



**УСТРОЙСТВО
ЧИСЛОВОГО ПРОГРАММНОГО УПРАВЛЕНИЯ
NC-110, NC-310
NC-200, NC-201, NC-201M, NC-210, NC-220, NC-230
NC-301, NC-302**

Руководство по характеристике

АННОТАЦИЯ

Документ «Руководство по характеристике» (версия ВЗ.4) распространяется на устройства числового программного управления (УЧПУ) **NC-110, NC-200, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230, NC-301, NC-302** и **NC-310** (далее УЧПУ серии NC-XXX). В данном документе рассмотрены вопросы характеристики и инициализации УЧПУ в системе «УЧПУ – ОБЪЕКТ УПРАВЛЕНИЯ».

Понимание рассматриваемых вопросов необходимо специалистам, занимающимся подготовкой УЧПУ для работы с конкретным объектом управления, т. е. тем специалистам, которые специфицируют конфигурацию УЧПУ для конкретного пользователя.

СОДЕРЖАНИЕ

1. ОБЩИЕ СВЕДЕНИЯ О ХАРАКТЕРИЗАЦИИ.....	5
1.1. НАЗНАЧЕНИЕ ХАРАКТЕРИЗАЦИИ.....	5
1.2. УРОВНИ ХАРАКТЕРИЗАЦИИ.....	5
1.3. ФАЙЛЫ ХАРАКТЕРИЗАЦИИ.....	6
1.3.1. Назначение файлов характеристики	6
1.3.2. Структура файлов характеристики	7
2. КОДИРОВАНИЕ, УСТАНОВКА, КОПИРОВАНИЕ И ВОССТАНОВЛЕНИЕ SW УЧПУ	9
2.1. КОДИРОВАНИЕ ВЕРСИЙ SW УЧПУ	9
2.2. УСТАНОВКА SW УЧПУ НА FLASH DISK.....	9
2.3. СОЗДАНИЕ УСТРОЙСТВА USB-FLASH «СЕРВОДИСК».....	10
2.4. СОЗДАНИЕ АРХИВА ФАЙЛОВ DOM.....	11
2.4.1. Особенности использования USB-flash «СЕРВОДИСК»	12
2.4.1. Меню	12
3. ХАРАКТЕРИЗАЦИЯ СИСТЕМНОГО УРОВНЯ.....	15
3.1. СВЕДЕНИЯ О СОЗДАНИИ И ЗАГРУЗКЕ ФАЙЛА FCRSYS	15
3.2. СТАНДАРТНЫЙ ФАЙЛ FCRSYS	15
3.3. РАБОЧИЙ ФАЙЛ FCRSYS.....	16
3.3.1. Инициализация рабочего файла FCRSYS.....	16
3.3.2. Секция 1	16
3.3.3. Секция 2	17
3.3.4. Секция 3	19
3.3.5. Секция 4	19
3.3.5.1. Определение подключённых устройств FDD	20
3.3.5.2. Определение подключённых устройств HDD	20
3.3.5.3. Определение подключённого устройства Disk On Chip	21
3.3.5.4. Определение драйверов поддержки USB-устройств.	22
4. ХАРАКТЕРИЗАЦИЯ ОСЕЙ. ФАЙЛ AXCFIL	24
4.1. ОБЩИЕ СВЕДЕНИЯ О ФАЙЛЕ AXCFIL.....	24
4.1.1. Правила характеристики осей.....	24
4.2. СОДЕРЖАНИЕ СЕКЦИЙ ФАЙЛА AXCFIL	25
4.2.1. Свойства файла AXCFIL.....	25
4.2.1.1. Типы осей	25
4.2.2. Свойства секций файла AXCFIL	31
4.2.3. Секция 1	32
4.2.3.1. Инструкция NBP	32
4.2.3.2. Инструкция TIM	33
4.2.3.3. Инструкция PRO	34
4.2.3.4. Инструкция COM.....	34
4.2.3.5. Инструкция INx.....	35
4.2.3.6. Инструкция CAS	36
4.2.3.7. Инструкция ACC (ECDF).....	37
4.2.4. Секция 2	41
4.2.4.1. Инструкция PRO	42
4.2.4.2. Инструкция NAS.....	42
4.2.4.3. Инструкция TPA (общий случай).....	42
4.2.4.4. Инструкция TPA (попеременное управление осями от одного канала датчика). Версии SW с расширением Р-ПД	44
4.2.4.5. Инструкция NTC.....	45
4.2.4.6. Инструкция RAP	48
4.2.4.7. Инструкция GAS	49
4.2.4.8. Инструкция GAS (ECDF)	50
4.2.4.1. Инструкция VGAS	51
4.2.4.2. Инструкция PAS	52
4.2.4.3. Инструкция PAS (ECDF).....	54
4.2.4.4. Инструкция SKW	56
4.2.4.5. Инструкция MCZ (общий случай).....	58
4.2.4.6. Инструкция MCZ (TPA=...+40000H).....	60
4.2.4.7. Инструкция POS	62
4.2.4.8. Инструкция SRV	63

4.2.4.9.	Инструкция MAN.....	63
4.2.4.10.	Инструкция MSS	64
4.2.4.11.	Инструкция GMxx.....	64
4.2.4.12.	Инструкция GMxx (ECDF).....	65
4.2.4.13.	Инструкция FRC (ECDF)	67
4.2.4.14.	Инструкция LOP	69
4.2.4.15.	Инструкция LOn (ECDF)	70
4.2.4.16.	Инструкция MFC	72
4.2.4.17.	Инструкция AFC.....	72
4.2.4.18.	Инструкция CUB (ECDF).....	73
4.2.4.19.	Инструкция SWx (ECDF)	74
4.2.4.20.	Инструкция TSM	75
4.2.4.21.	Инструкция ASM.....	76
4.2.4.22.	Инструкция POM.....	77
4.2.4.23.	Инструкция POM (ECDF).....	78
4.2.4.24.	Инструкция ADP	79
4.2.4.25.	Инструкция ZNO	80
4.2.4.26.	Инструкция FBF	81
4.2.4.27.	Инструкция HWC	83
4.2.4.28.	Инструкция HWC (с версии x.86)	84
4.2.4.29.	Инструкция ATR.....	87
4.2.4.30.	Инструкция GRA	88
4.2.4.31.	Инструкция FOPP.....	88
4.2.4.32.	Инструкция FTP.....	90
4.2.4.33.	Инструкция ADG.....	92
4.2.4.34.	Инструкция LAS	97
4.2.4.35.	Инструкция FQF	99
4.2.5.	Секция 3	100
4.2.5.1.	Инструкция PRO.....	101
4.2.5.2.	Инструкция NAS.....	102
4.2.5.3.	Инструкция PAS	102
4.2.5.4.	Инструкция Exxx.....	103
4.2.5.5.	Инструкция NM0	103
4.2.5.6.	Инструкция NM0 (ECDF).....	105
4.2.5.7.	Инструкция NAC	105
4.2.5.8.	Инструкция FEG	106
4.3.	ПРИМЕРЫ ФАЙЛА AXCFIL.....	107
4.3.1.	Пример характеризации управления двумя процессами.....	107
4.3.2.	Пример характеризации управления одним процессом	109
5.	ХАРАКТЕРИЗАЦИЯ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА. ФАЙЛ PGCFIL	112
5.1.	ТЕХНОЛОГИЧЕСКИЙ ПРОЦЕСС	112
5.2.	ОСОБЕННОСТИ ФОРМАТИРОВАННЫХ ФАЙЛОВ	112
5.3.	ЗАГРУЗКА ФАЙЛА PGCFIL.....	113
5.4.	СОДЕРЖАНИЕ ФАЙЛА PGCFIL	113
5.5.	СЕКЦИЯ 1	114
5.5.1.	Инструкция TRI.....	115
5.6.	СЕКЦИЯ 2	116
5.6.1.	Инструкция PRO.....	118
5.6.2.	Инструкция SIM.....	118
5.7.	СЕКЦИЯ 3	119
5.7.1.	Инструкция JCL.....	120
5.8.	СЕКЦИЯ 4	121
5.8.1.	Инструкция PRO.....	121
5.8.2.	Инструкция ASS.....	121
5.8.3.	Инструкция NPL	124
5.8.4.	Инструкция NDD.....	125
5.8.5.	Инструкция PRF	125
5.8.6.	Инструкция FIL	126
5.8.7.	Инструкция FLC	126
5.8.8.	Инструкция STR.....	127
5.8.9.	Инструкция SER	127
5.8.10.	Инструкция CHN	128
5.8.11.	Инструкция SCR	128
5.9.	СЕКЦИЯ 5	129
5.9.1.	Инструкция PRO.....	129
5.9.2.	Инструкция NIP.....	129

