

# МОДЕЛЬ СИСТЕМЫ УПРАВЛЕНИЯ НА ОСНОВЕ ОБРАБОТКИ И АНАЛИЗА ДАННЫХ В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ ИНСТРУМЕНТАМИ MATLAB

Высоцкий Александр Александрович  
Руководитель: Чертков Валерий Михайлович, аспирант  
Учреждение образования «Полоцкий государственный университет»

*Рассматривается структурная схема и практическая реализация модели системы управления на основе обработки и анализа данных в режиме реального времени. Рассмотрены интерфейсы взаимного обмена потоками данных системы MATLAB и внешнем оборудовании. Представлен пример использования обработки данных в режиме реального времени с применением системных объектов MATLAB.*

## Введение

Системы связи и управления являются наиболее важными аспектами при работе с программно-аппаратными комплексами обработки и анализа получаемой информации. Прямая связь с оборудованием и периферийными устройствами позволяет быстро тестировать и верифицировать алгоритмы, а также вносить необходимые изменения в разрабатываемые проекты. Программное обеспечение MATLAB предоставляет множество различных способов взаимодействия со сложным оборудованием и их периферией.

Система компьютерной математики MATLAB фирмы MathWorks является интерактивной средой, в которой используется язык высокого уровня для численных вычислений, визуализации и программирования. На основании этой системы, можно разрабатывать алгоритмы, анализировать данные, создавать модели и приложения [5].

Использование цифровых осциллографов и аналогово-цифровых преобразователей совместно с программой математического моделирования MATLAB открывает большое количество перспективных возможностей в обработке и последующем использовании измеряемых данных для проектирования систем различного назначения.

Основной сложностью при реализации процесса обмена данными являются передача потока измеренных в процессе работы данных в рабочее пространство MATLAB и обеспечение аппаратной и программной совместимости подключаемого оборудования и персонального компьютера (ПК).

## Постановка задачи

В управляющих системах часто возникает задача обработки большого объема информации в реальном времени, который формируется на основе данных, полученных в результате измерений. Такая обработка практически невозможна без дорогостоящего физического прототипа или оборудования. Однако даже при наличии такого оборудования разработка, тестирование и верификация алгоритмов может быть весьма рискованной и повлечь за собой даже его повреждение. Более надежным и менее дорогостоящим является метод построения модели целевой системы и симуляция ее работы (например в системе MATLAB) в различных режимах работы до запуска на реальном оборудовании. [1]

Структурная схема простой модели управления на основе анализа и обработки данных в реальном времени представлена на Рисунке 1.

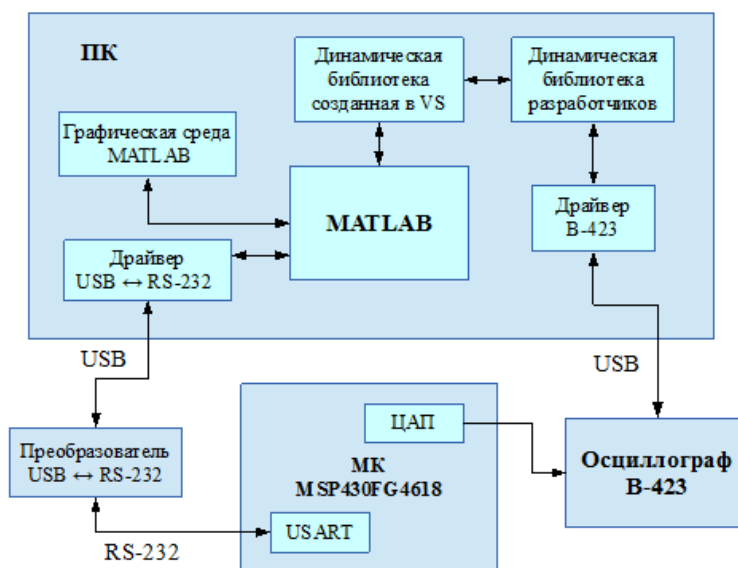


Рисунок 1 – Структурная схема программно-аппаратного комплекса.

