

ПРАКТИКА ИСПОЛЬЗОВАНИЯ ДИНАМИЧЕСКИХ БИБЛИОТЕК АЦП L-CARD E-502 ДЛЯ АВТОМАТИЗАЦИИ ЭКСПЕРИМЕНТОВ В СРЕДЕ MATLAB

*Мацак И.С., к.т.н., в.н.с. ПАО «Ракетно-космическая корпорация «Энергия» им. С.П. Королёва»,
e-mail: ismatsak@mail.ru;*

*Капранов В.В., инженер ПАО «Ракетно-космическая корпорация «Энергия» им. С.П. Королёва»
Воропаев Р.А., инженер ПАО «Ракетно-космическая корпорация «Энергия» им. С.П. Королёва»*

THE PRACTICE OF USING OF ADC L-CARD E-502 DYNAMIC LIBRARIES FOR AUTOMATED TESTING IN MATLAB

Matsak I.S., Kapranov V.V., Voropaev R.A.

The solution of the problem of integration ADC L-card E-502 to MATLAB environment using dynamic libraries. Matlab script performing the cycle data acquisition from analog channels of ADC and source of Matlab subfunctions used in script are provided.

Key words: automated testing, L-card E-502, dynamic libraries, MATLAB, C-shared libraries, loadlibrary.

Ключевые слова: automated testing, L-card E-502, dynamic libraries, MATLAB, C-shared libraries, loadlibrary.

Введение

Настоящая работа посвящена вопросам интеграции динамических библиотек АЦП E-502 в Matlab. Представленные результаты позволяют расширить возможности построения измерительных систем на основе связки Matlab / АЦП E-502 и могут быть использованы в прикладных и исследовательских работах.

Программная среда Matlab имеет широкие возможности для взаимодействия с периферийным оборудованием [1], которые могут быть использованы для автоматизации научных экспериментов и создания измерительных систем для решения ряда прикладных задач. Взаимодействие может быть реализовано напрямую с помощью универсальной библиотеки для измерительного оборудования VISA (Virtual Instrument Software Architecture) или в совокупности с инструментальными драйверами IVI (Interchangeable Virtual Instrument) [2]. Расширение возможностей получения данных и управления периферией может быть достигнуто при помощи средств Instrument Control Toolbox [3]. Однако, перечень производителей измерительного оборудования, поддерживающего такой способ взаимодействия, ограничен, и этот подход не может в полной мере считаться универсальным. В частности, построение измерительной системы на основе АЦП E-502 производства «Л Кард» [4] с обработкой данных в среде Matlab не может быть реализовано с помощью описанных выше способов. Поэтому в ряде работ отдается предпочтение написанию кода на Python, например [5]. Хотя для некоторых моделей крейтов «Л-Кард» (например, LTR-EU-2-2) возможна настройка подключения с применением технологии OPC (Open Platform Communications) [6, 7], основной возможностью интеграции в измерительную систему на основе Matlab АЦП L-card E-502 является подключение Matlab к динамическим библиотекам, поставляемым

Представлено решение задачи по интеграции АЦП L-card E-502 в среду MATLAB с помощью подключения динамических библиотек. Приведены скрипт Matlab, осуществляющий циклический сбор данных с аналоговых каналов АЦП, и исходные коды подфункций, используемых в скрипте.

в комплекте с оборудованием [8]. Несмотря на некоторую сложность такого подхода, он может быть успешно реализован для выбранной модели АЦП и применен на практике.

При выборе Matlab в качестве среды для обработки и анализа данных отсутствие интеграции с измерительным оборудованием вынуждает проводить операции с измерительной информацией в несколько этапов, использовать набор программных средств разных производителей, экспортировать и импортировать данные, обеспечивать совместимость формата данных, что значительно усложняет проведение экспериментальных исследований. В качестве примеров научных исследований прикладного характера, где такой подход был использован, могут служить работы [9, 10].

Создание универсального скрипта для работы в среде MATLAB АЦП E-502 может быть интересно широкому кругу исследователей, занимающихся экспериментальной деятельностью и автоматизацией экспериментов. На основе рассматриваемого скрипта могут быть написаны функции и программы, выполняющий циклический сбор данных, а описанные принципы расширить функциональность программ до необходимого уровня, и задействовать цифровые каналы, а также выходы ЦАП.

