

## Создание приложений для NI CompactRIO в среде LabVIEW. Краткое руководство

В данном руководстве рассмотрен пример разработки приложения для CompactRIO в среде LabVIEW. В примере используется встраиваемая реконфигурируемая система CompactRIO (Reconfigurable Embedded System) для создания простейшего термостата. В процессе разработки этого приложения вы ознакомитесь с основными принципами и методами программирования в среде LabVIEW для платформы CompactRIO.

### Программное обеспечение

Для работы с CompactRIO Reconfigurable Embedded System необходимо следующее программное обеспечение:

- NI LabVIEW 8.0 или более поздняя версия
- NI LabVIEW Real-Time Module 8.0 или более поздняя версия
- NI LabVIEW FPGA Module 8.0 или более поздняя версия
- NI-RIO 2.0 или более поздняя версия

### Оборудование

Для выполнения упражнений данного руководства необходимо следующее оборудование (также может понадобиться перечисленная ниже документация):

- встраиваемая реконфигурируемая система CompactRIO
  - встраиваемый контроллер реального времени cRIO-9002/9004
  - реконфигурируемое шасси cRIO-9101, cRIO-9102, cRIO-9103, или cRIO-9104
  - Источник питания
  - Подключение к сети Ethernet
  - *Инструкция по эксплуатации (Operating Instructions)* контроллера cRIO-9002/9004 — содержит информацию о подключении контроллера к источнику питания и сети Ethernet.
  - *Инструкция по установке (Installation Instructions)* системы CompactRIO — содержит информацию об установке шасси и модулей ввода/вывода.
  - модуль термодатчика ввода NI 9211
  - модуль цифрового вывода NI 9474
  - *Инструкция по эксплуатации NI 9211* — содержит информацию о подключении термодатчика к модулю
  - *Инструкция по эксплуатации NI 9472/9474* — содержит информацию о подключении внешних устройств к модулю NI 9474

Если у вас есть не все перечисленное оборудование, например, нет модулей ввода/вывода, вы можете использовать другие модули, выполняя упражнения по аналогии.

### Задача

Система CompactRIO считывает показания температуры с термопары. Если текущая температура превышает установленный интервал, система включает вентилятор. Если же температура оказывается ниже установленного интервала — система включает нагреватель.

В результате выполнения упражнений, вы получите готовое приложение, позволяющее наблюдать текущее значение температуры и устанавливать интервал допустимых температур.

Создание и запуск приложения может быть реализован без реального вентилятора и нагревателя.

В процессе выполнения заданий будет создан новый проект LabVIEW (LabVIEW Project), виртуальные приборы (ВП) FPGA VI и host VI.

Проект LabVIEW — это файл, в котором содержится информация о виртуальных приборах и целевых устройствах (target), используемых в приложении.

FPGA VI загружается в целевое устройство ПЛИС — в данном случае это реконфигурируемое шасси. FPGA VI используется для чтения и записи данных в каналы модулей ввода/вывода. В нашем приложении FPGA VI также производит сравнение текущей температуры с заданным интервалом и, если необходимо, включает вентилятор или нагреватель.

