

<sup>1</sup>Канд. техн. наук, доцент, доцент кафедры автоматизации и компьютерно-интегрированных технологий Харьковского национального автомобильно-дорожного университета, Харьков, Украина

<sup>2</sup>Канд. техн. наук, доцент, доцент кафедры автоматизации и компьютерно-интегрированных технологий Харьковского национального автомобильно-дорожного университета, Харьков, Украина

<sup>3</sup>Инженер, младший научный сотрудник кафедры автоматизации и компьютерно-интегрированных технологий Харьковского национального автомобильно-дорожного университета, Харьков, Украина

## ПОВЫШЕНИЕ ТОЧНОСТИ ОЦЕНКИ СОСТОЯНИЯ ДИНАМИЧНЫХ ОБЪЕКТОВ КОМПЛЕКСОМ MATLAB-ARDUINO ПРИ ПРОЕКТИРОВАНИИ КИБЕР-ФИЗИЧЕСКИХ СИСТЕМ

Решена задача повышения эффективности взаимодействия MATLAB и Arduino при проектировании кибер-физических систем путем внесения изменений в реализацию стандартного протокола обмена со стороны Arduino. Предложено при запросе MATLAB на чтение данных с первого аналогового порта Arduino выполнять аналогово-цифровое преобразование данных со всех требуемых аналоговых портов с дальнейшей последовательной передачей полученных данных в MATLAB, что позволяет повысить качество управления динамичными процессами за счет уменьшения области неопределенности состояния многомерной быстродействующей системы управления. Кроме того, предложено функции предварительной обработки показаний датчиков выполнять средствами Arduino, что повышает гибкость кибер-физической системы за счет возможности изменения аппаратного обеспечения без изменения программного кода и протокола обмена MATLAB. Формальное описание взаимодействия MATLAB и Arduino позволяет реализовать протокол обмена с использованием беспроводных интерфейсов, микропроцессорных устройств и платформ, не совместимых с Arduino.

**Ключевые слова:** кибер-физическая система, MATLAB, Arduino, протокол обмена, период опроса.

### НОМЕНКЛАТУРА

CPS – кибер-физическая система;

$e$  – ошибка управления;

$k$  – момент дискретизации по времени;

$K_p$  – коэффициент усиления пропорционального канала регулятора;

$T_i, T_d$  – постоянные времени интегрирующего и дифференцирующего каналов регулятора;

$T_R$  – период чтения данных;

$T_s$  – период дискретизации по времени;

$T_w$  – период записи данных;

$u$  – управляющее воздействие;

$val$  – символьная переменная.

### ВВЕДЕНИЕ

В результате взаимодействия встроенных систем управления и сетевых технологий возникла новая технология управления объектами и процессами – кибер-физическая система (cyber-physical system, CPS), которая характеризуется тесной интеграцией и координацией между вычислительными и физическими процессами при помощи сетевых технологий [1]. CPS широко внедряются в самые разнообразные отрасли производства: транспорт, энергетику, машиностроение, робототехнику, металлургию, а также в биомедицинские системы.

Проектирование CPS требует наличия более надежных моделей физических процессов, протекающих в системах управления [2, 3, 4]. От того, как модель соотносится с реальностью, зависит работоспособность CPS.

В промышленной практике аппаратные и программные составляющие многих систем управления до сих пор разрабатываются отдельно, без учета их взаимодействия между собой и с физическим миром. И уже после разработки системы управления, проверки ее на моделях,

устраняется влияние различного рода неопределенностей путем использования специальных методов настройки. Этот процесс является трудоемким и дорогостоящим, а с усложнением систем – практически неосуществимым.

Альтернативным подходом к проектированию CPS является использование технологии модельно-ориентированного проектирования [4]. Ее суть заключается в том, что вместо физических прототипов в специализированном программном пакете создается имитационная модель объекта управления, на которой отрабатывается и совершенствуется разрабатываемый алгоритм управления. В случае достижения алгоритмом поставленной задачи выполняется автоматическая генерация программного кода для целевой платформы. Чаще всего такое проектирование проводят в средах LabVIEW компании National Instruments и MATLAB/Simulink компании Mathworks [5, 6]. Возможности этих продуктов существенно расширяются при использовании связи с физическим миром, для чего в них предусмотрены коммуникационные функции. Различные датчики и исполнительные устройства могут подключаться через COM порт или USB. Это позволяет перейти от имитационных моделей к гибридным, в которых сочетаются как модели сложных объектов, так и реальные физические устройства.

Однако использование указанных функций при необходимости реализации протоколов обмена вызывает определенные трудности, и до недавнего времени было доступно лишь специалистам узкого профиля, поэтому встречалось довольно редко.

Ситуация изменилась с появлением платформы Arduino, которая существенно упрощает разработку CPS. В частности, унифицированные функции взаимодействия с сервером на Arduino получили поддержку в MATLAB путем разработки приложения «Arduino

Support from MATLAB» [7]. Данная работа направлена на повышение эффективности взаимодействия MATLAB и платформы Arduino при гибридном (модель/устройство) подходе к проектированию систем управления.

### 1 ПОСТАНОВКА ЗАДАЧИ

Привлечение MATLAB позволяет не только упростить процедуру проектирования, но и решать достаточно сложные задачи управления, используя ПЭВМ с установленным MATLAB в качестве ПЭВМ верхнего уровня, в то время как платформа Arduino решает задачи по получению и преобразованию информации о значениях контролируемых параметров системы управления и реализует протокол обмена доступа к серверу (рис. 1).

Реализация текстового обмена вида запрос/ответ между MATLAB и Arduino осуществляется по последовательному USB интерфейсу с использованием механизма виртуальных COM-портов (рис. 2) при помощи команд, сходных с командами среды Arduino. Известны реализации серверных протоколов [7], где для повышения гибкости управления каждая команда оперирует одним из 14 цифровых или одним из 6 аналоговых выводов. В текстовых запросах/ответах каждому порту (pin 0...13) Arduino ставится в соответствие символьное значение a...n переменной val (табл. 1).

