

Стандарт обмена ElVinto для ЧПУ.

Версия 11.

Дата модификации: 08.01.2023

Добавлен п.8

Дата модификации: 03.01.2023

Добавлены функции G27, W72, W73 (п.3)

Добавлены коды ошибок ALARMNUM (п.4)

Добавлен п.7

Дата модификации: 21.10.2022

Изменен формат функции STATE (п.5)

Дата модификации: 18.10.2022

Добавлены функции M06, W70, W71 (п.3)

Добавлены функции внешнего приложения CHECKCHANGETOOL, READY (п.4)

Добавлены функции контроллера станка TOOL, READY, PROGRAM_END (п.5)

Изменен формат функции GCODE (п.4)

Дата модификации: 13.10.2022

Добавлены функции по работе с SD-картами

Дата модификации: 15.09.2022

Некоторые изменения

Дата модификации: 18.09.2022

Добавлены функции в п.3

Текущий стандарт предназначен для обмена данными между станком ЧПУ и внешним приложением. Для передачи могут использоваться интерфейсы Bluetooth, RS232 или UART.

Взаимодействие осуществляется таким образом, что инициатором команд может выступать как внешнее приложение, так и сам станок. Команды, которые не понятны внешнему приложению, должны игнорироваться. Команды, которые не понятны станку – отражаются ошибкой.

1. Формат передачи данных ElVinto.

При передачи данных используется следующий формат команд:

```
CMD, PARM1<#VALUE1, PARM2<#VALUE2, ...;
```

, где:

CMD – название команды

PARM1, PARM2, ... – название параметра

VALUE1, VALUE2, ... – значение параметра

Все команды отделяются знаком «;» («точка с запятой»).

Параметры команды от названия команды и других параметров отделяются знаком «,» («запятая»).

Параметры команды от значения этого параметра отделяется знаком «<» («меньше»).

Если используется текст не содержащий вышеуказанных служебных символов, он должен начинаться со знака «#» («решетка»), при этом этот знак игнорируется и не считается значащим.

Если используется текст или двоичные данные, которые могут содержать вышеуказанные служебные символы, то знак «#» («решетка») не используется, а весь текст кодируется Base64. Последовательность расположения параметров не имеет значения. Команды и параметры чувствительны к регистру.

2. Передача данных.

При начальной инициализации, станку нужно передать команду «INIT;» для выполнения начальной инициализации.

Дальнейшая передача всех команд осуществляется следующим образом:

Сначала на станок передаются контрольные данные для следующей за ней команды, например, передача команды «G28;» (вместе со знаком «точка с запятой») :

```
CRC,VALUE<#КонтрольнаяСумма,MD5<#ХэшMD5,ID<#IDкоманды;  
GCODE,DATA<#G28;
```

, где:

CRC – команда передачи контрольной суммы

КонтрольнаяСумма – простой алгоритм подсчета контрольной суммы следующей далее команды «G28;»

ХэшMD5 – Хэш сумма MD5 следующей далее команды «G28;»

IDкоманды – Порядковый номер передаваемой команды (каждая новая команда должна инкрементировать это значение, иначе не будет принята)

Алгоритм расчета простой контрольной суммы в формате платы STM32:

```
long getCRC16(const char *data, int len)  
{  
    long crcShort = 0;  
  
    for (int i = 0; i < len; i++) {  
        long f = ((long) data[i]) << 1;  
  
        f = f ^ ((f & 0xff00) >> 8);  
    }  
}
```

```

        crcShort += (f & 0xffff);
        crcShort &= 0xffff;
    }

    return crcShort;
}

