

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное
учреждение высшего образования
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
И УПРАВЛЕНИЯ В ТЕХНИЧЕСКИХ СИСТЕМАХ

Кафедра информатики и управления в технических системах

СБОР ИНФОРМАЦИИ ОТ ДАТЧИКОВ IOT

Методические указания
к выполнению лабораторной работы
по дисциплине
"Разработка IoT приложений и сервисов"
для студентов направления подготовки
27.04.04 – Управление в технических системах



Севастополь
СевГУ
2021

УДК 004.7:681.586(076.5)
ББК 32.96-04:32.973.202я73
С232

Р е ц е н з е н т:

В. Н. Мирянова – доцент кафедры "Информатика и управление в технических системах", кандидат техн. наук, доцент

Составитель: А. Е. Осадченко

С232 Сбор информации от датчиков IoT : методические указания к выполнению лабораторной работы по дисциплине "Разработка IoT приложений и сервисов" для студентов направления подготовки 27.04.04 – Управление в технических системах / Севастопольский государственный университет, Институт информационных технологий и управления в технических системах, кафедра информатики и управления в технических системах ; сост. А. Е. Осадченко. – Севастополь : СевГУ, 2021. – 12 с. – Текст электронный.

В методических указаниях приводится описание лабораторной работы, целью которой является моделирование системы сбора информации от датчиков для интернета вещей.

УДК 004.7:681.586 (076.5)
ББК 32.96-04:32.973.202я73

Методические указания рассмотрены и утверждены на заседании кафедры "Информатика и управления в технических системах", протокол № 9 от 15 июня 2021 г.

Методические указания рассмотрены и рекомендованы к изданию на заседании ученого совета Института информационных технологий и управления в технических системах, протокол № 6 от 16 июня 2021 г.

СОДЕРЖАНИЕ

1. ЦЕЛЬ РАБОТЫ.....	4
2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ	4
2.1 УСТРОЙСТВА ИНТЕРНЕТА ВЕЩЕЙ И ИХ ВЗАИМОДЕЙСТВИЕ	4
2.2 ПОДГОТОВКА ПРОГРАММ ДЛЯ МИКРОКОНТРОЛЛЕРОВ	6
2.3 МОДЕЛИРОВАНИЕ СБОРА ИНФОРМАЦИИ ОТ ДАТЧИКОВ	7
2.4 ПРИМЕР МОДЕЛИРОВАНИЯ СБОРА И ОБРАБОТКИ ИНФОРМАЦИИ ОТ ДАТЧИКА ТЕМПЕРАТУРЫ	10
3. ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ	10
4. СОДЕРЖАНИЕ ОТЧЕТА О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ	11
5. ПОРЯДОК ЗАЩИТЫ РАБОТЫ	11
6. КОНТРОЛЬНЫЕ ВОПРОСЫ.....	11
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	12

1. ЦЕЛЬ РАБОТЫ

Моделирование системы сбора информации от датчиков для интернета вещей.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1. Устройства интернета вещей и их взаимодействие

Интернет вещей (англ. Internet of things, IoT) — концепция вычислительной сети физических предметов («вещей»), оснащённых встроенными технологиями для взаимодействия друг с другом или с внешней средой, рассматривающая организацию таких сетей как явление, способное перестроить экономические и общественные процессы, исключающее из части действий и операций необходимость участия человека.

На рис. 1 показаны типы устройств IoT и их взаимодействие [1].

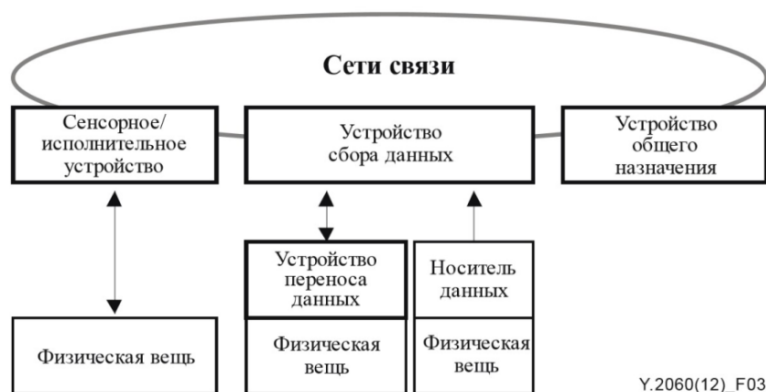


Рис. 1 – Типы устройств IoT и их взаимодействие

Сенсорные и исполнительные устройства взаимодействуют с физическими вещами в окружающей среде.