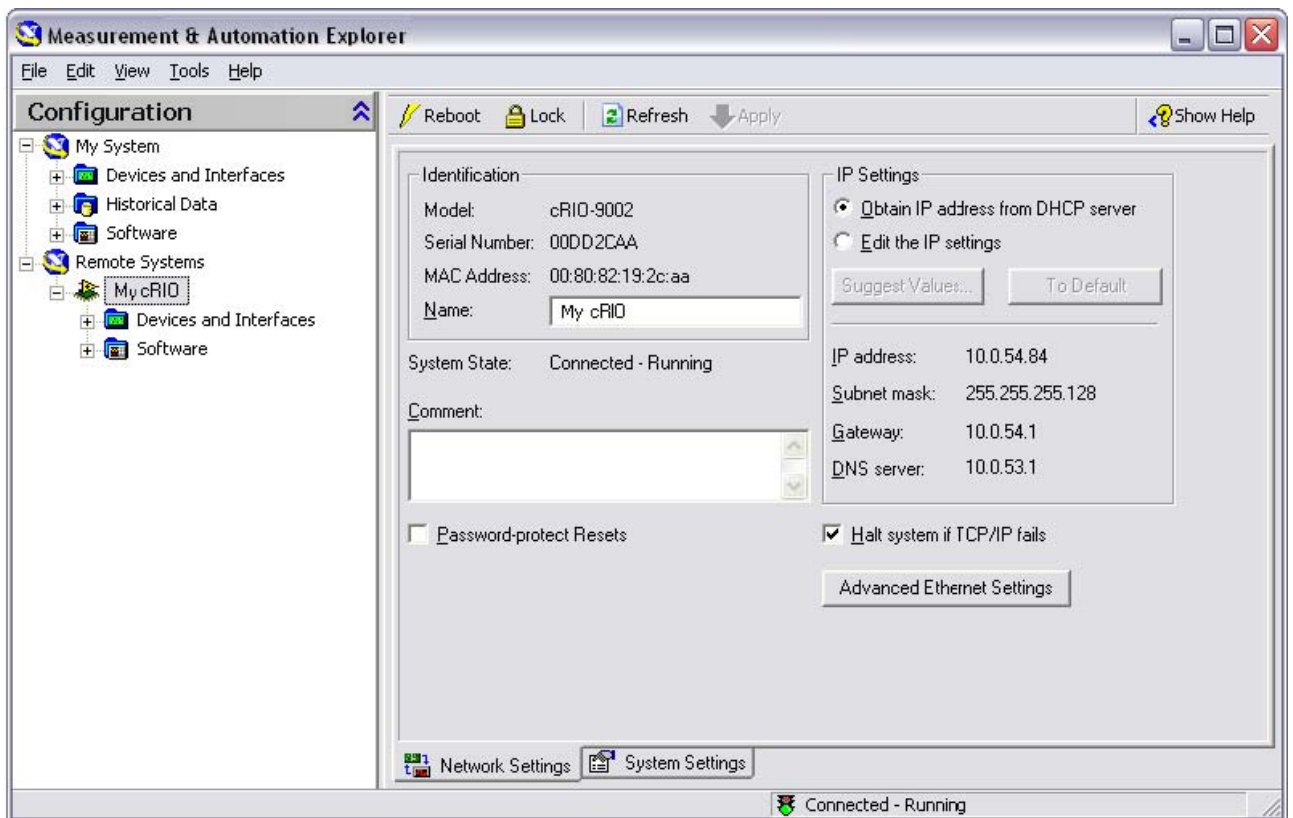
Host VI обменивается данными с FPGA VI. Host VI может выполняться как на контроллере реального времени, так и на компьютере под управлением Windows. В нашем приложении host VI будет использоваться для преобразования полученных в FPGA VI значений в стандартные единицы измерения, а также для задания рабочего интервала температур.

## Настройка оборудования

Чтобы подготовить оборудование к работе, выполните следующие действия:

1. Установите контроллер cRIO-9002/9004 на шасси cRIO-910x.
2. Установите модуль ввода/вывода NI 9211 в слот 1 шасси и модуль NI 9474 в слот 2
3. Подключите термодатчик к каналу 0 модуля NI 9211. Подробнее см. инструкцию по эксплуатации модуля NI 9211.
4. Канал 0 модуля термодатчика будем использовать для считывания показаний температуры.
5. Канал 0 модуля цифрового вывода будем использовать для управления вентилятором. Канал 1 — для управления нагревателем.
6. Подключите источник питания к контроллеру и установите Ethernet-соединение в соответствии с инструкцией по эксплуатации CompactRIO cRIO-9002/9004.
7. Чтобы задать или изменить IP-адрес контроллера, выполните следующие действия:
  - a. Отключите питание контроллера и установите переключатель **SAFE MODE** на его корпусе в положение On.
  - b. Включите питание контроллера и запустите программу Measurement & Automation Explorer (MAX) на вашем компьютере.
  - c. В панели **Configuration** выберите **Remote Systems»0.0.0.0**
  - d. Чтобы задать IP-адрес, в разделе IP Settings отметьте пункт **Obtain IP address from DHCP server** либо **Edit the IP settings** и введите статический IP-адрес.

На рисунке представлено окно настройки MAX:



## Создание проекта в LabVIEW

Для организации и управления виртуальными приборами и целевыми устройствами приложения используется LabVIEW Project Explorer. Например, чтобы работать с FPGA VI, в проект необходимо добавить целевое устройство FPGA.

