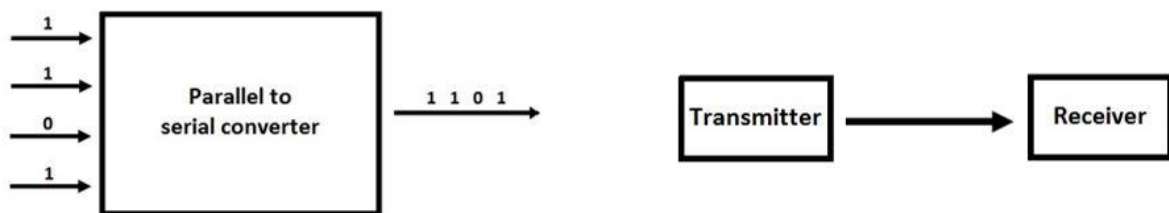


Руководство к лабораторному практикуму: Введение в цифровую электронику

С использованием платы Digilent Digital Electronics для NI ELVIS III



Left: *Parallel-in Serial-out Transmitter*

Right: *Serial Transmitter and Receiver*

Лабораторная работа 18:
Проект №3 – Цифровая связь

© 2018 National Instruments

All rights reserved. Данный ресурс и любые его части не могут быть скопированы или в любой форме воспроизведены иным способом без письменного разрешения издателя.

National Instruments относится с уважением к чужой интеллектуальной собственности и призывает к этому же своих читателей. Данный ресурс защищен законами об охране авторских прав и прав на интеллектуальную собственность. Вы имеете право передавать программное обеспечение и прочие материалы, разработанные с помощью описанного в данном ресурсе программного обеспечения, третьим лицам в соответствии с условиями приобретенной вами лицензии и другими законодательными ограничениями.

LabVIEW и National Instruments являются торговыми марками National Instruments.

Названия других упомянутых торговых марок и изделий являются собственностью их правообладателей.

Дополнительные ограничения ответственности: Читатель принимает все риски от использования данного ресурса и всей информации, теорий и программ, содержащихся или описанных в нем. Данный ресурс может содержать технические неточности, типографические ошибки, прочие ошибки и упущения, и устаревшую информацию. Ни автор, ни издатель не несут ответственности за любые ошибки или неточности, за обновление любой информации и за любые нарушения патентного права и прочих прав на интеллектуальную собственность.

Ни автор, ни издатель не дают никаких гарантий, включая, но не ограничиваясь, любую гарантию на достаточность ресурса и любой информации, теорий или программ, содержащихся или описанных в нем, и любую гарантию, что использование любой информации, теорий или программ, содержащихся или описанных в ресурсе, не нарушит любое патентное право или иное право на интеллектуальную собственность. РЕСУРС ПОСТАВЛЯЕТСЯ "КАК ЕСТЬ". ИЗДАТЕЛЬ ЗАЯВЛЯЕТ ОБ ОТКАЗЕ ОТ ЛЮБЫХ ГАРАНТИЙ, ЯВНО ВЫРАЖЕННЫЕ ИЛИ ПОДРАЗУМЕВАЕМЫХ, ВКЛЮЧАЯ, НО НЕ ОГРАНИЧИВАЯСЬ, ЛЮБЫЕ ПОДРАЗУМЕВАЕМЫЕ ГАРАНТИИ ТОВАРНОГО СОСТОЯНИЯ, ПРИГОДНОСТИ ДЛЯ КОНКРЕТНОЙ ЦЕЛИ И НЕНАРУШЕНИЯ ПРАВ ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ.

Издатель или автор не предоставляют прав или лицензий под любым патентным правом или иным правом на интеллектуальную собственность прямо, косвенно или лишением права на возражение.

НИ ПРИ КАКИХ ОБСТОЯТЕЛЬСТВАХ ИЗДАТЕЛЬ ИЛИ АВТОР НЕ НЕСУТ ОТВЕТСТВЕННОСТИ ЗА ПРЯМЫЕ, КОСВЕННЫЕ, ОСОБЫЕ, СЛУЧАЙНЫЕ, ЭКОНОМИЧЕСКИЕ ИЛИ ВТОРИЧНЫЕ УБЫТКИ, ПОНЕСЕННЫЕ ИЗ-ЗА ИСПОЛЬЗОВАНИЕ ЭТОГО РЕСУРСА ИЛИ ЛЮБОЙ ИНФОРМАЦИИ, ТЕОРИЙ ИЛИ ПРОГРАММ, СОДЕРЖАЩИХСЯ ИЛИ ОПИСАННЫХ В НЕМ, ДАЖЕ БУДУЧИ ПРЕДУПРЕЖДЕННЫ О ВОЗМОЖНОСТИ ПОДОБНЫХ УБЫТКОВ, И ДАЖЕ ЕСЛИ УБЫТКИ ВЫЗВАНЫ НЕБРЕЖНОСТЬЮ ИЗДАТЕЛЯ, АВТОРА ИЛИ ИНЫХ ЛИЦ, Применимое законодательство может не разрешить исключение или ограничение случайных или косвенных убытков, поэтому приведенные выше ограничения или исключения могут вас не касаться.

Лабораторная работа 18: Проект №3 – Цифровая СВЯЗЬ

В этой лабораторной работе вы примените знания и навыки, приобретенные в курсе лабораторных работ по цифровой электронике, для разработки и реализации устройства цифровой связи. Вы используете триггеры, счетчики, регистры сдвига и полупроводниковую память. Система связи будет отображать сообщения, а также обладать различными дополнительными функциями, которые вы выберете.

Цель работы

В этой лабораторной работе студенты должны:

1. Разработать передатчик и/или приемник для создания полноценной системы связи.
2. Исследовать обмен данными в параллельном и последовательном коде.
3. Соединять триггеры в регистры сдвига.
4. Научиться использовать функциональный генератор в качестве источника тактовых импульсов.
5. Получить возможность использовать стандартную схему кодирования символов или создать свою собственную схему для передачи более сложных сообщений.
6. Исследовать процесс шифрования и дешифрования сообщений.

Необходимые инструментальные средства и технологии

Платформа: NI ELVIS III

Измерительные приборы, необходимые в этой лабораторной работе:

- Функциональный генератор

Примечание: для работы с приборами требуется набор кабелей и аксессуаров для NI ELVIS III (приобретается отдельно).

- ✓ Руководство пользователя:
<http://www.ni.com/en-us/support/model.ni-elvis-iii.html>
- ✓ Учебные пособия:
https://www.youtube.com/playlist?list=PLvcPIuVaUMIWm8ziaSxv0gwtshBA2dh_M
- ✓ Установка поддержки программной лицевой панели:
<http://www.ni.com/documentation/en/ni-elvis-iii/latest/getting-started/installing-the-soft-front-panel/>

Технические средства:
Плата Digilent Digital Electronics Board for NI ELVIS III

- ✓ Руководство пользователя:
NI Digital Electronics Board:
<http://www.ni.com/pdf/manuals/376627b.pdf>

Электронные компоненты

- Проводники

Программное обеспечение:
NI Multisim 14.0.1 Education Version или выше

- ✓ Установка Multisim:
http://www.ni.com/gate/gb/GB_ACA_DEMICEVALMULTISIM/US
- ✓ Справка Multisim Help:
<http://www.ni.com/multisim/technical-resources/>

Программное обеспечение:
NI LabVIEW FPGA Vivado 2014.4

- ✓ Установка:
<http://www.ni.com/download/labview-fpga-module-2015-sp1/5920/en/>
Примечание: драйвер Digilent (Инсталлятор, указанный выше автоматически загружает на ваш компьютер инсталлятор, указанный ниже)
- ✓ Перейдите по ссылке:
C:\NIFPGA\programs\Vivado2014_4\data\xicom\cable_drivers\nt64\digilent
- ✓ Установите: install_digilent.exe

Ожидаемые результаты

В этой лабораторной работе вы должны собрать для отчета:

- Скриншот готовой схемы
- (необязательно) Зарисовки и скриншоты схем шифрования и дешифрования
- Видео работающей схемы
- Скриншоты схемы после внесения любых изменений
- Ответы на вопросы из раздела *Заключение*

Преподавателю, скорее всего, необходимо предъявить полный отчет о работе. Узнайте у вашего преподавателя, есть ли конкретные требования к отчету или шаблон для его оформления.