В качестве осциллографа будем использовать осциллограф компании АУРИС В-423. Готовых решений

для использования его совместно с MATLAB производитель не дает. Однако в комплекте с программным обеспечением (ПО) поставляются динамические библиотеки (\*.dll) для разработки стороннего ПО по использованию приобретенного оборудования.

В роли управляемого генератора выступил микроконтроллер (МК) фирмы Texas Instruments серии MSP430FG4816. Этот высокоинтегрированный МК позволяет использовать практически любые сочетания периферийных устройств семейства MSP430. Удобство подключения разнообразных интегрированных периферийных устройств и других приборов предполагает достаточно широкие возможности в области разработки оборудования и делает микроконтроллер идеальной платформой для ее основы. [2]

Перейдем к реализации системы, представленной на рисунке 1.

### 1. Интерфейсы связи с внешним оборудованием

Пакет MATLAB включает различные интерфейсы для получения доступа к данным, клиентам и серверам и внешним подпрограммам, написанных на других языках программирования. К самым широко используемым интерфейсам относятся:

- COM. MATLAB предоставляет доступ к функциям, позволяющим создавать, манипулировать и удалять COM-объекты (как клиентские, так и серверные).

- NET. MATLAB в Microsoft Windows предоставляет доступ к программной платформе .NET Framework. Имеется возможность загружать .NET сборки (Assemblies) и работать с объектами .NET классов из среды MATLAB. [3]

- DDE. MATLAB содержит функции, которые позволяют ему получать доступ к другим приложениям среды Windows, равно как и этим приложениям получать доступ к данным MATLAB.

- COM-порт. Интерфейс для последовательного порта пакета MATLAB обеспечивает прямой доступ к периферийным устройствам и научному оборудованию, подключающемуся к компьютеру через последовательный порт (COM-порт).

- DLL. Интерфейс MATLAB, относящийся к общим DLL библиотекам позволяет вызывать функции, находящиеся в обычных динамически подключаемых библиотеках, прямо из MATLAB. При этом также функции должны иметь C-интерфейс.

Для подключения осциллографа В-423 к компьютеру применяется интерфейс связи USB, причем передачу данных по которому от осциллографа к MATLAB можно осуществить по двум технологиям NET и DLL. Стоит отметить, что прямое использование динамической библиотеки (DLL) внешних функций (OSCB423Lib.dll) для данного осциллографа в MATLAB не оказалось возможным, в связи с неизвестной структурой библиотеки разработчиков. Поэтому для рассматриваемой нами модели управления осциллографом мы применили технологию NET, для которой была написана собственная динамическая библиотека простых функций (Osc.dll) на языке C# в среде Visual Studio 2013. Написанная библиотека является согласующим звеном между динамической библиотекой (OSCB423Lib.dll) разработчиков и MATLAB. В составе созданной библиотеки находятся только 2 функции: *Setting()* — для настройки основных параметров осциллографа и *Measuring()* — для произведения измерения и возвращения полученных данных в MATLAB.

Для управления микроконтроллером выбран интерфейс COM-порт, который обеспечивает простое прямое подключение по интерфейсу связи RS-232. В следствии того, что система разрабатывалась на ноутбуке, а не на персональном компьютере в систему был введен дополнительный интерфейс - преобразователь USB-RS232.

### 2. Использование МК для работы в качестве управляемого генератора

В качестве генератора сигнала нами был использован периферийный модуль 10-ти битного ЦАП, а для управления применялся встроенный периферийный универсальный синхронно-асинхронный приемопередатчик USART. Разработанный программный код для МК протестирован с помощью программного обеспечения Code Composer Studio v6. [2]

### 3. Инициализация осциллографа для измерений

Подключение библиотеки по технологии NET в MATLAB происходит следующей командой в консоли приложения:

```
>> NET.addAssembly('путь_к_библиотеке\Osc.dll');
```

Создание NET объекта, соответствующего классу из нашей динамической библиотеки (Osc.dll) происходит с помощью команды:

```
>> obj = Osc.Program;
```

### 4. Выполнение измерения и передача потока полученных данных

Настройка осциллографа для измерения осуществляется командой:

```
>> obj.Setting();
```

Измерение и передача потока полученных данных в рабочее пространство MATLAB производится командой:

```
>> X = double(obj.Meas(ScaleX, ScaleY));
```

Для настройки развертки выбирают переменные *ScaleX* и *ScaleY*, значение которых будут определять соответственно количество выборок за секунду и развертку по напряжению. Причем при выборе этих значений переменных *ScaleX* и *ScaleY* используются величины, которые предопределены в библиотеке разработчиков.