Основные команды, которые содержит стандартный текстовый протокол обмена MATLAB и Arduino, реализованный в MATLAB и в скетче adiosrv\_UNO.ino для Arduino (настройка линий цифрового ввода-вывода, собственно цифровой ввод-вывод, аналоговый ввод и псевдоаналоговый вывод ШИМ), содержат код команды (функции) и строку параметров (рис. 3):

```
< function><pin>
<function><pin><val>
```

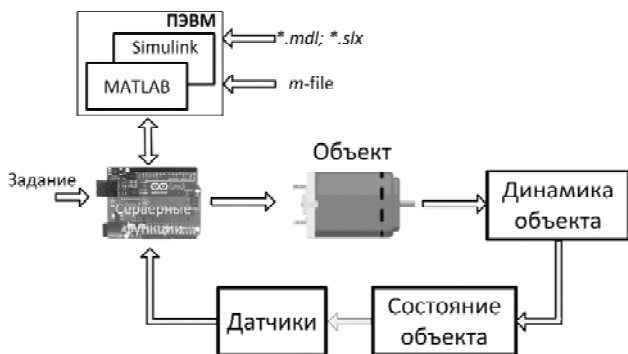


Рисунок 1 – Структура системы управления с Arduino



Рисунок 2 – Структурная схема взаимодействия MATLAB и Arduino

Таблица 1 – Соответствие номера порта символьной переменной val

val	a	b	c	d	e	f	g	h	i	j	k	l	m	n
pin	0	1	2	3	4	5	6	7	8	9	10	11	12	13

$function \in \{ '0' \dots '4' \} = \{ 0x30 \dots 0x34 \} = \{ 48 \dots 52 \}$  – код запроса/ команды MATLAB;

$pin \in \{ 'a' \dots 'n' \} = \{ 0x61 \dots 0x6E \} = \{ 97 \dots 110 \}$  – символьные обозначения цифровых линий pin0 ... pin13;

$pin \in \{ 'a' \dots 'f' \} = \{ 0x61 \dots 0x66 \} = \{ 97 \dots 102 \}$  – символьные обозначения аналоговых линий A0 ... A5;

$val \in \{ 0 \dots 255 \}$  – байтовые значения псевдоаналогового вывода (ШИМ).

$val \in \{ '0', '1' \}$  – бинарное значение для задания режима или состояния линии цифрового ввода-вывода.

Например, команда '0e1' устанавливает цифровой порт 4 ('e') в состояние вывода; команда '3b' дает указание на выполнение аналого-цифрового преобразования сигнала с A1 ('b'), а команда '4fz' выдает через pin 5 ('f') параметр ШИМ 122 ('z' в ASCII равен 122).

Однако использование рассмотренного стандартного протокола обмена MATLAB – Arduino выявило следующие недостатки, которые затрудняют его применение при исследовании и управлении динамичными объектами в режиме реального времени.

1) Непостоянство периода опроса портов в зависимости от загрузки ПЭВМ. В то же время при реализации обратной связи информацию от датчиков о состоянии объекта необходимо получать со строгой привязкой ко времени.

Кроме того, последовательный опрос нескольких сигналов, определяющих состояние многомерной системы, приводит к значительному временному сдвигу получаемых отсчетов ( $\tau_1, \tau_2$  на рис. 4), который также существенно зависит от скорости работы и загрузки компьютера и изменяется во времени (среднее время записи/ считывания с одного порта составляет 15 мс). Это затрудняет оценку состояния быстро меняющихся, динамичных процессов. Подобная проблема была также описана в [8–11]. Практически целесообразно, чтобы информация от всех портов поступала одновременно за одно обращение, либо, при последовательном чтении, соответствовала состояниям объекта в предельно близкие моменты времени.

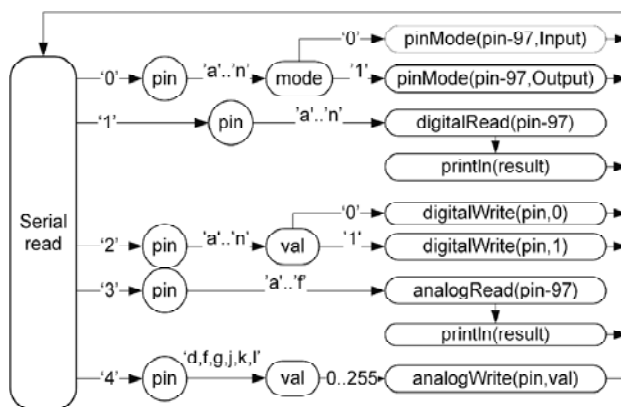


Рисунок 3 – Графическое представление протокола обмена MATLAB – Arduino

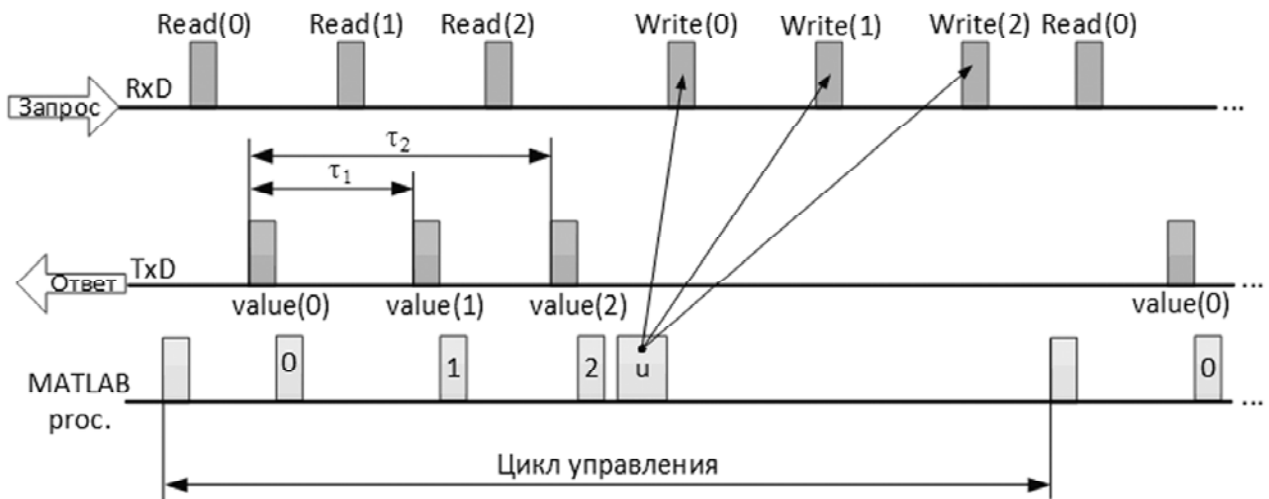


Рисунок 4 – Задержка определения состояния объекта при последовательном опросе портов Arduino

2) При использовании датчиков возникает необходимость предварительной обработки их показаний. Выполнять ее средствами MATLAB нецелесообразно, т.к. в этом случае имеет место жесткая привязка программы MATLAB к аппаратной части системы управления.

