

УДК 338.2

РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ СБОРА ДАННЫХ С НАГРУЗОЧНОГО СТЕНДА

И.С. Кузнецов, студент гр. МРб-121, IV курс
Научный руководитель: И.С. Сыркин, к.т.н., доцент
Кузбасский Государственный Технический Университет
имени Т.Ф. Горбачева
г. Кемерово

В лаборатории Горюче – Смазочных Материалов, которая принадлежит кафедре «Автомобили и автомобильное хозяйство» находится нагрузочный стенд с дизельным двигателем Д37М. Данный лабораторный стенд используется для определения качества топлива, путем нагружения двигателя с использованием гидравлического тормоза. Без использования средств автоматизации для проведения опытов и снятия данных требуется 5-6 человек, при этом снимаемые данные могут быть ошибочны, так как человек может допустить ошибку ввиду человеческого фактора или пропустить некоторые выходные значения. Чтобы не допустить таких ошибок, данный стенд автоматизируется с использованием микроконтроллера Arduino Nano V3.0. двух датчиков угла поворотов и датчика расхода топлива. Автоматизация данного нагрузочного стенда позволит быстро и качественно выполнять заказы предприятий, проводить лабораторные работы или научные опыты, которые не будут требовать такого количества людей для снятия данных с нагрузочного стенда.

Целью данного проекта является разработка программного обеспечения и автоматизированной системы для снятия данных с нагрузочного стенда с двигателем Д37М. Для достижения поставленной цели решались следующие задачи:

- Подбор датчиков и микроконтроллера, которые удовлетворяют требованиям стенда;
- Объединение всех компонентов в единую систему;
- Разработка программного обеспечения, необходимого для сбора данных с датчиков и вывод их на экран монитора компьютера.

Для того чтобы обеспечить питание испытываемого двигателя топливом стенд оборудован системой питания и приборами, измеряющими расход топлива. Топливные баки обычно размещают в том же помещении, где и тормозной стенд. Без использования средств автоматизации измерение расхода топлива обычно производится объемным или весовым способом. При этом необходимо определять удельный вес топлива и учитывать температурную поправку.

Краткие характеристики микроконтроллера приведены в Таблице 1.

Таблица 1 – Характеристики Arduino Nano V3.0.

Параметр	Описание (значение)
Микроконтроллер	Atmel ATmega 168 или ATmega 328
Рабочее напряжение (логический уровень)	5В
Входное напряжение (рекомендуемое)	7-12В
Входное напряжение (предельное)	6-20В
Цифровые Входы/Выходы	14 (6 из них могут использоваться как выходы ШИМ)
Аналоговые выходы	8
Постоянный ток вход/выход	40мА
Флэш-память	16 Кб (ATmega168) или 32 Кб (ATmega328), при этом 2 Кб используются для загрузчика
ОЗУ	1Кб (ATmega168) или 2Кб (ATmega328)
EEPROM	512 байт (ATmega168) или 1 Кб (ATmega328)
Тактовая частота	16 МГц
Размеры	1,85 на 4,2 см

На платформе Arduino Nano установлено несколько устройств необходимые для осуществления связи с компьютером, другими устройствами Arduino или микроконтроллерами. ATmega 168 и ATmega 328 поддерживают последовательный интерфейс UART TTL (5В), осуществляемый выводами 0 (RX) и 1 (TX). Установленная на плате микросхема FTDI FT232RL направляет данный интерфейс через USB порт, а драйверы FTDI (включенные в программу Arduino) предоставляют виртуальный Com порт программе на компьютере. Мониторинг последовательной шины (SerialMonitor) программы Arduino позволяет посылать текстовые данные при подключении к платформе. Светодиоды RX и TX на платформе будут мигать при передаче данных через микросхему FTDI или USB подключении (но не при использовании последовательной передачи через выводы 0 и 1). С использованием библиотеки Software Serial возможно создать последовательную передачу данных через любой из цифровых выводов Nano. ATmega 168 и ATmega 328 поддерживают интерфейсы I2C (TWI) и SPI. В Arduino включена библиотека Wire для удобства использования шины I2C.

Платформа программируется посредством программного обеспечения Arduino. Из меню Tools в разделе Board выбирается нужная нам плата согласно установленному микроконтроллеру. Микроконтроллеры ATmega

168 и ATmega 328 поставляются с записанным загрузчиком, облегчающим запись новых программ без использования внешних программаторов. Связь осуществляется оригинальным протоколом STK500. Также возможно не использовать загрузчик и запрограммировать микроконтроллер через выводы блока ICSP, но это уже будет внутрисхемное программирование.

Краткие характеристики энкодеров приведены в таблице 2.

Таблица 2 – Характеристики энкодеров

Параметр	Описание (значение)
Электропроводка	Зеленый – А фаза Белый – В фаза Красный – VCC (+) Черный – заземление (-)
Выход	Квадратурный импульсный выход
Тип выхода	Открытый коллектор
Назначение	Для измерения скорости вращения объекта, ускорение объекта, угла поворота и измерения длины
Область применения	Используют для интеллектуального управления различного водоизмещения, автоматически фиксированной длины
Особенности	Небольшой размер, легкий вес, невысокая стоимость

Краткие характеристики датчика расхода топлива модели OF05ZAT приведены в таблице 3.