Устройства сбора данных считывают информацию из физических вещей или записывают её на физические вещи, взаимодействуя с устройствами переноса данных или носителями данных, подключенными или связанными с физическим объектом.

Физический экземпляр элемента IoT представляет собой объект, который имеет микроконтроллер и софт для управления, содержит датчик для измерения каких-либо физических параметров либо исполнительный механизм, работой которого можно управлять.

Датчики устройств IoT обеспечивают взаимодействие физического и виртуального мира, собирая аналоговые данные и преобразуя их в цифровую форму. Чтобы передать собранную информацию, датчики подключаются к сети и взаимодействуют с серверами и шлюзами, используя различные сетевые протоколы. На рис. 2 показаны некоторые типы датчиков [2].

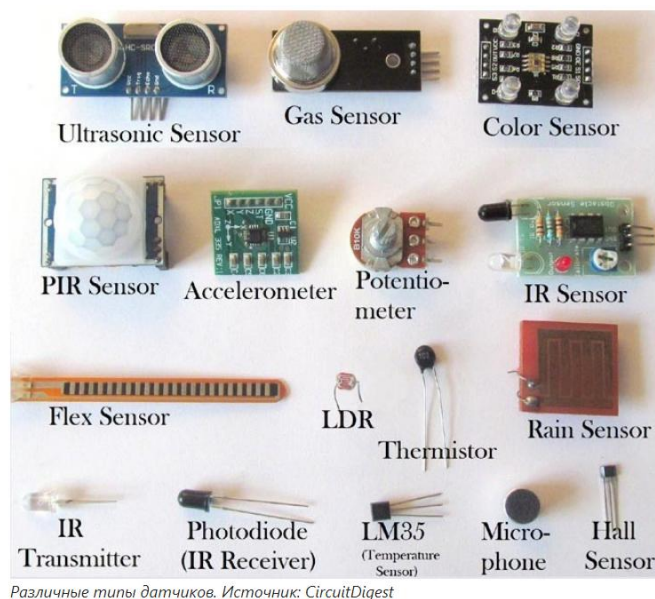


Рис. 2 – Датчики для устройств IoT

Для обработки и преобразования данных в устройствах IoT используют микроконтроллеры различных типов. На рис. 3. Показан внешний вид контроллера Arduino Nano [2] построенного на базе микроконтроллера ATmega 328 [3], модель которого будет использоваться в лабораторной работе для изучения системы сбора информации от датчиков IoT.



Рис. 3 – Контроллер Arduino Nano

Контроллер Arduino Nano имеет 14 цифровых выводов, каждый из которых, может настраиваться как вход или выход. Выводы работают при напряжении 5 В. Каждый вывод имеет нагрузочный резистор (стандартно отключен) 20-50 кОм и может пропускать до 40 мА. Некоторые выводы имеют особые функции.

Последовательная шина – выводы 0 (RX) и 1 (TX). Выводы используются для получения (RX) и передачи (TX) данных TTL. Данные выводы подключены к соответствующим выводам микросхемы последовательной шины FTDI USB-to-TTL.

Внешнее прерывание - выводы 2 и 3. Данные выводы могут быть сконфигурированы на вызов прерывания либо на младшем значении, либо на переднем или заднем фронте, или при изменении значения.

ШИМ– выводы 3, 5, 6, 9, 10, и 11. Любой из выводов обеспечивает ШИМ с разрешением 8 бит.

SPI – выводы 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Посредством данных

выводов осуществляется связь SPI, которая, хотя и поддерживается аппаратной частью, не включена в язык Arduino.

