

Пирометр на ПЛИС

Часть 2

Павел Редькин (ppr1@yandex.ru)

Во второй части статьи, посвящённой описанию работы пирометра на базе промышленного инфракрасного датчика температуры, рассматриваются проект ПЛИС, наладка и регулировка устройства, а также рассказывается об осуществлении обмена данными с ПК.

Проект ПЛИС

Проект ПЛИС пирометра написан на алгоритмическом языке Verilog HDL. Он включает в себя следующие модули: модуль верхнего уровня (файл TEMPER_IR_LCD0802.v), модуль ФАПЧ (файл ALT_PLL.v), модуль подавления дребезжания кнопок (файл noise_filter_but.v), модуль формирования сигналов озвучивания кнопок (файл buzzer_but.v), модуль обмена данными с датчиком температуры семейства MLX90614 (файл MLX90614_wr_rd.v), модуль расчёта CRC (файл CRC8_D40.v), модуль обмена данными с микросхемой FRAM-памяти FM24LC16 (файл FM24LC16_wr_rd.v), модуль передачи байта данных через UART (файл UART_

byte_tx.v), модуль приёма байта данных через UART (файл UART_byte_rx.v), модуль двухступенчатого триггера для формирования положительных импульсов длительностью один период синхросигнала по фронту входного сигнала (файл Trigger_2st_front.v), модуль преобразования чисел из двоичного в двоично-десятичное четырёхразрядное представление (файл binar_BCD_XXXX.v).

Общая структура проекта ПЛИС достаточно полно иллюстрируется структурной схемой устройства [1]. Модуль верхнего уровня включает в себя несколько контроллеров, не выделенных в проекте в отдельные файлы: контроллер ЖКИ, контроллер обработки сигналов кнопок,

контроллер формирования звуковых сигналов, контроллер передачи блока данных через UART, контроллер приёма блока данных через UART, контроллер вычислений и сравнения результата с порогом (вычислитель). В состав вычислителя также входит контроллер вычисления значения CRC и его сравнения со значением CRC, принятым из датчика.

Описанный ранее алгоритм вычисления температуры датчика с некоторыми вариациями реализован в вычислителе. Поскольку все математические операции в этом проекте являются целочисленными, то для достижения требуемых точности и разрешающей способности измерителя они осуществляются над исходными значениями температуры T_A и T_0 , предварительно умноженными на 10. Дробные части этих исходных значений, а также дробные части результатов операций над ними при вычислениях отбрасываются. Таким образом, в качестве конечного результата вычислений получается целочисленное значение температуры, представленное в единицах $0,1^\circ\text{C}$. Это значение в десятичном виде с десятичной запятой перед младшим разрядом отображается на ЖКИ, а также выдаётся через UART. Указанным способом обеспечивается разрешающая способность вычислителя, равная $0,1^\circ\text{C}$. Такое разрешение является достаточным для обеспечения точности измерений, равной $0,2^\circ\text{C}$.

Вместо деления на 50 в проекте ПЛИС реализовано умножение на 41943 с последующим делением на 2097152. Пара чисел 41943 и 2097152 подобрана из расчёта $41943/2097152=0,0199999\approx 0,02=1/50$. Это сделано для обеспечения максимальной точности вычислений. Операция умножения над целочисленными сомножителями в проекте ПЛИС даёт целый результат без появления каких-либо ошибок, однако при делении целых чисел в ПЛИС дробная часть частного будет автоматически отброшена, и это приведёт к появлению ошибки округления, которая будет тем значительнее, чем меньше делимое. Вместо этого исходное значение температуры умножается на 41943, а вместо деления на 2097152 используется логический сдвиг вправо на 21 разряд, поскольку $2097152=2^{21}$. Применённый способ целочисленного деления

Листинг 1

```
// деление результата умножения на 2097152 = 2**21,
// что эквивалентно лог сдвигу вправо на 21 разряд
//result_ - результат умножения слова данных из датчика на 419430
Temper_kelvin_x10_ish <= (result_ >> 21);
```