5.9.3.	Инструкция DPM.....	130
5.9.4.	Инструкция SMC.....	131
5.9.5.	Инструкция TOF.....	131
5.9.6.	Инструкция GXX.....	132
5.9.7.	Инструкция PRC.....	133
5.9.7.1.	Особенности расчёта значения <i>so_pre</i>	134
5.9.8.	Инструкция SWP.....	135
5.9.9.	Инструкция NAM.....	137
5.9.10.	Инструкция NPD.....	137
5.9.11.	Инструкция G70.....	137
5.9.12.	Инструкция MBR.....	138
5.9.13.	Инструкция TAS.....	138
5.9.14.	Инструкция INU.....	140
5.9.15.	Инструкция CAX (ECDF).....	140
5.9.16.	Инструкция QFD.....	141
5.9.17.	Инструкция TGA.....	141
5.10.	СЕКЦИЯ 6.....	142
5.10.1.	Инструкция PRO.....	142
5.10.2.	Инструкция MAS.....	142
5.10.3.	Инструкция FRO.....	143
5.10.3.1.	Инструкция FRO для УЧПУ NC-301 и NC-302	143
5.10.3.2.	Инструкция FRO для УЧПУ серии NC-XXX.....	143
5.10.4.	Инструкция SSO.....	144
5.10.5.	Инструкция FMO.....	145
5.11.	ПРИМЕР ФАЙЛА PGCFIL для УЧПУ NC-110.....	146
6.	ХАРАКТЕРИЗАЦИЯ ЛОГИКИ. ФАЙЛ IOCFIL	148
6.1.	СЕКЦИЯ 1.....	148
6.1.1.	Инструкция ALM.....	148
6.1.2.	Инструкция INx.....	149
6.1.3.	Инструкция OUx.....	150
6.1.4.	Инструкция CLO.....	150
6.1.5.	Инструкция SPL.....	151
6.2.	СЕКЦИЯ 2.....	152
6.2.1.	Инструкция PRO.....	152
6.2.2.	Инструкция Mxx.....	152
6.2.3.	Инструкция GPS.....	154
6.3.	СЕКЦИЯ 3.....	155
6.3.1.	Инструкция PRO.....	155
6.3.2.	Инструкция ASM.....	156
6.3.3.	Инструкция TAx.....	156
6.3.4.	Инструкция ASx.....	158
6.3.5.	Инструкция UCDA.....	159
6.3.6.	Инструкция ADV.....	159
6.3.6.1.	Управление штурвалом/штурвалами.....	159
6.3.6.2.	Управление двумя штурвалами	161
6.3.7.	Инструкция CWD.....	163
6.3.8.	Инструкция ADC.....	164
6.3.9.	Инструкция DAC.....	164
6.4.	СЕКЦИЯ 4.....	165
6.4.1.	Инструкция Txx.....	166
6.5.	ПРИМЕР ФАЙЛА IOCFIL	166
7.	ИНДИКАЦИЯ ОШИБОК ХАРАКТЕРИЗАЦИИ ФУНКЦИОНАЛЬНОГО УРОВНЯ	168
7.1.	ВИДЕОКАДР ДИАГНОСТИКИ ФАЙЛОВ ХАРАКТЕРИЗАЦИИ ФУНКЦИОНАЛЬНОГО УРОВНЯ	168
7.2.	ОШИБКИ ИНИЦИАЛИЗАЦИИ ФАЙЛОВ ХАРАКТЕРИЗАЦИИ ФУНКЦИОНАЛЬНОГО УРОВНЯ	169
7.2.1.	Синтаксические и семантические ошибки	169
7.2.2.	Ошибки характеристизации осей - файл AXCFIL.....	171
7.2.3.	Ошибки при управлении осями	178
7.2.4.	Ошибки характеристизации процесса - файл PGCFIL.....	178
7.2.5.	Ошибки характеристизации логики - файл IOCFIL.....	180
8.	НАСТРОЙКА СЕТИ	182
8.1.	ОСОБЕННОСТИ НАСТРОЙКИ СЕТИ	182

8.2.	НАСТРОЙКА ПАРАМЕТРОВ СЕТИ В ФАЙЛЕ NETCFG.INI	182
8.2.1.	Инструкция ip	182
8.2.2.	Инструкция mask	183
8.2.3.	Инструкция user	183
8.2.4.	Инструкция password	184
8.2.5.	Инструкция lancard	184
8.2.6.	Инструкция namepc	185
8.2.7.	Инструкция group	185
8.2.8.	Инструкция syspass	186
8.3.	НАСТРОЙКА ПАРАМЕТРОВ СЕТИ В ФАЙЛЕ USER.INI	186
8.4.	НАСТРОЙКА СЕТЕВЫХ РЕСУРСОВ В ФАЙЛЕ ADMIN.INI	187
8.5.	НАСТРОЙКА ВНЕШНИХ СЕТЕВЫХ РЕСУРСОВ В ФАЙЛЕ SERVERS.INI	188
8.6.	НАСТРОЙКА ДОСТУПА К ДРУГОЙ СЕТИ В ФАЙЛЕ GATECFG.INI	189
9.	КОДЫ СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ РТА И ГТА	191
9.1.	НАЗНАЧЕНИЕ КОДОВ РТА И ГТА	191
9.1.1.	Код РТА.....	191
9.1.2.	Код ГТА	192
	ПРИЛОЖЕНИЕ А Особенности управления ошей Приложение к инструкции FQF.....	194
	ПРИЛОЖЕНИЕ В Особенности управления ошей.....	198

1. ОБЩИЕ СВЕДЕНИЯ О ХАРАКТЕРИЗАЦИИ

1.1. Назначение характеристики

Каждое УЧПУ серии **NC-XXX** представляет собой промышленный компьютер, имеющий набор периферийных модулей для управления оборудованием. Для подготовки такого промышленного компьютера к работе в конкретной системе необходимо выполнить установку параметров и характеристик управляемого оборудования, а также аппаратных и программных модулей самого промышленного компьютера, т.е. необходимо выполнить характеристику, что позволит использовать устройство как УЧПУ. Характеристика осуществляется через запись указанных параметров в файлы характеристики, создание которых описано в данном документе.