## 2 ОБЗОР ЛИТЕРАТУРЫ

Совместное использование MATLAB/Simulink и Arduino при проектировании кибер-физических систем в литературе освещено достаточно широко. Так в [9, 10, 12, 13] приведены примеры систем управления скоростью вращения вала двигателя постоянного тока как сервопривода различных объектов. В работе [14] представлено микропроцессорное устройство контроля параметров трехфазной сети по текущей информации от датчиков тока и напряжения. Устройство реализовано на плате Arduino Galileo. Для анализа точности выполняемых расчетов в среде MATLAB/Simulink разработана компьютерная модель микропроцессорного устройства, учитывающая дискретные преобразования аналоговых входных сигналов токов и напряжений по времени и уровню, измеренных непрерывными датчиками в трехфазной сети.

В [15] описана система управления уровнем жидкости в резервуаре на основе аппарата нечеткой логики. При этом для синтеза нечеткого регулятора использовались возможности MATLAB, а плата Arduino служила для получения данных от датчиков и генерации ШИМ-сигнала для управления насосом.

В [16] предложена система управления движением мобильного робота на основе Arduino, для программирования которой используются возможности MATLAB/Simulink.

Однако приведенные выше, и подавляющее большинство других публикаций посвящены описанию процедуры проектирования и принципов работы соответствующих систем управления и практически не рассматривают процессы, возникающие при взаимодействии Arduino и MATLAB. Результаты экспериментов с платой Arduino для оценки реализуемой ею точности измерения временных интервалов представлены в [11]. Однако в этой работе не предлагается методов повышения эффективности использования Arduino для управления объектами в реальном масштабе времени.

## 3 МАТЕРИАЛЫ И МЕТОДЫ

Рассмотренные выше недостатки стандартного протокола обмена между MATLAB и Arduino снижают эффективность использования гибридного подхода к проектированию систем управления. В связи с этим ниже предлагается модифицированный протокол обмена между MATLAB и Arduino, позволяющий устранить указанные недостатки.

Синхронизацию считываемых с портов данных предлагается осуществлять следующим образом. При получении запроса от MATLAB на чтение данных с первого аналогового порта (рис. 5), Arduino производит опрос всех требуемых аналоговых входов, тем самым формируется выборка данных, которые затем, при поступлении соответствующего запроса (функция '3') передаются в MATLAB. И хотя в Arduino опрос осуществляется также последовательно и занимает определенное время (считывание значения с аналогового входа занимает 14 тактов частоты синхронизации АЦП с периодом 8мкс, тогда  $T_{\text{ацп}} = 148 = 112 \text{ мкс}$ ), это значительно меньше, чем при программном опросе с помощью MATLAB (около 17 мс).

На рис. 5 жирными (красными) стрелками показано прохождение потоков данных между Arduino, MATLAB и в приложениях \*.m, \*.ino.

Функцию предварительной обработки показаний с датчиков предлагается возложить на платформу Arduino, тем более что она имеет значительные вычислительные возможности. Так, после выполнения аналогово-цифрового преобразования данных со всех требуемых аналоговых портов, Arduino производит обработку полученного кода и определяет численные значения контролируемых параметров в соответствующих единицах измерения, а затем передает полученные данные из Arduino на ПЭВМ через последовательный порт. Приведенная процедура позволяет более гибко подходить к выбору технических средств получения информации о состоянии физических параметров системы управления. При этом нет необходимости вносить изменения в программу MATLAB.

С учетом всего сказанного выше в данном параграфе, соответствующий рис. 5 фрагмент модернизированного протокола обмена MATLAB-Arduino отображается рис. 6.

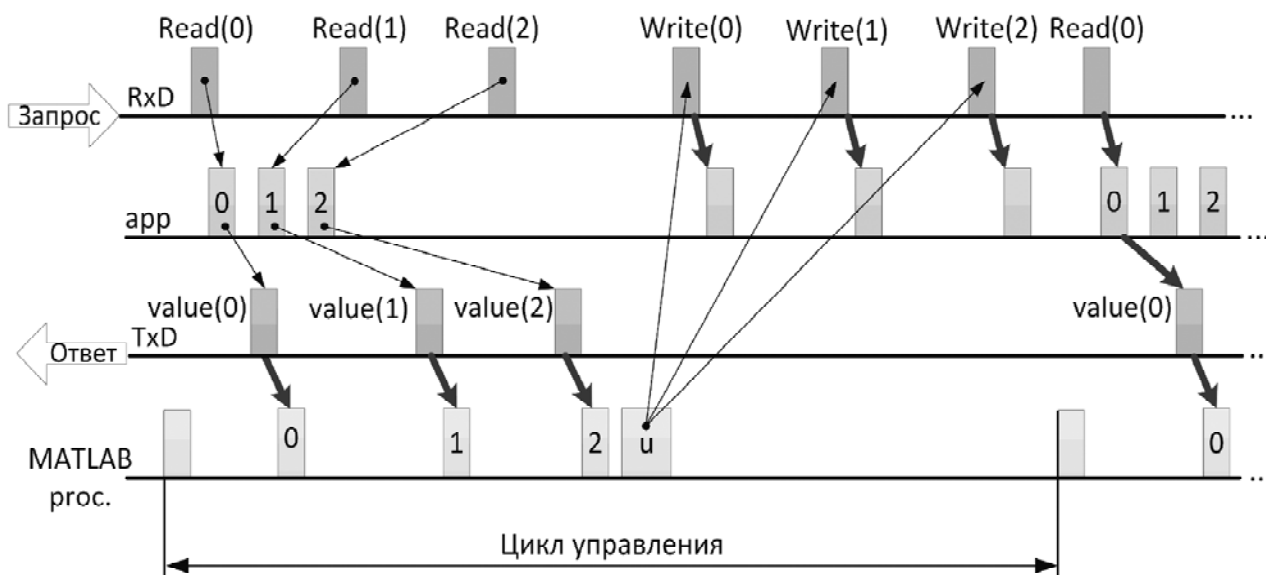


Рисунок 5 – Диаграмма опроса/ответа обмена при синхронизации считывания аналоговых сигналов