Листинг 2

```
// Преобразование результата вычитания
// (в регистре result_sub) из дополнительного кода в прямой код
// с выводом на индикацию знака получившегося числа (значение
// температуры)
// Результат преобразования (Temper_decimal) направляется на преобразование
// из двоичного в двоично-десятичный вид
always@(posedge clk_pll)
begin
if (result_sub[15] == 0) // результат вычитания положительный
// (знаковый разряд 0)
begin
Temper_decimal <= result_sub; // преобразование не требуется,
// результат готов
if (Temper_decimal == 0)
begin
REG_LCD_STR1_POZ0 <= " "; // индикация нулевого
// результата
end
else
begin
REG_LCD_STR1_POZ0 <= "+"; // индикация
// положительного результата
end
end
else // результат вычитания отрицательный
// (знаковый разряд 1)
begin
// преобразование отрицательного результата
// из дополнительного в прямой код
Temper_decimal <= ~(result_sub - 1);
// вычитание из числа 1 и
// побитовая инверсия разности
// запись преобразованного результата в прямом коде
REG_LCD_STR1_POZ0 <= "-"; // индикация
// отрицательного результата
end
end
```

не даёт никакой ошибки, за исключением пренебрежимо малой ошибки округления числа 0,0199999 до 0,02.

Таким образом, итоговая формула вычисления значения температуры окружающей среды $T_{АТОГ}$ представленного в единицах $0,1^{\circ}\text{C}$, с учётом предварительного умножения на 10 реализована в вычислителе проекта ПЛИС в следующем виде:

$$T_{АТОГ} = \frac{(T_A \times 419430)}{2097152} - 2732 [0,1^{\circ}\text{C}]. \quad (2)$$

Температура объекта $T_{О ПОГ}$ вычисляется в проекте ПЛИС по формуле, аналогичной (2), с подстановкой значения T_{O} вместо T_A .

Вычисление значений $T_{АТОГ}$ и $T_{О ПОГ}$ в проекте ПЛИС осуществляется в контроллере вычислений и сравнения результата с порогом (в вычислителе), структурная схема которого приведена на рисунке 8.

Для обеспечения умножения на 419430 в проекте ПЛИС используется стандартная мегафункция умножителя LPM_MULT из утилиты MegaWizard Plug-In Manager среды Quartus II. Умножитель представляет собой комбинационное устройство, на которое не подаются никакие входные синхросигналы. Разрядность сомножителей выбрана, исходя из их максимальных возможных значений операндов умножителя.

Деление на 2097152, как уже было сказано, реализуется путём логического сдвига вправо с помощью языковой конструкции, представленной в листинге 1.

Для обеспечения вычитания числа 2732 в проекте ПЛИС используется стандартная мегафункция вычитателя LPM_ADD_SUB из утилиты MegaWizard Plug-In Manager среды Quartus II. В качестве вычитаемого в этой мегафункции используется константа – десятичное число 2732. Уменьшаемое и разность в мегафункции вычитателя должны быть представлены в дополнительном коде. Для всегда положительного уменьшаемого, которое исходно представлено в прямом коде, указанное требование не играет роли, поскольку у положительных чисел представление в дополнительном и прямом коде совпадает. Преобразование разности из дополнительного в прямой код может потребоваться только в случае отрицательной разности, поскольку дальнейшие операции над разностью в вычислителе требуют представления данных в прямом коде. Преобразование кода разности с генерацией знака получившегося результата осуществляется с помощью языковой конструкции, представленной в листинге 2.