1.1 Сведения из теории

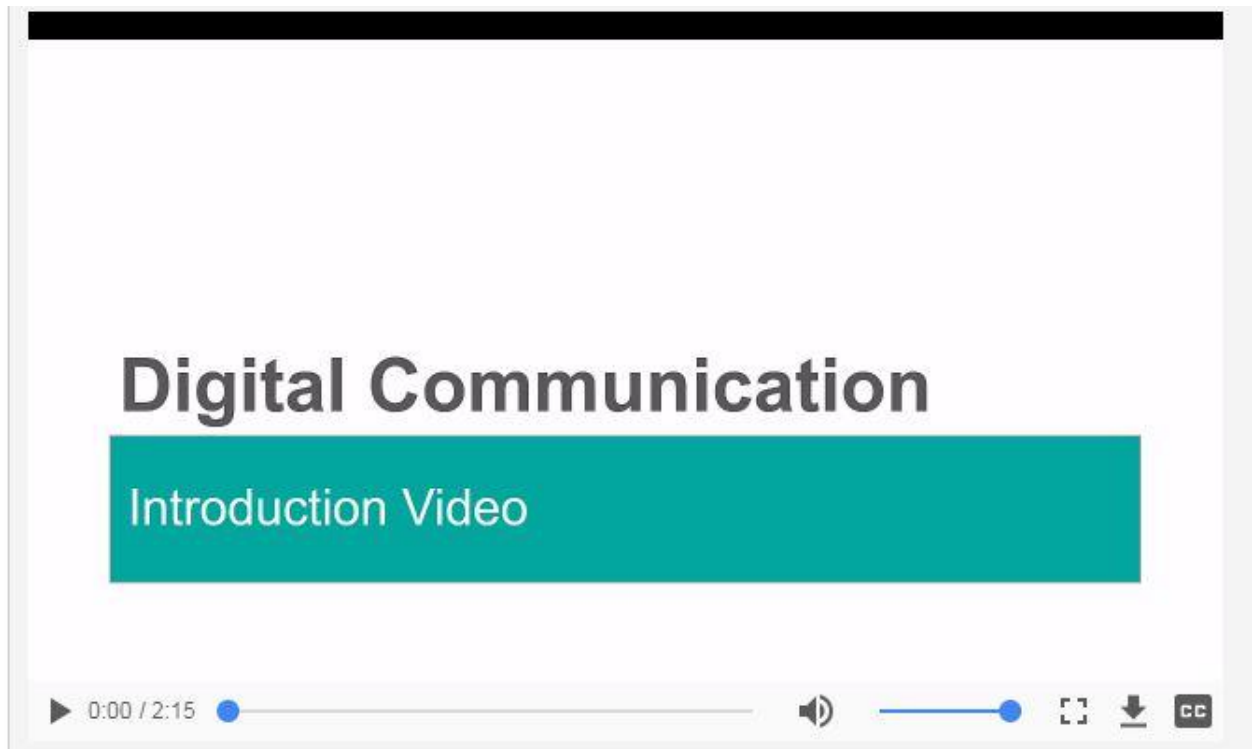


Рисунок 1-1 Скриншот видео Просмотр видео здесь: <https://youtu.be/HKjR2VnWFqw>

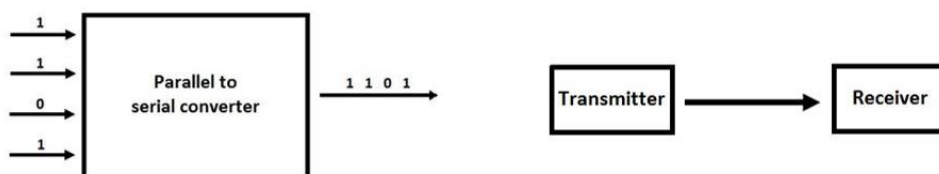


Краткое содержание видео

- В этом видео обсуждается использование полученных вами знаний и навыков для разработки и реализация устройства цифровой связи
- Устройства связи передают и принимают сообщения, используя методы цифрового обмена данными
- Обмен данными может выполняться в последовательном или параллельном коде
- Шифрование – это мера обеспечения безопасности, при которой содержанием цифровых сообщений манипулируют так, чтобы прочесть сообщение мог только тот, у кого есть ключ дешифрования

Обмен данными в последовательном и параллельном коде

Обмен данными в последовательном коде – это метод *цифровой связи*, при котором отдельные биты передаются по одному каналу связи по очереди, тем самым создавая длинный поток данных. Часто передатчики данных в последовательном коде получают все биты сообщения одновременно, а затем преобразуют данные с нескольких входов в один поток данных для передачи.

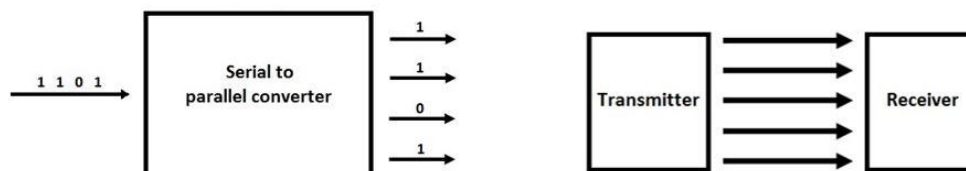


Left: *Parallel-in Serial-out Transmitter*

Right: *Serial Transmitter and Receiver*

Рисунок 1-2 Слева: передатчик с параллельным входом и последовательным выходом; справа: передатчик и приемник данных в последовательном коде

При *обмене данными в параллельном коде* все биты сообщения передается одновременно, каждый бит по собственному каналу. В системах обмена данными в параллельном коде для каждого бита информации необходима своя линия связи. Например, для передачи 8-битного сообщения передатчик должен быть соединен с приемником восемью каналами.



Left: *Serial-in Parallel-out Transmitter*

Right: *Parallel Transmitter and Receiver*

Рисунок 1-2 Слева: передатчик с последовательным входом и параллельным выходом; справа: передатчик и приемник данных в параллельном коде

Поскольку для обмена данными в параллельном коде требуется несколько линий связи, и, следовательно, больше материала, это более дорогой способ передачи данных на большие расстояния. Поэтому:

- Для передачи данных на большие расстояния чаще используется обмен данными в последовательном коде.
- Обмен данными в параллельном коде часто используется при передаче информации на очень короткие расстояния, например, в оперативной памяти компьютера или в интегральных схемах.

Хотя свои преимущества есть у обоих типов обмена данными, обмен информацией в последовательном коде становится все более популярным в различных областях (например, в электронике и компьютерной технике). Это происходит потому, что по мере развития технологий скорость передачи данных в последовательном коде растет, а способность сохранять целостность сигнала у нее выше, чем у передачи данных в параллельном коде.

Шифрование и дешифрование

Шифрование – это процесс манипулирования цифровым сообщением с помощью программных алгоритмов, чтобы не допустить прочтение сообщения без ключа дешифрования. Шифрование – это метод обеспечения безопасности, часто применяемый военными и правительством для уменьшения рисков прочтения перехваченной секретной информации.

Для дешифровки сообщения необходимо знать *ключ дешифрования*, созданный человеком, зашифровавшим сообщение. Правильный ключ дешифрования манипулирует зашифрованным сообщением, преобразуя его в исходное сообщение. Если вы решите добавить в данную лабораторную работу шифрование и дешифрование, вы будете использовать 8-разрядный двоичный ключ и логические элементы для шифрования и дешифрования сообщения.

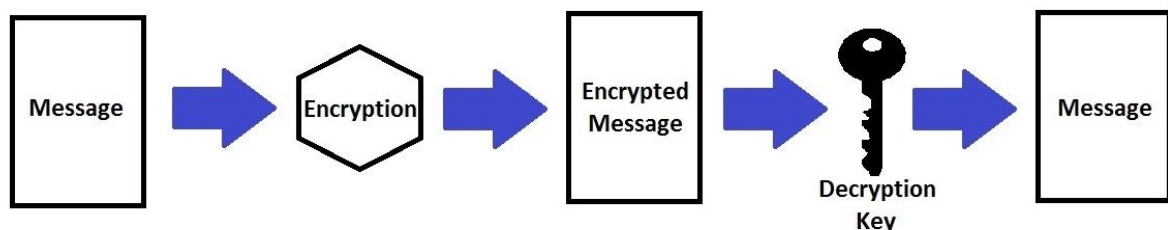


Рисунок 1-4 Процесс шифрования

Message – сообщение, Encryption – шифрование, Encrypted Message – зашифрованное сообщение, Decryption Key – ключ дешифрования

В этой работе ваша цель заключается в том, чтобы построить схему, которая генерирует, отправляет, принимает и отображает 8-разрядные сообщения в последовательном коде. В идеале необходимо обмениваться данными между двумя платами Digital Electronics, но вы можете отправлять и принимать сообщения с помощью одной платы. Кроме того, вы будете использовать программируемый функциональный генератор в качестве источника тактовых импульсов. Он заменит системный генератор тактовых импульсов платы Digital Electronics, частота которого (128 МГц) слишком велика.

