

Подход к разработке контроллера движения CAN сервоприводов на основе ARM микрокомпьютеров¹

Г.М. Мартинов,
д.т.н., проф., martinov@ncsystems.ru,
Акрам Аль Хури,
асп., инж., akramalkhoury@gmail.com,
МГТУ «СТАНКИН», г. Москва

Цель работы - обеспечить механизм взаимодействия между высокоуровневым контроллером движения, выполняющим функции системы ЧПУ, и низкоуровневым СПШ приводом. Этот механизм должен быть надежным, недорогим, простым в использовании и гибким.

Ensure a connection mechanism between a high-level motion controller that performs the functions of CNC system and a low-level servo-drive (SPSH). This mechanism should be reliable, inexpensive, easy to use and flexible.

В настоящее время существует большой спрос на использование настольных станков с ЧПУ, которые имеют низкую цену, достаточно широкий диапазон функциональных возможностей, и могут использоваться в образовательных целях. Эти станки могут выполнять основные технологические переходы по изготовлению деталей, такие как, например, фрезерование и сверление легких сплавов, где нет необходимости выполнять высокоскоростную и высокоточную обработку. В этой работе предложена модель системы ЧПУ для настольных станков и проанализированы ее компоненты и эффективность. Модель состоит из трех основных слоев: системы ЧПУ (прикладной уровень), микрокомпьютера на основе ARM в качестве CAN-контроллера (соединительный слой) и сервоприводных шаговых двигателей (исполнительный слой).

СПШ – это высокопроизводительный сервопривод с векторным управлением по скорости и моменту. Интегрированный сервопривод СПШ объединяет в одном компактном корпусе все необходимое для обеспечения управления движением ротора шагового двигателя в различных режимах работы. Стоимость нового интегрированного сервопривода СПШ существенно ниже аналоговых систем, ими можно управлять непосредственно через протокол CAN, а использование микрокомпьютерами семейства Raspberry PI дает гибкость управления.

CAN (англ. Controller Area Network) – это широко распространенные в промышленной автоматизации контроллеры и трансиверы, используемые при организации процессов управления в реальном времени, в технологиях «умного дома» и многих других областях.

Основное преимущество технологии CAN - очень высокая степень надежности: сеть выживает в самых суровых условиях, причем статистическая вероятность отказа - порядка одного случая на десять лет.

Применение протокола CAN актуально в самых разнообразных приложениях, где требуется применение распределенной системы контроллеров.

1. СПШ сервоприводы

Интегрированный сервопривод СПШ объединяет в одном компактном корпусе все необходимое для обеспечения управления движением ротора шагового двигателя в различных режимах работы.

Гибкая система настройки и встроенный контроллер позволяют использовать сервоприводы в различных областях, а благодаря стандартным фланцам NEMA и компактным размерам СПШ может быть использован и при модернизации оборудования [2].

Конструктивно сервопривод СПШ можно разделить на следующие основные блоки (рис. 1):

- 1) гибридный шаговый двигатель с габаритами NEMA 23 и NEMA 34;
- 2) преобразователь частоты на основе высокопроизводительного DSP процессора;
- 3) блок управления (сервоконтроллер и программируемый логический контроллер в одном корпусе);
- 4) датчик позиции вала мотора.



рис. 1 Конструкция сервопривода СПШ

Применение сервопривода СПШ повышает точность изготовления изделий, ведь аналогом сервопривода СПШ могут выступать системы, основанные на шаговых двигателях, где пропуск шагов (дискретных угловых перемещений) существенно отражается на производственном процессе. В сервоприводе СПШ пропуск шагов исключен. Таким образом, в отличие от сервоприводов, собранных на основе шаговых двигателей, сервопривод СПШ имеет более высокие динамические характеристики и точность обработки деталей, при использовании СПШ увеличивается.

¹ Работа выполнена при поддержке Минобрнауки России в рамках выполнения государственного задания (№ 2.1237.2017/4.6).

2. Специфика задач, связанных с ЧПУ

Система ЧПУ оснащена средствами для создания технологических комплексов, подключаемых к сетям цехов. Открытая модульная масштабируемая архитектура управления позволяет производителям станков и конечным пользователям расширять функциональности систем, просто интегрируя в них новые программные и аппаратные решения.

В настоящее время решения, доступные на рынке, дороги и имеют избыточные функции, а большинство из них закрыты. Это приводит к сложности и невозможности использования существующих решений для дешевого сегмента промышленного производства [1].

Чтобы реализовать СПШ сервоприводы в системах CNC, которые также поддерживают протокол CAN, необходимо специфицировать предъявляемые к ним требования. Речь здесь идет о техническом решении, которое может доставлять правильные команды от контроллера СЧПУ к этим приводам [5].