Файлы характеристики содержат параметры и характеристики, значения которых специфицируют конфигурацию УЧПУ конкретного пользователя. Из этих файлов УЧПУ получает всю информацию, требуемую для функционирования **SW**, управляющего технологическим процессом обработки деталей.

После завершения процедуры характеристики УЧПУ ещё не способно управлять конкретным оборудованием. Для этого необходимо создать программу управления вспомогательными механизмами станка, которая называется программой логики управляемого оборудования (ПЛ). ПЛ разрабатывается на языке PLC. Описание языка PLC, методы отладки, компилирования и активизации ПЛ приведены в документе «Программирование интерфейса PLC».

1.2. Уровни характеристики

1.2.1. Характеристика УЧПУ имеет два уровня:

- системный;
- функциональный.

Системный уровень характеристики является приоритетным по отношению к функциональному уровню, поэтому он должен быть выполнен в первую очередь. Системный уровень характеристики выполняет установку параметров, определяющих доступ к физически существующим рабочим каталогам, расположенным на любом из используемых дисковых устройств памяти: **FLASH** диске, **HDD**, **HDD** сервера (при работе УЧПУ в локальной сети) или **FDD**, а также определяет список рабочих файлов функционального назначения для УЧПУ.

Функциональный уровень характеристики устанавливает параметры, которые задают конкретные функции (персонализируют) УЧПУ, и этим ориентируют его на управление определённым технологическим процессом, т. е. специфицируют УЧПУ.

Характеристика выполняется следующими файлами:

- 1) **FCSYS**;
- 2) **AXCFIL**;

3) **PGCFIL**;

4) **IOCFIL**.

Каждый из этих файлов используется при конкретном уровне характеристики. На рисунке 1.1 показана связь между уровнями характеристики и соответствующими им файлами.



Рисунок 1.1 - Связь между уровнями и файлами характеристики

1.3. файлы характеристики

1.3.1. Назначение файлов характеристики

Назначение файлов характеристики:

- системный уровень:

FCRSYS - содержит информацию об именах устройств памяти на **FLASH, HDD, FDD** или **HDD** сервера, список рабочих файлов характеристики функционального назначения в УЧПУ для объекта управления, а также данные для определения дисковых устройств, подключённых к УЧПУ;

- функциональный уровень:

AXCFIL - содержит параметры управления осями;

PGCFIL - содержит информацию о характеристиках управления технологическим процессом обработки деталей для объекта управления;

IOCFIL - содержит параметры ПЛ, позволяющие персонализировать УЧПУ для конкретного применения. ПЛ является промежуточным звеном между программным обеспечением и объектом управления.

Все файлы характеристики читаются и интерпретируются **SW** УЧПУ по его включению (при начальной загрузке). Если появилась необходимость модифицировать какой-нибудь из файлов, то модифицированный файл интерпретируется **SW** УЧПУ только после полной перезагрузки программного обеспечения тремя клавишам **Ctrl+Alt+Del**, или выключения и последующего включения УЧПУ. По включению УЧПУ файлы характеристики тестируются, в случае обнаружения ошибок на дисплее визуализируются соответствующие сообщения. Визуализация ошибок осуществляется на двух видеокадрах диагностики:

- на первом - ошибки файла **FCRSYS**;
- на втором - ошибки файлов: **AXCFIL**, **PGCFIL**, **IOCFIL**.

1.3.2. Структура файлов характеристики

Все файлы характеристики имеют одинаковую файловую структуру независимо от типа файла. Однако структура записей в файлах отличается в зависимости от содержащейся в них информации. На рисунке 1.2 представлена структура файла характеристики.

Файл состоит из определённого количества секций (***1÷*n**). Секции включают в себя одну или несколько инструкций. Инструкции файлов функционального уровня имеют одну и ту же структуру записей, а записи инструкций файла **FCRSYS** не имеют общей структуры и изменяются в зависимости от секции, в которую они входят. Структура и формат записей приводятся в данном документе при описании конкретного файла.

Каждая запись содержит информацию о компонентах аппаратных или программных модулей УЧПУ. Для удобства чтения информации перед записями в содержание файлов можно вводить комментарии. Комментарий состоит из строки алфавитно-цифровых символов, первым знаком которой должен быть символ «;». Каждая секция файла начинается символом «*», после которого записывается её порядковый номер. Последняя секция также заканчивается символом «*» (только **FCRSYS**).

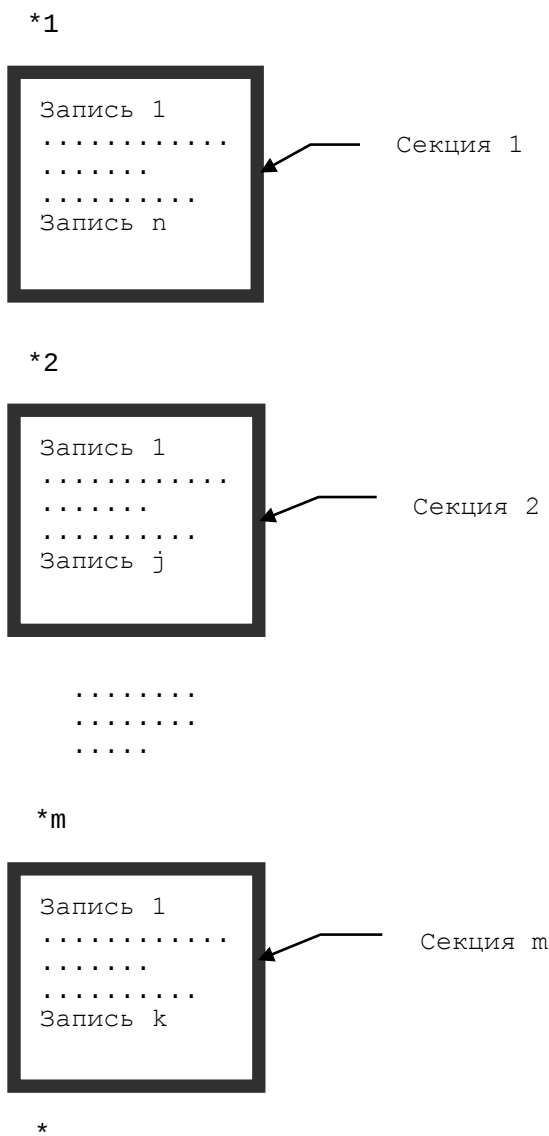


Рисунок 1.2 - Структура файла характеристики

2. КОДИРОВАНИЕ, УСТАНОВКА, КОПИРОВАНИЕ И ВОССТАНОВЛЕНИЕ SW УЧПУ

2.1. Кодирование версий SW УЧПУ

SW УЧПУ имеет варианты исполнения. Версия SW подлежит согласованию с изготовителем УЧПУ при оформлении заказа.

Версия SW УЧПУ типа NC имеет обозначение:

Z(xxx).XX[.Y] «extension» ,

где:

Z(xxx) - цифровой код типа УЧПУ:

- 1** - NC-100;
- 2** - NC-110;
- 3** - NC-200, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230;
- 4** - NC-310, NC-301, NC-302

(xxx) - обозначение особенности типа УЧПУ, например: **(230E)**, которое обозначает SW, используемое для УЧПУ **NC-230** с портом SSB.

