рад/с достигается снова при $t = 10$ с.

В функциональной схеме управления скоростью двигателя постоянного тока, созданной в Simulink, заменим некоторые из блоков Simulink схемами, созданными в PSpice. На рис. 28 показана функциональная схема системы.

На рис. 29 показана функциональная схема Simulink, которая включает блок SLPS, созданный с помощью PSpice.

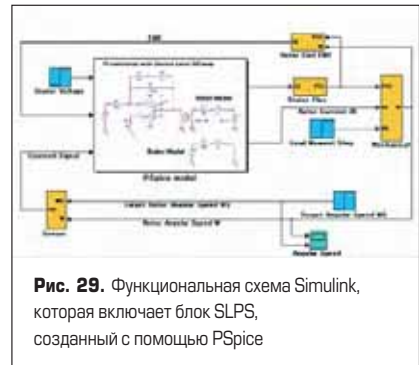


Рис. 29. Функциональная схема Simulink, которая включает блок SLPS, созданный с помощью PSpice

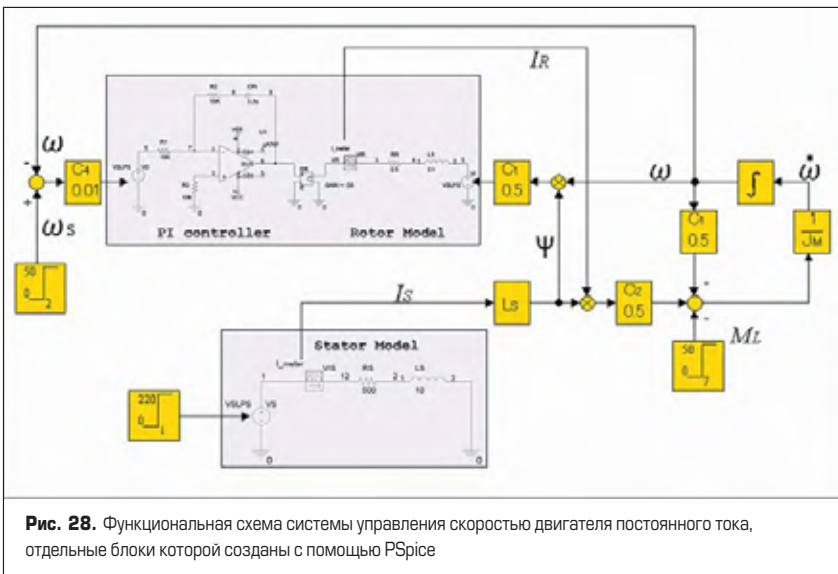


Рис. 28. Функциональная схема системы управления скоростью двигателя постоянного тока, отдельные блоки которой созданы с помощью PSpice

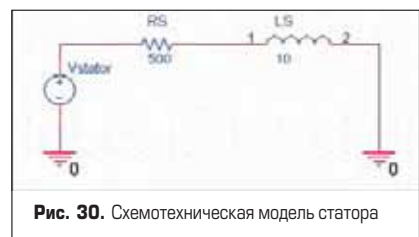


Рис. 30. Схематехническая модель статора

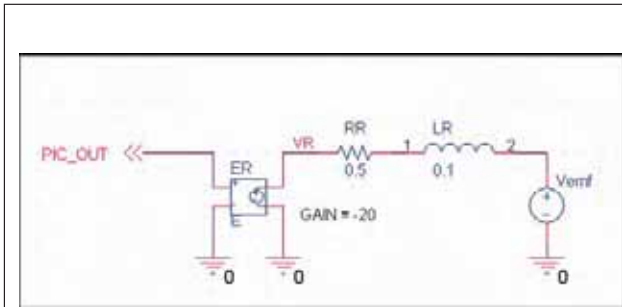


Рис. 31. Схематехническая модель ротора

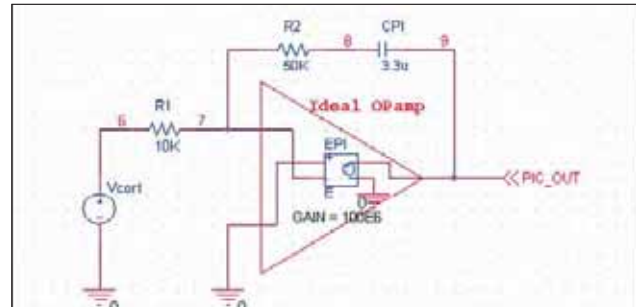


Рис. 32. Модель контроллера пропорционально-интегрального регулирования со схематехнической моделью в виде идеального ОУ