Основные требования, предъявляемые к механизму подключения оборудования сторонних производителей [6]:

- возможность подключения промышленных устройств различного рода (электроприводы, устройства ввода-вывода, контроллеры автоматизации, ...)
- возможность подключения промышленных устройств различных стандартов связи (Sercos, EtherCAT, CAN ...)
- независимость времени цикла обмена данными в сети непосредственно от времени цикла работы любого конечного устройства.

Следовательно, основной контроллер ЧПУ должен соответствовать этим требованиям для обеспечения подключения стороннего CAN-оборудования:

Микрокомпьютеры и микроконтроллеры программировании с открытым исходным кодом широко применяются в области в том числе, благодаря их способности адаптироваться к любой части приложений.

Идея состоит в том, чтобы использовать Raspberry Pi3 в качестве промежуточного устройства между контроллером ЧПУ и сервоприводом СПШ. Здесь речь идет о аппаратно-программном решении. Raspberry Pi3 имеет операционную систему (на базе Linux), которая делает ее мощным инструментом для обработки потоков программ и других типов процедур синхронизации.

Реализация физического уровня, обеспечивающего обмен данными, требует аппаратного решения, поэтому была использована микросхема MCP2515 как трансивер данных по CAN

3. CAN протоколы для управления приводами СПШ

В автоматизированной промышленной установке, шина CAN чаще всего используется как часть распределенной системы управления, соединяющей жизненно важные системы, которые могут распространяться по всему объекту. Как правило, система ЧПУ передает команды через интерфейс шины CAN на датчики, исполнительные механизмы, двигатели или другие механические системы, которые выполняют требуемое действие.

Шина CAN является популярным выбором, где скорость, низкая стоимость и простота установки для подключения систем контроля доступа, безопасности и охраны окружающей среды.

Интегрированная коммуникация с CAN-шиной предлагает ряд преимуществ для промышленного управления:

- скорость передачи данных CAN выше, чем традиционная аналоговая связь, поскольку одновременно можно отправлять несколько сообщений всем подключенным устройствам и исполнительным устройствам,
- благодаря своей двухпроводной структуре шина CAN обеспечивает повышенную гибкость установки и обслуживания. Системы с CAN-подключением не только содержат значительно меньше проводов, что упрощает их установку, но добавление новых компонентов в систему требует гораздо меньше настроек,
- гораздо меньше чувствительность к магнитным помехам, чем аналоговая проводка.
- более низкие затраты на оборудование и минимальные требования к обработке сигналов делают CAN идеальным решением для встроенных приложений, требующих многопроцессорной связи по бюджету.

3.1. Протокол CAN реального времени

В текущей реализации используется стандартный формат сообщения, состоящий из 11-ти битного идентификатора (ID10-ID0) и данных, длина которых варьируется от 0 до 8 байт. Идентификатор сообщения в шине CAN определяет приоритет команды. CAN шина обладает высокой детерминированностью – это означает, что готовое к отправке сообщение с наивысшим приоритетом начнет передаваться немедленно, если линия свободна, и не позже завершения передачи текущего сообщения [2].

Таблица 1

Формат передаваемого сообщения

ID10 – ID6	ID5 – ID3	ID2-ID0	8*8 bits
Команда	Адрес источника	Адрес назначения	Данные

В рамках одной сети необходимо, чтобы каждое устройство имело уникальный адрес в диапазоне от 0 до 6. Адрес 7 используется как широковещательный. Любая посылка, переданная со значением поля «Адрес назначения», равным 7, будет воспринята всеми приводами.

3.2. Технологический протокол

Технологический протокол предназначен для настройки привода и анализа его работы. С помощью данного протокола пользователь может осуществить доступ к любым параметрам как на запись, так и на чтение.

В отличие от протокола управления приводами, технологический протокол не является протоколом реального времени, поэтому передается с низким приоритетом и обрабатывается в приводе потоком, имеющим наименьший приоритет. Технологический протокол передается в поле данных D0-D6 сообщения протокола управления приводами по шине CAN.

Формат сообщения технологического протокола

Байт данных в сообщении CAN	D0	D1	D2	D3	D4	D5	D6
Формат технологического протокола	Команда технологического протокола		Адрес параметра		Значение параметра		

4. Архитектурная модель используемого решения

Предлагаемая модель состоит из CAN-мастера, реализованного на основе программы С, на Raspberry PI3. Эта программа обеспечивает соединение между системой ЧПУ и приводами. Пакеты, которые поступают из системы управления ЧПУ через Ethernet и соответствующие сообщения, будут сформированы в Raspberry PI3, чтобы отправить их на приводы через шину CAN [3] [4].