LED – вывод 13. Встроенный светодиод, подключенный к цифровому выводу 13. Если значение на выводе имеет высокий потенциал, то светодиод горит.

На платформе Nano установлены 8 аналоговых входов, каждый разрешением 10 бит. Стандартно выводы имеют диапазон измерения до 5 В относительно земли.

Некоторые выводы имеют дополнительные функции.

I2C – выводы A4 (SDA) и A5 (SCL). Посредством выводов осуществляется связь I2C (TWI).

2.2. Подготовка программ для микроконтроллеров

Для создания ПО контроллеров существуют различные среды разработки. Рассмотрим для примера Arduino IDE — программную среду разработки, использующая C++ и предназначенную для программирования всех плат Ардуино [4] (рис. 4).

Используя эту программную среду, можно, основываясь лишь на знаниях C++, решать самые разные задачи, связанные с программированием и моделированием.

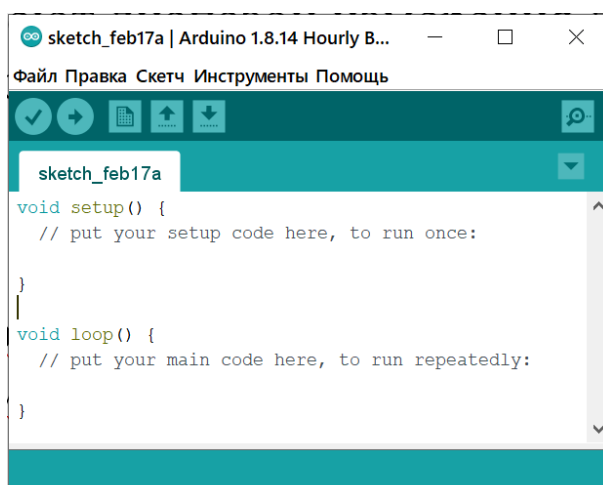


Рис.4 – Arduino IDE

Программная среда Arduino IDE позволяет осуществлять ввод программного кода через последовательный порт компьютера в плату Arduino, при этом программный код сохраняется во временном файле с расширением **hex** (обычно в каталоге TMP) и может быть в дальнейшем использован в некоторых системах моделирования (например, Proteus). Также программный код можно получить при помощи экспорта бинарного файла как показано на рис. 5.

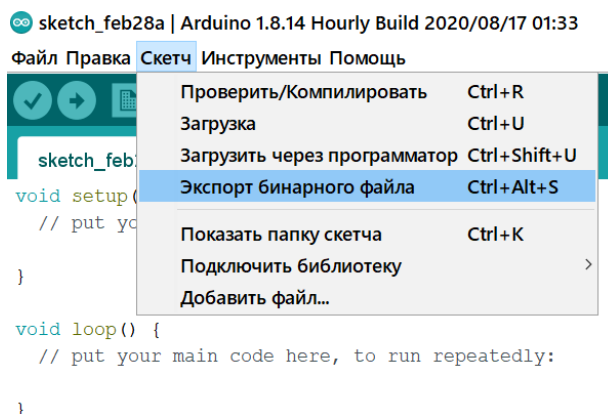


Рис. 5 – Экспорт бинарного файла

2.3. Моделирование сбора информации от датчиков

Рассмотрим возможности системы автоматизированного проектирования электронных схем Proteus Design Suite, которая является разработкой компании Labcenter Electronics (Великобритания) [5].

Proteus позволяет проводить моделирование работы программируемых устройств: датчиков, микропроцессоров, микроконтроллеров, в том числе плат Arduino. Библиотека компонентов системы содержит справочные данные большого числа выпускаемой электронной компонентов, кроме того, возможно подключение внешних библиотек. Дополнительно в пакет PROTEUS VSM входит система проектирования печатных плат. Пакет является коммерческим. Бесплатная ознакомительная версия характеризуется полной функциональностью, но не имеет возможности сохранения файлов.

В Proteus 8 версии есть встроенные библиотеки arduino, которые можно подключить при создании нового проекта на вкладке “from development board” как показано на рис. 6.