Настройка Matlab для взаимодействия с АЦП

Работа с динамическими библиотеками в Matlab требует установки драйвера устройства, самих библиотек и компилятора. Для установки библиотек e502api и x502api и драйвера АЦП E-502 был использован стандартный установщик от производителя АЦП [11]. Установка MinGW compiler осуществлялась с помощью инструмента «Add-Ons» на вкладке «Home» интерфейса Matlab [12]. При

Таблица 1. Пример заполнения файла E502_CONFIG.xlsx для конфигурации аналоговых каналов АЦП E-502

Логический канал	Физический канал	Режим измерения	Диапазон измерения	Усреднение	Использование
0	0	1	5	36	1
1	1	1	0	36	1
2	2	2	1	36	1
3	3	2	2	36	1
4	4	0	3	36	1

использовании версии Matlab 2017a и ранее следует обратиться к видеопроцедуре [13], и перед установкой MinGW скопировать папки bin и bugreport, расположенные в прикрепленном архиве [14], в директорию установки Matlab.

Алгоритм сбора данных с аналоговых каналов АЦП

Типичный алгоритм для организации синхронного получения данных с выбранных аналоговых каналов АЦП согласно [15] имеет основные следующие шаги:

- X502_StreamsEnable() – разрешение нужных синхронных потоков;
 - X502_StreamsStart() – запуск синхронных потоков;
 - X502_Recv() – чтение принятых данных из модуля;
 - X502_ProcessAdcData() – обработка прочитанных данных;
 - X502_StreamsStop() – останов синхронных потоков.
- Реализация этого алгоритма в среде Matlab рассмотрена авторами ниже.

Конфигурация каналов

Конфигурация каналов АЦП осуществляется путем задания значений параметров канала в отдельном файле E502_CONFIG.xlsx, который помещается в директорию с основным скриптом циклического сбора данных E502_CYCLE_IN. В данной работе эта директория задана как C:\MATLAB. Пример конфигурации приведен в табл. 1. Задание логических и физических каналов, диапазонов и режимов измерения, параметров усреднения выполняется в соответствии с [15, с. 31, с. 34, с.53].

Реализация модулей скрипта

Для получения данных с АЦП в Matlab необходимо прописать 12 основных подфункций и скриптов, которым будет обращаться основной скрипт внутри себя. При этом загрузка библиотек с помощью скрипта load502api может быть проведена еще до запуска основного скрипта, и, в таком случае, исключена из него.

Подключение и вызов функций динамических библиотек производится с помощью набора функций и методов Matlab [16]: loadlibrary, calllib и libpointer. Дополнительные функции – libfunctions и libfunctionsview – позволяют корректно задавать переменные функций библиотек. Для правильной работы скрипта должны быть прописаны следующие основные функции:

- загрузка библиотек:

```
function load502api
    %Добавление в перечень директорию с
    программами и скриптами Matlab
    addpath('C:\MATLAB');
    %Добавление в перечень директорию с
    библиотеками e502api и x502api;
    addpath('C:\Program Files (x86)\L-
    Card\lpcie\include');
    %Отключение сообщений о потенциаль-
    ных конфликтах
    warning('off','all');
    %Загрузка библиотек e502api и x502api
    if not(libisloaded('x502api'))
        loadlibrary('x502api')
    end
    if not(libisloaded('e502api'))
        loadlibrary('e502api')
    end

    %Проверка успешности загрузки библиотек
    может осуществляться с помощью функции
    libisloaded.
```

- открытие порта:

```
function E502_OpenUsb
    err =
    calllib('e502api','E502_OpenUsb', hnd,
    card_serial);
    if err ~= 0
        s =
    calllib('x502api','X502_GetErrorString',
    err);
        fprintf('Ошибка %1$d функции
    E502_OpenUsb: %2$s \n', err, s)
    end
end

%Dля контроля корректности работы под-
    функций каждые скрипт и функция снабжа-
    ются стандартным блоком сообщения об
    ошибках, общий вид которого:

    if err ~= 0 % или в некоторых случаях
    err < 0
        s =
    calllib('x502api','X502_GetErrorString',
    err);
        fprintf('Ошибка %1$d функции
    (Имя функции) : %2$s \n', err, s)
    end

    %В следующих скриптах эти блоки будут
    опущены ввиду своей однотипности, одна-
    ко в исходном алгоритме [17] они сохра-
    нены.
```

- закрытие порта:

```
function X502_Close
    err = calllib('x502api',
```

```
'X502_Close', hnd);
end
```

– освобождение описателя:

```
function X502_Free
    err = calllib('x502api', 'X502_Free',
hnd);
end
```

– конфигурация АЦП:

```
function X502_Configure
    err =
calllib('x502api', 'X502_Configure', hnd,
0);
end
```

– задание количества каналов:

```
function X502_SetLChannelCount
    err =
calllib('x502api', 'X502_SetLChannelCount'
, hnd, nk);
end
```

– разрешение потока:

```
function X502_StreamsEnable
    err =
calllib('x502api', 'X502_StreamsEnable',
hnd, 1);
end
```

– запуск потока:

```
function X502_StreamsStart
    err =
calllib('x502api', 'X502_StreamsStart',
hnd);
end
```