XX - порядковый номер версии SW;

Y - номер изменения версии (необязательный);

«extension» - буквенный код, определяющий набор встраиваемых функций:

- **Р** - русифицированные версии;
- **Е** - версии на английском языке;
- **ПД** - переключение одного канала датчика между несколькими осями.

2.2. Установка SW УЧПУ на FLASH DISK

В фирме-изготовителе SW УЧПУ устанавливают на **FLASH DISK** типа **DISK ON MODULE (DOM)**.

Для установки SW УЧПУ, если диском «C:» определён **HDD** или **FLASH DISK** типа **DOM**, необходимо выполнить:

- 1) установки в **SETUP** для определения диска:

STANDART CMOS SETUP			
HARD DISKS	TYPE		MODE
Primary Master	AUTO		AUTO
Primary Slave	AUTO		AUTO

- 2) конфигурирование жёсткого диска или **DOM** для работы в **MS-DOS** программой **fdisk.exe**;
- 3) форматирование жёсткого диска или **DOM** программой **format.exe**;
- 4) установку файлов **MS-DOS**;
- 5) установку файлов УЧПУ:

- характеристику системного уровня (файл **FCRSYS**);
- характеристику функционального уровня (файлы **AXCFIL**, **PGCFIL**, **IOCFIL**);
- программу логики станка (ПЛ).

В обязательный комплект поставки УЧПУ входит **USB-FLASH «СЕРВОДИСК»** с резервной копией файлов **FLASH DISK**, установленных в ООО «Балт-Систем», в архиве **FLASH.RAR**. Архив **FLASH.RAR** может состоять из нескольких томов-архивов с именем **FLASH**. Используя устройство **USB-FLASH «СЕРВОДИСК»** пользователь может восстановить исходное **SW**, на **FLASH DISK** УЧПУ, если такая необходимость возникнет. Комплект для восстановления исходного **SW** включает:

- **USB-FLASH «СЕРВОДИСК»** с архивом **FLASH** в каталоге **BACKUP**.

На устройстве **USB-FLASH «СЕРВОДИСК»** записаны программы, которые помогут пользователю создать рабочие дискеты с резервной копией архивного файла **FLASH** после привязки УЧПУ к станку, а также восстановить данные на **FLASH DISK** с резервной копии в случае необходимости.

Фирма-изготовитель гарантирует на момент поставки отсутствие вирусов в УЧПУ и на других носителях информации, поставляемых с УЧПУ.

ВНИМАНИЕ! При работе с внешними носителями файлов необходимо быть уверенными, что они не содержат вирусов.

2.3. Создание устройства **USB-FLASH «СЕРВОДИСК»**.

В комплект УЧПУ входит устройство **USB-FLASH «СЕРВОДИСК»**. Назначение этого устройства описано в п.2.4.

Если устройство **USB-FLASH «СЕРВОДИСК»** утеряно, то его можно создать самостоятельно. Для создания устройства **«СЕРВОДИСК»** необходимо обратиться к ООО «Балт-Систем» и получить архив **USB_SERVODISK_NC-XXX_dos710_2013_V1.2.rar** или **USB_SERVODISK_UP1G_GRUB_2014_V1.0.rar**, распаковать его; пароль: 123456 и по инструкции "Создание устройства **USB-FLASH СЕРВОДИСК.doc**" создать **«СЕРВОДИСК»**.

После распаковки архива **USB_SERVODISK_NC-XXX_dos710_2013_V1.2.rar** полученные файлы и каталоги рекомендуется копировать на **USB-flash «СЕРВОДИСК»**, если его объем менее 512Мб, и он будет использован в УЧПУ, материнские платы которых имеют интерфейс V1.1.

После распаковки архива **USB_SERVODISK_UP1G_GRUB_2014_V1.0.rar** полученные файлы и каталоги рекомендуется копировать на USB-flash «СЕРВОДИСК», если его объем более 512Мб, и он будет использован в УЧПУ, материнские платы которых имеют интерфейс V2.0.

2.4. Создание архива файлов DOM

Основное назначение устройства USB-flash «СЕРВОДИСК» - это создание резервного архива файлов, расположенных на **DOM** и восстановление файлов на **DOM**. Кроме этого устройства USB-flash «СЕРВОДИСК» позволяет работать с файлами пользователя, например, выполнять обновление версии ПрО в УЧПУ, копировать и изменять управляющие программы и другие файлы, а также тестировать **DOM**.

Местоположением архива файлов **DOM** на устройстве USB-flash «СЕРВОДИСК» является каталог BACKUP.

Архив файлов **DOM** в каталоге **BACKUP** может состоять из нескольких томов с именами: **FLASH.RAR**, **FLASH.R00**, **FLASH.R01** и т.д. с максимальным объемом 1.44Мб.

ВНИМАНИЕ !

1. Создание архива **DOM** должна выполнять организация, выполняющая подключение УЧПУ к станку, после проведения:
 - пуско-наладочных работ,
 - обновления версии ПрО,
 - изменения файлов характеристики или ПЛ,
 - изменения других файлов, используемых для работы станка.

При этом устройство USB-flash «СЕРВОДИСК» с архивом файлов **DOM** должно быть передано заказчику станка.

Организация, выполняющая подключение УЧПУ к станку, должна хранить копии архивов с файлами **DOM** в течение всего гарантийного срока, данного на станок.

2. Создание архива **DOM** должен выполнять заказчик станка самостоятельно в случае, если он выполнил в УЧПУ изменения, перечисленные в п.1.
3. Заказчик станка должен надежно хранить архив файлов **DOM**. Устройство USB-flash «СЕРВОДИСК» не является надежным местом хранения файлов.
4. До выполнения любых действий с архивом файлов **DOM** требуется проверить их целостность, а в случае выполнения действий по восстановлению файлов **DOM** дополнительно убедиться, что архив содержит информацию для данного станка.

2.4.1. Особенности использования USB-flash «СЕРВОДИСК»

Использование устройства USB-flash «СЕРВОДИСК» будет доступно при условии, если загрузка УЧПУ выполняется с этого же устройства. Для этого необходимо выполнить следующие действия:

1. Загрузите УЧПУ в режиме SETUP и выполните установки для загрузки УЧПУ с соответствующего дискового устройства;
2. Установите USB-flash «СЕРВОДИСК» в разъем «USB1» или «USB2» на модуле «CPU»;
3. Перезагрузите УЧПУ;

ВНИМАНИЕ! Если после перезапуска УЧПУ на экране появляется строка «**Press F11 for BBS POPUP**», то сразу нажмите клавишу «**F11**», и после вывода на экран меню выбора устройства для загрузки выберите загрузку с USB устройства.

2.4.1. Меню

Устройство «СЕРВОДИСК» предназначено для работы вне программы управления станком и имеет следующее меню:

1. Создать архив *DiskOnModule (DOM)* на СЕРВОДИСК (каталог *BACKUP*)
2. Работать с файлами (*Norton Commander*)
3. Антивирус для DOS
4. Проверка СЕРВОДИСК (*Scandisk*)
5. Проверка DOM (*Scandisk*)
6. FDISK для DOM (*HDD, CompactFlash*)
7. Форматировать *DOM* и ВОССТАНОВИТЬ из СЕРВОДИСК (каталог *BACKUP*)
8. Язык меню СЕРВОДИСК (*Language menu SERVODISK*) (переход в меню выбора языка)

Меню выбора языка позволяет выбрать язык основного меню «СЕРВОДИСК», и имеет следующие варианты для выбора:

1. Русский (*Russian*)
2. Английский (*English*)
3. Основное меню

Примечание.