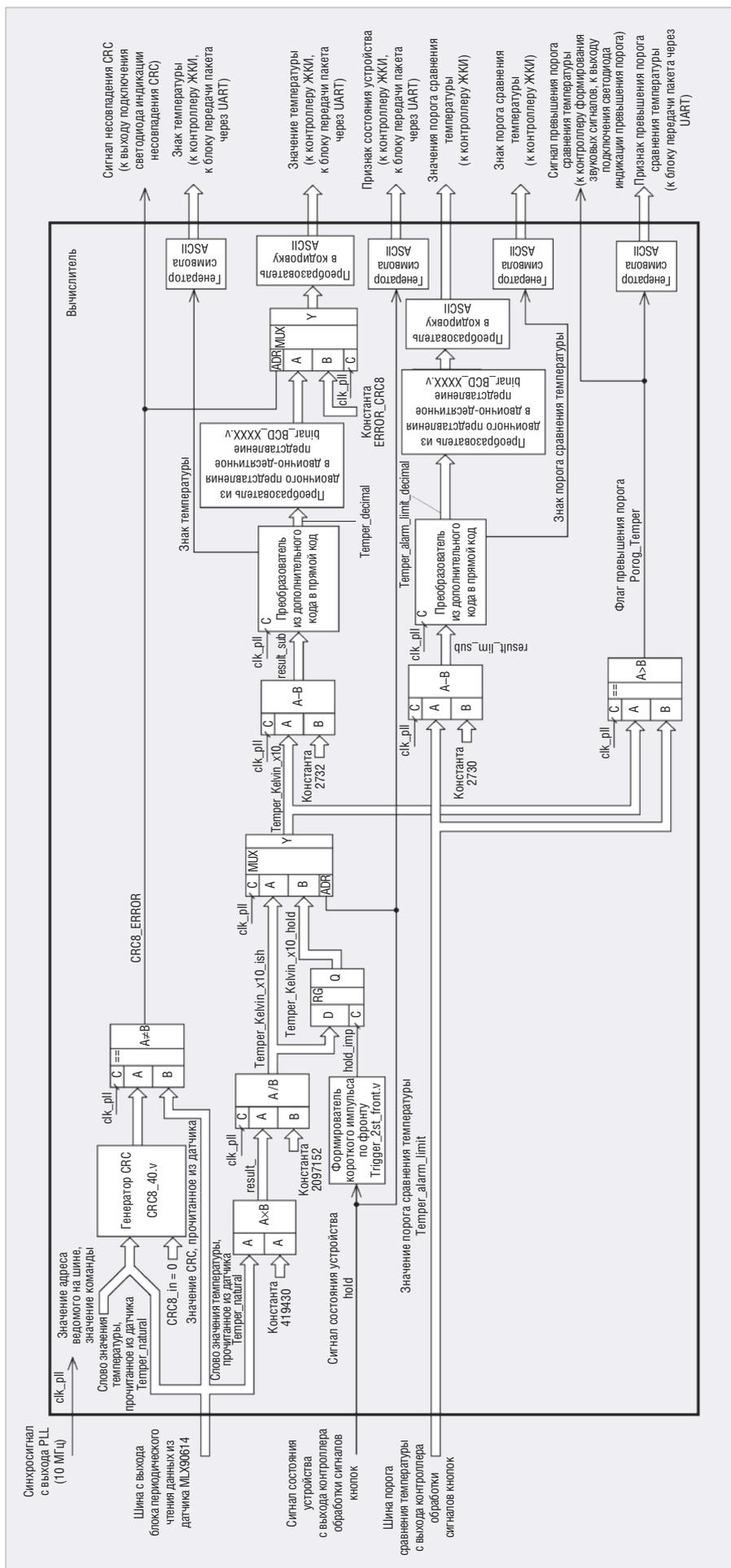


Рис. 8. Структурная схема вычислителя

Листинг 3

```
// преобразование из двоичного в десятичный вид
always @(*)
begin
    Temp0_data <= binar / 1000;
    Temp1_data <= binar % 1000;
    Temp2_data <= Temp1_data / 100;
    Temp3_data <= Temp1_data % 100;
    Temp4_data <= Temp3_data / 10;
    Temp5_data <= Temp3_data % 10;
end

assign BCD_TIS_ = Temp0_data;
assign BCD_SOT_ = Temp2_data;
assign BCD_DES_ = Temp4_data;
assign BCD_EDI_ = Temp5_data;
```