– останов потока:

```
function X502_StreamsStop
    err =
calllib('x502api', 'X502_StreamsStop',
hnd);
end
```

– обработка полученных данных:

```
function X502_ProcessAdcData
    err =
calllib('x502api', 'X502_ProcessAdcData',
hnd, rcv_buf, adc_data, rcv_size, 1);
end
```

– получение данных обработанных каналов:

```
function X502_GetRecvReadyCount
    a = 1;
    rdy_cnt = libpointer('uint32Ptr', a);
    ots =
calllib('x502api', 'X502_GetRecvReadyCount'
, hnd, rdy_cnt);
    rdy_cnt.Value;
end
```

– конфигурация каналов:

```
function SetLChannel(lch, phy_ch, mode,
range, avg)
    err =
calllib('x502api', 'X502_SetLChannel',
hnd, lch, phy_ch, mode, range, avg);
end
```

Скрипт циклического опроса аналоговых каналов АЦП

Скрипт E502_CYCLE_IN непрерывно выводит значения в вольтах с АЦП, пока не будет нажата клавиша 'q' на клавиатуре [17]. Конфигурация каналов АЦП производится с помощью таблицы E502_CONFIG.xlsx. Каждый выводимый скриптом столбец соответствует значению напряжения на логическом канале АЦП в порядке возрастания номеров.

```
function E502_CYCLE_IN
% Определение директорий, загрузка библиотек, указание серийного номера АЦП
load502api;
% Создание описателя модуля
hnd = calllib('x502api', 'X502_Create');
if class(hnd) ~= 'lib.pointer'
fprintf('Ошибка создания описателя модуля')
end
% Открытие порта
E502_OpenUsb;
% Считывание таблицы с установками каналов:
M = readmatrix('E502_CONFIG.xlsx', 'Range', 'B3:G18');
% Выделение каналов, отмеченных как используемые
R=M(M(:,6)>0,1:5);
% Определение количество используемых каналов:
nk = height(R);
% Задание количества этого числа каналов АЦП:
X502_SetLChannelCount;
% Конфигурация АЦП в соответствии с таблицей E502_CONFIG.xlsx
for ii = 1:nk
    SetLChannel(R(ii,1), R(ii,2), R(ii,3), R(ii,4), R(ii,5));
end
%Задание частоты АЦП, частоты кадров и количества значений в блоке
Fadc = 100; Faf = 5; nb = 1;
%Расчет таймаута приема блока (мс)
READ_TIMEOUT =1000/Faf*nb;
%Установка частот АЦП с помощью указателей
Fadc = libpointer('doublePtr',Fadc); Faf = libpointer('doublePtr',Faf);
calllib('x502api', 'X502_SetAdcFreq', hnd, Fadc, Faf);
%Конфигурация АЦП
X502_Configure;
%Определение количества отсчетов за блок и создание
READ_BLOCK_SIZE
=nk*int32(Faf.Value*READ_TIMEOUT/1000);
buf = ones(1,READ_BLOCK_SIZE);
rcv_buf = libpointer('uint32Ptr',buf);
data = ones(1,READ_BLOCK_SIZE);
adc_data = libpointer('doublePtr',data);
%Разрешение потока от АЦП
X502_StreamsEnable;
%Запуск потока от АЦП
X502_StreamsStart;
% Создание бесконечного цикла считывание, прерываемого нажатием клавиши 'q', который был сделан на основе [17]
hf=figure('position',[0 0 eps eps], 'menubar', 'none');
while 1
```

```

%Получение данных (по таймауту)
rcv_size =
calllib('x502api','X502_Recv', hnd, rcv_buf,
READ_BLOCK_SIZE, READ_TIMEOUT);
rcv_buf.Value;
% Обработка сырые данные от аналоговых
входов и пересчитываем в вольты
X502_ProcessAdcData;
%ВЫВОД ДАННЫХ
A = adc_data.Value;
% Разбираем массив данных (каждый
столбец - данные с одного канала)
% Очередность логических каналов согл-
сано таблице от 0 до последнего
B = reshape(A, [nk,
READ_BLOCK_SIZE/nk] );
% Отображение данных
disp(B)
% Условие выхода из цикла (нажатие
'q')
if
strcmp(get(hf, 'currentcharacter'), 'q')
close(hf)
break
end
figure(hf)
drawnow
end
%=====
%Останов потока данных
X502_StreamsStop;
%Закрытие порта
X502_Close;
% Освобождение указателя
X502_Free;
% =====
% Подфункции
function load502api
function E502_OpenUsb
function X502_Close
function X502_Free
function X502_Configure
function X502_SetLChannelCount
function X502_StreamsEnable
function X502_StreamsStart
function X502_StreamsStop
function X502_ProcessAdcData
function X502_GetRecvReadyCount
function SetLChannel(lch, phy_ch, mode,
range, avg)
end