```

3. Команды G-кода принимаемые станком.

```

#define CMD_G00          0          // Быстрое позиционирование
#define CMD_G01          1          // Линейное перемещение на подаче
#define CMD_G02          2          // Арка по часовой стрелке
#define CMD_G03          3          // Арка против часовой стрелки

#define CMD_G17          17         // НЕ ИСПОЛЬЗУЕТСЯ. Установить координатную
плоскость XY для функций G02, G03
#define CMD_G18          18         // НЕ ИСПОЛЬЗУЕТСЯ. Установить координатную
плоскость ZX для функций G02, G03
#define CMD_G19          19         // НЕ ИСПОЛЬЗУЕТСЯ. Установить координатную
плоскость YZ для функций G02, G03

#define CMD_G21          21         // Установить метрическую систему
(установлена всегда, дюймовой не существует)

#define CMD_G27          27         // Аналогично G29, но выполняется только один
раз с момента включения станка.
#define CMD_G28          28         // Отвести в плоскость безопасности по Z,
переместить в позицию [0, 0] по X, Y
#define CMD_G29          29         // Переместить в начальную позицию, сбросить
координаты, а затем вернуться в исходную точку. Нужно, например, для определения
текущих координат.

#define CMD_G40          40         // Отключить компенсацию инструмента
#define CMD_G41          41         // НЕ ИСПОЛЬЗУЕТСЯ. Компенсации инструмента
слева не существует
#define CMD_G42          42         // НЕ ИСПОЛЬЗУЕТСЯ. Компенсации инструмента
справа не существует

#define CMD_G54          54         // НЕ ИСПОЛЬЗУЕТСЯ. Координатная система
только одна

#define CMD_G75          75         // Запустить ось A в режиме токарной
обработки рад/сек
#define CMD_G76          76         // Запустить ось A в режиме токарной
обработки об/мин

#define CMD_G80          80         // Отключить все циклы
#define CMD_G81          81         // Цикл сверления
#define CMD_G81_99      81.99     // Цикл сверления (следующие операции сверления
внутри цикла)
#define CMD_G83          83         // Цикл поэтапного сверления
#define CMD_G83_99      83.99     // Цикл поэтапного сверления (следующие операции
сверления внутри цикла)

#define CMD_G90          90         // Установить абсолютную систему координат
#define CMD_G91          91         // Установить относительную систему
координат
#define CMD_G92          92         // Установить текущие абсолютные координаты

// ElVinto commands
// {{{

```

```

#define CMD_G100      100      // Фрезерование плоскости в направлении X...X,Y -
координата диагонального угла
#define CMD_G101      101      // Фрезерование плоскости в направлении Y
// }}}

#define CMD_G300      300      // Функция отладки кода

#define CMD_M01        1        // НЕ ИСПОЛЬЗУЕТСЯ. Остановки программы не
происходит. Если для смены инструмента надо использовать M06.
#define CMD_M03        3        // Включить шпиндель/лазер
#define CMD_M04        4        // Включить шпиндель/лазер
#define CMD_M05        5        // Выключить шпиндель/лазер

#define CMD_M06        6        // Сменить инструмент. Переводит шпиндель в
плоскость безопасности (параметр станка safetyZ)

#define CMD_M07        7        // Включить плазму
#define CMD_M08        8        // Выключить плазму

#define CMD_M30        30        // Конец программы со сбросом модальных
функций

#define CMD_M76        76        // НЕ ИСПОЛЬЗУЕТСЯ. Постпроцессор
инициализации плазмы. Ничего не делает
#define CMD_M77        77        // НЕ ИСПОЛЬЗУЕТСЯ. Постпроцессор
инициализации плазмы. Ничего не делает

#define CMD_M99        99        // Конец программы
#define CMD_M200       200        // Сброс текущих координат
#define CMD_M201       201        // Установить процент ускорения/торможения по
умолчанию (нужно для длительных вращений оси A)
#define CMD_M210       210        // Сброс координаты Z в ноль (нужно для ручной
корректировки высоты инструмента)

#define CMD_W70        70        // Запомнить флаг абсолютной системы
координат
#define CMD_W71        71        // Восстановить флаг абсолютной системы
координат
#define CMD_W72        72        // Включить контроль рабочего поля
(03.01.2023)
#define CMD_W73        73        // Отключить контроль рабочего поля
(03.01.2023)
#define CMD_W90        90        // Отправлять состояние станка во время
микроопераций
#define CMD_W91        91        // Не отправлять состояние станка во время
микроопераций
#define CMD_W93        93        // Игнорировать функцию G28
#define CMD_W94        94        // Отменить игнорирование функции G28
#define CMD_W95        95        // Получение текущих координат
#define CMD_W96        96        // Получение данных о концевых выключателях
#define CMD_W97        97        // Получение данных о станке
#define CMD_W98        98        // Нажата кнопка "Emergency"