Вы можете также добавить возможность передавать простые слова и фразы с помощью *схемы кодирования символов*. Кроме того, вы можете увеличить безопасность ваших сообщений, используя шифрования и дешифрование. Вы можете передавать сообщение между двумя платами Digital Electronics. Наконец, вместо передачи 8-битных сообщений вы можете модифицировать схему так, чтобы она передавала 7-битные сигналы, соответствующие сегментам семисегментного индикатора.

Перед началом выполнения работы рекомендуется освежить в памяти следующие работы:

- Лабораторная работа 10: Триггеры
- Лабораторная работа 11: Счетчики
- Лабораторная работа 13: Регистры сдвига
- Лабораторная работа 14: Полупроводниковая память

Исследование и планирование

- Осмысление и обобщение следующих ключевых слов поможет вам лучше понять, что потребуется при выполнении данной работы.
- Вы сможете вернуться сюда и добавить новую информацию в любой момент.

1-1 Передача данных:

1-2 Телекоммуникации:

1-3 Передача данных в последовательном коде:

1-4 Передача данных в параллельном коде:

1-5 Регистр сдвига:

1-6 Триггер (различные типы, доступные в Multisim, их функционирование):

1-7 Счетчик (типы, доступные в Multisim, типы выходных сигналов, функции сброса и синхронизации):

1-8 Функциональный генератор:

Ответы на следующие вопросы помогут вам подготовиться к проектированию. Они также гарантируют, что в своих исследованиях вы вспомните всю необходимую информацию, прежде чем начнете разрабатывать систему связи.

1-9 Составьте контрольный список всех функций, которые должны быть реализованы в вашей схеме. Включите в список также все дополнительные функции, которые планируете реализовать.

1-10 Опишите отличия обмена данными в последовательном и параллельном коде? Приведите достоинства и недостатки каждого способа и объясните, почему в этой работе используется передача данных в последовательном коде.

1-11 Как вы будете проектировать схему для передачи нескольких 8-битных двоичных сообщений?

1-12а У передатчика должен быть параллельный вход и последовательный выход. Объясните, как работает триггер, и определите, сколько триггеров вам потребуется для передачи 8-разрядных символов (8)?

1-12b Какой тип триггеров вам потребуется?

1-13 В этой работе вы используете функциональный генератор, чтобы иметь возможность управления частотой тактовых импульсов. В чем преимущества использования функционального генератора по сравнению с простой модификацией системного генератора тактовых импульсов платы Digital Electronics?

1.2 Разработка системы связи

На приведенной ниже блок-схеме показано, как будут соединены различные части вашей системы связи. Используйте ее в качестве рекомендации при соединении различных компонентов, описанных в инструкциях ниже.

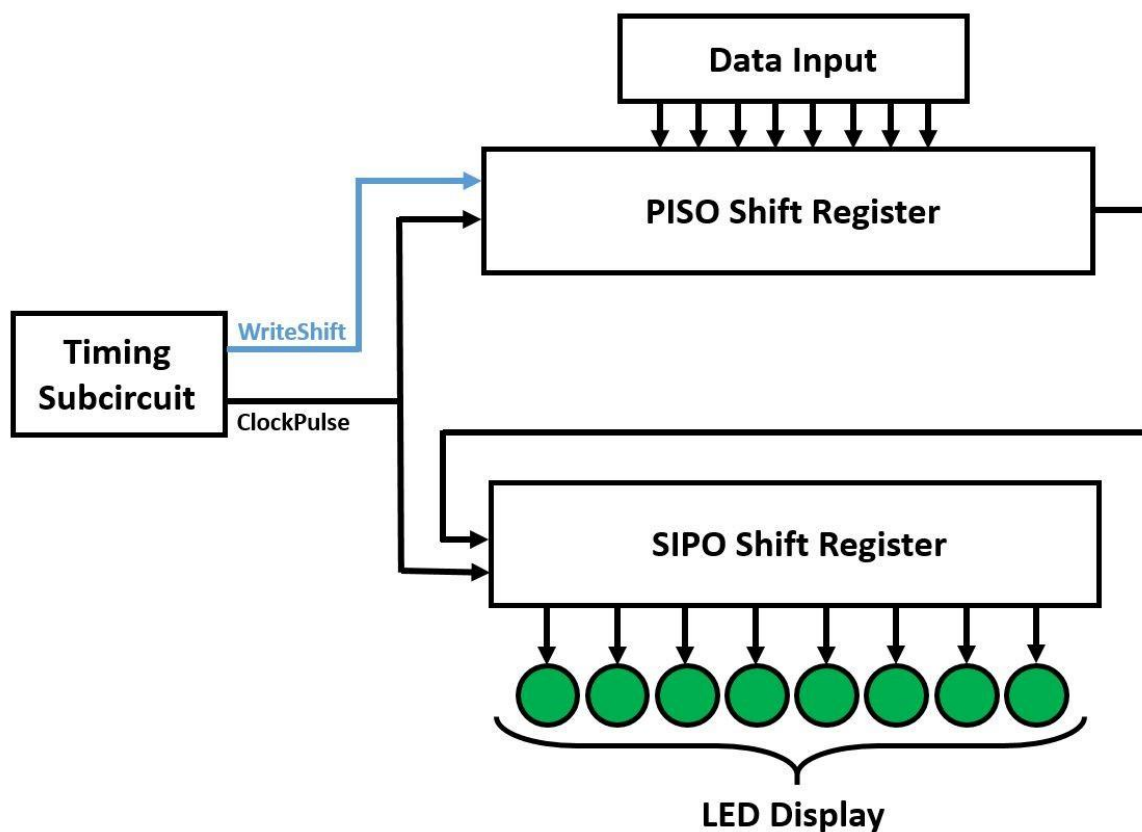


Рисунок 1-5 Блок-схема системы связи

Timing Subcircuit – подплата тактирования; WriteShift – сдвиг для записи;
ClockPulse – тактовые импульсы; Data Input – входные данные;
PISO Shift Register – регистр сдвига с параллельным входом и последовательным выходом;
SIPO Shift Register – регистр сдвига с последовательным входом и параллельным выходом;
LED Display – светодиодный индикатор

Начало работы

1. Откройте **новый** проект PLD.
2. Создайте следующие подсистемы PLD в основном рабочем пространстве PLD: **Timing**, **PISO Shift Register**, и **SIPO Shift Register**.

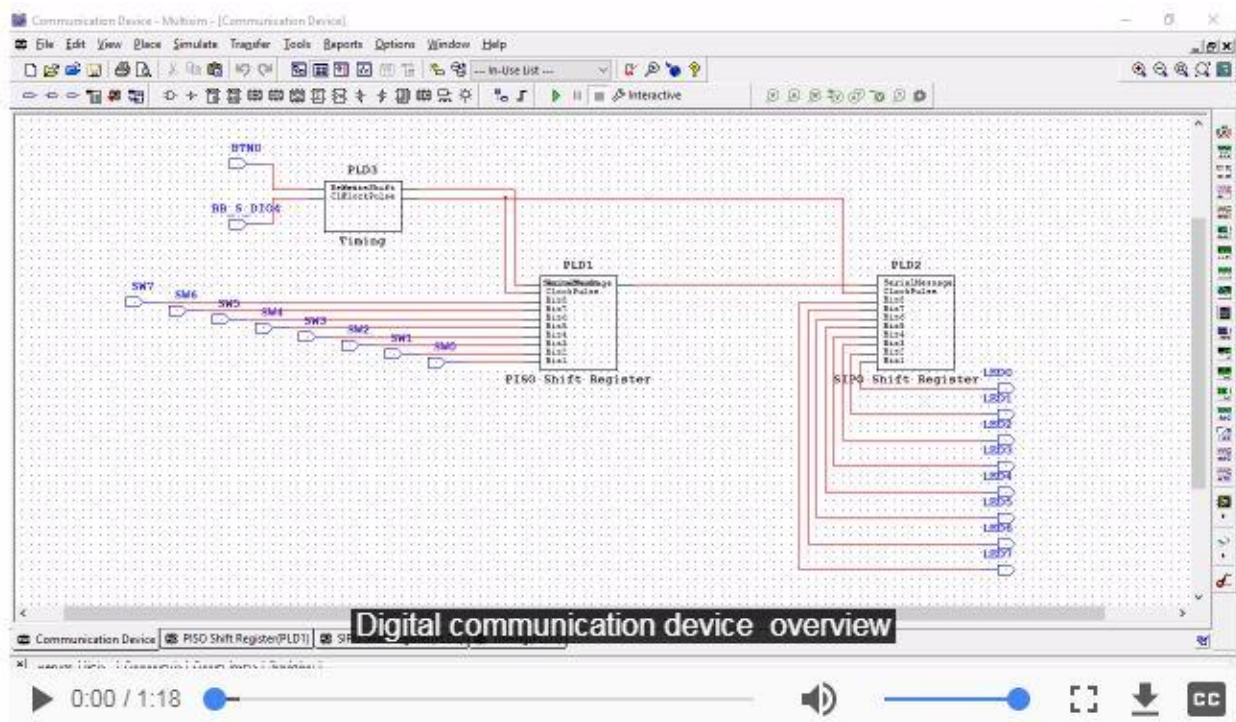


Рисунок 1-6 Видеообзор системы связи. Просмотр видео здесь: https://cf-ts.mythinkscape.com/video/Digital_Communication_Overview_no_cursor.mp4