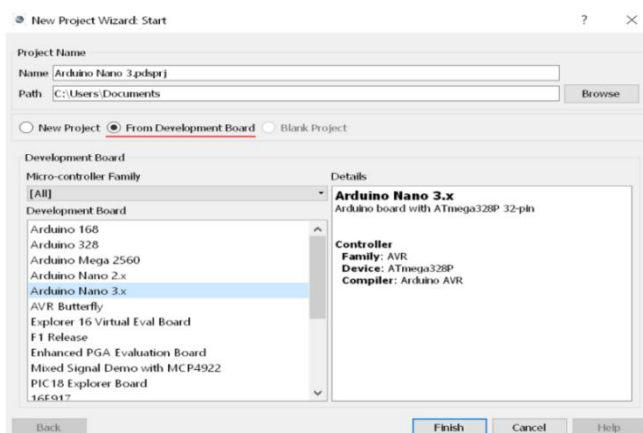


Рис. 6 – Подключение библиотек Arduino при создании проекта

На рис. 7 показано рабочее поле системы PROTEUS с моделью платы Arduino Nano из встроенной библиотеки.

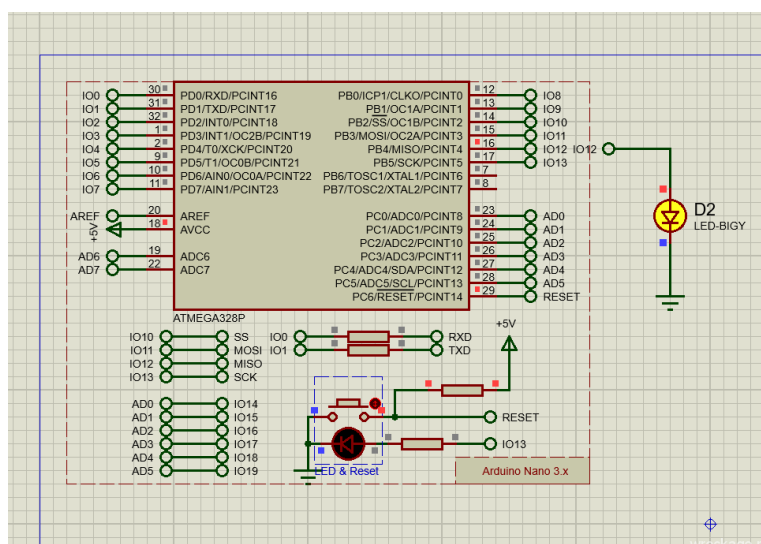


Рис. 7 – Модель платы Arduino Nano

Существуют интернет ресурсы, на которых также можно получить библиотеки элементов для различных систем моделирования. Например, проект THE ENGINEERING PROJECTS[6] предоставляет библиотеки плат Arduino для PROTEUS, которые более удобны в использовании.

На рис. 8 показано рабочее поле системы PROTEUS с платами Arduino из библиотеки THE ENGINEERING PROJECTS.