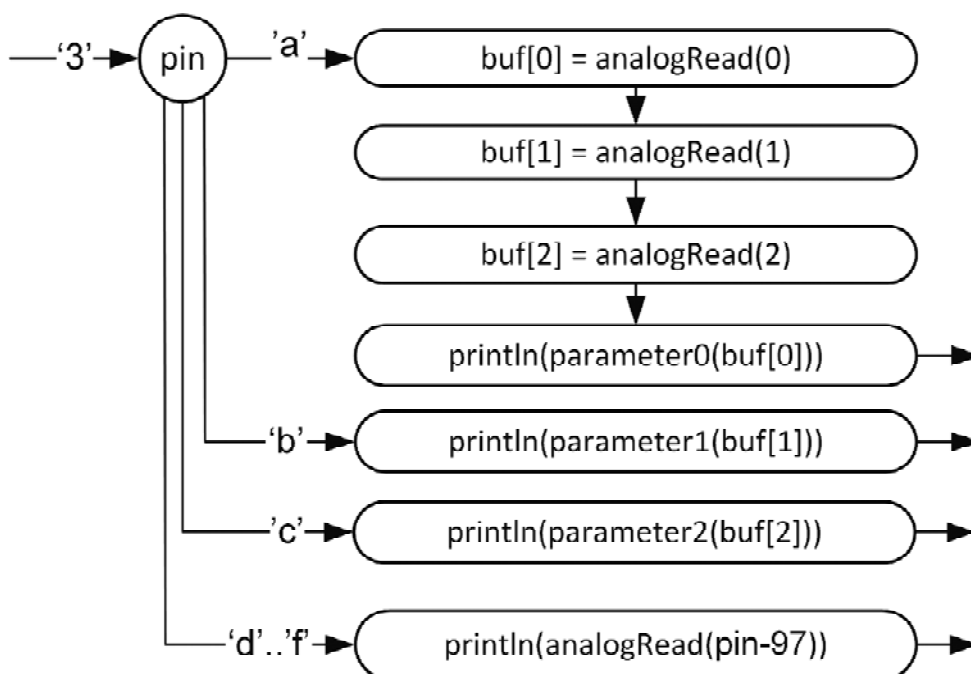


Рисунок 6 – Фрагмент модифицированного протокола обмена MATLAB-Arduino

На рис. 6 функция '3a' запрашивает результат аналого-цифрового преобразования сигнала с входа A0 ('a'), но последовательно выполняется преобразование для A0, A1, A2 и результаты сохраняются в буфере памяти (buf[0..2]) и впоследствии подвергаются обработке с помощью различных, в общем случае, функций `parameter0(buf[0])`, `parameter1(buf[1])`, `parameter2(buf[2])`. Преобразование обработанного кода buf[0] в текстовый формат и отправка по последовательному интерфейсу в ПЭВМ реализует функция `println(parameter0(buf[0]))`. По запросам MATLAB '3b' и '3c' возвращаются результаты преобразования buf[1] и buf[2]. В то же время по запросам '3d'..'3f' выполняется аналого-цифровое преобразование сигналов с входов A3..A5 и возвращаются результаты в текстовом виде – `println(analogRead(pin-97))`.

Например, если 'a' ... 'c' (A0..A2) – входы для датчиков углового положения с разными характеристиками, то при указанной организации протокола в MATLAB возвращаются результаты преобразования напряжение-угол, скрывающие отличия датчиков. Аналогично можно организовать обработку для 1..6 произвольных аналоговых входов.

При необходимости жесткой цикличности опроса портов, например, при считывании информации с датчиков, функции отсчета временных интервалов целесообразно, как и буферизации данных, целесообразно возложить на Arduino. В этом случае MATLAB будет получать поток данных с переменной цикличностью, но массив полученных данных привязан к временной шкале.

#### 4 ЭКСПЕРИМЕНТЫ

Для подтверждения наличия описанных в п.1 недостатков стандартного протокола обмена MATLAB-Arduino, а также проверки эффективности предлагаемой модификации протокола проведен ряд экспериментов на стенде, предназначенном для управления двигателем постоянного тока (ДПТ) посредством MATLAB и Arduino. Стенд (рис. 7) состоит из двигателя постоянного тока DC напряжением питания 5..12 В, ПЭВМ с MATLAB 2011b, Arduino Uno (MCU) с микроконтроллером Atmega328 с 32 кБ флеш-памяти (Flash), 1кБ ЭСППЗУ (EEP ROM) и 2 кБ SRAM, электронного ключа (ES), датчика скорости вращения двигателя (S). В качестве ПЭВМ использовался ноутбук с процессором Intel Core i3-330M 2,13 ГГц, ОЗУ 3ГБ и операционной системой Windows 7 Professional x64.

Задание желаемой скорости  $V_3$  вращения осуществляется напряжением  $U(V_3) = 0..5В$  с выхода потенциометра, а в качестве датчика скорости используется датчик Холла, частота импульсов  $f(V)$  на выходе которого определяется скоростью  $V$  вращения электродвигателя с 12-полюсным магнитом на валу. Преобразователь частота-напряжение ( $f/U$ ) формирует напряжение  $U(V)$ , пропорциональное скорости вращения двигателя. Управление двигателем производится посредством ШИМ-сигнала (PWM), параметры которого определяются на основе ПИД закона (1):

$$u_k = (u_{k-1} + K_p(e_k - e_{k-1}) + \frac{K_p T_s}{T_i} e_k + \frac{K_p T_d}{T_s} (e_k - 2e_{k-1} + e_{k-2})) \cdot \frac{255}{1023}. \quad (1)$$

При расчете параметров ПИД-регулятора использовались встроенные возможности MATLAB.