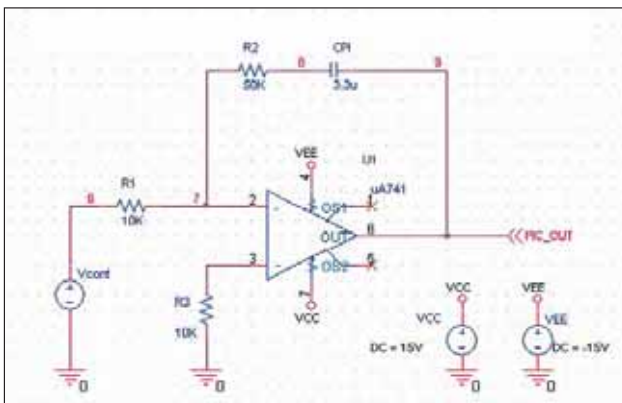


Рис. 34. Схема контроллера на операционном усилителе А741

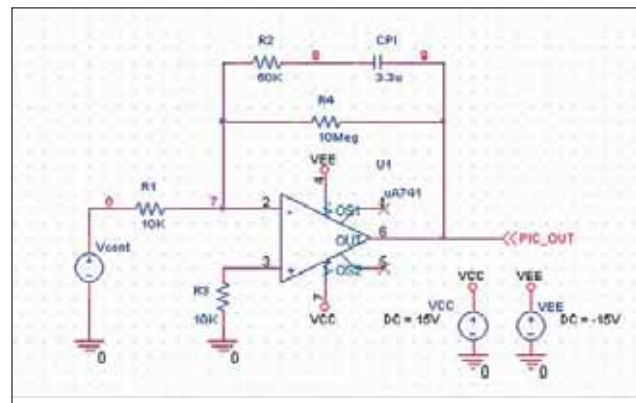


Рис. 36. Схема контроллера на основе ОУ с обратной связью по постоянному току

рального регулирования $V(9)$ (PIC_OUT) задается уравнением:

$$V(9) = -\left(\frac{1}{R_1 C} \int V_D dt + \frac{R_2}{R_1} V_D \right),$$

с коэффициентами:

$$\frac{1}{RC} = \frac{1}{10 \text{ кОм} \times 3,3 \text{ мкФ}} = 30,3,$$

$$\frac{R_2}{R_1} = \frac{50 \text{ кОм}}{10 \text{ кОм}} = 5.$$

Напряжение $V(9)$ управляет источником напряжения E_R , который имеет коэффициент передачи, равный -20 . Поэтому мы получаем:

$$V(3) = V_R = -20 \times V(9) = 606 \int V_D dt + 100 V_D.$$

Результаты моделирования показаны на рис. 33.

Мы теперь заменим идеальную модель ОУ в контроллере на модель реального ОУ-А741 (рис. 34).

После включения через 1 с двигатель начинает вращаться в противоположном направлении (рис. 35). Очевидно, модель контроллера с реальным ОУ ведет себя иначе, чем контроллер с идеальной моделью ОУ. Это происходит потому, что выход ОУ насыщается из-за входного напряжения смещения и очень высокого усиления при разомкнутой цепи обратной связи по постоянному току.

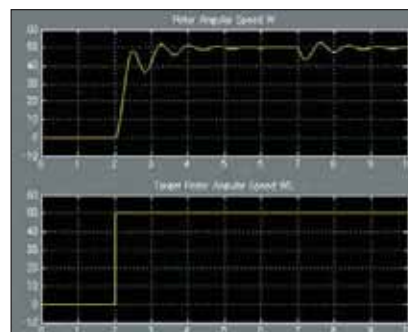


Рис. 33. Графики угловых скоростей при использовании контроллера на идеальном операционном усилителе

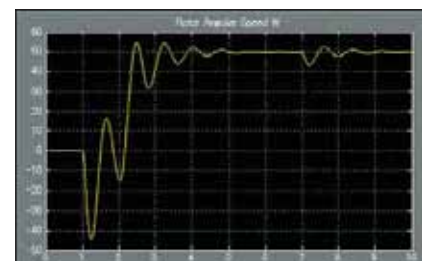


Рис. 35. Графики угловых скоростей при использовании контроллера, выполненного на операционном усилителе А741



Рис. 37. Результаты совместного моделирования Simulink-PSpice схемы управления скоростью двигателя постоянного тока с контроллером на основе ОУ с обратной связью по постоянному току ($R_4 = 10 \text{ МОм}$)

Для устранения проблемы со смещением вставим в цепь обратной связи резистор 10 МОм и ограничим усиление по постоянному току до 1000 (рис. 36).