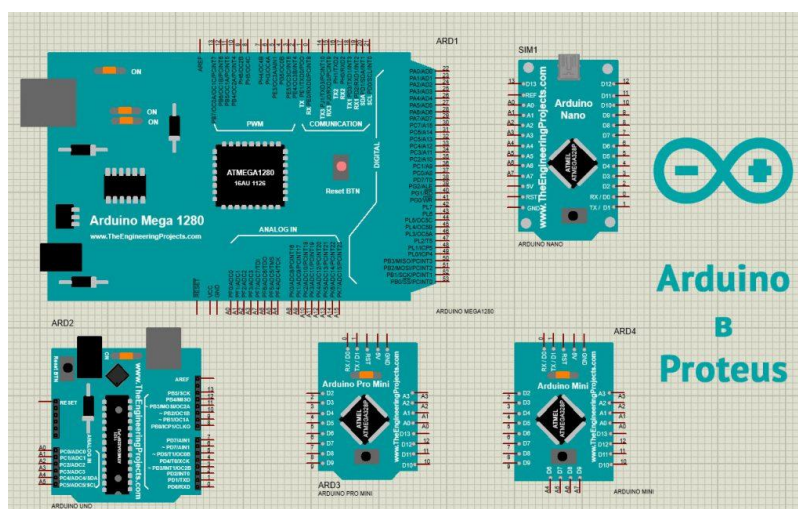


Рис. 8 – Платы Arduino из библиотеки THE ENGINEERING PROJECTS

Для установки библиотек в PROTEUS требуется скачать архив "Library for Proteus" который содержит файлы ArduinoTEP.LIB и ArduinoTEP.IDX. Файлы библиотек необходимо скопировать в каталог LIBRARY программы PROTEUS (путь может быть разным, например "C:\Labcenter Electronics\Proteus 8 Professional\LIBRARY"), после чего библиотеки будут доступны в программе, как показано на рис. 9.

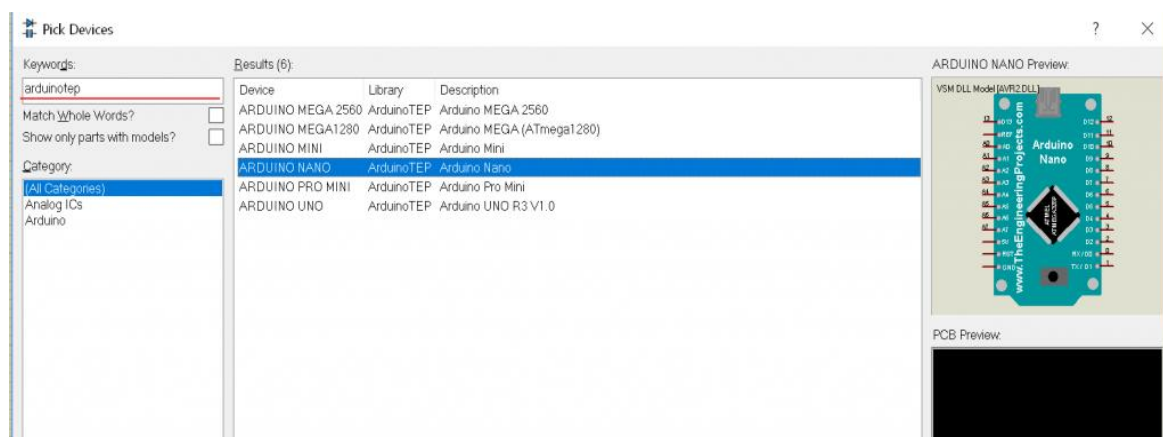


Рис. 9 – Платы Arduino в библиотеке PROTEUS

Для моделирования работы микроконтроллера необходимо в свойствах компонента указать путь к файлу с кодом для микроконтроллера. На рис. 10 показан пример моделирования работы платы Arduino UNO с кодом, полученным при компиляции демонстрационной программы Blink из среды Arduino IDE.