Взаимодействие MATLAB с Arduino осуществлялось на основе приведенного в п. 3 модифицированного протокола обмена.

#### 5 РЕЗУЛЬТАТЫ

Рис. 8 отображает осциллограммы обмена между MATLAB и Arduino UNO посредством стандартного протокола. Период обмена включает: посылку MATLAB команды чтения аналогового входа A0 на вход последовательного обмена Arduino (RxD); прием значения (value), передаваемого по линии TxD Arduino; посылку команды аналогового вывода на pin 9, на которую ответ не передается. На рис. 8 видно изменение периода опроса D, зависящее от загрузки компьютера. Горизонтальная развертка составляет 2,5 мс/деление, вертикальная – 5В/деление, т.е. период  $T_1$  составляет 12,5 мс, а период  $T_2$  – 10 мс.

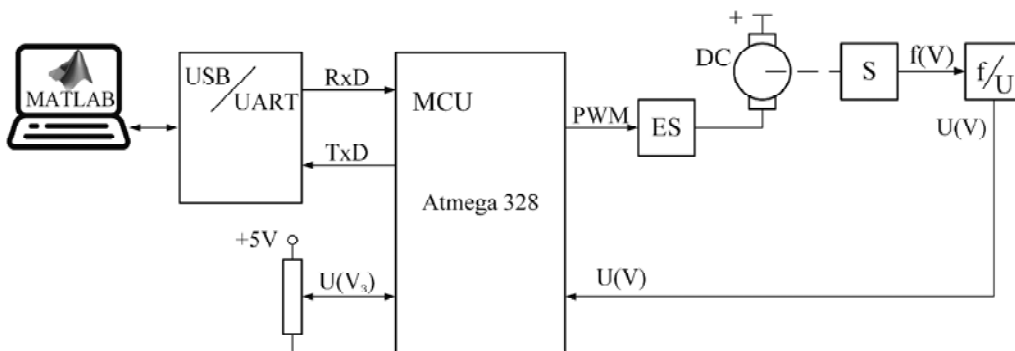


Рисунок 7 – Структурная схема стенда

Результаты серии экспериментов с обращением к аналоговым портам Arduino через MATLAB свидетельствуют о том, что основной вклад в вариацию периодов обмена вносит период чтения  $T_R$  (рис. 8), который, при обращении к одному порту, изменяется случайным образом в диапазоне  $T_R = 8 \div 14$ мс. В то же время, период записи  $T_W$  изменяется не так заметно и в среднем составляет около 3 мс. С увеличением количества задействованных портов период их опроса увеличивается практически пропорционально.

Динамику системы управления двигателем при действии внешней нагрузки отображает рис. 9, где  $U_x = U(V_3)/5 \cdot 1023$  – результат преобразования напряжения  $U(V_3)$ , снимаемого с переменного резистора, в код 0..1023;  $U_v = U(V)/5 \cdot 1023$  – результат преобразования напряжения  $U(V)$ , снимаемого с датчика скорости в код 0..1023; pwm – параметр ШИМ в диапазоне 0..255; int – выход интегрирующего канала ПИД-регулятора.

#### 6 ОБСУЖДЕНИЕ

Результаты экспериментальных исследований подтверждают преимущества предложенного в данной работе протокола. Реализация процедуры синхронизации считываемых с портов данных позволяет на два порядка (с 15 ÷ 17 мс до 112 мкс) уменьшить время работы с аналоговыми портами Arduino. Это, а также уменьшение зависимости системы управления от непостоянства периода опроса портов Arduino, приводит к уменьшению неопределенности относительно области возможных состояний объекта. Очевидно, что при управлении инерционными объектами полученный результат не окажет заметного влияния на качество управления, однако при управлении быстро меняющимися, динамичными процессами, использование предложенного протокола дает ощутимый эффект.

#### ВЫВОДЫ

Распространению кибер-физических систем управления способствует применение современных методов модельно-ориентированного проектирования. При этом важную роль играет использование недорогих, относительно простых в использовании и в то же время эффективных периферийных устройств, обеспечивающих коммуникацию между моделью объекта или процесса управления и физическим миром. Одной из таких устройств является платформа Arduino.