Выполните следующие действия, чтобы создать проект LabVIEW для реконфигурируемой встраиваемой системы CompactRIO и добавить в него модули ввода/вывода:

1. Запустите LabVIEW.
2. В окне **Getting Started** выберите пункт **Empty Project** (пустой проект), чтобы открыть окно **Project Explorer**.
3. Кликните правой кнопкой мыши по названию **untitled project** в окне **Project Explorer** и выберите **New»Targets and Devices**. Появится диалоговое окно **Add Targets and Devices** (добавление целевых устройств и модулей).
4. Выберите пункт **Existing target** (имеющееся целевое устройство), откроется список доступных целевых устройств.
5. В списке отметьте **CompactRIO Reconfigurable Embedded Controller** и нажмите **OK**.
6. Кликните правой кнопкой мыши по названию контроллера, появившемуся в окне **Project Explorer** и выберите **New»Targets and Devices**. Появится диалоговое окно **Add Targets and Devices**.
7. Выберите пункт **Existing target**, откроется список доступных целевых устройств.
8. Выберите реконфигурируемое шасси (chassis), например **cRIO-9101**, и нажмите **OK**.
9. Кликните правой кнопкой мыши по названию шасси, появившемуся в окне **Project Explorer** и выберите **New»C Series Modules** (модули ввода/вывода серии C). Появится диалоговое окно **Add Targets and Devices**.

10. Выберите пункт **Existing target**, откроется список доступных модулей ввода/вывода.
11. Выберите модуль NI 9211 и нажмите **ОК**.
12. Повторите шаги с 9 по 11, чтобы добавить в проект модуль NI 9474.

### Настройка модулей ввода/вывода

Чтобы провести настройку модулей ввода/вывода, выполните следующие действия:

1. Кликните правой кнопкой мыши по названию модуля NI 9211 в окне **Project Explorer** и выберите пункт **Properties** (Свойства). Появится диалоговое окно **C Series Module Properties** (Свойства модуля серии C).
2. Введите `thermocouple` в поле **Name** (имя).
3. Выберите **Slot 1** в меню **Location**, если этот пункт еще не выбран.
4. Нажмите **ОК**.
5. Кликните правой кнопкой мыши по названию модуля NI 9474 в окне **Project Explorer** и выберите пункт **Properties**. Появится диалоговое окно **C Series Module Properties**.
6. Введите `digital output` в поле **Name**.
7. Выберите пункт **Slot 2** в меню **Location**, если этот пункт еще не выбран.
8. Нажмите **ОК**.
9. В главном меню окна **Project Explorer** выберите **File»Save Project** (Сохранить проект). Откроется диалоговое окно **Name the Project** (Введите имя проекта).
10. В поле **File name** введите `crio thermostat`.
11. Нажмите **ОК**.

### Добавление элементов ввода/вывода в проект.

Чтобы использовать функции ввода/вывода в FPGA VI, в проект необходимо добавить элементы ввода/вывода.

Выполните следующие действия, чтобы добавить элементы ввода/вывода, соответствующие модулям NI 9211 и NI 9474:

#### Модуль термопарного ввода:

1. В окне **Project Explorer** кликните правой кнопкой мыши по названию целевого устройства FPGA и выберите **New»FPGA I/O**. Появится окно **FPGA I/O**.
2. Раскройте пункт **Analog Input** (аналоговый ввод) в левой части окна, затем пункт **thermocouple**.
3. Выберите **TC0** и нажмите **Add** (Добавить).
4. Выберите **CJC** и нажмите **Add**.
5. Выберите **Autozero** и нажмите **Add**.
6. Кликните по названию канала **TC0** в столбце **Name** таблицы **New FPGA I/O** (Новый элемент ввода/вывода FPGA) и введите `actual temperature`.
7. Нажмите **ОК**.

#### Модуль цифрового вывода:

1. В окне **Project Explorer** кликните правой кнопкой мыши по названию целевого устройства FPGA и выберите **New»FPGA I/O**. Появится окно **FPGA I/O**.
2. Раскройте пункт **Digital Line Output** в левой части окна, затем пункт **digital output**.
3. Выберите **DO0** и нажмите **Add**.
4. Выберите **DO1** и нажмите **Add**.
5. Кликните на имя канала **DO0** в столбце **Name** таблицы **New FPGA I/O** и введите `fan control`.
6. Кликните на имя канала **DO1** в столбце **Name** таблицы **New FPGA I/O** и введите `heater control`.
7. Нажмите **OK**.
8. В главном меню окна **Project Explorer** выберите **File»Save Project**.

Теперь можно приступить к созданию FPGA VI.