Листинг 4

```
// Направление данных значения температуры на индикацию в ЖКИ
// Если установлен флаг ошибки CRC (CRC8_ERROR),
// то вместо цифрового значения данных отображаются
// символы ERROR_CRC8 (прочерки)
always@(posedge clk_pll)
begin
    if (CRC8_ERROR)
    begin
        REG_LCD_STR1_POZ1 <= ERROR_CRC8;
        REG_LCD_STR1_POZ2 <= ERROR_CRC8;
        REG_LCD_STR1_POZ3 <= ERROR_CRC8;
        REG_LCD_STR1_POZ4 <= ", ";
        REG_LCD_STR1_POZ5 <= ERROR_CRC8;
    end
    else
    begin
        REG_LCD_STR1_POZ1 <= Temper_tis + 48;
        REG_LCD_STR1_POZ2 <= Temper_sot + 48;
        REG_LCD_STR1_POZ3 <= Temper_des + 48;
        REG_LCD_STR1_POZ4 <= ", ";
        REG_LCD_STR1_POZ5 <= Temper_edi + 48;
    end
    REG_LCD_STR1_POZ6 <= GRAD_C;
    REG_LCD_STR1_POZ7 <= "C";
end
```

да – повышенный расход ресурсов ПЛИС.

Затем в вычислителе производится преобразование полученного набора байтовых двоично-десятичных значений тысяч, сотен, десятков и единиц °C в ASCII-код, которое заключается в прибавлении к каждому байту десятичного числа 48, добавление в нужное место десятичной запятой и отправка полученного набора байт на индикацию в ЖКИ и канал передачи UART.

Подготовка ASCII-данных для ЖКИ выполняется с помощью языковой конструкции, представленной в листинге 4.

В этой конструкции анализируется результат сравнения значения CRC, поступившего из датчика, и значения CRC, вычисленного в ПЛИС с помощью собственного генератора. Модуль генератора CRC для пятибайтового блока входных данных на языке Verilog HDL, содержащийся в файле CRC8_D40.v, создавался не автором статьи, а с помощью online-генератора кода HDL, доступно в интернете по ссылке [2]. Внешний вид интернет-страницы этого online-генератора с введенными параметрами описываемого датчика (полином, размер блока данных в битах) показан на рисунке 9. При нажатии кнопки Download на этой странице автоматически создается и предлагается к загрузке файл, содержащий код генератора CRC на языке Verilog HDL с заданными параметрами.

Помимо собственно вычислений значения температуры, заданного пользователем порога сравнения, и их передачи в контроллер ЖКИ, в вычислителе производится сравнение текущего вычисленного значения температуры с этим порогом. При превышении порога одноразрядная переменная Porog_Temper в вычислителе принимает значение 1, в противном случае – значение 0. Кроме того, при превышении порога байтовая переменная Porog_Temper_UART принимает значение R, в противном случае – значение S. Значение переменной Porog_Temper используется в дальнейшем в качестве строба для генерации прерывистого звукового сигнала высокого тона, сигнализирующего о превышении порога, а также (после инверсии) – в качестве сигнала управления светодиодом сигнализации превышения порога. Значение переменной Porog_Temper_UART выводится на ЖКИ, а также передается через UART во внешнее устройство в составе информационной посылки.

Для сравнения с порогом в вычислителе используется умноженное на 10 значе-

The screenshot shows a web interface for generating a CRC polynomial. At the top, it says 'Specify the generator polynomial by clicking on the polynomial terms, or select a predefined value from the list below.' There is a grid of polynomial terms from x^0 to x^{31} . Below the grid, there is a dropdown menu for 'Select a predefined value:' with 'CRC8 ATM HEC' selected. Underneath, there is a 'Data Width' section with a text input field containing '40'. Below that is an 'Output Language' section with radio buttons for 'VHDL' and 'Verilog', where 'Verilog' is selected. At the bottom left, there is a 'Download' button.

Рис. 9. Интернет-страница online-генератора CRC

Далее в вычислителе производится преобразование полученных в прямом коде значений $T_{АЛОГ}$ или $T_{ПОГ}$ из двоичного в двоично-десятичное представление с помощью содержащейся в модуле binar_BCD_XXXX.v языковой конструкции, представленной в листинге 3.

Эта полностью комбинационная схема преобразования демонстрирует решение задачи «в лоб», при котором осуществление деления и получение остатка от деления возлагается на компилятор среды разработки и происходит прозрачно для разработчика. Недосток такого подхо-

ние температуры в единицах 0,1°K, хранящееся в регистре Temper_kelvin_x10.