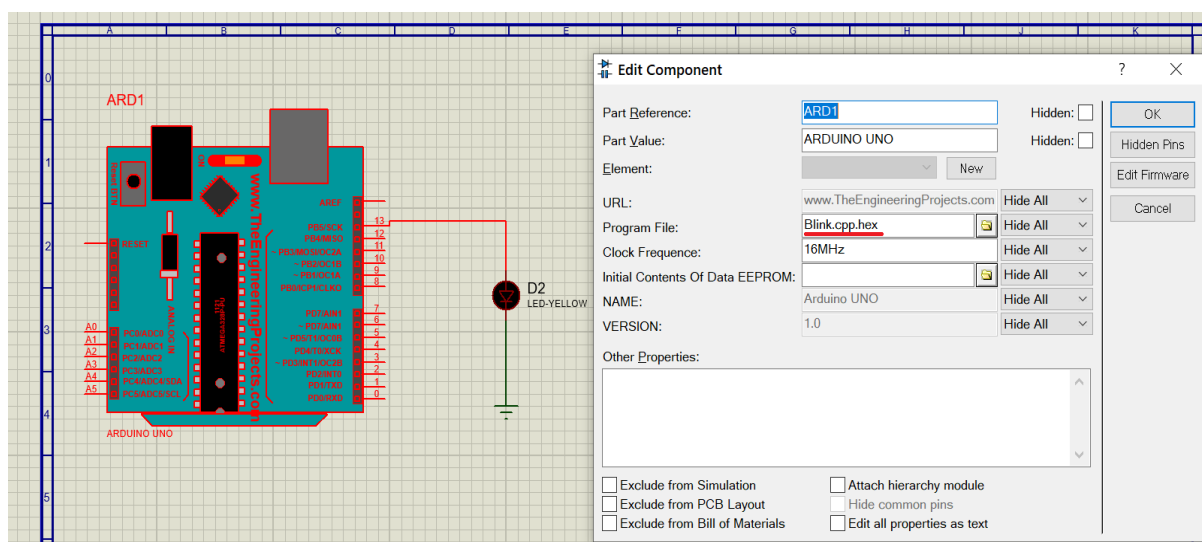


Рис. 10 – Настройка свойств компонента PROTEUS

Стандартная библиотека системы PROTEUS содержит некоторые датчики, например цифровой термометр ds18b20 компании Dallas Semiconductor, при необходимости можно добавить сторонние библиотеки датчиков других типов (газа, огня, инфракрасного излучения и др.). Проект THE ENGINEERING PROJECTS[6] предоставляет библиотеки различных датчиков, установка которых производится таким же образом, как и для платы микроконтроллера.

Необходимо скачать архив, который содержит файлы с расширениями LIB и IDX. Файлы библиотек должны быть размещены в каталоге LIBRARY программы PROTEUS. Для некоторых датчиков требуется в свойствах указать путь к файлу с расширением **hex**, который содержит его программный код, как показано на рис. 11. Этот файл обычно находится в одном архиве с файлами библиотеки датчика.

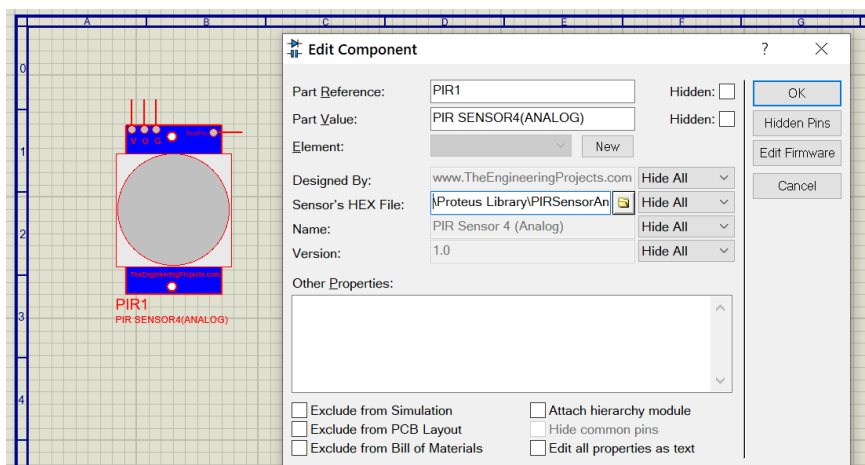


Рис. 11 – Настройка свойств датчика в PROTEUS

2.4. Пример моделирования сбора и обработки информации от датчика температуры

Для моделирования системы мониторинга температуры используется датчик DS18B20. Система должна выводить через последовательный порт информацию о текущей температуре на виртуальный терминал, а также индицировать повышенную температуру (более 27 °C) при помощи красного светодиода и пониженную температуру (менее 22 °C) при помощи синего светодиода.

Для работы с датчиком DS18B20 в среде Arduino IDE можно использовать библиотеку OneWire из проекта Arduino Library List [7]. Архив библиотеки также содержит примеры программ для работы с некоторыми датчиками по интерфейсу 1-Wire.

Схема моделирования обработки данных от датчика температуры в PROTEUS показана на рис. 12.