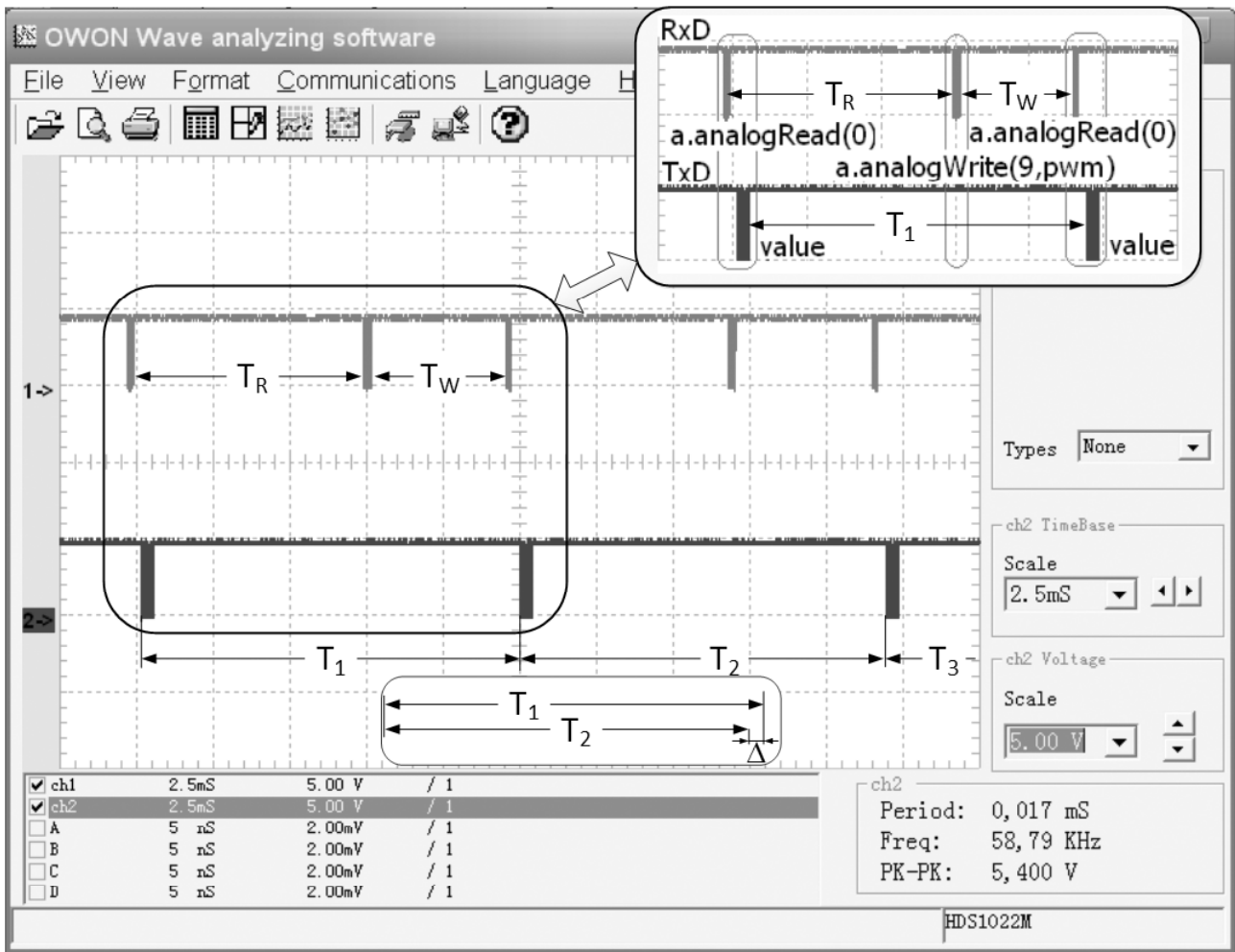


Рисунок 8 – Осциллограммы обмена между MATLAB и Arduino

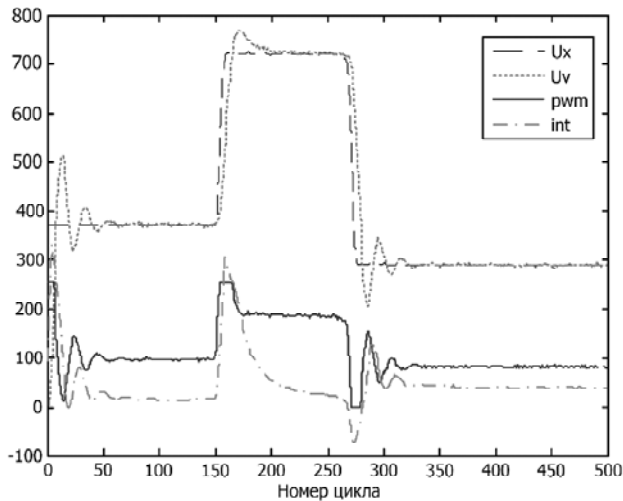


Рисунок 9 – Поведение системы управления двигателем

Однако использование стандартного протокола Arduino для обмена данными между ПЭВМ и физическим миром имеет определенные недостатки, связанные с непостоянством периода времени между получением выборки данных, невозможностью параллельного получения данных о состоянии объекта, а также нерациональным использованием ресурсов ПЭВМ.

Предложенный в данной работе протокол обмена данными между MATLAB и Arduino позволяет оптимизировать процесс взаимодействия системы управления с объектом и повысить качество управления динамичными многомерными процессами за счет уменьшения запаздывания в определении состояния многомерных объектов при последовательном обращении к аналоговым портам Arduino.

Выполнение предварительной обработки показаний датчиков средствами Arduino не влияет непосредственно на качество управления, однако повышает гибкость CPS за счет возможности изменения аппаратного обеспечения без изменения программного кода MATLAB.

Рассмотренный протокол может быть модифицирован для применения с другими средами моделирования, например, LabVIEW, а также реализован на основе иных платформ (Raspberry и проч.) и микроконтроллерах известных семейств – PIC, MSP430, MCS51, ARM, MIPS и т.д. Последовательный обмен может быть построен с использованием беспроводных Bluetooth (HC05 и др.), WiFi (ESP8266 и др), XBee и прочих модемов. Это расширяет область применения предлагаемых подходов.

#### СПИСОК ЛИТЕРАТУРЫ

1. Sangiovanni-Vincentelli A. Taming Dr. Frankenstein: Contract-Based Design for Cyber-Physical Systems / A. Sangiovanni-