## 5. Цифровая обработка потоковых данных

Использование технологии потоковой и покадровой обработки, позволяет ускорить моделирование благодаря буферизации входных данных во фреймы и одновременной обработки множества семплов данных. Ускорение моделирования достигается также и за счет сосредоточения фиксированного процесса над обработкой сразу многих данных. В MATLAB потоковая обработка сигналов достигается на основании использования системных объектов (System objects) для представления управляемых данными алгоритмов, инструментов ввода и вывода данных из моделей. Системные объекты косвенным образом управляют многими параметрами работы системы такими, как индексирование данных, буферизация и регулирование состояния алгоритмов. В одной программе можно объединять системные объекты со стандартными функциями и операторами MATLAB.

Приведем пример использования системного объекта `dsp.FFT()`, который вычисляет дискретное преобразование Фурье (ДПФ) из отсчетов входного сигнала. Системный объект может использовать один из нескольких типов алгоритма быстрого преобразования Фурье (FFT). При создании системного объекта как правило указывают параметры: длину входного дискретного сигнала, расчетное окно, битреверсивный порядок выходных отсчетов, тип алгоритма. Создание системного объекта в консоли MATLAB выглядит следующим образом [4]:

```
>> hfft = dsp.FFT('FFTLenghSource', 'Property', 'FFTLengh', 1024, 'BitReversedOutput', false, 'WrapInput', true, 'FFTImplementation', 'auto');
```

Использование объекта для вычислений

```
>> Y = step(hfft, y);
```

Здесь *y* - это входной вектор данных или кадр в потоке данных. *Y* – расчетный вектор для этого кадра БПФ.

## 6. Представление результатов измерений

В результате проделанных вычислений обычно получается большой объем данных, который трудно анализировать без наглядной визуализации. Графические возможности MATLAB в этом случае являются мощными и разнообразными. Их часто называют высокоуровневой графикой. Это название отражает тот факт, что пользователю нет никакой необходимости вникать во все тонкости и подробности непосредственной работы с графикой. При этом в случае необходимости, графическое представление обрабатываемых данных всегда можно легко представить в виде отдельного Windows-приложения.

На рисунке 3 представлен пример графического отображения результата обработки и анализа потока данных, измеренного с помощью осциллографа.

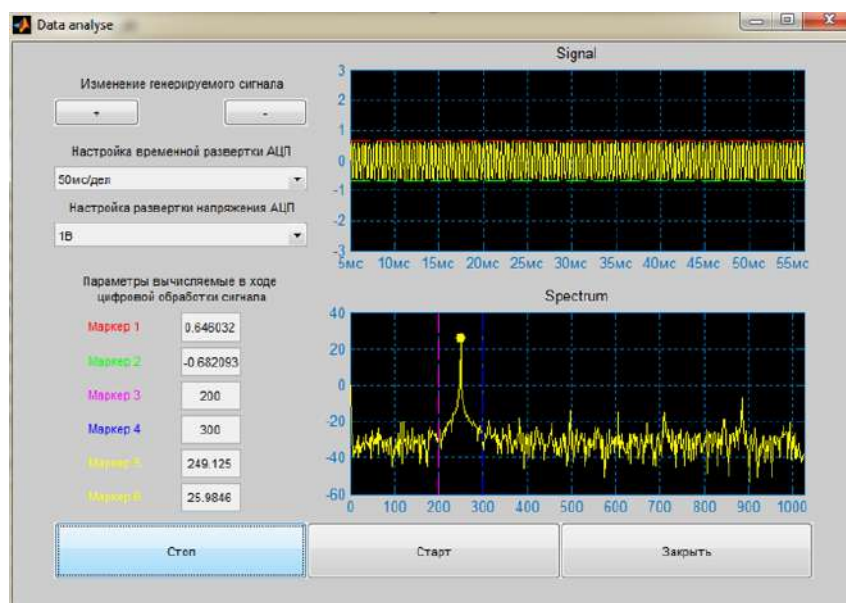


Рисунок 3 - Графический интерфейс MATLAB.