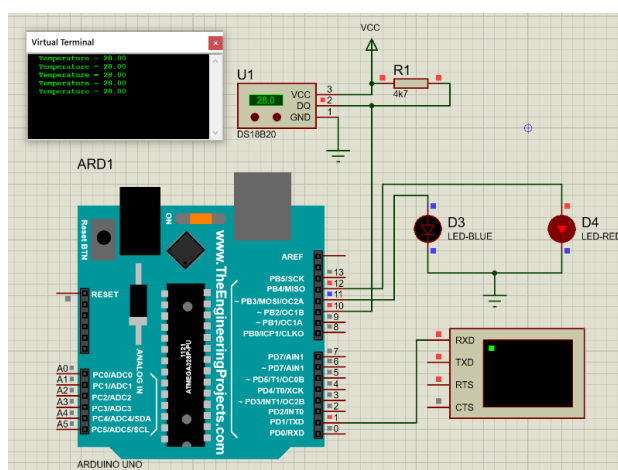


Рис. 12 – Моделирование обработки данных от датчика температуры

3. ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ

Необходимо построить программную модель приема информации от датчиков различных типов в соответствии с вариантами задания, которые приведены в таблице 1. Система должна выводить через последовательный порт информацию о состоянии датчика на виртуальный терминал, а также индицировать превышение

температуры над заданным значением при помощи красного светодиода. Номер варианта задания соответствует номеру студента в списке группы.

Таблица 1 – Варианты заданий

№ варианта	Заданное значение для датчика температуры	Датчик № 2
1	5 °C	природного газа
2	10 °C	пирометр
3	15 °C	влажности
4	20 °C	газа CO ₂
5	25 °C	задымления
6	30 °C	природного газа
7	35 °C	пирометр
8	40 °C	влажности
9	45 °C	газа CO ₂
10	50 °C	задымления
11	55 °C	природного газа
12	60 °C	природного газа
13	65 °C	пирометр
14	70 °C	влажности

4. СОДЕРЖАНИЕ ОТЧЕТА О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ

Отчёт должен содержать:

- титульный лист;
- цель лабораторной работы;
- схемы экспериментов;
- программный код для микроконтроллера с пояснениями;
- выводы.

Отчёт по лабораторной работе должен быть выполнен в соответствии требованиями ГОСТ.

5. ПОРЯДОК ЗАЩИТЫ РАБОТЫ

Защита производится при наличии отчета и состоит в демонстрации работающей программной модели и ответах на контрольные вопросы.

6. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое интернет вещей?
2. Что такое физический экземпляр элемента IoT?
3. Какие задачи выполняют датчики устройств IoT?
4. Для каких целей используют микроконтроллеры в IoT?
5. Что такое программная среда Arduino IDE?
6. Как осуществляется ввод программного кода в плату Arduino?
7. Для каких целей используют Proteus Design Suite?
8. Что такое библиотека элементов?

9. Как установить дополнительные библиотеки элементов в систему моделирования?
10. Приведите примеры датчиков IoT.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Электронный ресурс МСЭ-Т Y.2060. Глобальная информационная инфраструктура, аспекты протокола интернет и сети последующих поколений. — URL: https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-Y.2060-201206-I!!PDF-R&type=items (дата обращения: 10.06.2021).
2. Электронный ресурс для Arduino Nano. — URL: <http://arduino.ru/Hardware/ArduinoBoardNano> (дата обращения: 10.06.2021).
3. Электронный ресурс для ATMEGA328P. — URL: <https://www.microchip.com/wwwproducts/en/ATMEGA328P> (дата обращения: 10.06.2021).
4. Электронный ресурс для Arduino IDE: программная среда для разработки под Ардуино. — URL: <https://arduinoplus.ru/arduino-ide-opisanie-gde-skachat/> (дата обращения: 10.06.2021).
5. Электронный ресурс для системы автоматизированного проектирования электронных схем Proteus Design Suite. — URL: <https://www.labcenter.com> (дата обращения: 10.06.2021).
6. Электронный ресурс THE ENGINEERING PROJECTS. — URL: <http://www.theengineeringprojects.com/2015/12/arduino-library-proteus-simulation.html> (дата обращения: 10.06.2021).
7. Электронный ресурс Arduino Library List: OneWire. — URL: <https://www.arduinolibraries.info/libraries/one-wire> (дата обращения: 10.06.2021).

В авторской редакции