```

Заключение

В работе предложен способ подключения к АЦП L-card E-502 к среде расчетов MATLAB и подробно рассмотрена техническая реализация этого способа, что упрощает создание информационно-измерительных систем и обеспечивает возможность автоматизации физических экспериментов. Решение позволяет получать и обрабатывать данные с одной среде, в полной мере используя преимущества и простоту языка программирования Matlab для ускорения проведения исследовательских и прикладных работ. Исходный код функции на языке Matlab E502_CYCLE_IN.m и файл EXCEL для конфигурации АЦП E502_CONFIG.xlsx рас-

положены на файлообменнике сайта Mathworks [18].

Литература

1. Взаимодействие MATLAB с оборудованием. URL: <https://exponenta.ru/news/vzaimodejstvie-matlab-s-oborudovaniem>.
2. Взаимодействие MATLAB с оборудованием: 07. Измерительное оборудование. URL: <https://exponenta.ru/news/video-vzaimodejstvie-matlab-s-oborudovaniem-7-izmeritelnoe-oborudovanie>.
3. Instrument Control Toolbox. URL: <https://exponenta.ru/instrument-control-toolbox> (дата обращения 19.05.2021)
4. E-502. Многофункциональный модуль АЦП / ЦАП E-502 с интерфейсами USB&Ethernet. URL: www.lcard.ru/products/external/e-502 (дата обращения 19.05.2021)
5. Верзунов С.Н. Разработка программного компонента трассоискателя на базе устройства сбора данных Л Кард E502 / С.Н. Верзунов, И.В. Бочкарев. Электронные системы и компоненты. 2018. №2 (39).
6. OPC Toolbox. URL: <https://exponenta.ru/opc-toolbox>.
7. Взаимодействие MATLAB с оборудованием: 14. Подключение оборудования L-card при помощи OPC. – URL: <https://exponenta.ru/news/vzaimodejstvie-matlab-s-oborudovaniem-14-podklyuchenie-oborudovaniya-l-card-pri-pomoshchi-opc>.
8. Взаимодействие MATLAB с оборудованием: 15. Подключение оборудования L-card при интеграции dll. – URL: <https://exponenta.ru/news/vzaimodejstvie-matlab-s-oborudovaniem-15-podklyuchenie-oborudovaniya-l-card-pri-integracii-dll>.
9. Суслов Д.А. Влияние переходных режимов на нестационарные вихревые явления в модели отсасывающей трубы гидротурбины / Д.А. Суслов, И.В. Литвинов, С.И. Шторк, Е.Ю. Гореликов. Сибирский физический журнал. 2019. Т. 14. № 4. С. 55-68. DOI 10.25205/2541-9447-2019-14-4-55-68.
10. Когогин Д.А. Разработка автоматизированного приемного комплекса для исследования характеристик радиоволн, распространяющихся через ионосферу / Д.А. Когогин, И. А. Насыров. Труды Международной Байкальской молодежной научной школы по фундаментальной физике и Конференции молодых ученых «Взаимодействие полей и излучения с веществом», Иркутск, 19-24 сентября 2011 года. Иркутск: Редакционно-издательский отдел Института солнечно-земной физики СО РАН, 2011. С. 278-279.
11. L-Card L502/E502 SDK. Версия 1.1.19. URL: www.lcard.ru/download/lpcie_setup.exe.
12. FAQ: How do I install the MinGW compiler? URL: www.mathworks.com/matlabcentral/answers/311290-faq-how-do-i-install-the-mingw-compiler (дата обращения 19.05.2021)
13. Lum C. How to install the MinGW-w64 compiler in Matlab? [Видео] URL: <https://youtu.be/12XRTUqG94Y>.
14. Installation of some MATLAB Support Packages fails with «Download Error» message (1741173). URL: www.mathworks.com/support/bugreports/1741173
15. Борисов А. Со временные устройства сбора данных L502 и E502: Руководство программиста / А. Борисов. М.: ООО «Л Кард», 2020. Ревизия 1.1.10. 127 с. URL: www.lcard.ru/download/x502api.pdf.
16. C Libraries in MATLAB URL: www.mathworks.com/help/matlab/c-library-functions.html (дата обращения 19.05.2021)
17. How do I break out of an infinite loop without terminating the subsequent routine? Matlab Answers. URL: <https://www.mathworks.com/matlabcentral/answers/98491-how-do-i-break-out-of-an-infinite-loop-without-terminating-the-subsequent-routine>.
18. Matsak, I.S. L-CARD E-502 Matlab code. 2021. URL: <https://www.mathworks.com/matlabcentral/fileexchange/10204>.