На графический интерфейс пользователя налаживаются:

- график, который представляет данные полученные от осциллографа в виде временного представления сигнала;

- график, который отображает обработанные данные от осциллографа с помощью системного объекта в виде спектра.

Анализ данных происходит посредством четырех маркеров, которые контролируют максимальное и минимальное значение амплитуды сигнала и максимальную относительную мощность сигнала и частоту гармоник, на которой она рассчитана.

Маркер 1 – это максимальная амплитуда сигнала, выраженная в Вольтах

Маркер 2 – это минимальная амплитуда сигнала, выраженная в Вольтах

Маркер 3 и Маркер 4 – это настройка диапазона частот по управлению

Маркер 5 и Маркер 6 – это положение точки на спектральном графике

## 7. Определение управляющего воздействия на генератор сигнала

Определение необходимости воздействия на генератор сигнала MATLAB принимает на основе анализа данных от расставленных маркеров 3 и 4. (см рис 3) и расчетных значений проставленных в маркерах 1, 2, 5 и 6.

Задача разработанной системы управления обеспечить удержание частоты генерируемого колебания в пределах установленных с помощью маркеров 3 и 4. Система может регулировать амплитуду колебания и частоту посредством передачи команд МК через интерфейс RS-232. Алгоритм принятия решения по для выбора той или иной команды на данном этапе достаточно прост:

- команда по увеличению напряжения выдается при низких значениях измеряемого напряжения;

- команда изменения развертки по напряжению выдается в том случае если на протяжении одной секунды с команда по увеличению напряжения не срабатывает;

- команда изменения развертки по временной оси выдается в случае, если при заданной частоте дискретизации диапазон частот, определяемый маркерами будет либо в начале, либо в середине всего диапазона частот,

- команда на увеличение или уменьшения частоты генерируемого колебания выдается в зависимости от положения расчетного значения частоты относительно диапазона, определяемого маркерами 3 и 4.

## Заключение

В результате проделанной работы нами была разработана структурная схема модели системы управления на основе обработки и анализа данных в режиме реального времени.

Реализована интеграция осциллографа В-423 фирмы «Аурис» посредством использования технологии .NET для получения потока данных.

Написан собственный графический интерфейс пользователя по отображению и анализу полученных данных от осциллографа в реальном времени и управления генератором в автоматическом и ручном режиме.

Представлен пример использования обработки данных в режиме реального времени с применением системных объектов MATLAB.

Созданная нами модель охватывает практически все основные этапы разработки систем управления на основе данных полученных в режиме реального времени.

## Список литературы

1. Pravallika Vinnakota. Управление двигателем с помощью Arduino: Пример информационного моделирования и разработки системы управления. [Электронный ресурс]. – Режим доступа: <http://matlab.ru/articles/motor-control-with-arduino.pdf>. Дата доступа 10.10.2014.

2. Семейство микроконтроллеров MSP430x2xx. Архитектура, программирование, разработка приложений/ пер. с англ. Евстифеева А. В. — М. : Додэка-XXI, 2010. — 544 с.: ил. — (Серия «Мировая электроника»). — ISBN 978-5-94120-221-8.

3. NET.addAssembly. [Электронный ресурс]. – Режим доступа: <http://www.mathworks.com/help/matlab/ref/net.addassembly.html>. Дата доступа 20.09.2014.

4. MATLAB и быстрое преобразование Фурье. [Электронный ресурс]. – Режим доступа: <http://habrahabr.ru/post/112068/>. Дата доступа 04.09.2014.

5. MathWorks. Accelerating the pace of engineering and science [Electronic resource] / The MathWorks, Inc., 2013. – Mode of access: <http://www.mathworks.com/index.html/>. – Date of access: 28.02.2013.