```

4. Команды принимаемые станком.

CRC16,VALUE<#КонтрольнаяСумма,MD5<#Хэш;

Передача контрольной суммы и хэш-суммы для следующей за этой командой. Все команды должны предваряться этой командой, иначе команды приняты не будут.

INIT;

Начальная инициализация станка. При этом происходит также сброс счетчика ID для принимаемых команд

GCODE,DATA<#КадрGкода;

Передача одного кадра G-кода. Если требуется передать пакет сразу из нескольких кадров, то их следует отделить знаком «|» (вертикальный разделитель)

RESETALERT;

Сброс аварийной ситуации на станке.

READDIR;

Получение списка файлов находящихся на SD-карте в корневой директории. Сначала отправляется команда DIR_BEGIN, затем список команд FILE соответствующих каждому файлу, затем DIR_END. В случае ошибки станок передает соответствующие команды. Более подробно о командах передаваемых станком в п.5.

RUN,NAME<#ИмяФайла;

Запуск программы G-кода находящейся в файле на SD-карте.
NAME

CHECKCHANGETOOL;

Запросить у станка, не требуется ли смена инструмента. Есть требуется – отправить в ответ команду TOOL (см.п.6)

READY;

Сообщить станку, что внешнее приложение проинициализировано и готово для передачи данных станку. В ответ ждет от станка аналогичную команду.

5. Команды отправляемые станком.

Метод передачи от станка к внешней системе:

CRC16, VALUE<#КонтрольнаяСумма, MD5<#Хэш;

Передача контрольной суммы и хэш-суммы для следующей за этой команды

RECEIVED, ID<#IDкоманды;

Станок отвечает, что переданная команда успешно принята в работу и начала выполняться

ID – ID – команды, которая начала выполняться

OK, ID<#IDкоманды;

Станок отвечает, что переданная команда полностью выполнена

ID – ID – команды, которая выполнена

ERROR, ID<#IDкоманды, NUM<#num;

Станок отвечает, что переданная команда вызвала ошибку

ID – ID – команды, которая вызвала ошибку

NUM – Номер ошибки

STATE, TYPE<#type, MX<#mx, MY<#my, MZ<#mz, MA<#ma, MB<#mb, MC<#mc, FEED<#feed, SPEED<#speed [ALARMNUM<#alarmnum, AXISNUM<#axisnum], ISABS<#ДаНет;

TYPE – Тип текущего состояния станка

idle – ожидание принятия команд

run – выполнение команды, другие команды не

принимаются

cnscalarm - произошла аварийная ситуация в масштабе всего станка, например, нажата кнопка «Emergency»
axisalarm - произошла аварийная ситуация в масштабе оси, например, сработал концевой выключатель
MX, MY, MZ, MA, MB, MC - текущие координаты станка. В некоторых режимах работы станка актуальные координаты будут только после завершения выполнения текущей команды.
FEED - Текущая подача
SPEED - Текущая скорость вращения шпинделя

ALARMNUM - номер аварийной ситуации

- 1 - Сработал начальный концевой выключатель
- 2 - Сработал конечный концевой выключатель
- 3 - Сбой работы начального концевого выключателя
(концевой выключатель передает промежуточный уровень)
- 4 - Сбой работы конечного концевого выключателя
(концевой выключатель передает промежуточный уровень)
- 5 - Нарушение физических границ станка со стороны начала физической оси
- 6 - Нарушение физических границ станка со стороны окончания физической оси
- 100 - Нажата кнопка «Emergency»