- Программы **Scandisk**, **Norton Commander**, **Fdisk** в документации к УЧПУ не описываются.
- Программа **DEBUG** описана в приложении «Руководства оператора».

Выбранный номер пункта в меню определяет загружаемую программу и дальнейшие действия с УЧПУ.

ВНИМАНИЕ! Чтобы не потерять параметры управления станком необходимо до выбора п.7 выполнить следующие действия:

- а) Выполнить п.1 – «Создать архив *DiskOnModule (DOM)* на СЕРВОДИСК (каталог *BACKUP*)»;
- б) Скопировать все архивные файлы из каталога **BACKUP «СЕРВОДИСК»** в офисный компьютер в каталог с номером данного станка.

1. Пункт меню «Создать архив *DiskOnModule (DOM)* на СЕРВОДИСК (каталог *BACKUP*)» предназначен для сохранения

файлов устройства **DOM** в архиве **FLASH** в каталоге **BACKUP** на «**СЕРВОДИСК**». Полученный архив будет использован при выполнении п.7. «**Форматировать DOM и ВОССТАНОВИТЬ из СЕРВОДИСК (каталог BACKUP)**».

Примечание. Перед созданием новых архивных томов **FLASH.R??** в каталоге **C:\BACKUP** существующие архивные тома будут сохранены под именами **FLASH.O??**. Эти тома можно будет использовать для восстановления **DOM** предыдущей версией архива, если файлы **FLASH.R??** оказались поломанными. Перед использованием файлов **FLASH.O??** для восстановления с помощью п.8 – «**Форматировать DOM и ВОССТАНОВИТЬ из СЕРВОДИСК (каталог BACKUP)**» их необходимо переименовать в имена **FLASH.R??** с помощью программы **Norton Commander**.

После успешного выполнения данного пункта меню на экране УЧПУ должно быть следующее сообщение:

```
*****
*
*      РЕЗЕРВНАЯ КОПИЯ устройства Disk On Module (DOM) ГОТОВА.
*
*
*
*
*
*****
* Для работы станка с DOM:
*
*      - выключите УЧПУ;
*
*      - выньте загрузочный СЕРВОДИСК;
*
*      - включите УЧПУ.
*
*
*****
* - Для продолжения работы с СЕРВОДИСК, нажмите любую клавишу.
*
*****
```

ВНИМАНИЕ! Для гарантии сохранности многотомного архива **FLASH** скопируйте все его тома на персональный компьютер в каталог данного станка.

2. Пункт 2 «Работа с файлами ...» позволяет выполнить работу с файлами в **MS-DOS** в файловой оболочке **Norton Commander**.
3. Пункты 3 «Антивирус для DOS».
4. Пункты 4 «Проверка СЕРВОДИСК (Scandisk)» выполняется по желанию оператора для проверки файловой структуры **USB-FLASH** и его рабочей поверхности.
5. Пункты 5 «Проверка DOM (Scandisk)» выполняется по желанию оператора для проверки файловой структуры **DOM** и его рабочей поверхности.
6. Пункты 6 «**FDISK для DOM (HDD, CompactFlash)**» обычно выполняется только для новых устройств памяти **DOM/HDD/CompactFlash**. Основное назначение, выполняемое утилитой **fdisk**, состоит в разделении дискового пространства устройств памяти на логические диски и формирование в каждом из них файловой структуры. Для 32-х разрядной **MS-DOS 7.10** рекомендуется устанавливать **FAT32**.
7. Пункт 7 «**Форматировать DOM и ВОССТАНОВИТЬ ФАЙЛЫ из СЕРВОДИСК**» позволяет восстановить файлы устройства **DOM**. Данные для восстановления должны находиться в архиве

FLASH.RAR (архив может иметь несколько томов **FLASH.RAR**, **FLASH.R00** и т.д.), расположенном в каталоге **BACKUP** на «СЕРВОДИСК».

ВНИМАНИЕ! Не выбирайте в меню пункт 7, если каталог **BACKUP** пустой, или в этом каталоге архив **FLASH.RAR** не соответствует, восстанавливаемому станку.

После успешного выполнения данного пункта меню на экране УЧПУ должно быть следующее сообщение:

```
*****
**
*                               Восстановление      устройства      DOM      завершено.
*                               *
*
*****
*                               -      Для      работы      станка      с      DOM:
*                               *
*                               -      выключите      УЧПУ;
*                               *
*                               -      выньте      загрузочный      СЕРВОДИСК
*                               *
*                               -      включите      УЧПУ.
*                               *
*****
*                               Для продолжения работы с СЕРВОДИСК, нажмите любую клавишу.
*                               *
*****
**
```

8. Пункт 8 «Язык меню СЕРВОДИСК (Language menu SERVODISK) (переход в меню выбора языка)» позволяет изменить язык меню «СЕРВОДИСК».

3. ХАРАКТЕРИЗАЦИЯ СИСТЕМНОГО УРОВНЯ

3.1. Сведения о создании и загрузке файла FCRSYS

Файл **FCRSYS** должен быть создан при первом включении УЧПУ. Этот файл можно создать повторно или отредактировать любым внешним редактором **ASCII** или встроенным редактором УЧПУ. Создать новый файл **FCRSYS** или открыть на редактирование уже существующий файл **FCRSYS** можно в режиме «**COMMAND**» командой **EDI,FCRSYS/MP0**, а также используя меню режима «**COMMAND**»: **F2 (MODIF) -> F1 (EDI) -> F7 (MP0) ->** установить в списке файлов курсор на имя файла **FCRSYS** или набрать в строке редактирования имя **FCRSYS** и завершить ввод команды по клавише **ENTER**.

Примечание. Порядок набора команды, используя функциональные клавиши меню режима «**COMMAND**», записывается от верхнего уровня меню, на который можно выйти, нажимая несколько раз клавишу **ESC**.

Имя файла **FCRSYS** зарезервировано в **SW** и не должно изменяться пользователем. Файл **FCRSYS** должен быть расположен в текущем каталоге, из которого загружается файл **VIDMEM.EXE** или **CNC.EXE**. Обычно эти файлы и файл **FCRSYS** расположены в каталоге **C:\CNC32\MP0 (C:\CNC\MP0)**.

По окончании редактирования файла для его активизации нажмите одновременно три клавиши **Ctrl+Alt+Del** или выключите и снова включите УЧПУ. При запуске **SW** выполняется диагностика введенного файла, результаты которой визуализируются на дисплее.

В случае обнаружения синтаксических ошибок или ошибок формата записей необходимо произвести корректировку файла либо редактором **SW** (см. «Руководство оператора»), либо любым другим текстовым редактором.

Примечание. Перезапуск УЧПУ по клавишам **Ctrl+Alt+Del** будет выполнен, если реле «**SPEPN**» выключено (см. «Руководство по программированию интерфейса PLC»).