Результаты совместного моделирования схемы управления скоростью двигателя постоянного тока с контроллером на основе ОУ с обратной связью по постоянному току в Simulink и PSpice показаны на рис. 37.

Поскольку выходное напряжение смещения операционного усилителя уменьшается, начальный переходный процесс также уменьшается. Отрицательным эффектом уменьшения усиления по постоянному току является то, что угловая скорость в установившемся режиме немного отличается от 50 рад/с.

В заключение отметим, что объединение схематехнического моделирования в программе OrCAD с функциональным моделированием

в программе MATLAB/Simulink обладает рядом достоинств, к которым следует отнести:

- моделирование с идеальными моделями, позволяющими доказать работоспособность моделируемого устройства;
- моделирование радиоэлектронных проектов с использованием моделей компонентов PSpice;
- большая библиотека компонентов для PSpice и блоков для Simulink;
- полный доступ к окружающей среде PSpice для проектирования и отладки;
- полный доступ к MATLAB для осуществления анализа и визуализация данных. ■

Литература

1. Раевич В. Д. Система проектирования OrCAD 9.2. М.: Солон-Р. 2001.
2. Дьяконов В.П. MATLAB 6/6.1/6.5 + Simulink 4.1/5 в математике и моделировании. Полное руководство пользователя. М.: СОЛОН-Пресс. 2003.
3. Златин И. Моделирование на функциональном уровне в OrCAD 9.2 // Компоненты и технологии. 2003. № 3–4.
4. Златин И. В Монте-Карло с OrCAD 9.2 // Компоненты и технологии. 2003. № 5.
5. Златин И. Графический анализ результатов моделирования в OrCAD 9.2 // Компоненты и технологии. 2003. № 7.
6. Златин И. Программа Advanced Analysis и режим анализа Smoke в PSD 15.0 и OrCAD 10.0 // Компоненты и технологии. 2005. № 4.
7. Златин И. Программа Advanced Analysis и режим анализа Sensitivity в PSD 15.0 и OrCAD 10.0 // Компоненты и технологии. 2005. № 5.
8. Златин И. Программа Advanced Analysis и режим анализа Optimizer в PSD 15.0 и OrCAD 10.0 // Компоненты и технологии. 2005. № 6, 8.
9. Златин И. Программа Advanced Analysis и режим анализа Monte Carlo в PSD 15.0 и OrCAD 10.0 // Компоненты и технологии. 2005. № 9.
10. Златин И. Программа Advanced Analysis и режим анализа Troubleshooting в PSD 15.0 и OrCAD 10.0 // Компоненты и технологии. 2006. № 1.
11. Златин И. Программа Advanced Analysis и режим анализа Parametric Plotter в OrCAD 10.5 // Компоненты и технологии. 2006. № 2.
12. Златин И. OrCAD 10.5 для начинающих пользователей // Компоненты и технологии. 2006. № 3, 4.
13. PSpice SLPS Interface User's Guide (Version 2.5). Cybernet Systems Co., Ltd. 2005.

В ближайшее время планируется выпуск серии магнитно-изолированных драйверов компании IXYS

Новые драйверы подходят для pin-to-pin замены традиционных оптически изолированных драйверов HCPL3180, HCPL3120, а также HCPL316J.

Компания IXYS объявила о скором выпуске на рынок серии магнитно-изолированных драйверов для IGBT и MOSFET-транзисторов. Новая серия IXM2 является прямой заменой таких продуктов, как HCPL3120 и HCPL3180. Также планируется выпустить драйвер для замены HCPL316J, но с небольшими отличиями,

определяющими основные преимущества, например, разработка компании IXYS имеет цифровой вход, в то время как выход останется такой же, как и у 316J. Одними из основных особенностей магнитно-изолированных драйверов являются более высокая — до 50 кВ/мкс, по сравнению с традиционными оптронами, а также более широкие диапазоны рабочих температур от –55 до +125 °С.

www.components.argussoft.ru