Для синхронизации вычислителя используется синхросигнал с частотой 10 МГц, поступающий с выхода системы ФАПЧ (PLL). Процессы вычислений над входными данными и преобразования полученных результатов занимают в вычислителе всего несколько тактов этого синхросигнала.

Проект ПЛИС разработан в среде Quartus II бесплатной web-версии 13.0, его конфигурация занимает около 80% ёмкости ПЛИС. Файл с расширением .jic, предназначенный для записи в микросхему конфигурационной памяти ПЛИС, в составе проекта уже содержится. Архив проекта ПЛИС TEMPER_IR_LCD0802.zip доступен в дополнительных материалах к статье на сайте журнала www.soel.ru.

НАЛАДКА И РЕГУЛИРОВКА ПИРОМЕТРА

Правильно собранное устройство в наладке не нуждается и начинает работать сразу же. При включении питания на ЖКИ в течение 2 с отображается начальная заставка – надпись «InfraRed» и номер версии проекта ПЛИС, затем в течение 2 с отображается надпись «Load» с нарастающим заполнением экрана точками, сопровождаемым серией коротких звуковых сигналов. Это имитация начальной загрузки встроенного программного обеспечения (шутка разработчика – ПО в устройстве отсутствует). Затем пирометр переходит в состояние текущих измерений в последнем заданном перед выключением режиме. Однако при первом включении пирометра микросхема памяти FRAM DD4 FM24LC16B в области хранения значения порога ещё чиста, поэтому на ЖКИ будет отображаться нечитаемое значение порога сравнения температуры. Необходимо с помощью нескольких нажатий на кнопки «++», «—», «+», «←» добиться перехода текущего значения порога через верхнюю или нижнюю границы, после чего текущее значение порога станет отображаться корректно. То же самое относится и к значению режима устройства, задаваемому кнопкой MODE. Задав с помощью кнопок управления требуемые корректные значения порога и режима, необходимо выключить питание устройства и через несколько секунд включить его снова для контроля правильности сохранения этих значений в FRAM. При повторном включении последние заданные значения порога и режима в устрой-