3.2. Стандартный файл FCRSYS

При создании персонализированного файла **FCRSYS** можно воспользоваться стандартным файлом. Этот стандартный файл необходимо адаптировать (модифицировать) для использования его в конкретном УЧПУ.

Стандартный файл **FCRSYS** имеет вид:

```
*1
MP0=C:/CNC32/MP0
MP1=C:/CNC32/MP1
MP2=C:/CNC32/MP2
MP3=C:/CNC32/MP3
MP4=C:/CNC32/MP4
MP5=C:/CNC32/MP5
MP6=A:
```

```

*2
AXCONF, AXCFIL/MP0
IOCONF, IOCFIL/MP0
PGCONF, PGCFIL/MP0
FILCMD, FILCMD/MP0
FILMAS, FILMOV/MP0
FILMS1, RUMES1/MP0
FILMS2, RUMES2/MP0
FILMS3, RUMES3/MP0
FILMS4, RUMES4/MP0
FILMS5, RUMES5/MP0
FORMAT, FORMAT/MP0
*4
USB=CRD, CRD, ,
*
```

3.3. Рабочий файл FCRSYS

3.3.1. Инициализация рабочего файла FCRSYS

Рабочий файл **FCRSYS** состоит из четырёх секций, каждая из которых содержит информацию об именах и размещении рабочих каталогов, функциональных файлов и файлов сообщений УЧПУ. Вся информация, содержащаяся в файле, диагностируется, интерпретируется и активизируется (в случае отсутствия ошибок) по включению УЧПУ или при перезагрузке программы тремя клавишами **Ctrl+Alt+Del**.

Если файл **FCRSYS** отсутствует в УЧПУ, или обнаружены ошибки при его диагностике, работа УЧПУ останавливается, и на экране в течение 2-х секунд появляется текст: ***ОСТАНОВ ИНИЦИАЛИЗАЦИИ*** и сообщение об ошибке: **«Сообщение 1__21»**. После этого в системе выполняется автоматический переход в режим **«COMMAND»**. Создание файла **FCRSYS** или его редактирование выполняется командой: **EDI,FCRSYS/MP0**.

3.3.2. Секция 1

Секция 1 объявляет соответствие между логическим именем пути и путём доступа к физически существующему каталогу **DOS**. Длина пути к физически существующему каталогу **DOS** составляет 255 символов.

УЧПУ использует логические имена пути **MP0, MP1, MP2, MP3, MP4, MP5, MP6**, которым могут соответствовать любые каталоги на любом дисковом устройстве, подключенном к этому УЧПУ.

Формат записи секции 1:

```
MPx=disk_name [\catalog_name\...\ catalog_name] ,
```

или:

```
MPx=\\resource_name [\catalog_name\...\ catalog_name] ,
```

где:

MPx - логическое имя пути (**x=0,1,2,3,4,5,6**), используемое для доступа к файлам, расположенным на дисковом устройстве;

Disk_name - имя дискового устройства (**C:**, **A:** или **B:**, **D:** и т. д.);
 Значение «Диск» для логических имен пути **MP0** и **MP1** рекомендуется всегда записывать именем диска **C:**, например:
MP0=C:\CNC32\MP0
MP1=C:\CNC32\MP1

Resource_name - имя сетевого ресурса или его IP-адрес;
Catalog_name - имя каталога на дисковом устройстве или на сетевом ресурсе;
**** - символ разделителя между именами. Разделителем имен может быть символ «\».

Пример

Запись секции 1:

```
*1
MP0=C:\CNC32\MP0
MP1=C:\CNC32\MP1
MP2=C:\CNC32\MP2
MP3=C:\CNC32\MP3
MP4=C:\CNC32\MP4
MP5=D:
MP6=A:
```

ВНИМАНИЕ !

1. Вся информация в файле **FCRSYS** должна быть набрана прописными буквами.
2. Символ разделителя «\» в конце записей в секции 1 не ставится.

Определение логических имён пути **MP0** и **MP1** в секции 1 файла **FCRSYS** является обязательным.

В каталоге, путь к которому записан в строке **MP0=...**, рекомендуется размещать файлы, указанные в секции 2 этого руководства и руководства «Программирование интерфейса PLC», и не рекомендуется размещать файлы управляющих программ для обработки деталей.

Если в секции 1 обнаружены синтаксические ошибки, на дисплее визуализируется информация: «Сообщение 1__...».

3.3.3. Секция 2

Секция 2 содержит логические имена файлов функционального назначения, системных сообщений и технологического процесса. В памяти УЧПУ содержится таблица постоянных логических имён (максимум 15 имён), которые пользователь не может изменять. Через запятую справа от постоянного логического имени файла пользователь должен записать имя своего файла и место его размещения. Соответствие между постоянными логическими именами файлов **SW** УЧПУ и именами файлов, создаваемых пользователем, и их функциональным назначением представлено в таблице 3.1.

Таблица 3.1 - Постоянные логические имена файлов пользователя

Постоянное логическое имя файла	Имя файла пользователя	Назначение файла	Команда создания файла
FILMS1	RUMES1	Файл сообщений RUMES1 содержит ошибки, визуализируемые при диагностике файла FCRSYS и ошибки JCL , индицируемые в режиме « COMMAND ».	EDI
FILMS2	RUMES2	Файл сообщений RUMES2 содержит сообщения, визуализируемые при работе в режиме « COMMAND ».	EDI
FILMS3	RUMES3	Файл сообщений RUMES3 содержит ошибки ввода/вывода в режиме « COMMAND » и в режиме « УПРАВЛЕНИЕ СТАНКОМ ».	EDI
FILMS4	RUMES4	Файл сообщений RUMES4 содержит сообщения и ошибки программирования в режиме « УПРАВЛЕНИЕ СТАНКОМ ».	EDI
FILMS5	MESSAG	Файл сообщений RUMES5 содержит сообщения, визуализируемые из ПЛ.	EDI
FORMAT	FORMAT	Файл FORMAT предназначен для записи в него строки формата, используемой для создания форматированных файлов: начальных точек, корректоров и срока службы инструмента.	EDI
AXCONF	AXCFIL	Файл AXCFIL содержит параметры управления осями	EDI
PGCONF	PGCFIL	Файл PGCFIL содержит параметры технологического процесса, выполняемого на станке	EDI
IOCONF	IOCFIL	Файл IOCFIL содержит параметры для ПЛ станка	EDI
FILCMD	FILCMD	Файл FILCMD содержит имена УП и команды, активизируемые по запросу из ПЛ	EDI
FILMAS	FILMOV	Файл FILMOV содержит кадры, выполняемые по запросу из ПЛ (в соответствии с документом «Программирование интерфейса PLC»)	EDI

Примечание.

1. Определение логических имён **AXCONF**, **PGCONF**, **IOCONF** и **FORMAT** обязательно в секции 2.
2. Если файлы сообщений не введены пользователем в память УЧПУ, или их имена не определены в секции 2 с логическими именами **FILMS1** – **FILMS5**, или текст сообщения отсутствует в файле на нужной строке, то сообщения об ошибках будут визуализироваться на дисплее в следующем виде:

Сообщение N_XXX ,

где:

N - номер файла сообщений об ошибках;

XXX - номер ошибки (номер строки в файле сообщений).

Пример.