Обзор компонентов:

- Подсхема Timing:
<https://mythinkscape.com/labs/v2/23305/steps/21204#Timing%20SubC>
- Подсхема PISO Shift Register (передатчик):
<https://mythinkscape.com/labs/v2/23305/steps/21204#PISO%20SubC>
- Подсхема SIPO Shift Register (приемник):
<https://mythinkscape.com/labs/v2/23305/steps/21204#SIPO%20SubC>
- Входные и выходные контакты:
<https://mythinkscape.com/labs/v2/23305/steps/21204#I/O%20Connectors>

Подсхема Timing:

3. В подсхеме Timing разместите следующие компоненты:
 - 1 двоичный счетчик (CNTR_7BIN)

Примечание: Он гарантирует, что на сдвиговый регистр с параллельным входом и последовательным выходом (PISO) будет подано необходимое количество тактовых импульсов.

- 1 D-триггер (FF_D_CO)

Примечание: Он управляет выходом WriteShift.

- 3 инвертора , 1 логический элемент AND с 3 входами и 1 логический элемент OR
4. Поместите следующие контакты и дайте им имена:
 - Входные контакты **Clk** и **Reset**
 - Выходные контакты **ClockPulse** и **WriteShift**
 5. Соберите подсхему таймирования, как показано на рисунке ниже:

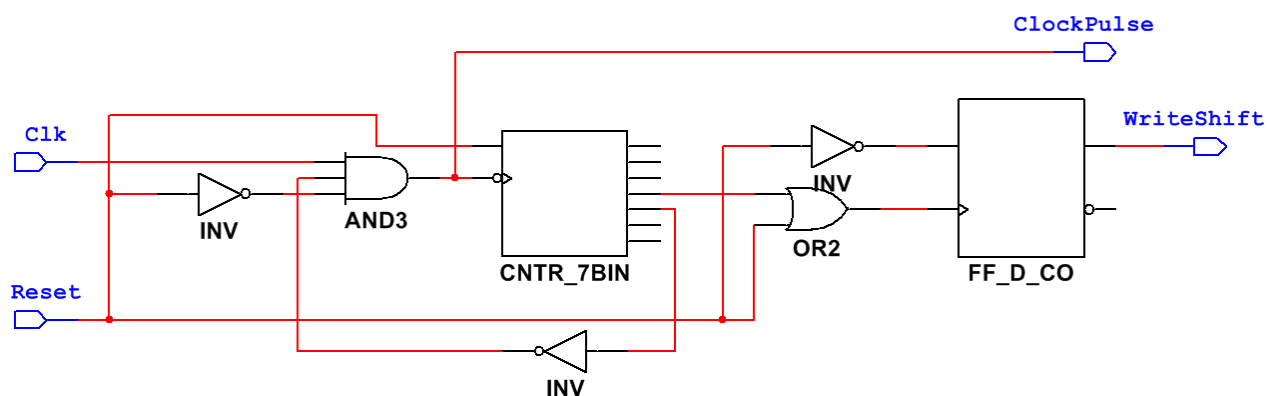


Рисунок 1-7 Подсхема Timing

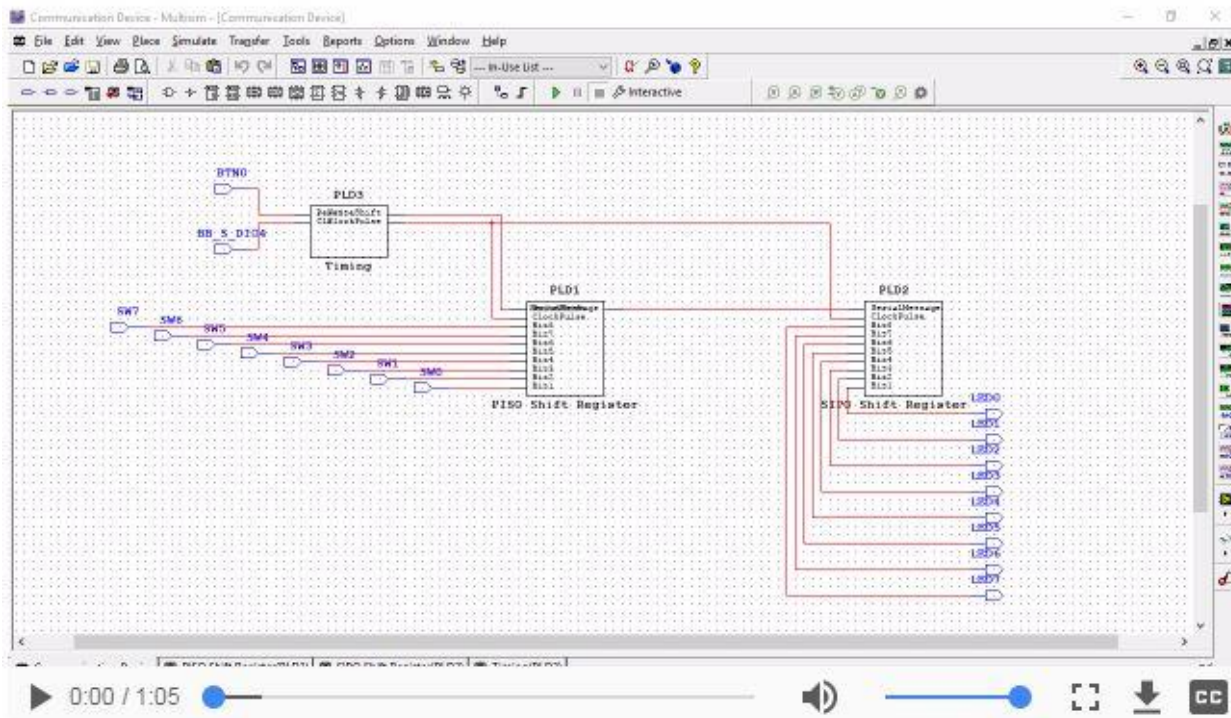


Рисунок 1-8 Видеообзор подсхемы Timing. Просмотр видео здесь: https://cf-ts.mythinkscape.com/video/Timing_no_cursor.mp4

Подсхема PISO Shift Register (передатчик):

6. В подсхеме **PISO Shift Register** разместите следующие компоненты:
 - 1 инвертор (INV)
 - Он обеспечивает правильное функционирование схемы.
 - 21 логический элемент NAND с 2 входами
 - Эти элементы организуют данные сообщения и обеспечивают их правильный ввод в триггеры.
 - 8 D-триггеров (FF_D_CO)
 - Триггеры сохраняют данные сообщения для передачи их в правильном порядке.
7. Поместите следующие контакты и дайте им имена:
 - Входы триггеров: **Bit1 – Bit8**
 - Выход регистра сдвига: **SerialMessage**
 - Вход тактовых импульсов: **ClockPulse**
 - Вход записи/сдвига: **WriteShift**
8. Соедините компоненты в 8-разрядный регистр сдвига с параллельным входом и последовательным выходом (PISO), как показано на рисунке ниже:

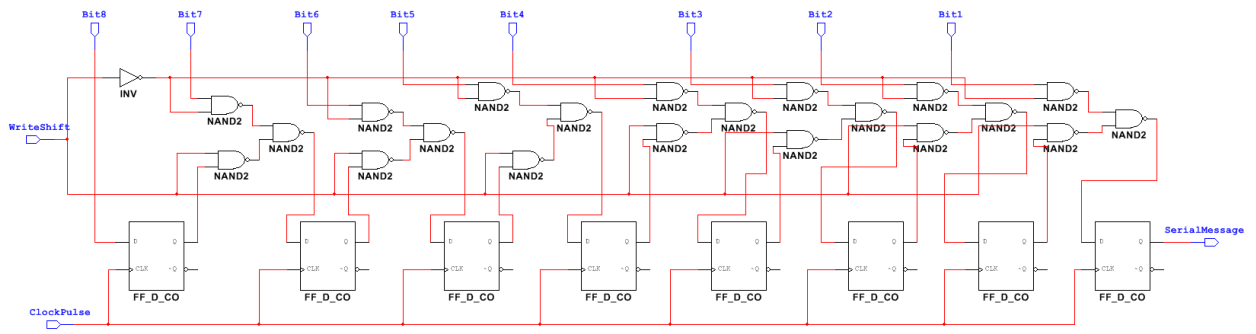


Рисунок 1-9 Схема 8-разрядного регистра сдвига с параллельным входом и последовательным выходом (PISO)

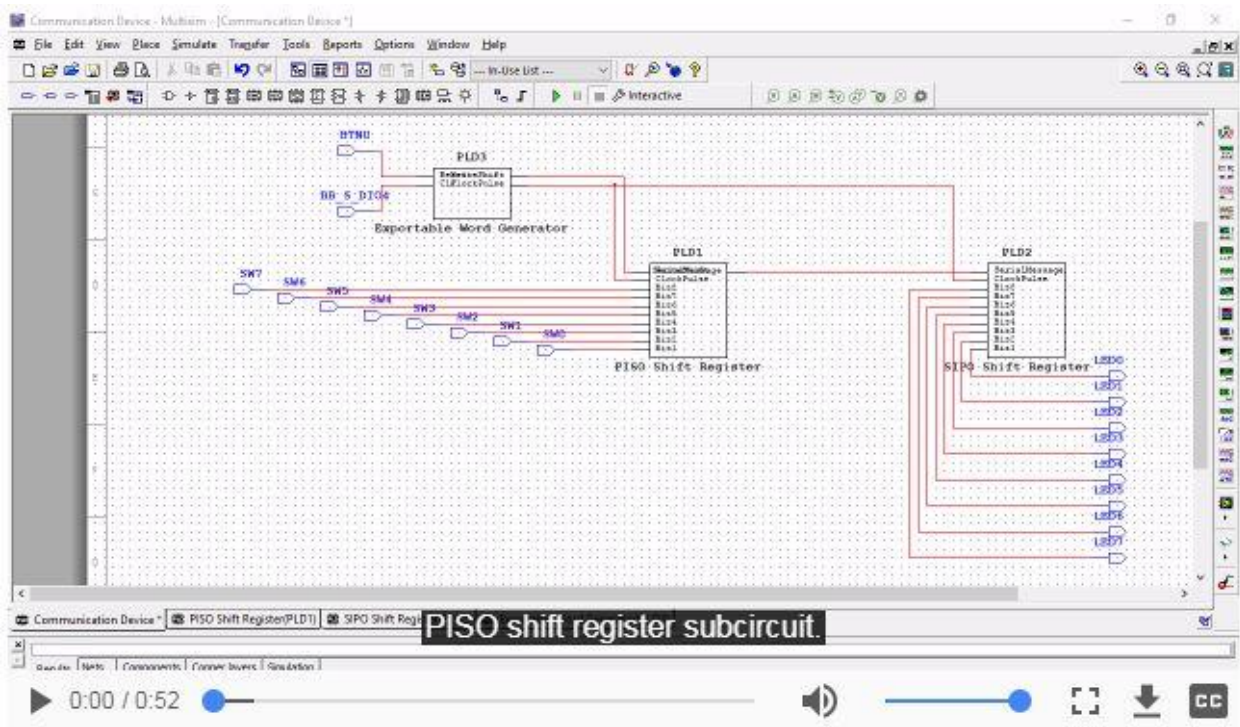


Рисунок 1-10 Видеообзор схемы 8-разрядного регистра сдвига (PISO). Просмотр видео здесь: https://cf-ts.mythinkscape.com/video/PISO_Shift_register_no_cursor.mp4

Подсхема SIPO Shift Register (приемник):

9. В подсхему **SIPO Shift Register** поместите 8 D-триггеров (FF_D_CO). Эти триггеры получают переданные данные в правильном порядке и сохраняют их для чтения сообщения.
10. Поместите следующие контакты и дайте им имена:
 - Выходы триггеров: **Bit1 – Bit8**
 - Вход регистра сдвига: **SerialMessage**
 - Вход тактовых импульсов: **ClockPulse**
11. Соедините компоненты в 8-разрядный регистр сдвига с последовательным входом и параллельным выходом (SIPO), как показано на рисунке ниже:

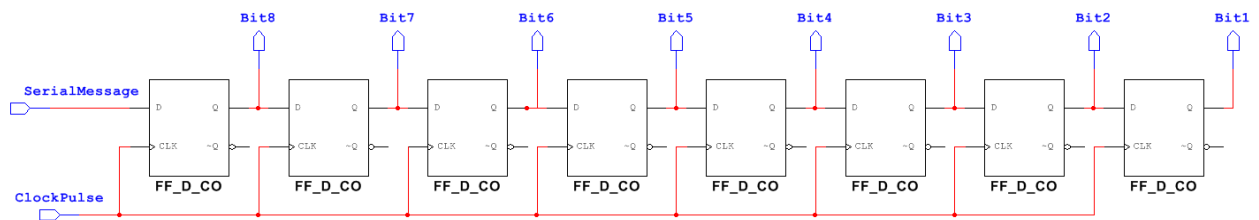


Рисунок 1-11 Схема 8-разрядного регистра сдвига с последовательным входом и параллельным выходом (SIPO)

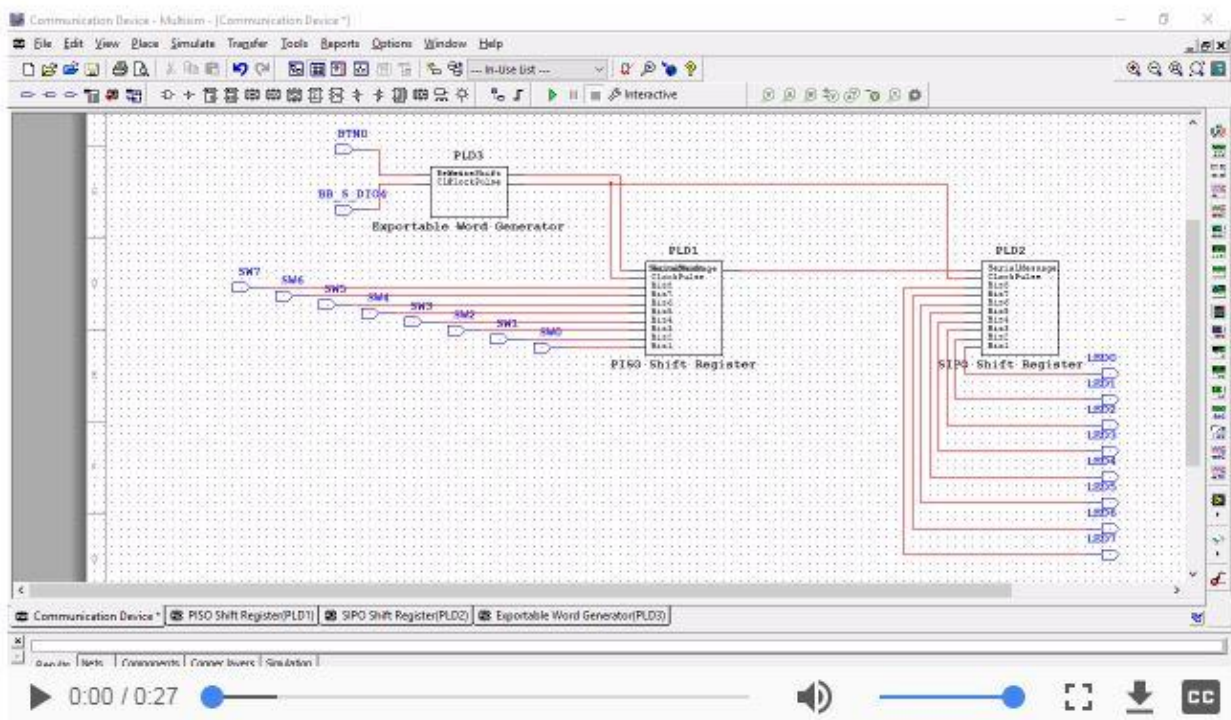


Рисунок 1-12 Видеообзор схемы 8-разрядного регистра сдвига (SIPO). Просмотр видео здесь: https://cf-ts.mythinkscape.com/video/SIPO_Shift_Register_no_cursor.mp4

Входные и выходные контакты

12. В главной схеме добавьте **контакт LED connector** к каждому выходу регистра сдвига SIPO. Светодиод 0 должен отображать бит 1, светодиод 1 – бит 2 и т.д.
13. Добавьте 8 контактов для переключателей (**SW0 – SW7**) для каждого входа регистра сдвига PISO. SW0 должен задавать бит 1, SW1 – разряд 2 и т.д.
14. Добавьте входные контакты **BTN0** и **BB_S_DIO4** и подключите их к подсхеме Timing, как показано на рисунке ниже.