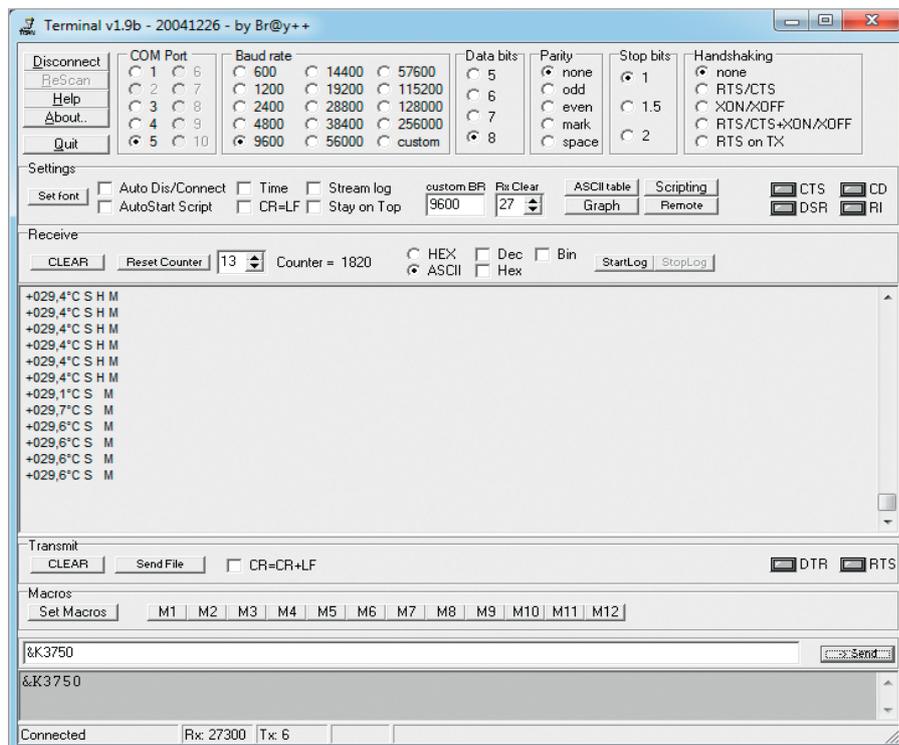


Рис. 10. Окно терминальной программы Terminal с принятыми от пирометра данными

стве должны сохраниться. Если этого не происходит, необходимо проверить правильность подключения к плате Core EP2C5 микросхемы FRAM DD4. Значение состояния удержания в FRAM не сохраняется, по умолчанию после включения питания устройство всегда переходит в состояние текущих измерений.

Нажимать кнопку SB7 «Сброс» для запуска пирометра не требуется – она является технологической.

Регулировку пирометра следует начинать с контроля и, при необходимости, с задания выходного напряжения +4,2 В стабилизатора DA4 путём подбора сопротивлений резисторов R28, R29. Затем следует перейти к заданию контрастности ЖКИ HG1 путём подстройки резистора R15 и заданию яркости подсветки ЖКИ путём подбора сопротивления резистора R14. После этого необходимо задать желаемую громкость звука излучателя BF1 путём подбора сопротивления резистора R16. Правильность подключения кнопок управления SB1...SB6 и их исправность можно определить по наличию короткого звукового сигнала с частотой 500 Гц, генерируемого при нажатии на каждую из них. Ток через лазерный диод HL6 ограничивается резистором R13, смонтированным на корпусе диода.

В случае если по условиям эксплуатации пирометра с помощью него предполагается измерять температуру небольших объектов, необходимо произвести юстировку оптических осей темпера-

турного датчика и лазерного маркера. В авторском варианте устройства лазерный маркер на несущем кронштейне при юстировке остаётся неподвижным, а положение температурного датчика подстраивается под него с помощью металлических шайб, подкладываемых под элементы его крепления (см. рис. 4 [1]). Для проведения юстировки удобно использовать какой-либо точечный объект, имеющий температуру, резко отличающуюся от температуры окружающей среды, например жало паяльника.

ОБМЕН ДАННЫМИ С ПК

Для обеспечения обмена данными между пирометром и ПК через интерфейс USB можно использовать любую из распространённых программ эмуляции простого терминала, например Terminal, CnCTerm, ModemSE, Extem, X-CTU и др., поддерживающих обмен ПК с внешними устройствами через COM-порт. В данном случае обмен физически производится через порт USB ПК, но программно для обеспечения этого обмена в ПК создаётся виртуальный COM-порт, который и следует указать в настройках терминальной программы. Создание виртуального COM-порта в ОС Windows будет произведено автоматически в ходе установки драйвера микросхемы PL2303, который можно загрузить по ссылке [3].

На рисунке 10 показано окно терминальной программы Terminal v1.9b,

Команды управления пирометром через UART

Название команды	Формат команды в кодировке ASCII	
Задание режима	&Rxzzz,	где & – признак начала команды вообще; R – признак начала команды задания режима; x – задаваемый режим: ● A – среда; ● M – объект с маркером; ● N – объект без маркера; zzz – три произвольных (но обязательных) байта
Задание порога сигнализации	&Kuuuu,	где & – признак начала команды вообще; K – признак начала команды задания порога; uuuu – четырёхзначное значение порога, задаваемое в 0,1°K
Задание состояния	&Nxzzz,	где & – признак начала команды вообще; N – признак начала команды задания состояния; x – задаваемое состояние: ● 0 – выключить состояние удержания; ● 1 – включить состояние удержания; zzz – три произвольных (но обязательных) байта

отображающее поступающие из пирометра данные. В состав выдаваемой устройством посылки данных входят знак и значение температуры в °C; признак превышения (символ R) или неперевышения (символ S) этой температурой текущего значения порога; текущий режим устройства (символы A, M или N); состояние удержания (символ H) при его наличии. Для управления пирометром с ПК используются команды, перечисленные в таблице. Все символы в командах латинские. Заметим, что каждая команда должна передаваться в пирометр без каких-либо дополнительных символов, например возврата каретки или перевода строки, в конце.