Сообщение 1__02

Формат записи секции 2:

log_name, file_name/MPx ,

где:

log_name - постоянное логическое имя файла, используемое **SW** УЧПУ; имя состоит из шести алфавитно-цифровых знаков, первый из которых является буквой;

file_name - имя файла, присвоенное пользователем; имя состоит из шести алфавитно-цифровых знаков, первый из которых является буквой;

MPx - логическое имя пути (**x=0÷6**) к файлу, имя которого записано в параметре **file_name**.

Пример

Запись секции 2:

```
*2
AXCONF,AXCFIL/MP0
IOCONF,IOCFIL/MP0
PGCONF,PGCFIL/MP0
FORMAT,FORMAT/MP0
FILCMD,FILCMD/MP0
FILMAS,FILMOV/MP0
FILMS1,RUMES1/MP0
FILMS2,RUMES2/MP0
FILMS3,RUMES3/MP0
FILMS4,RUMES4/MP0
FILMS5,MESSAG/MP0
```

3.3.4. Секция 3

Секция 3 объявляет логические имена жёстких дисков (**HDD**) УЧПУ. **SW** УЧПУ будет выполнять периодические обращения к объявленным именам дисков во избежание их перехода в режим уменьшения потребления электроэнергии.

Формат записи секции 3:

HDD=[log_name_HDD1][log_name_HDD2][log_name_HDD3] ,

где:

log_name(_HDD1,_HDD2,_HDD3) - имена HDD.

Примеры

HDD=C

HDD=CDF

3.3.5. Секция 4

Секция 4 предназначена для определения дисковых и **USB** устройств, драйверы которых должны использоваться при работе УЧПУ.

Данная секция может быть записана для версий **SW**, начиная с номера **Z.73** (**Z = 2, 3, 4**), например: 3.73P или 2.73РИВ.

3.3.5.1. Определение подключённых устройств FDD.

Определение подключённых к УЧПУ устройств **FDD** (накопителей на гибких магнитных дискетах) выполняется в инструкции **FLP**.

Формат:

FLP=log_name_FDD1,log_name_FDD2 ,

где:

log_name_FDD1 - логическое имя **FDD1** может быть определено именами «A» или «B»;

log_name_FDD2 - логическое имя **FDD2** может быть определено именами «A» или «B».

Примечание. Если инструкция **FLP** не записана в данной секции, то по умолчанию можно использовать устройство **FDD** только под именем «A», например, в секции 1 можно записать инструкцию: **MP6=A:**

Примеры

- 1) **FLP=A,B** или **FLP=B,A** - в УЧПУ подключены два устройства **FDD** с именами «A» и «B». В этом случае в секции 1 можно записать две инструкции: **MP5=A:** и/или **MP6=B:**.
- 2) **FLP=,B** или **FLP=B,** - в УЧПУ подключено устройство **FDD** с именем «B». В этом случае в секции 1 можно записать инструкцию: **MP6=B:**.
- 3) **FLP=,** или **FLP=,** - в УЧПУ не подключены устройства **FDD**. В этом случае записи в секции 1 **MP5=A:** и/или **MP6=B:** - не активны.

3.3.5.2. Определение подключённых устройств HDD

Определение дисков, подключённых к УЧПУ по интерфейсам **IDE** или **SATA** в режиме **Compatible**.

Формат:

IDE=PM,PS,SM,SS ,

где:

PM,PS,SM,SS могут принимать значения «HDD», «CD» или быть пустыми.

- «HDD» - имя, активирующее работу драйвера для жесткого диска, подключенного к **IDE**-порту.
- «CD» - имя, активирующее работу драйвера для CD-ROM/DVD-ROM, подключенного к **IDE**-порту.

ПРИМЕЧАНИЯ

- 1) Запись значения «HDD» для параметра **PM** активирует в системе работу с диском, обнаруженного в **BIOS** на месте **Primary Master (IDE Master)**.
- 2) **PM** не может иметь значение «CD».
- 3) Если инструкция **IDE** не записана, то по умолчанию активируются следующие значения: **IDE=HDD,,,**
- 4) Запись значения «HDD» или «CD» для параметра **PS** активирует в системе работу с диском, обнаруженного в **BIOS** на месте **Primary Slave (IDE Slave)**.
- 5) Запись значения «HDD» или «CD» для параметров **SM** и **SS** активирует в системе работу с дисками, обнаруженными в **BIOS** на месте **Secondary Master** или **Secondary Slave**. Эти параметры обычно не используются в УЧПУ и могут иметь значения при установке эмулятора УЧПУ на ПК.
- 6) Если имя «HDD» или «CD» записано с ошибкой, то это устройство не активизируется, например: **IDE=HDD,DD**. Значение для **PS** здесь записано с ошибкой и драйвер диска на месте **Primary Slave** не активизируется.
- 7) Запятые после последнего указанного имени драйвера можно не записывать, например: **IDE=,HDD**

Пример

IDE=HDD,HDD,CD - в УЧПУ подключены три устройства:

```
PM = HDD (Primary Master);
PS = HDD (Primary Slave);
SM = CD  (Secondary Master).
```

3.3.5.3. Определение подключённого устройства Disk On Chip.

Определение подключённого к УЧПУ устройства **Disk On Chip (DOC)** выполняется в инструкции **DOC**.

Формат:

DOC=DOC ,

где:

DOC - инструкция **DOC=DOC** записывается только в том случае, если **BIOS** процессорной платы при его загрузке определил наличие устройства **Disk On Chip**.

ВНИМАНИЕ! АДРЕСНОЕ ОКНО ДЛЯ РАБОТЫ **DOC** УСТАНОВЛИВАЕТСЯ ПЕРЕМЫЧКАМИ НА ПРОЦЕССОРНОЙ ПЛАТЕ И ДЛЯ ВЕРСИЙ **SW** ДОЛЖНО ИМЕТЬ АДРЕС **0: D000**.

Примечание. Если слово **DOC** записано с ошибкой или отсутствует, то это устройство не активизируется, например: **DOC=DOS**.

Пример

DOC=DOC - в УЧПУ подключено устройство **Disk On Chip**.

3.3.5.4. Определение драйверов поддержки USB-устройств.

Определение драйвера поддержки USB-устройства для каждого USB порта УЧПУ, выполняется в инструкции **USB**.

Формат:

USB=USB_driv_name,USB_driv_name,USB_driv_name,USB_driv_name,

где:

USB_driv_name - имя драйвера для каждого **USB** порта. **USB_driv_name** может иметь следующие имена:

- CD** - имя драйвера для **USB**-порта, к которому подключается устройство **CD-ROM**;
- CRD** - имя универсального драйвера **Card Reader** для **USB**-порта (драйвер поддерживает работу USB порта со следующими устройствами: **USB-FLASH**, **USB-FDD** или **USB-HDD**);
- FLOPPY** - имя драйвера для **USB**-порта, к которому подключается устройство **USB-FDD**;
- FLASH** - имя драйвера для **USB**-порта, к которому подключается устройство **USB-FLASH**.