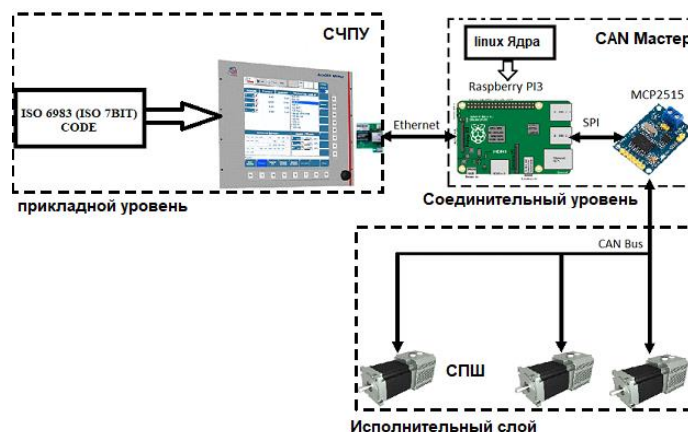


рис. 2 Предлагаемая системная модель

В рамках работы были запрограммированы основные части, управляющие приводами СПШ на основе сообщений, полученных от системы ЧПУ. Были использованы библиотеки SocketCAN, которые обеспечили быстродействие и надежность получения и отправки пакетов по CAN.

SocketCAN — это набор драйверов с открытым исходным кодом CAN и сетевой стек, предоставленный Volkswagen Research для ядра Linux. Ранее известна как CAN-стандарт низкого уровня Low Level CAN Framework (LLCF). Концепция SocketCAN использует модель сетевых устройств, которая позволяет нескольким приложениям одновременно обращаться к одному CAN-устройству. Кроме того, одно приложение может одновременно обращаться к нескольким сетям CAN.

Приложение сначала устанавливает свой доступ к интерфейсу CAN, инициализируя сокет (подобно тому, как в TCP / IP-связи), а затем привязывая (binding) этот сокет к интерфейсу (или всем интерфейсам, если приложение так желает). После привязки сокет может затем использоваться как UDP socket через read и write функции.

Следующая структура была использована для формирования желаемого сообщения CAN, которое будет отправлено на сервопривод. Где id - CAN-11-битный идентификатор, а data - 8 * 8-битные CAN-данные.

```
struct CAN_MSG {
  unsigned int id;
  unsigned char data[8];
};
```

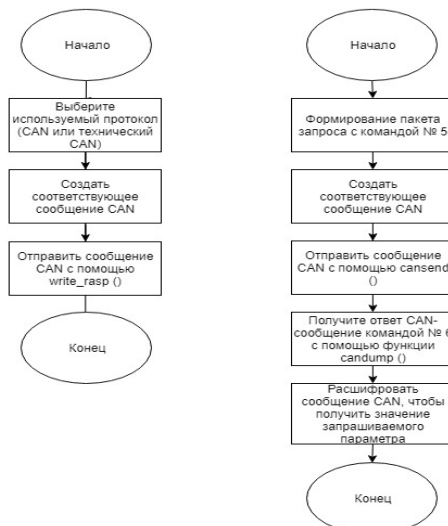


рис. 3 Процедуры чтение\запись

5. Результаты измерения перемещений, задаваемых внешнему контроллеру CAN на стенде с сервоприводами СПШ-10 из тестового приложения системы ЧПУ АксиОМА Контроль

Представленная на рисунке 4 осциллограмма отображает: разницу между конечным и текущим положением, отклик на команду перемещения, перемещение на заданное расстояние в ручном режиме без перерегулирования и соответствует заданному.