При получении корректной команды устройство выдаёт короткий звуковой сигнал с частотой 500 Гц. Параметры обмена через виртуальный COM-порт:

- скорость 9600 бит/с;
- 8 бит в байте;
- контроль чётности отсутствует.

При переключении режимов устройства кнопкой MODE в первый момент (несколько сотен мс) после этого значение измеренной температуры на ЖКИ может заменяться прочерками. Так проявляется работа модуля расчёта CRC из состава проекта ПЛИС, который блокирует поступление на индикацию и в канал передачи недостоверных, с его точки зрения, данных.

Несколько слов о практическом применении пирометра. Изготовленное в авторском варианте устройство используется для бесконтактного измерения температуры электрорадиоэлементов. При наладке и ремонте электронных устройств, состоящих, например, из большого количества микросхем, их повышенная или пониженная относительно нормы температура часто позволяет судить об их исправности. При этом норма для каждой конкретной микросхемы может быть определена путём измерения её температуры в исправном устройстве и занесения показаний в так называемую карту тепловых режимов. Измерение температуры корпусов микросхем производится на расстоянии 2–3 см.

Кроме того, возможно применение устройства и для других бытовых и промышленных целей, например бесконтактного измерения температуры агрегатов, жидкостей, смесей, растворов и т. п. в строительстве, химии, физике, кулинарии и других отраслях.

ЛИТЕРАТУРА

1. Редькин П. Пирометр на ПЛИС. Часть 1. Современная электроника. 2019. № 7.
2. <https://www.easics.be/webtools/crctool>
3. https://www.waveshare.com/wiki/File:PL2303_Windows_Driver.7z 

НОВОСТИ МИРА

ЧЕТЫРЕ ЗАБЛУЖДЕНИЯ ОТНОСИТЕЛЬНО «КИТАЙСКОГО КАЧЕСТВА»

Китайские производители довели мастерство изготовления одной и той же продукции, но с разным качеством до совершенства. Закупочные цены на функционально аналогичный продукт в Китае могут различаться в десятки раз. Китай не просто производит дешёвый товар – он адаптирует продукт под покупательную способность экономики той или иной страны. У компаний, закупающих материалы и комплектующие в Китае, существуют заблуждения относительно того, что такое «китайское качество» и каким образом им можно управлять.

Заблуждение первое: иностранные компании могут управлять качеством продукции своих китайских поставщиков. На самом деле это обычно невозможно. Массовое производство в Китае часто осуществляется без прямого контроля со стороны зарубежного клиента. В этих условиях единствен-

ным полноценным участником процесса обеспечения качества продукта в Китае становится сам китайский завод.

Заблуждение второе: можно найти завод в Китае, выпускающий некачественную дешёвую продукцию, и заставить его производить качественно. На самом деле качество и уровень фабрики – тождественные понятия: выбирая фабрику, вы выбираете и качество. Исключения бывают только в тех случаях, когда технологический уровень заказчика гораздо выше уровня китайского производителя.

Заблуждение третье: можно поставить контролёра над каждым рабочим или отбирать только хорошую продукцию во время приёмки товара перед отгрузкой. На самом деле это не так. Чаще всего продукцию для развивающихся стран собирают вручную, и невозможно гарантировать постоянство качества из-за человеческого фактора.

Заблуждение четвёртое: китайские производители способны выпускать и хоро-



шую, и плохую продукцию на одних и тех же технологических площадках. На самом деле китайские производители соотносят качество и цену с определёнными рынками сбыта. Поэтому первым делом закупщик должен задать китайскому производителю вопрос: «С какими рынками вы работаете?» Ответ скажет многое о ценовом диапазоне и о том, на какое качество фабрика ориентируется.

ВЕДОМОСТИ

КОМПЛЕКСНЫЕ ПОСТАВКИ ИБП



ПОСТАВКА, ПУСКОНАЛАДКА, ИНТЕГРАЦИЯ

Широкий ассортимент ИБП, включая модели:

- для альтернативной энергетики
- для приложений с нестабильным основным питанием



НОВОСТИ МИРА

**Электроника Транспорт 2020:
ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
ДЛЯ ПАССАЖИРСКОГО ТРАНСПОРТА**

27–29 мая 2020 года в Москве в КВЦ «Сокольники» пройдёт 14-я международная выставка информационных технологий и электроники для пассажирского транспорта «Электроника-Транспорт 2020».