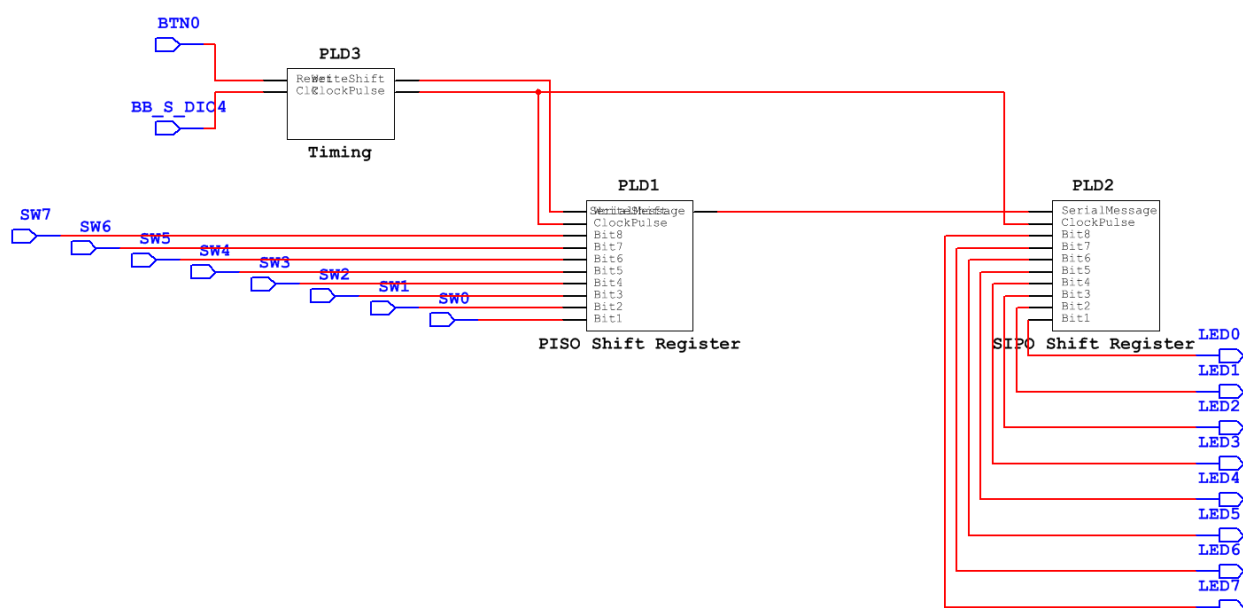


Рисунок 1-13 Схема соединений подсхем

1.3 Настройка функционального генератора

Порядок выполнения:

1. Подключите разъем BNC кабеля к **порту FGEN** платформы NI ELVIS III. Подключите красную клипсу к проводнику и вставьте проводник в гнездо **BB_S_DIO4**. Аналогичным образом подключите черную клипсу к проводнику и вставьте его в гнездо **GND**.
2. Установите следующие параметры генератора функций:
 - Waveform: **Square**
 - Frequency: **1 024 Hz**
 - Amplitude: **3,00 Vpp**
3. Убедитесь, что функциональный генератор правильно распознает устройство (**NI ELVIS III**) и маршрут сигнала (**FGEN BNC**). Питание платы ELVIS должно быть включено.

Используйте следующий рисунок для проверки выполненных настроек.

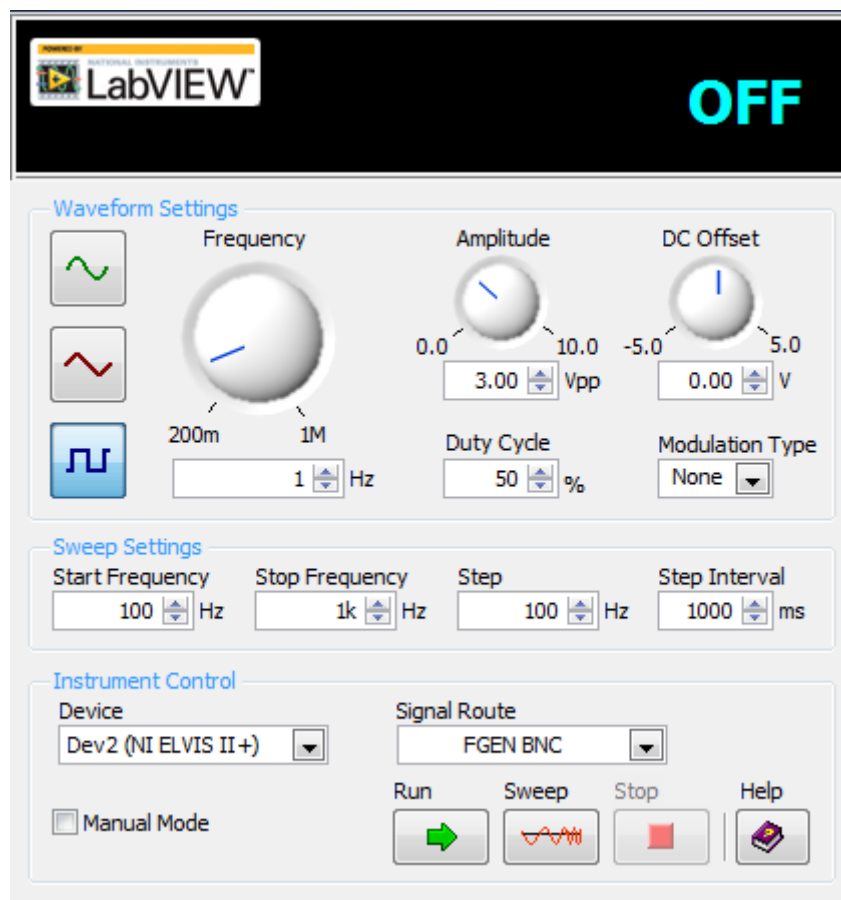


Рисунок 1- 14 Настройки генератора

1.4 Сборка системы связи

1. Соедините различные подсхемы в соответствии с блок-схемой, изображенной на рисунке 1-5.
2. Экспортируйте схему на плату Digital Electronics. Не забудьте перезагрузить плату перед экспортом схемы, чтобы изменения были приняты.

Примечание: Сделайте скриншот, фотографию или рисунок вашей окончательной схемы и всех подсхем и включите изображения в отчет.

1.5 Тестирование и решение проблем

В этом разделе приведена последовательность тестирования вашей системы связи. В конце этого раздела вы найдете руководство, оно поможет решить проблемы, которые могут возникнуть в процессе тестирования.

Совет: добавьте интерактивные цифровые константы и пробники в вашу схему PLD, чтобы иметь возможность тестировать внесенные вами изменения без необходимости экспорта схемы на плату Digital Electronics.

Порядок выполнения:

1. Убедитесь, что ваш проект PLD экспортирован на плату Digital Electronics.
2. Какое сообщение вы будете отправлять? Установите переключатели на плате соответствующим образом.
3. Запустите функциональный генератор.
4. Запишите отображаемое на светодиодном индикаторе сообщение. После вывода сообщения на индикатор обмен данными останавливается. Совпадает ли полученное сообщение с отправленным?
5. Сформируйте другое сообщение, изменив положение переключателей, и нажмите кнопку сброса. На светодиодном индикаторе отображается новое сообщение?

Что необходимо всегда проверять:

- Включено ли питание платы Digital Electronics?
- Вы выключили и включили питание платы Digital Electronics, прежде чем разворачивать схему?
 - Для перезагрузки необходимо выключить и включить плату.

Схема PLD не развернулась/ неправильно развернулась на плате Digital Electronics.