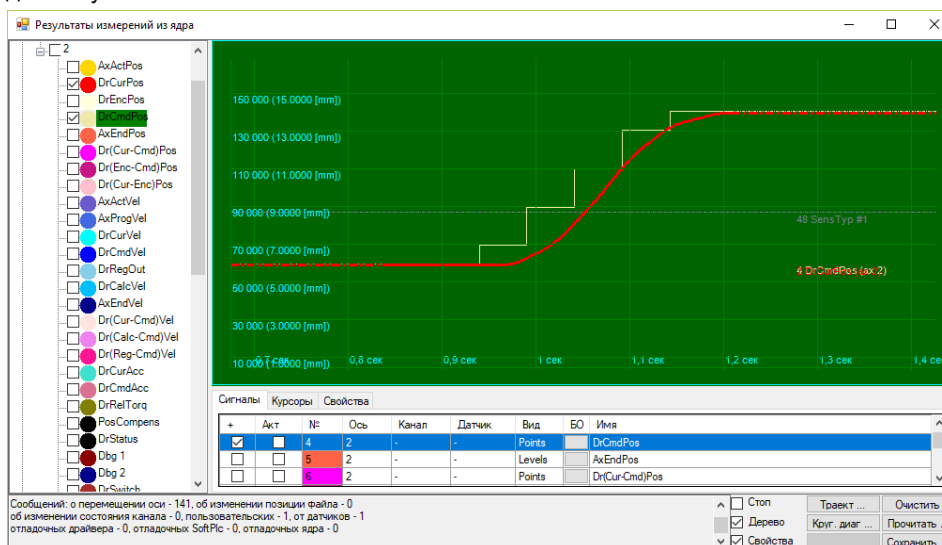


рис. 4 Результаты измерений, полученные из ядра СЧПУ

При тестировании разрабатываемого решения было выявлено, что динамические показатели соответствуют требуемым показателям, что показано на рисунке 5.

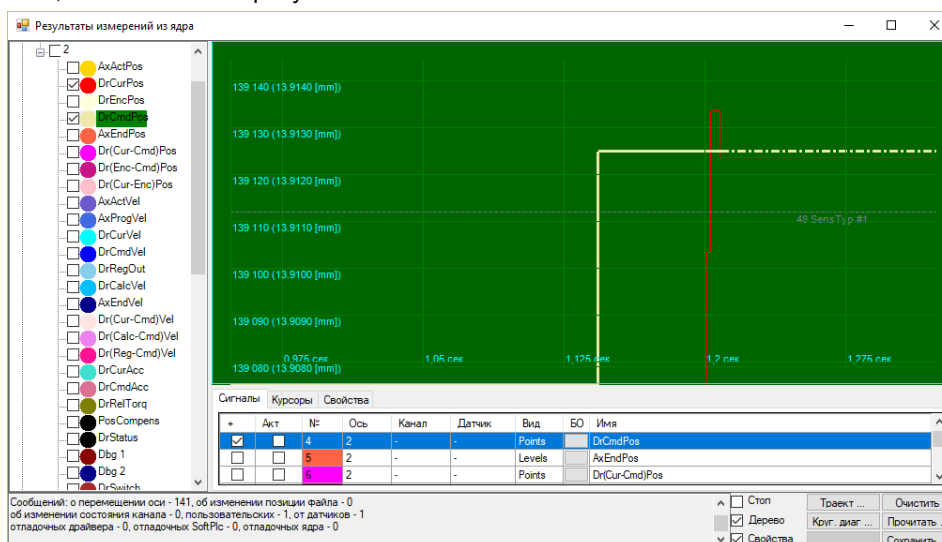


рис. 5 Результаты тестирования

Не высокая точность объясняется использованием грубого датчика позиции сервопривода, что будет исключено в последующей работе, запланированной для управления сервоприводами, где будут использоваться команды положения вместо команд скорости, и, следовательно, система будет считать шаги, которые необходимы для обеспечения заданной позиции.

Литература

1. Martinov, G., Sokolov, S., Martinova, L., Grigoryev, A. and Nikishechkin, P. (2017). Approach to the Diagnosis and Configuration of Servo Drives in Heterogeneous Machine Control Systems. In: 8th International Conference, ICSI. Fukuoka, Japan, pp.586-594.
2. Интегрированный сервопривод СПШ 10: Руководство пользователя // ЗАО «Сервотехника». 2010
3. Сосонкин В.Л., Мартинов Г.М. Системы числового программного управления: Учеб. Пособие. – М.: Логос, 2005.- 296с.
4. Денисенко В.В. Компьютерное управление технологическим процессом, экспериментом, оборудованием. – М: Горячая линия – Телеком, 2009. – 608 с., ил.
5. МЕТОД ДЕКОМПОЗИЦИИ И СИНТЕЗА СПЕЦИАЛИЗИРОВАННЫХ СИСТЕМ ЧПУ Мартинов Г.М., Козак Н.В., Нежметдинов Р.А., Григорьев А.С., Обухов А.И., Мартинова Л.И. Автоматизация в промышленности. 2013. № 5. С. 08-14.
6. ПРАКТИЧЕСКИЕ АСПЕКТЫ РЕАЛИЗАЦИИ МОДУЛЕЙ ОТКРЫТОЙ СИСТЕМЫ ЧПУ Мартинова Л.И., Мартинов Г.М. Автотракторное электрооборудование. 2002. № 3. С. 31-37.