Выставка собирает поставщиков комплексных решений для управления городским движением, автоматизации движения пассажир-

ского транспорта, учёта пассажиропотока, а также разработчиков систем безопасности, видеонаблюдения, информационной безопасности, оборудования для оплаты проезда, навигационных и информационных сервисов, средств оперативно-диспетчерской связи, автоматизации и телемеханики, диагностических и измерительных приборов для транспортной отрасли. Важную часть экспозиции занимают электронные и электротехнические компоненты и модули для транспортной аппаратуры.

Новые технологии обработки и передачи данных, Интернет вещей, облачные сервисы, безбарьерные системы оплаты проезда, тестирование автономных транспортных систем активно развиваются во всём мире, предлагая пассажирам кардинально иной уровень услуг. Цифровые технологии используются на всех участках работы: от вывода информации о расписании, маршрутах и тарифах до эксплуатации и ремонта подвижного состава, обслуживания и диагностики объектов транспортной инфраструктуры. В нашей стране активно формируется информационное поле пассажирских сервисов, включающее в себя навигацию, билетные решения, выбор маршрута поездки, предоставление доступа к сети «Интернет» и сопутствующих услуг.

В 2020 году выставка пройдёт в составе Российской недели общественного транспорта одновременно с выставкой «ЭлектроТранс» и автобусным салоном CityBus. На мероприятие приглашаются специалисты IT-департаментов организаций, осуществляющих городские и пригородные пассажирские перевозки: метрополитенов, автотранспортных предприятий, пригородных пассажирских компаний, а также вокзалов и автостанций, центров организации дорожного движения, разработчики инновационных сервисов для транспортной отрасли.

В программе «Электроника-Транспорт 2020» запланированы следующие мероприятия:

- 13-я всероссийская конференция «Развитие технологий оплаты проезда на общественном транспорте»;
- конференция «Электронные компоненты и модули для транспортного приборостроения и транспортных систем»;
- круглый стол «Новые технологии для обеспечения безопасности движения»;
- круглый стол «Современные решения для обеспечения информационной и антитеррористической безопасности»;
- круглый стол «Пассажирские сервисы как стимул развития пассажирских перевозок»;
- семинары и презентации участников выставки, технические визиты на объекты транспортной инфраструктуры Москвы и Московской области.

Традиционно в рамках выставки проводится конкурс перспективных разработок «Зелёный Свет», в котором участвуют инновационные разработки 2019–2020 годов. По итогам конкурса экспертный совет выносит оценку каждому представленному образцу, определяет потенциал его применения в транспортном комплексе.

www.e-transport.ru




ХОРОШО ПОД СОЛНЦЕМ, ЕСЛИ ТЫ LITEMAX!

Дисплеи сверхвысокой яркости

- ЖК-дисплеи серии DURAPIXEL™ с яркостью от 800 до 2000 кд/м²
- Размеры по диагонали от 6,5" до 60"
- Разрешение от 640×480 до 1910×1080 (FHD)
- Угол обзора 178° (во всех плоскостях)
- Диапазон рабочих температур (некоторых моделей) –30...+85°C
- Возможна установка сенсорного экрана, защитного стекла
- Разнообразные конструктивные исполнения
- Ресурс до 70 000 часов

PROSOFT®

ОФИЦИАЛЬНЫЙ ДИСТРИБЬЮТОР

(495) 234-0636 • INFO@PROSOFT.RU • WWW.PROSOFT.RU



Реклама



Электроника Транспорт 2020

14-я специализированная выставка электроники и информационных технологий для пассажирского транспорта и транспортной инфраструктуры

Проводится в рамках Российской недели общественного транспорта www.publictransportweek.ru



27-29 МАЯ, МОСКВА
КВЦ «СОКОЛЬНИКИ»
WWW.E-TRANSPORT.RU