### Создание FPGA VI

Чтобы создать FPGA VI, необходимо создать новый специальный виртуальный прибор для целевого устройства FPGA, затем скомпилировать его и загрузить в целевое устройство.

Мы будем использовать FPGA VI для чтения и записи данных в каналы ввода/вывода модулей.

### Добавление ВП для целевого устройства FPGA в проект

Чтобы добавить в проект виртуальный прибор для нужного целевого устройства FPGA, выполните следующие действия:

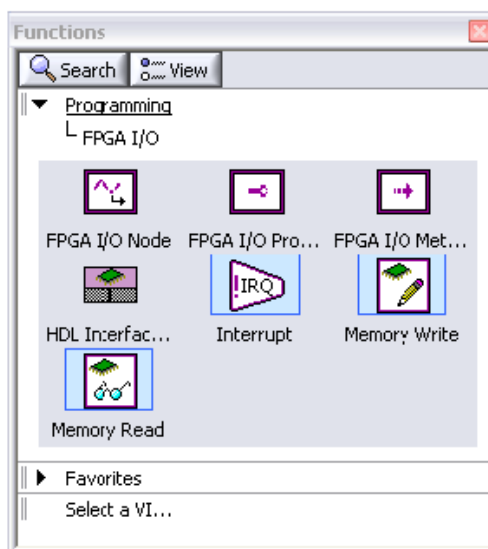
1. В окне **Project Explorer** кликните правой кнопкой мыши по названию целевого устройства FPGA и выберите **New»VI**. Появится окно нового виртуального прибора.
2. В главном меню лицевой панели или блок-диаграммы выберите **File»Save As**, откроется диалоговое окно **Name the VI**.
3. Введите `crio thermostat (fpga)` в поле **File name**.
4. Нажмите **OK**.

### Считывание показаний температуры.

Выполните следующие действия для создания ВП `crio thermostat (fpga) VI`.

1. Поместите **While Loop** на блок-диаграмму `crio thermostat (fpga) VI`.
2. Кликните правой кнопкой мыши по терминалу условия выхода и выберите в меню **Create Control**.
3. Откройте палитру функций и поместите узел **FPGA I/O Node** в цикл **While Loop**.
4. Кликните правой кнопкой мыши по узлу **FPGA I/O Node** и выберите **Select FPGA I/O»Analog Input»thermocouple input»actual temperature**.
5. Кликните правой кнопкой мыши по выходу **actual temperature** узла **FPGA I/O Node** и выберите **Create»Indicator**.
6. Назовите индикатор `actual temperature`.

На рисунке изображена палитра функций FPGA I/O:

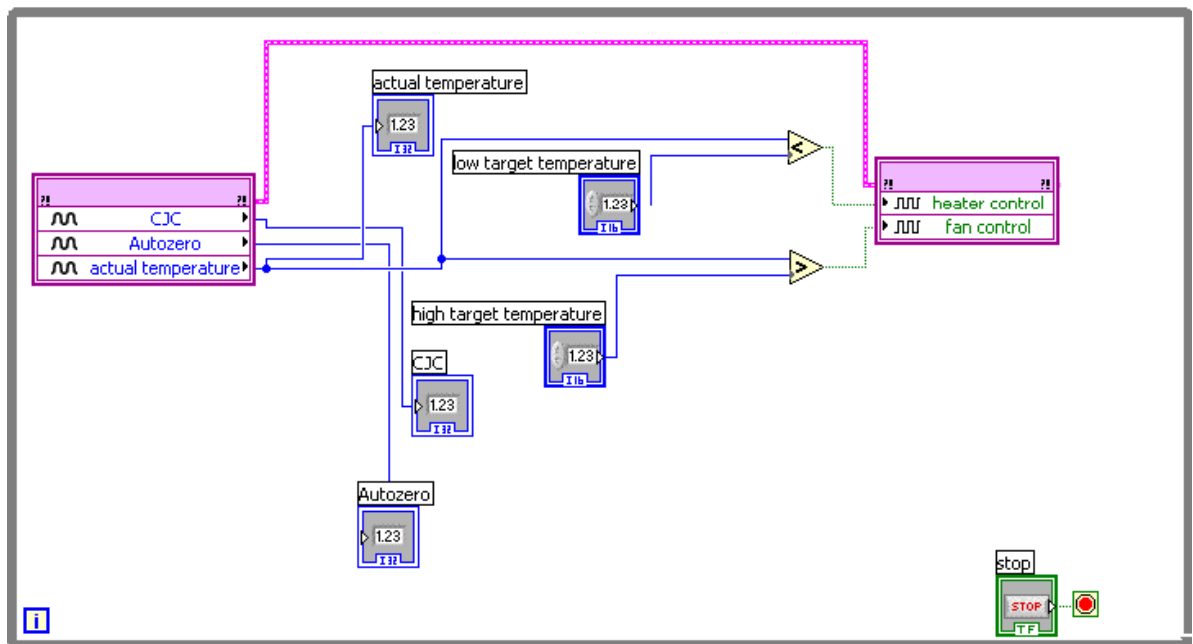


## Чтение данных из каналов CJC и Autozero

Для обеспечения точности измерений, необходимо считывать данные с каналов CJC (компенсация холодного спая) и autozero (автоматический уровень нуля) термопарного модуля CompactRIO. В модуле NI 9211 это внутренние каналы, они физически недоступны, и их нельзя увидеть. Считывание данных из этих каналов аналогично считыванию температуры в разделе *Считывание показаний температуры*. Можно использовать те же самые элементы FPGA I/O Node.

1. Кликните правой кнопкой мыши по узлу FPGA I/O Node, который вы ранее поместили на блок-диаграмму и выберите **Add Element**.
2. Кликните правой кнопкой мыши по новому элементу узла FPGA I/O Node и выберите **Select FPGA I/O»Analog Input»thermocouple input»Autozero**.
3. Кликните правой кнопкой мыши по узлу FPGA I/O Node и выберите **Add Element**.
4. Кликните правой кнопкой мыши по новому элементу в FPGA I/O Node и выберите **Select FPGA I/O»Analog Input»thermocouple input»CJC**.
5. Кликните правой кнопкой мыши по выходу **Autozero** узла FPGA I/O Node и выберите **Create» Indicator**. Повторите эти действия, чтобы создать индикатор элемента **CJC**. Назовите эти индикаторы **autozero** and **CJC** соответственно.

На рисунке показана блок-диаграмма ВП, считывающего показания из каналов autozero, CJC и actual temperature.



Данные из каналов autozero и CJC будут использованы в host VI для преобразования бинарных данных, полученных с термопарного входа, в стандартные единицы температуры.

## Управление температурой.

Чтобы осуществлять управление температуры, FPGA VI должен сравнивать текущую температуру с заданным диапазоном и определять, когда нужно включить вентилятор или нагреватель.

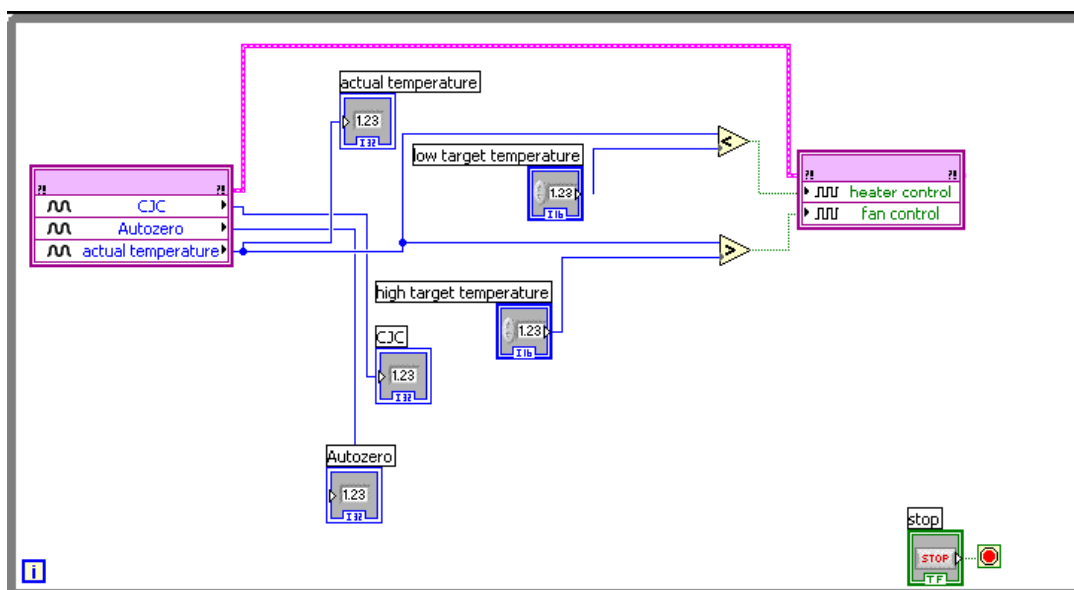
Выполните следующие действия, чтобы добавить в ВП возможность сравнения показаний температур и управления работой вентилятора и нагревателя:

1. Поместите два объекта numeric control (числовой элемент управления) на лицевую панель. Назовите их **low target temperature** и **high target temperature**. Элемент управления **low target temperature** будет задавать нижний предел температурного диапазона, а **high target temperature** – верхний.
2. Поместите узлы **Less?** (меньше?) и **Greater?** (больше?) в цикл **While loop** на блок-диаграмме.
3. Соедините выход **actual temperature** узла **FPGA I/O Node** с входом **x** функций **Less?** и **Greater?**.
4. Соедините выход **low target temperature** с входом **y** функции **Less?** Соедините выход **high target temperature** с входом **y** функции **Greater?**
5. Поместите еще один узел **FPGA I/O Node** в цикл **While Loop** справа от первого **FPGA I/O Node**.
6. Кликните правой кнопкой мыши по новому узлу **FPGA I/O Node** и выберите **Select FPGA I/O»Digital Line Output»digital output»fan control**.
7. Кликните правой кнопкой мыши по узлу **FPGA I/O Node** и выберите **Add Element**.
8. Кликните правой кнопкой по новому элементу **FPGA I/O Node** и выберите **Select FPGA I/O»Digital Line Output»digital output»heater control**.
9. Кликните правой кнопкой мыши по элементу **fan control** и выберите **Change to Write**.
10. Соедините выход **x<y?** функции **Less?** с входом **heater control** второго узла **FPGA**

I/O Node. Соедините выход  $x > y?$  функции Greater? с входом **fan control**.

11. Соедините выход **Error Out** первого узла FPGA I/O Node с входом **Error In** второго узла FPGA I/O Node.

На рисунке 4 изображена модифицированная блок-диаграмма ВП:



## Компиляция и загрузка FPGA VI.

Выполните следующие действия для компиляции и загрузки FPGA VI в целевое устройство FPGA.

1. Кликните правой кнопкой мыши в окне **Project Explorer** по названию целевого устройства FPGA (реконфигурируемое шасси), и выберите **Properties**. Появится диалоговое окно **FPGA Target Properties**.
2. Отметьте пункт **Run when loaded to FPGA**.
3. Выберите раздел **Top-Level Clock (максимальная частота)** в левой части диалогового окна **FPGA Target Properties**.
4. Убедитесь, что выбрана default clock (частота по умолчанию).
5. Нажмите **OK**.
6. Откройте ВП **crio thermostat (fpga)** и нажмите **Run**.
7. Нажмите **OK** в диалоговом окне **Beginning Compile**.
8. Нажмите **OK** в диалоговом окне **Successful Compile Report** по окончании компиляции.

Теперь ВП запущен на целевом устройстве FPGA. Вводите другие значения температуры на лицевой панели VI. Если приложение включает вентилятор, загорается диод LED0 на модуле NI 9474.

При включении нагревателя загорается диод LED1 на модуле NI 9474.

## Создание Host VI для CompactRIO

Мы будем использовать host VI для взаимодействия с FPGA VI и преобразования бинарных данных в температурные единицы. Host VI может быть запущен как на контроллере реального времени, так и на компьютере под управлением Windows. Выполните следующие действия для создания host VI.



1. Запустите LabVIEW.
2. Кликните на **crio thermostat.lvproj** в разделе **Open** окна **Getting Started**.
3. В окне **Project Explorer** кликните правой кнопкой мыши по названию базовой системы, например **My Computer** или **RT CompactRIO Target (0.0.0.0)**, и выберите **New»VI**.
4. Выберите **File»Save As** откроется окно **Name the VI**.
5. Введите crio thermostat (host) в поле **File name**.
6. Нажмите **OK**.
7. Поместите узел **Open FPGA VI Reference** на блок-диаграмму нового ВП. Мы будем использовать эту функцию для реализации связи с ВП, который загружается на целевое устройство FPGA. Функция **Open FPGA VI Reference** должна использоваться в каждом **host VI**.
8. Кликните правой кнопкой мыши по узлу **Open FPGA VI Reference** и выберите **Select VI (Выбрать ВП)**. Появится диалоговое окно **Select VI**.
9. Найдите **crio thermostat (fpga) VI** и нажмите **OK**.
10. Поместите цикл **While Loop** на блок-диаграмму, справа от узла **Open FPGA VI Reference**.
11. Поместите узел **Close FPGA VI Reference** справа от цикла **While Loop**. Узел **Close FPGA VI Reference** должен использоваться в каждом **host VI**.
12. Поместите узел **Read/Write Control** внутрь цикла **While Loop**.
13. Соедините выход **FPGA VI Reference Out** узла **Open FPGA VI Reference** с входом **FPGA VI Reference In** узла **Read/Write Control**.

