**ПЛАЗМОРЕЗ V1.0.1a**

**ПРОТОКОЛ УПРАВЛЕНИЯ**

**Версия 1.0.0**

Оглавление

1. Структура команд TCP-соединения………………………………………………………………………………………3
2. Команды TCP-соединения…………………………………………………………………………………………………5
3. Структура UDP пакета……………………………………………………………………………………………………..10

**Структура команд TCP-соединения**

**При установке параметров:**

XCD = a, b, c…z;

Где:

X – ось;

CD – команда;

a, b, c…z – задаваемые значения [целые числа];

“,” – разделитель устанавливаемых значений;

“;” – конец команды.

Ответ при положительной обработке команды (ось и команда распознаны, а значения в заданных пределах):

>;

Ответ при отрицательной обработке команды (ось или команда не распознаны, или значения не в заданных пределах):

?>;

**При мониторинге:**

XCD;

Где:

X – ось;

CD – команда.

Ответ при положительной обработке команды (ось и команда распознаны):

a, b, c…z;

Где a, b, c…z – считанные значения.

Ответ при отрицательной обработке команды (ось или команда не распознаны):

?>;

**Оси:**

X – горизонтальная ось (малая);

Y – вертикальная ось (большая);

A – все оси.

**Команды TCP-соединения**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Команда | Функция | Доступ | | Описание | Оси | | |
| Запись | Чтение | X | Y | A |
| 1 | SP | Установка рабочей скорости | ˅ |  | Запись максимальной движения оси, [мм/с]\*100 | ˅ | ˅ | ˅ |
| 2 | AC | Установка скорости разгона | ˅ |  | Запись ускорения оси, [мм/с2]\*100 | ˅ | ˅ | ˅ |
| 3 | DC | Установка скорости торможения | ˅ |  | Запись замедления оси, [мм/с2]\*100 | ˅ | ˅ | ˅ |
| 4 | AP | Установка целевой координаты | ˅ |  | Запись целевой координаты относительно начала, [мм]\*100 | ˅ | ˅ | ˅ |
| 5 | BG | Начало движения в целевую координату | ˅ |  | Команда выполняется только после калибровки и при заданных параметрах SP, AC, DC, AP | ˅ | ˅ | ˅ |
| 6 | ST | Медленная остановка двигателя | ˅ |  | После записи команды двигатель начинает торможение до полной остановки | ˅ | ˅ | ˅ |
| 7 | AB | Экстренная остановка двигателя | ˅ |  | Мгновенна остановка двигателя в критических ситуациях | ˅ | ˅ | ˅ |
| 8 | GP | Чтение EM, PS, VL, DP, PE, MS |  | ˅ | Возвращает значения статусных регистров микроконтроллера. При использовании команды с осью A возвращает сначала значения оси X, потом Y | ˅ | ˅ | ˅ |
| 9 | GS | Возвращает MF |  | ˅ | Возвращает значение регистра MF | ˅ | ˅ | ˅ |
| 10 | AO | Установка значений SP, AC, DC, AP | ˅ |  | Запись всех параметров оси.  При использовании команды с осью A(X и Y одновременно), значения задаются сразу на две оси. Например: AAO=190,40,30,600; | ˅ | ˅ | ˅ |
| 11 | PP | Установка целевых координат для двух осей | ˅ |  | Запись целевых координат для X и Y осей, [мм]\*100. Можно использовать с любой осью: X, Y, A | ˅ | ˅ | ˅ |
| 12 | GH | Начало калибровки | ˅ |  | Установка осей в начало координат, выравнивание по концевым выключателям и подсчет коэффициентов | ˅ | ˅ | ˅ |
| 13 | NA | Сетевой адрес микроконтроллера | ˅ | ˅ | Запись и чтение сетевого адреса микроконтроллера. Например, для записи (адрес по умолчанию) “XNA=192.168.16.251;”, для чтения “XNA;”. Записанный адрес устанавливается после перезагрузки.  Можно использовать любую ось | ˅ | ˅ | ˅ |
| 14 | NP | Номер порта микроконтроллера | ˅ | ˅ | Запись и чтение сетевого порта микроконтроллера. Например, для записи (порт по умолчанию) “XNP=4000;”.  Можно использовать любую ось | ˅ | ˅ | ˅ |
| 15 | UA | Сетевой адрес, на который отправляются отчеты по протоколу UDP | ˅ | ˅ | Запись и чтение сетевого адреса, на который периодически отправляется отчет состояния устройства | ˅ | ˅ | ˅ |
| 16 | UP | Номер порта, на который отправляются отчеты по протоколу UDP | ˅ | ˅ | Запись и чтение сетевого порта, на который периодически отправляется отчет состояния устройства | ˅ | ˅ | ˅ |
| 17 | UI | Интервал времени, через который отправляются отчеты по UDP | ˅ | ˅ | Запись и чтение интервала времени, через который периодически отправляется отчет состояния устройства, [мс] | ˅ | ˅ | ˅ |
| 18 | PS | Текущее положение |  | ˅ | Возвращает значение текущего положения (среднее для Y) оси/осей, [мм]\*100 | ˅ | ˅ | ˅ |
| 19 | VL | Текущая скорость |  | ˅ | Возвращает значение текущей скорости (среднее для Y) оси/осей, [мм/с]\*100 | ˅ | ˅ | ˅ |
| 20 | DP | Целевая координата |  | ˅ | Возвращает значение целевой координаты оси/осей, [мм]\*100 | ˅ | ˅ | ˅ |
| 21 | PE | Ошибка позиционирования |  | ˅ | Возвращает разность значений текущих координат энкодера и счетчика оси/осей, [мм]\*100 | ˅ | ˅ | ˅ |
| 22 | MF | Последняя ошибка |  | ˅ | Возвращает код последней ошибки:  0 – ошибок не было;  1 – ошибка драйвера (не та частота?);  2 – получена команда экстренного торможения;  3 – большая ошибка позиционирования (разница показаний  4 – двигатель заторможен;  5 – ошибка подключения энкодера (отсчеты в обратную сторону);  6 – энкодер отключен/не работает. | ˅ | ˅ | ˅ |
| 23 | MS | Текущий статус движения двигателя оси |  | ˅ | Возвращает битовую маску текущего состояния оси:  Бит 0 – в движении;  Бит 1 – неподвижен;  Бит 2 – ускоряется;  Бит 3 – замедляется;  Бит 4 – ожидает команды для начала движения;  Бит 5 – в целевой точке;  Бит 6 – проведена калибровка;  Бит 6-7 – reserved. | ˅ | ˅ | ˅ |
| 24 | EM | Причина последней остановки |  | ˅ | EM = 0 – все еще в движении;  EM = 1 – нормальная остановка двигателя;  EM = 2 – сработал передний концевой выключатель;  EM = 3 – сработал задний концевой выключатель;  EM = 4 – достигнут заданный предел конца координат;  EM = 5 – достигнут предел начала координат;  EM = 6 – произошла ошибка;  EM = 7 – получена команда остановки; | ˅ | ˅ | ˅ |
| 25 | RESET | Перезагрузка микроконтроллера |  | ˅ | Команда перезагрузки микроконтроллера может использоваться с любой осью | ˅ | ˅ | ˅ |