AXISNUM - номер оси, на которой возникла аварийная ситуация

ISABS - Y - если текущая система перемещения абсолютная
(после выполнения функции G90)

N - если текущая система перемещения относительная
(после выполнения функции G91)

DIR_BEGIN;

Начало передачи списка файлов корневого каталога SD-карты станка

FILE,NAME<#ИмяФайла;

Команда передачи имени файла, находящегося в корневом каталоге SD-карты станка

NAME - Имя файла

END_BEGIN;

Конец передачи списка файлов корневого каталога SD-карты станка

SDD_NO;

Ошибка - станок не смог обнаружить SD-карту в картридере

SDD_OK,TYPE<#ТипSDкарты,FS<#ФайловаяСистема,ROOT<#корень;
Результат доступа к SD-карте
TYPE - Тип SD-карты (SD1, SD2, SDHC, UNKNOWN)
FS - Тип файловой системы - FAT или NO. В случае «NO»
чтение файлов невозможно
ROOT - Результат чтения корневого каталога - ОК или NO. В
случае «NO» чтение файлов невозможно

TOOL,NUM<#НомерИнструментаВПрограммеGкода;
Во время выполнения программы G-кода, потребовалась смена
инструмента. Если ЧПУ не имеет средство автоматической замены
инструмента, то требуется сообщить пользователю.
NUM - Новый номер инструмента в программе G-кода который
требуется установить на станке.

READY;
Ответ на запрос аналогичной команды от внешнего приложения,
что станок проинициализирован и готов для получения команд

PROGRAM_END;
Сообщает внешнему приложению, что завершено выполнение G-кода
запущенного из файла на SD-карте

6. Особенности передачи данных.

После передачи основной команды на станок, он должен ответить командой RECEIVE, что команда успешно принята. В ином случае приложение должно отправлять эту команду с интервалом 200-400 мс, предполагая, что передача команды не состоялась.

После принятия команды, станок перед ответом RECEIVE должен запомнить ID команды, которую принял, и более не принимать ни одну команду с ID меньше или равным текущему. Это исключает ситуацию, когда одна и та же команда будет принята дважды. Станок у себя сбрасывает текущий ID в момент инициализации или при выполнении команды «Конец программы».

7. Контроль за перемещениями станка.

Контроль за перемещениями включает в себя контроль на срабатывание концевых выключателей и контроль границ рабочего поля. При включении станка любой контроль за перемещениями отсутствует. Для включения контроля рабочего поля нужно выполнить команду G28 или W72. Для отключения контроля рабочего поля нужно выполнить команду W73. Контроль на срабатывание концевых выключателей используется станком только во время выполнения команды G28.

Контроль физических границ станка необходим чтобы в случае сбоя при перемещениях (выпадение шагов, ручная корректировка положения позиционера и т.п.) каретки станка не могли выйти за пределы рабочей области и привести к поломке станка.

Контроль физических границ станка автоматически включается после выполнения команд G28 или G29, после выполнения которых станок запоминает начало физических координат и исходя из информации о физических размерах рабочего поля по каждой оси, контролирует нахождение позиционера внутри поля. Если позиционер выходит за пределы этого поля, то это приводит к аварийной ситуации (п.4, ALARMNUM).

8. Нестандартные команды G-кода

G27 – Аналогично G29, но выполняется только один раз с момента включения станка (остальные попытки выполнить команду игнорируются). После перезапуска станка команду G27 можно успешно выполнить снова. Необходима чтобы после включения станка при помощи концевых выключателей станок определил границы рабочего поля.

G28 – Возвратиться в начальную точку концевых выключателей. Сначала отводится по оси Z, затем Y, затем X.

G29 – Выполняется команда G28, а затем возвращается в исходную точку в обратном порядке. Сначала отводятся одновременно по осям X и Y, а затем по Z.