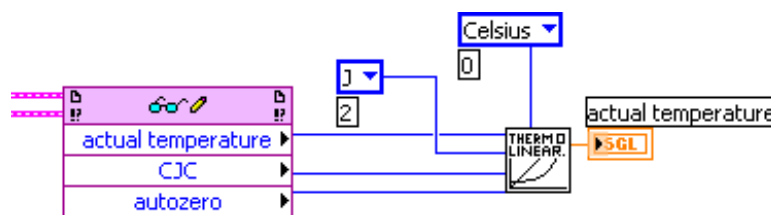
### **Считывание и преобразование текущего значения температуры.**

Выполните следующие действия для считывания фактического значения температуры и его преобразования в стандартные единицы.

1. Кликните правой кнопкой мыши на выход **Unselected** узла **Read/Write Control** и выберите **actual temperature**. Этот выход соответствует индикатору **actual temperature**, который вы создали в **FPGA VI**. Выход **actual temperature** возвращает данные бинарного типа. Эти данные нужно преобразовать в стандартные единицы температуры.
2. Выберите пункт **Select a VI** в палитре функций. Появится диалоговое окно **Choose the VI to open**.
3. Найдите директорию `LabVIEW 8.0\examples\CompactRIO\Module Specific Examples\Analog Input\NI 9211\NI 9211 Support Files`
4. Выберите файл **Convert to Temperature (NI 9211).vi** и нажмите **OK**. Поместите ВП в цикл **While Loop**.
5. Соедините выход **actual temperature** узла **Read/Write Control** с входом **Thermocouple Channel** виртуального прибора **Convert to Temperature (NI 9211) VI**.
6. Кликните правой кнопкой мыши по узлу **Read/Write Control** и выберите **Add Control**.

- Кликните правой кнопкой мыши по элементу **Unselected** и выберите **CJC**. Соедините выход **CJC** узла Read/Write Control с входом **CJC Channel** ВП Convert to Temperature (NI 9211) VI.
- Добавьте выход **autozero** узла Read/Write Control, аналогичным образом. Соедините выход **autozero** функции Read/Write Control с входом **Autozero** ВП Convert to Temperature (NI 9211) VI.
- Создайте константы для входов **Temperature Units** и **Thermocouple Type** ВП Convert to Temperature (NI 9211) VI.
- Кликните правой кнопкой мыши по выходу **Linearized Temperature** ВП Convert to Temperature (NI 9211) VI и выберите **Create»Indicator**. Переименуйте индикатор в **actual temperature**.

На рисунке представлена блок-диаграмма ВП, считывающего и преобразующего фактическое значение температуры:



Теперь Вы можете считывать показания температуры. Далее необходимо реализовать запись требуемого диапазона температур в целевое устройство.