- Vincentelli, W. Damm, R. Passerone // *European Journal of Control.* – 2012. – Vol. 18, № 3. – P. 217–238. DOI:10.3166/ejc.18.217-238.
2. Черняк Л. Киберфизические системы на старте / Л. Черняк // *Открытые системы. СУБД.* – 2014. – № 02 (198) – С. 10–13.
  3. Park Kyung-Joon Cyber-physical systems: Milestones and research challenges / Kyung-Joon Park, Rong Zheng, Xue Liu // *Computer Communications* – 2012. – № 36. – P. 1–7. DOI 10.1016/j.comcom.2012.09.006.
  4. Introduction to Embedded Systems – A Cyber-Physical Systems Approach / E. A. Lee and S. A. Seshia. Second Edition. [Electronic resource]. – Access mode: <http://LeeSeshia.org>, 2015.
  5. Jensen J. C. A model-based design methodology for cyber-physical systems / J. C. Jensen, D. H. Chang, E. A. Lee // *Wireless Communications and Mobile Computing Conference (IWCMC): 7th International conference, Istanbul, 4–8 July 2011: proceedings.* – IEEE, 2011. – P. 1666–1671. DOI: 10.1109/IWCMC.2011.5982785.
  6. Модельно-ориентированное проектирование программного обеспечения для встраиваемых систем в среде Matlab/Simulink / [Г. К. Топораш, А. В. Мазур, Д. А. Ковальчук, А. А. Пушкин] // *Автоматизація технологічних і бізнес-процесів.* – 2014. – Т. 17, № 17. – С. 26–29.
  7. Практические стратегии для перехода на модельно-ориентированное проектирование встроенных приложений / [Д. Венси, Д. Эрик, К. Ларри, Р. Винод] // *Компоненты и технологии.* – 2011. – Т. 10, №. 123. – С. 172–180.
  8. Arduino Support from MATLAB [Electronic resource]. – Access mode: <http://www.mathworks.com/hardware-support/arduino-matlab.html>.
  9. Al-Busaidi A. M. Development of an educational environment for online control of a biped robot using MATLAB and Arduino / A. M. Al-Busaidi // *Mechatronics: 9th France-Japan & 7th Europe-Asia Congress on and Research and Education in Mechatronics (REM), 21–23 Nov. 2012, 13th Int'l Workshop on.* – IEEE, 2012. – P. 337–344. DOI: 10.1109/MECATRONICS.2012.6451030.
  10. Barber R. Control Practices using Simulink with Arduino as Low Cost Hardware / R. Barber, M. Horra, J. Crespo. // *The 10th IFAC Symposium Advances in Control Education, August 28-30, 2013: proceedings.* – University of Sheffield, Sheffield, UK, 2013. – P. 250–255. DOI: 10.3182/20130828-3-UK-2039.00061.
  11. D'Ausilio A. Arduino: A low-cost multipurpose lab equipment / Alessandro D'Ausilio // *Behavior Research Methods.* – 2012. – Vol. 44, Issue 2. – P. 305–313. DOI 10.3758/s13428-011-0163-z
  12. Online Performance Optimization of a DC Motor Driving a Variable Pitch Propeller / [R. Cohen, D. Miculescu, K. Reilley and other] // *arXiv preprint arXiv:1310.0133.* – 2013.
  13. Bawa D. Fuzzy control based solar tracker using Arduino Uno / D. Bawa, C.Y. Patil // *International Journal of Engineering and Innovative Technology (IJEIT).* – 2013. – Vol. 2, Issue 12. – P. 179–187.
  14. Омельченко Е. Я. Устройство контроля трехфазной сети на основе Arduino-совместимых микроконтроллеров / Е. Я. Омельченко, А. В. Белый и др. // *Электротехника.* – 2014. – Т. 1, № 1. – С. 17–22.
  15. Krivić S. Design and implementation of fuzzy controller on embedded computer for water level control / S. Krivić, M. Hujdur, A. Mrzić, S. Konjicija // *MIPRO 2012: 35th International Convention, Opatija, Croatia, 21–25 May 2012: proceedings.* – IEEE, 2012. – P. 1747–1751.
  16. Gartseev I. B. A low-cost real-time mobile robot platform (ArEduBot) to support project-based learning in robotics & mechatronics / I. B. Gartseev, L. F. Lee, V. N. Krovi // *2nd International Conference on Robotics in Education (RiE 2011), Vienna, Austria, September 15–16, 2011: proceedings* [Electronic resource]. – Access mode: [http://mechatronics.eng.buffalo.edu/publications/conference/ICRE2011\\_IlyaLeeKrovi\\_ArEduBot.pdf](http://mechatronics.eng.buffalo.edu/publications/conference/ICRE2011_IlyaLeeKrovi_ArEduBot.pdf)

Статья поступила в редакцию 28.10.2015.  
После доработки 09.11.2015.

Гурко О. Г.<sup>1</sup>, Плахтеев А. П.<sup>2</sup>, Плахтеев П. А.<sup>3</sup>

<sup>1</sup>Канд. техн. наук, доцент, доцент кафедры автоматизації та комп'ютерно-інтегрованих технологій Харківського національного автомобільно-дорожнього університету, Харків, Україна

<sup>2</sup>Канд. техн. наук, доцент, доцент кафедри автоматизації та комп'ютерно-інтегрованих технологій Харківського національного автомобільно-дорожнього університету, Харків, Україна

<sup>3</sup>Інженер, молодший науковий співробітник кафедри автоматизації та комп'ютерно-інтегрованих технологій Харківського національного автомобільно-дорожнього університету, Харків, Україна

#### ПІДВИЩЕННЯ ТОЧНОСТІ ОЦІНКИ СТАНУ ДИНАМІЧНИХ ОБ'ЄКТІВ КОМПЛЕКСОМ MATLAB-ARDUINO ПРИ ПРОЕКТУВАННІ КІБЕР-ФІЗИЧНИХ СИСТЕМ

Розв'язано задачу підвищення ефективності взаємодії MATLAB і Arduino при проектуванні кібер-фізичних систем шляхом внесення змін у реалізацію стандартного протоколу обміну з боку Arduino. Запропоновано при запиті MATLAB на читання даних з першого аналогового порту Arduino виконувати аналогово-цифрове перетворення даних з усіх необхідних аналогових портів з подальшою послідовною передачею отриманих даних в MATLAB, що дозволяє підвищити якість управління динамічними процесами за рахунок зменшення області невизначеності стану багатовимірної швидкодіючої системи управління. Крім того, запропоновано функції попередньої обробки показань датчиків виконувати засобами Arduino, що підвищує гнучкість кібер-фізичної системи за рахунок можливості зміни апаратного забезпечення без зміни програмного коду і протоколу обміну MATLAB. Формальний опис взаємодії MATLAB і Arduino дозволяє реалізувати протокол обміну з використанням бездротових інтерфейсів, мікропроцесорних пристроїв та платформ, не сумісних з Arduino.