- Убедитесь, что включили питание NI ELVIS III и платы Digital Electronics.
- Убедитесь, что плата подключена к компьютеру и к источнику питания. Иногда порт USB на компьютере плохо работает, попробуйте использовать другой порт.
- Убедитесь, что выбрали правильную плату.
- Убедитесь, что выбрали правильные инструментальные средства Xilinx (32- или 64-разрядные в зависимости от операционной системы).

- Проверьте правильность выбора контактов PLD. Они должны совпадать с контактами платы Digital Electronics. Убедитесь также, что они правильно соединены в проекте и на плате.

Сообщение не получено, или получен только один бит (0 или 1):

- Убедитесь, что выход **WriteShift** подсхемы Timing подключен ко входу **WriteShift** передатчика.
- Убедитесь, что правильно установили переключатели на плате Digital Electronics.
- Убедитесь, что правильно соединили различные компоненты схемы.

Полученное сообщение не совпадает с отправленным:

- Убедитесь, что считываете отправленное и полученное сообщения в одном направлении.
- Убедитесь, что счетчик останавливает и передатчик, **И** приемник в нужное время (после 16 тактовых импульсов).
- Убедитесь, что **WriteShift** активируется после 16 тактовых импульсов.
- Убедитесь, что оба сдвиговых регистра собраны правильно.

После сброса схема перестает правильно отображать сообщения:

- Проверьте подключение контакта **Reset** в подсхеме Timing.
- Убедитесь, что активация Rest возвращает выход **WriteShift** в состояние с низким уровнем.

Примечание: если в этом разделе вы внесли в схему какие-то изменения, сделайте скриншоты, фотографии или рисунки модифицированных схем и включите изображения в окончательный отчет.

1.6 Дополнительные задания

В этой лабораторной работе вам предлагается четыре дополнительных задания. Выберите *ОДНУ* или *несколько* дополнительных функций для реализации в вашей системе цифровой связи:

- Шифрование и дешифрование сообщений
<https://mythinkscape.com/labs/v2/23305/steps/21217#Message%20Encryption%20Detection>
- Схема кодирования символов:
<https://mythinkscape.com/labs/v2/23305/steps/21217#Character%20Encoding%20Scheme>
- Обмен данными между двумя платами Digital Electronics:
<https://mythinkscape.com/labs/v2/23305/steps/21217#Communicating%20Between%20Two%20DSDB%20Boards>
- Семисегментный индикатор:
<https://mythinkscape.com/labs/v2/23305/steps/21217#SSD>

Шифрование и дешифрование сообщений

В этом разделе рассматривается реализация дополнительной функциональности в разработанной вами системе связи. Шифрование и дешифрование позволяет защитить ваше сообщения от прочтения лицами, у которых нет ключа дешифрования. Обсуждаются этапы исследования, планирования и реализации этой дополнительной функции.

Осмысление и обобщение следующих ключевых слов поможет вам лучше понять, что потребуется при выполнении этой части лабораторной работы. Вы сможете вернуться сюда и добавить новую информацию в любой момент.

1 -14 Шифрование:

1 -15 Дешифрование:

1 -16 Ключ дешифрования:

Ответы на следующие вопросы помогут вам подготовиться к разработке схемы, а также гарантируют, что вы изучите всю необходимую информацию, прежде чем начнете проектировать систему связи.

1-17 Какие функции должны выполняться окончательной схемой?

1-18 Для реализации функций шифрования и дешифрования в вашей системы каждый бит, формируемый на выходе передатчика и поступающий на вход приемника, должен проходить через логический элемент вместе с ключом. Для выбора логического элемента рассмотрите таблицы истинности различных элементов. Необходим такой логический элемент, который реализует: $A + B = C$ при шифровании и $C + B = A$ при дешифровании, где **A** – ваше исходное сообщение, **B** – ключ, а **C** – зашифрованное сообщение (как показано на рисунке ниже). Имея это в виду, какой логический элемент вы выберете?

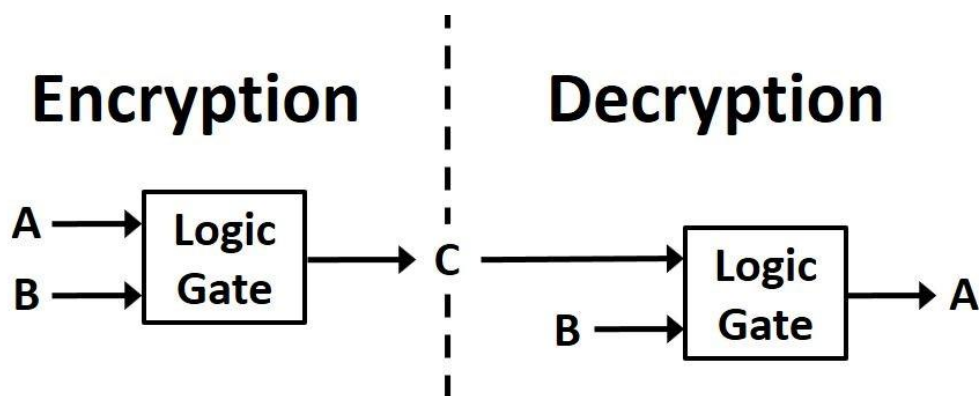


Рисунок 1-15 Логика шифрования и дешифрования

Encryption – шифрование, Decryption – дешифрование, Logic Gate – логический элемент

1-19 Между какими компонентами вы вставите шифрование?

1-20 Между какими компонентами вы вставите дешифрование?

- Нарисуйте *схему шифрования* и покажите, как подключить ее к уже разработанной схеме.
- Нарисуйте *схему дешифрования* и покажите, как подключить ее к уже разработанной схеме.

Примечание: Включите ваши рисунки в отчет.

Выполните следующие указания для реализации дополнительных функций в вашу исходную схему:

1. В сдвиговом регистре PISO поместите логический элемент между входом бита сообщения и входами элементов NAND.
2. Поместите логический элемент между выходом D-триггеров сдвигового регистра SIPO и выходным контактом. На один вход этого элемента будет подаваться бит ключа, а на второй вход – бит с выхода триггера. Выходные сигналы этих элементов будут подаваться на выходные контакты.

Примечание: Убедитесь, что биты ключа в передатчике и приемнике подаются в одном и том же порядке.

Примечание: Убедитесь, что вводите ключ с помощью цифровых констант низкого и высокого уровня, а не интерактивных цифровых констант.

Примечание: Реализовав в схемах новые функции, сделайте их скриншоты и включите их в отчет.

Тестирование и решение проблем

1. Подключите цифровые пробники к выходам сдвигового регистра SIPO для замены в Multisim светодиодного индикатора.
2. Введите правильный код дешифрования и убедитесь, что отображается правильное сообщение.

3. Введите неправильный код дешифрования и убедитесь, что сообщение не отображается.

Используйте следующие вопросы для решения проблем и исправления ошибок в вашей схеме.

Полученное сообщение не совпадает с отправленным.

- Убедитесь, что ключ шифрования и ключ дешифрования совпадают и введены в правильном порядке.

Сообщение не отображается при вводе правильного ключа дешифрования

- Убедитесь, что логические элементы между линиями битов сообщений и сдвиговым регистром PISO, а также между сдвиговым регистром SIPO и светодиодным индикатором подключены правильно.

Примечание: Если вы внесли в вашу схему какие-то изменения, сделайте скриншоты обновленной схемы и загрузите их ниже.

Если хотите реализовать еще одну функцию, перейдите по ссылке: <https://mythinkscape.com/labs/v2/23305/steps/21217> .

Если вы завершили реализацию новых функций, перейдите в раздел *1.7 Заключительное тестирование*.

Схема кодирования символов

В этом разделе рассматривается реализация дополнительной функциональности в разработанной вами системе связи. Схема кодирования символов позволяет отправлять более содержательные сообщения. Обсуждаются этапы исследования, планирования и реализации этой дополнительной функции.

Осмысление и обобщение следующих ключевых слов поможет вам лучше понять, что потребуется при выполнении этой части лабораторной работы. Вы сможете вернуться сюда и добавить новую информацию в любой момент.

1-21 Схемы кодирования символов:

1-22 ASCII:

Ответы на следующие вопросы помогут вам подготовиться к проектированию схемы, а также гарантируют, что вы усвоите всю необходимую информацию, прежде чем начнете разрабатывать вашу систему связи.

1-23 В чем преимущества и недостатки использования схемы кодирования символов в вашем сообщении?

1-24 Вы можете использовать или стандартную схему кодирования ASCII, или создать свою собственную. Какую схему вы будете использовать?

1-25 Если вы используете стандартную схему, найдите в интернете таблицу преобразования и держите ее под рукой для последующего выполнения лабораторной работы. Если вы предпочитаете собственную схему, создайте свою таблицу преобразования для алфавита.