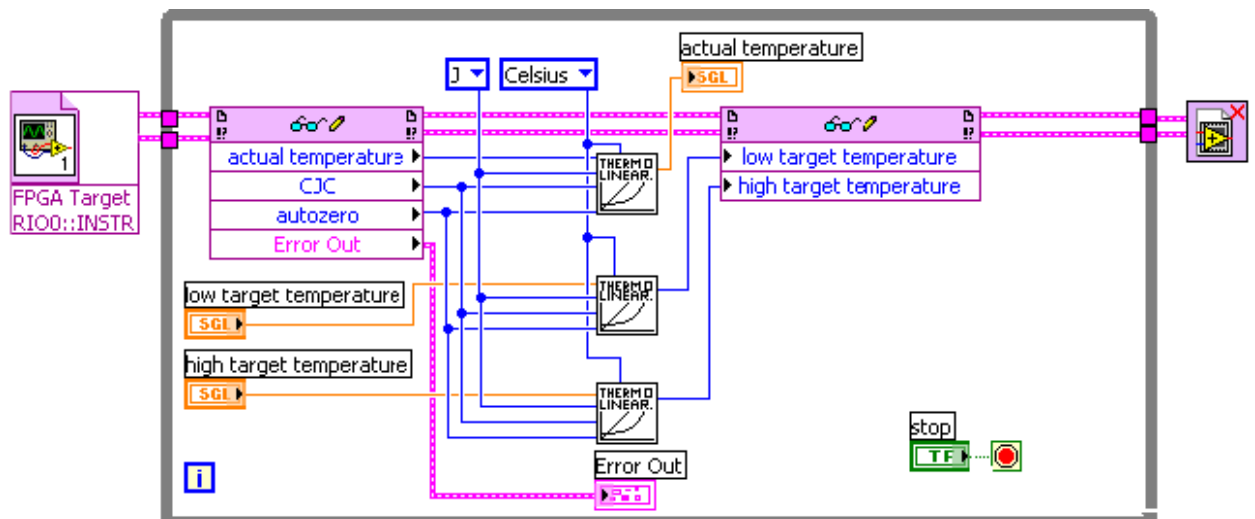
### Преобразование и запись требуемого диапазона температур.

В модули вывода CompactRIO должны передаваться бинарные данные. Если Вы хотите вводить данные в стандартных температурных единицах, вы должны преобразовать их в бинарный вид до того, как host VI передаст значения установленного диапазона в FPGA VI.

- Выберите **Select a VI** в палитре функций. Появится диалоговое окно **Choose the VI to open**.
- Найдите директорию LabVIEW 8.0\examples\CompactRIO\Module Specific Examples\Analog Input\NI 9211\NI 9211 Support Files.
- Выберите файл **Convert to Binary (NI 9211).vi** и нажмите **OK**. Поместите ВП в цикл While Loop.
- Поместите еще один ВП Convert to Binary (NI 9211) VI в цикл While Loop.
- Кликните правой кнопкой мыши по входу **Linearized Temperature** одного из ВП Convert to Binary (NI 9211) VI и выберите **Create»Control**. Назовите элемент управления **low target temperature**.
- Создайте элемент управления для входа **Linearized Temperature** для другого ВП Convert to Binary (NI 9211) VI и назовите его **high target temperature**.
- Соедините выходы **CJC** и **autozero** узла Read/Write Control с входами **CJC Channel** и **Autozero Channel** ВП Convert to Binary (NI 9211) VI
- Соедините константы **Thermocouple Type** и **Temperature Units** с соответствующими входами ВП Convert to Binary (NI 9211) VI.

9. Поместите другой узел Read/Write Control в цикл While Loop справа от первого. Соедините выход **FPGA VI Reference Out** первого узла Read/Write Control с входом **FPGA VI Reference In** второго узла Read/Write Control.
10. Кликните правой кнопкой мыши по элементу **Unselected** и выберите **low target temperature**. Соедините выход **Thermocouple Channel (Binary)** первого ВП Convert to Binary (NI 9211) VI с входом **low target temperature** узла Read/Write Control.
11. Кликните правой кнопкой мыши по новому узлу Read/Write Control и выберите **Add Control**. Кликните правой кнопкой мыши по выходу **Unselected** и выберите **high target temperature**. Соедините выход **Thermocouple Channel (Binary)** второго ВП Convert to Binary (NI 9211) VI с входом **high target temperature** узла Read/Write Control.
12. Соедините выход **FPGA VI Reference Out** второго узла Read/Write Control с входом **FPGA VI Reference In** узла Close FPGA VI Reference.
13. Соедините кластеры **error in** и **error out**.

На рисунке представлена блок-диаграмма готового host VI:



Разработка термостата на базе CompactRIO на этом завершена. Запустите host VI и поэкспериментируйте с различными значениями требуемого диапазона температур, чтобы посмотреть, как ведут себя диод модуля NI 9474 и индикаторы на лицевой панели. В то время как FPGA контролирует ввод/вывод данных и логику, host VI используется для наблюдения за текущей температурой и установки требуемого диапазона температур.

Автор: National Instruments

Источник: Файловый сервер National Instruments [Электронный ресурс]. – Режим доступа: [ftp://ftp.ni.com/pub/branches/russia/software/labview\\_compactrio.pdf](ftp://ftp.ni.com/pub/branches/russia/software/labview_compactrio.pdf)