**Ключові слова:** кібер-фізична система, MATLAB, Arduino, протокол обміну, період опитування.

Gurko A. G.<sup>1</sup>, Plakhteev A. P.<sup>2</sup>, Plakhteev P. A.<sup>3</sup>

<sup>1</sup>PhD, Associate professor, Associate professor of department of automation and computer-integrated technologies, Kharkiv National Automobile and Highway University, Kharkiv, Ukraine

<sup>2</sup>PhD, Associate professor, Associate professor of department of automation and computer-integrated technologies, Kharkiv National Automobile and Highway University, Kharkiv, Ukraine

<sup>3</sup>Engineer, Junior researcher of department of automation and computer-integrated technologies, Kharkiv National Automobile and Highway University, Kharkiv, Ukraine

#### ACCURACY INCREASE OF DYNAMIC OBJECTS STATE ESTIMATION BY A COMPLEX MATLAB-ARDUINO WHEN CYBER-PHYSICAL SYSTEMS DESIGNING

The problem of accuracy increase of MATLAB and Arduino interaction when cyber-physical systems designing by making changes into the implementation of the standard protocol of Arduino has been solved. It has been proposed to carry out the analog-digital conversion of all required analog pins with further sequential data transmission to MATLAB when requesting to read data from the first analog pin of Arduino.

It allows improving control quality of dynamic processes by reducing the uncertainty range of a high-speed multi-dimensional control system state. In addition, it has been offered to perform the preprocessing of sensors data by means of Arduino that increases the flexibility of cyber-physical systems due to the possibility of the hardware changing without MATLAB software and protocol changing. A formal description of MATLAB and Arduino interaction allows realizing the communications protocol using wireless interfaces as well as microprocessor units and platforms, which are not compatible with Arduino.

**Keywords:** cyber-physical system, MATLAB, Arduino, communications protocol, polling period.

## REFERENCES

1. Sangiovanni-Vincentelli A., Damm W., Passerone R. Taming Dr. Frankenstein: Contract-Based Design for Cyber-Physical Systems, *European Journal of Control*, 2012, Vol. 18, No. 3, pp. 217–238. DOI:10.3166/ejc.18.217-238.
2. Chernjak L. Kiberfizicheskie sistemy na starte, *Otkrytye sistemy. SUBD*, 2014, No. 02 (198), pp. 10–13.
3. Park, Kyung-Joon, Rong Zheng, Xue Liu Cyber-physical systems: Milestones and research challenges, *Computer Communications*, 2012, No. 36, pp. 1–7. DOI 10.1016/j.comcom.2012.09.006.
4. Lee E. A. and Seshia S. A. Introduction to Embedded Systems – A Cyber-Physical Systems Approach. Second Edition. [Electronic resource]. Access mode: <http://LeeSeshia.org>, 2015.
5. Jensen J. C., Chang D. H., Lee E. A. A model-based design methodology for cyber-physical systems, *Wireless Communications and Mobile Computing Conference (IWCMC): 7th International conference, Istanbul, 4–8 July 2011: proceedings. IEEE*, 2011, P. 1666–1671. DOI: 10.1109/IWCMC.2011.5982785.
6. Toporash G. K., Mazur A. V., Koval'chuk D. A., Pushkin A. A. Model'no-orientirovanoe proektirovanie programnogo obespechenija dlja vstraivaemyh sistem v srede Matlab/Simulink, *Avtomatizacija tehnologichnih i biznes-procesiv*, 2014, Vol. 17, No. 17, pp. 26–29.
7. Vensi D., Jerik D., Larri K., Vinod R. Prakticheskie strategii dlja perehoda na model'no-orientirovanoe proektirovanie vstroennyh prilozhenij, *Komponenty i tehnologii*, 2011, Vol. 10, No. 123, pp. 172–180.
8. Arduino Support from MATLAB [Electronic resource]. Access mode: <http://www.mathworks.com/hardware-support/arduino-matlab.html>.
9. Al-Busaidi A. M. Development of an educational environment for online control of a biped robot using MATLAB and Arduino, *Mechatronics: 9th France-Japan & 7th Europe-Asia Congress on and Research and Education in Mechatronics (REM), 21–23 Nov. 2012, 13th Int'l Workshop on. – IEEE, 2012*, pp. 337–344. DOI: 10.1109/MECATRONICS.2012.6451030.
10. Barber R., Horra M., Crespo J. Control Practices using Simulink with Arduino as Low Cost Hardware, *The 10th IEAC Symposium Advances in Control Education, August 28–30, 2013: proceedings. – University of Sheffield, Sheffield, UK, 2013*, pp. 250–255. DOI: 10.3182/20130828-3-UK-2039.00061.
11. D'Ausilio A. Arduino: A low-cost multipurpose lab equipment, *Behavior Research Methods*, 2012, Vol. 44, Issue 2, pp. 305–313. DOI 10.3758/s13428-011-0163-z
12. Cohen R., Miculescu D., Reilley K., Pakmehr M., Feron E. Online Performance Optimization of a DC Motor Driving a Variable Pitch Propeller, *arXiv preprint arXiv:1310.0133*, 2013.
13. Bawa D., Patil C. Y. Fuzzy control based solar tracker using Arduino Uno, *International Journal of Engineering and Innovative Technology (IJEIT)*, 2013, Vol. 2, Issue 12, pp. 179–187.
14. Omel'chenko E. Ja., Belyj A. V. i dr. Ustrojstvo kontrolja trehfaznoj seti na osnove Arduino-sovmestimyh mikrokontrollerov, *Jelektrotehnika*, 2014, Vol. 1, No. 1, pp. 17–22.
15. Krivić S., Hujdur M., Mrzić A., Konjicija S. Design and implementation of fuzzy controller on embedded computer for water level control, *MIPRO 2012: 35th International Convention, Opatija, Croatia, 21–25 May 2012: proceedings. IEEE, 2012*, pp. 1747–1751.
16. Gartsev I. B., Lee L. F., Krovi V. N. A low-cost real-time mobile robot platform (ArEduBot) to support project-based learning in robotics & mechatronics, *2nd International Conference on Robotics in Education (RiE 2011), Vienna, Austria, September 15–16, 2011: proceedings [Electronic resource]*. Access mode: [http://mechatronics.eng.buffalo.edu/publications/conference/ICRE2011\\_IlyaLeeKrovi\\_ArEduBot.pdf](http://mechatronics.eng.buffalo.edu/publications/conference/ICRE2011_IlyaLeeKrovi_ArEduBot.pdf)