Примечание: Включите вашу схему в отчет.

1-26 Понадобится ли вам вносить какие-то изменения в уже разработанную систему связи для реализации схемы кодирования символов?

1-27 Как представить буквы A, J, T и Z в выбранной вами схеме кодирования символов?

1-28 Как представить сообщение “hello” в выбранной вами схеме кодирования символов?

Используйте следующие указания для реализации дополнительной функциональности в вашей схеме.

Вы можете сверяться с данной таблицей для преобразования символов.

ASCII Code: Character to Binary

0	0011 0000	O	0100 1111	m	0110 1101
1	0011 0001	P	0101 0000	n	0110 1110
2	0011 0010	Q	0101 0001	o	0110 1111
3	0011 0011	R	0101 0010	p	0111 0000
4	0011 0100	S	0101 0011	q	0111 0001
5	0011 0101	T	0101 0100	r	0111 0010
6	0011 0110	U	0101 0101	s	0111 0011
7	0011 0111	V	0101 0110	t	0111 0100
8	0011 1000	W	0101 0111	u	0111 0101
9	0011 1001	X	0101 1000	v	0111 0110
A	0100 0001	Y	0101 1001	w	0111 0111
B	0100 0010	Z	0101 1010	x	0111 1000
C	0100 0011	a	0110 0001	y	0111 1001
D	0100 0100	b	0110 0010	z	0111 1010
E	0100 0101	c	0110 0011	.	0010 1110
F	0100 0110	d	0110 0100	,	0010 0111
G	0100 0111	e	0110 0101	:	0011 1010
H	0100 1000	f	0110 0110	;	0011 1011
I	0100 1001	g	0110 0111	?	0011 1111
J	0100 1010	h	0110 1000	!	0010 0001
K	0100 1011	I	0110 1001	'	0010 1100
L	0100 1100	j	0110 1010	"	0010 0010
M	0100 1101	k	0110 1011	{	0010 1000
N	0100 1110	l	0110 1100	}	0010 1001
				space	0010 0000

Рисунок 1-16 Код ASCII

Тестирование и решение проблем

- Преобразуйте двоичный выходной код приемника в соответствующие символы. Совпадает ли полученное сообщение с отправленным?

Используйте следующие вопросы для решения проблем и исправления ошибок в вашей схеме.

Полученное сообщение не совпадает с отправленным.

- Убедитесь, что правильно преобразуете двоичный код в символы.
- Проверьте, не внесли ли в исходную схему нежелательных изменений.

Если хотите реализовать еще одну дополнительную функцию, перейдите по *ссылке*:

Если вы завершили реализацию дополнительных функций, перейдите в раздел *1.7 Заключительное тестирование*.

Обмен данными между двумя платами Digital Electronics:

В этом разделе рассматривается реализация дополнительной функциональности в разработанной вами системе связи. Вы будете передавать сообщение между двумя платами Digital Electronics. Обсуждаются этапы исследования, планирования и реализации этой дополнительной функции.

Ответы на следующие вопросы помогут вам подготовиться к проектированию схемы, а также гарантируют, что вы усвоите всю необходимую информацию, прежде чем начнете разработку системы связи.

1-29 Чем схема на плате-передатчике будет отличаться от схемы на плате-приемнике?

1-30 Какие изменения необходимо внести в исходную схему для реализации рассматриваемой функции?

1-31 Как сигнал будет передаваться между двумя платами? Откуда и куда он будет поступать?

1-32 Как вы будете синхронизировать две платы?

Используйте следующие указания для реализации дополнительной функциональности в вашей схеме.

1. Модифицируйте схему на плате-передатчике.
2. Модифицируйте схему на плате-приемнике.

Тестирование и решение проблем

- Отправьте сообщение от одной платы Digital Electronics к другой. Совпадает ли полученное сообщение с отправленным?

Используйте следующие вопросы для решения проблем и исправления ошибок в вашей схеме.

Полученное сообщение не совпадает с отправленным.

- Проверьте соединения в вашей схеме.
- Убедитесь, что проводник, по которому передается сигнал, подключен к правильным входному и выходному контактам.
- Убедитесь, что две платы синхронизируются общим тактовым сигналом (от одного источника).

Примечание: Если вы внесли в вашу схему какие-то изменения, сделайте скриншоты, снимки или рисунки модифицированной схемы и включите изображения в отчет.

Если хотите реализовать еще одну дополнительную функцию, перейдите по ссылке: <https://mythinkscape.com/labs/v2/23305/steps/21217> .

Если вы завершили реализацию дополнительных функций, перейдите в раздел *1.7 Заключительное тестирование*.

Семисегментный индикатор

В этом разделе рассматривается реализация дополнительной функциональности в разработанной вами системе связи. Вы будете преобразовывать 8-битные сообщения в числа и отображать их на семисегментном индикаторе. Обсуждаются этапы исследования, планирования и реализации этой дополнительной функции.

Осмысление и обобщение следующих ключевых слов поможет вам лучше понять, что потребуется для выполнения этой части лабораторной работы. Вы сможете вернуться сюда и добавить новую информацию в любой момент.

1-33 Семисегментный индикатор:

Ответы на следующие вопросы помогут вам подготовиться к проектированию схемы, а также гарантируют, что вы усвоите всю необходимую информацию, прежде чем начнете разработку системы связи.

1-34 Сколько цифр вы хотите отображать?

1-35 Чем отличается отображение сообщения на 8-разрядном двоичном индикаторе и на семисегментном индикаторе? Какие изменения необходимо внести в вашу схему? Одинаково ли будет количество выходов у схем?

1-36 Как преобразовать данные для отображения чисел на семисегментном индикаторе?

1-37 Как вы будете формировать сигналы FPGA для управления индикаторами?

Используйте следующие указания для реализации дополнительных функций вашей исходной схемой:

1. Модифицируйте схему для отображения ваших сообщений на семисегментном индикаторе.

Тестирование и решение проблем

- Преобразуйте двоичный выходной код приемника в соответствующие числа. Совпадает ли полученное сообщение с отправленным?

Используйте следующие вопросы для решения проблем и исправления ошибок в вашей схеме.

Полученное сообщение не совпадает с отправленным.

- Убедитесь, что переключатели и контакты семисегментного индикатора подключены к правильным входам и выходам регистра сдвига.

Примечание: Если вы внесли в вашу схему какие-то изменения, сделайте скриншоты обновленной схемы и загрузите их ниже.

Если хотите реализовать еще одну дополнительную функцию, перейдите по ссылке: <https://mythinkscape.com/labs/v2/23305/steps/21217>.

Если вы завершили реализацию новых функций, перейдите в раздел *1.7 Заключительное тестирование*.

1.7 Заключительное тестирование.

В этом разделе вы выполните заключительное тестирование вашей схемы, чтобы продемонстрировать ее работоспособность, а также ответите на ряд вопросов, касающихся процесса проектирования. Перед демонстрацией убедитесь, что выполнили указания из подразделов *Тестирование и решение проблем*. Ваша схема должна быть полностью функционирующей.

- Запишите видео схемы при выполнении следующих тестов:
 - Отправьте сообщение длиной 10 символов или больше с передатчика на приемник.
 - Отобразил ли приемник правильное сообщение?
 - Опционально: Если вы использовали схему кодирования символов, преобразуйте двоичный код с выхода приемника обратно в символы. Совпадают ли они с отправленным сообщением?
 - Опционально: Если вы использовали семисегментный индикатор, преобразуйте двоичный код с выхода приемника обратно в цифры. Совпадают ли они с отправленным сообщением?

Примечание: Включите ваше видео в отчет.

1.8 Заключение

1-1 1-38 Объясните, что такое шифрование и дешифрование, и приведите задачи, в которых полезно их использовать (кроме перечисленных в разделе *Введение*).

1-39 Какой параметр задает скорость передачи битов от передатчика к приемнику?

1-40 Что изменилось бы, если бы вы использовали передачу данных в параллельном коде, а не передачу данных в последовательном коде? Сравните достоинства и недостатки этих методов передачи данных.

1-41 Сколько существует возможных ключей шифрования при длине ключа 8 бит?

1-42 (необязательно): Если вы использовали схему кодирования символов, почему выбрали именно такую схему? Не хотели бы вы изменить свое решение?

1-43 Какое неожиданное для вас затруднение возникло при работе над проектом? Как вы с ним справились?

1-44 Если бы вы могли заново спроектировать схему, какие изменения вы бы внесли? Изменился бы ваш подход к проектированию?