**Структура UDP пакета**

Структура, содержащая флаги калибровки и работоспособности основных составляющих позиционера:

typedef struct

{

uint8\_t switcherBitMask;

unsigned xCalibrationFlag:1;

unsigned yCalibrationFlag:1;

unsigned xEngineWorkingCapacity:1;

unsigned ylEngineWorkingCapacity:1;

unsigned yrEngineWorkingCapacity:1;

unsigned xEncoderWorkingCapacity:1;

unsigned ylEncoderWorkingCapacity:1;

unsigned yrEncoderWorkingCapacity:1;

uint16\_t reserved;

}statusBitMaskH;

Структура, содержащая текущие статусы движений осей (битовые маски):

typedef struct

{

uint8\_t xEngineMS;

uint8\_t yEngineMS;

uint16\_t resrved;

}statusMSH;

Структура пакета UDP:

typedef struct

{

uint32\_t identificator; // - идентификатор устройства

uint32\_t version; // - версия ПО

uint32\_t network\_address; // - записанный сетевой адрес микроконтроллера

uint32\_t network\_port; // - записанный порт сикроконтроллера

uint32\_t xME; // - статусный регистр МЕ оси Х

uint32\_t yME; // - статусный регистр МЕ оси Y

uint32\_t xPositionMm; // - текущая позиция оси Х в мм

uint32\_t yPositionMm; // - текущая позиция оси Y в мм

statusBitMaskH statusBitMask;// - структура, содержащая флаги калибровки и работоспособности двигателей и энкодеров

uint32\_t xPE; // - ошибка позиционирования оси X

uint32\_t yPE; // - ошибка позиционирования оси Y

statusMSH statusMS; // - структура, содержащая текущие статусы движений осей (битовые маски):

uint32\_t xSpeed; // - текущая скорость по энкодеру оси X

uint32\_t ySpeed; // - текущая скорость по энкодеру оси Y

uint32\_t switcherBitMask; // - битовая маска состояний концевых выключателей

}udpPacketH;

Битовые поля структуры состояний концевых выключателей:

struct switcherBitMask

{

uint8\_t X\_backwardSwitcher:1;

uint8\_t X\_forwardSwitcher:1;

uint8\_t Y\_backwardLeftSwitcher:1;

uint8\_t Y\_backwardRightSwitcher:1;

uint8\_t Y\_forwardLeftSwitcher:1;

uint8\_t Y\_forwardRightSwitcher:1;

uint8\_t reserved:2;

};