Таблица 3 – Характеристики датчика расхода топлива

Параметр	Описание (значение)
Расход	0 – 10 LPM (единица измерения скорости потока)
Присоединительная резьба	G1/2
Входное питание	3 – 12 VCD
Точность измерений	0,5%
Красный провод	Питание
Черный провод	Заземление
Желтый провод	Выход

Для того чтобы обеспечить сбор данных с датчиков и выведение их на экран монитора компьютера создавались две программы. Цель создания первой программы, написанной на языке Arduino, состоит в снятии данных с датчиков и преобразование этих данных в байтовые массивы. Для получения этих массивов ставятся следующие задачи:

- Считать данные с датчиков;
- Разбить полученные данные на байтовые массивы.

Данные в виде импульсных сигналов поступают с трех датчиков (расхода топлива, и двух энкодеров). Эти данные динамичны, то есть они обновляются каждый раз на протяжении работы датчиков, и чтобы получить действующие значения, необходимо сравнивать новые данные с предыдущими результатами, если они различны, то записываем новые результаты, таким образом, значения в массивах постоянно обновляются. Для простоты передачи данных в компьютер массивы сигналов разбиваются на байтовые массивы. Для этого для каждого датчика ставим условие, чтобы полученные значения сдвигались на 8, затем на 16 и далее на 24 байта; таким образом, получаем байтовые массивы, которые передаются через Com-порт в компьютер. После поступления массивов байтов, работу с ними начинает вести вторая программа. Цель создания второй программы – получение массивов в байтах из Com – порта, преобразование их в целые числа и вывод этих чисел на экран монитора компьютера. Для достижения данной цели разработана программа на языке программирования C# (Си Шарп). Для того чтобы получать измеренные данные с датчиков устанавливается связь компьютера с микроконтроллером с оптимальной скоростью обмена данными, с использованием соответствующих команд. После получения посылки данных состоящих из массивов байтов, необходимо выполнить обратное преобразование массивов байтов в целые числа. Но необходимо не просто считать данные и вывести их на экран монитора компьютера, а необходимо разделить эти данные на три массива, каждый из которых пополняется от своего датчика. Важным условием считывания и преобразования данных – обязательная последовательность байтов 255 255 255, после нахождения этой последовательности, берется последующий индекс поле данного условия и выделяется по четыре байта и сдвигаем сначала на 8 потом на 16 и наконец на 24 байта. То есть, выполняем обратное преобразование, и в результате получаем целые числа. Общая последовательность байтов состоит в том, что устанавливаем последовательность, какая четверка отвечает за какой датчик. Эта последовательность имеет вид: Энкодер 1, Энкодер 2, Датчик расхода топлива. После того как мы вывели числа на монитор, для лучшего усвоения информации строятся графики, каждый из которых отвечает за свой массив. Таким образом, мы получаем как численные массивы, так и их графическое представление в виде графиков. Интерфейс рабочей программы представлен на рисунке 1.

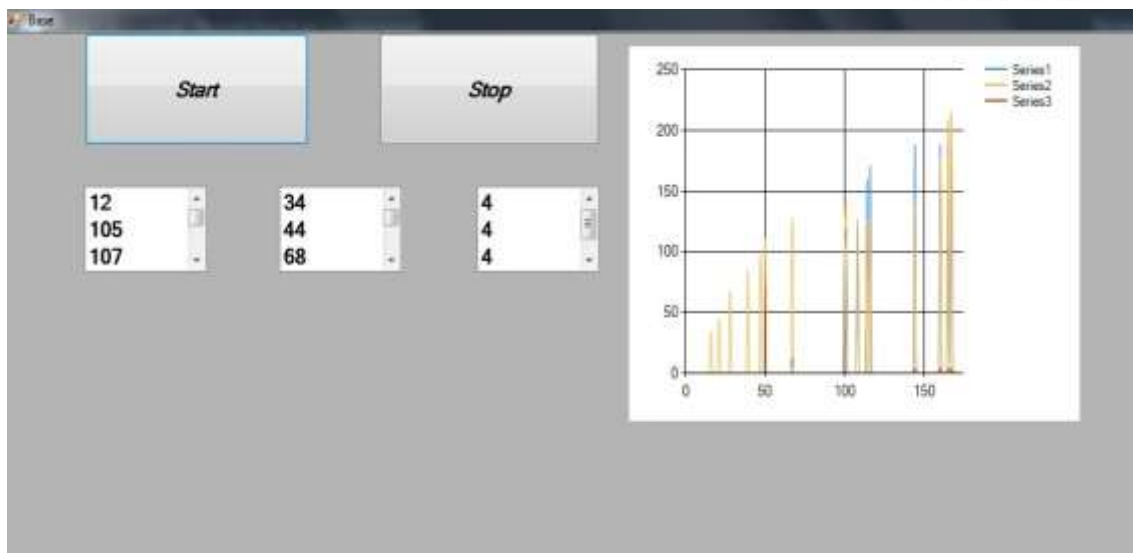


Рисунок 1 – Интерфейс рабочей программы

В конечном итоге получаем программное обеспечение и автоматизированную систему для сбора данных с нагрузочного стенда с двигателем ДЗ7М.

Список литературы:

1. Петров И.Е. Программируемые микроконтроллеры. Стандартные языки и инструменты/ Под.ред.проф. В.П. Дьяконова – М.:СОЛОН-Пресс, 2003 – 256 стр. – (Серия проектирования)
2. Герберд Шилдт С# полное руководство.: Пер. с англ. – М.: ООО «И.Д. Вильямс»,2011. – 1056 с.: ил. – Парал. тит. англ.