ПРИМЕЧАНИЯ

1. Если инструкция **USB** не записана в данной секции, то по умолчанию активизируется устройство, определённое именем **CRD**.
2. Если инструкция **USB** записана без указания имён **USB**-устройств, то порт **USB** отключён. Пример: **USB=**
3. Если в именах устройств (**CD**, **CRD**, **FLOPPY** или **FLASH**) допущена ошибка, то устройство, имя которого указано неправильно, не активизируется. Пример: **USB=CD,CRD,FLOPY,FLASH**. Значение **устройство_3** здесь записано с ошибкой и поэтому не активизируется.
4. Запятые после последнего указанного устройства можно не записывать. Пример: **USB=,HDD**

Для УЧПУ, имеющего только один **USB**-порт, инструкция **USB** обычно не записывается. При этом активизируется устройство **CRD**. Это устройство поддерживает работу устройств **CD**, **CRD**, **FLOPPY** и **FLASH**, подключаемых к **USB**-порту по очереди.

Для ПК, на котором установлен программный эмулятор и который имеет до 4-х **USB**-портов для каждого порта, можно назначить драйвер **USB**-устройства.

Пример

USB=CD,CRD,FLOPPY,FLASH

Определение подключённых устройств выполняется по умолчанию в том случае, если секция 4 не имеет инструкций или отсутствует. Список устройств, активных по умолчанию:

- **A;**
- **C (HDD Primary Master);**
- **USB (CRD Card Reader).**

4. ХАРАКТЕРИЗАЦИЯ ОСЕЙ. ФАЙЛ АХСF1L

4.1. Общие сведения о файле АХСF1L

4.1.1. Правила характеристики осей

SW управления осями, размещаемое в УЧПУ, выполняет две основных задачи:

- интерполяцию осей;
- управление приводами осей.

Задача интерполятора заключается в расчете координат движения на основе запрограммированных параметров. В системе присутствует четыре типа интерполяции:

- линейная;
- круговая;
- винтовая;
- сплайновая.

Задача управления приводом осей заключается в создании безошибочного движения всех непрерывных координатных осей по профилю, рассчитанному интерполятором этих осей, а также обеспечение движения осей от «точки к точке», рассчитанному для них отдельными интерполяторами.

Эти задачи реализованы на одном модуле **CPU**. При характеристике осей следует соблюдать следующие правила:

- 1) оси, интерполируемые совместно (непрерывные скоординированные оси), должны иметь один интерполятор;
- 2) шпиндель и непрерывные оси должны иметь один и тот же интерполятор;
- 3) ось «от точки к точке» должна быть объявлена в отдельном интерполяторе;
- 4) интерполятор непрерывных скоординированных осей каждого процесса может управлять максимально 8-ю осями. Обычно реальное количество осей в интерполяторе устанавливается не более количества доступных к использованию аппаратных разъемов датчиков и каналов ЦАП (ЦИП):
 - **NC-110 и NC-310:**
 - 7-ю осями + шпиндель с датчиком;
 - 8-ю осями + шпиндель без датчика;
 - **NC-230:**
 - 4-мя осями + шпиндель с датчиком;
 - 5-ю осями + шпиндель без датчика;
 - **NC-200, NC-210, NC-220, NC-301 и NC-302:**
 - 3-мя осями + шпиндель с датчиком;
 - 4-мя осями + шпиндель без датчика;
 - **NC-201, NC-201M и NC-202:**
 - 2-мя осями + шпиндель с датчиком;
 - 3-мя осями + шпиндель без датчика,

плюс количество виртуальных осей (до 3-х осей) в зависимости от конфигурации станка и технологии разработки УП. Типы осей рассмотрены при описании инструкции **TPA** файла **AXCFIL**.

4.2. Содержание секций файла **AXCFIL**

4.2.1. Свойства файла **AXCFIL**

Управление осями характеризует пользователь в файле **AXCFIL**. Этот файл состоит из 3-х секций. Каждая секция характеризует конкретную компоненту или группу компонент управления осями.

Создать или открыть на редактирование уже существующий файл **AXCFIL** можно в режиме «**COMMAND**» командой **EDI,AXCFIL/MP0**, а также функциональными клавишами режима «**COMMAND**»: **F2 (MODIF)** -> **F1 (EDI)** -> **F7 (MP0)** -> установить в списке файлов курсор на имя файла **AXCFIL** или набрать в строке редактирования имя **AXCFIL** и завершить ввод команды по клавише **ENTER**.

Примечание.

1. Имя файла **AXCFIL** в конкретном УЧПУ необходимо уточнить в секции 2 файла **FCRSYS/MP0**.
2. Порядок набора команды, используя функциональные клавиши меню режима «**COMMAND**», записывается от верхнего уровня меню, на который можно выйти, нажимая несколько раз клавишу **ESC**.

Все секции состоят из равного количества подсекций, которое определено количеством процессов; таким образом, записи (инструкции) могут повторяться в рамках одной секции для всех объявленных процессов.

Структура и формат записей приводятся в данном документе при описании конкретного файла.

4.2.1.1. Типы осей

Секция 2 файла **AXCFIL** назначает тип конфигурируемой оси. Список типов конфигурируемых осей следующий:

координатная ось - ось станка, которая может участвовать в интерполяции с другими осями такого же типа;

ось «от точки к точке» - это позиционная ось, которая не участвует в интерполяции скоординированных осей, а выполняет только позиционирование от одной точки к другой (от позиции к позиции). Ось должна иметь датчик положения. Движение этой оси управляется **SW** УЧПУ и может быть активизировано, как аналоговым напряжением, так и дискретным выходным сигналом. Например: дискретный поворотный стол, магазин инструментов или другой механизм, имеющий встроенный датчик положения;

В версиях **SW** без функций расширения (см. инструкцию **NBP**) оси «от точки к точке» могут получить задание на движение только от ПЛ.

В версиях **SW** с функциями расширения (см. инструкцию **NBP**) оси «от точки к точке» с ЦАП имеют возможность получить задание на движение от **SW** в режиме ручных перемещений («**MANU**», «**MANJ**», «**HOME**») по кнопке «**ПУСК**» и задание только от ПЛ в режимах «**MDI**», «**AUTO**», «**STEP**». Кроме этого, существует возможность сформировать особую конфигурацию для оси «от точки к точке», которая условно называется «магазин инструментов». Для конфигурации такой оси необходимо определить следующие параметры:

1. в инструкции **TPA** при определении типа оси должен быть записан только код: 2.
2. в инструкции **PAS** должно быть записано число позиций на один оборот оси «от точки к точке» (параметр2), отличное от нуля.
3. в инструкции **PAS** должно быть записано число оборотов двигателя на оборот оси «от точки к точке» (параметр3), отличное от нуля.
4. в инструкции **PAS** может быть записан номер позиции, с которого ось «от точки к точке» будет вести нумерацию позиций (параметр4).
5. в файле **IOCFIL** в секции 3 для этой оси должна быть записана инструкция **TAn**, в которой должны быть следующие установки:
 - ось имеет ЦАП;
 - ось является осью вращения;
 - должно быть объявлено ровно такое количество позиций, которое указано для этой оси во втором параметре инструкции **PAS**;

вращательная ось - координатная ось вращения, движение которой программируется в градусах;

переключаемая ось - координатная ось, имеющая общий канал управления с другой осью (общий канал ЦАП), наименование которой объявляется во втором параметре инструкции **TPA**. При программировании переключаемой оси вторая (отключенная) ось контролируется **SW** УЧПУ только на допуск позиционирования. Переключение одного канала ЦАП между осями выполняется интерфейсными сигналами ПЛ (см. «Руководство по программированию интерфейса PLC»);

ось шпинделя без датчика - ось шпинделя, контролируемая УЧПУ по скорости. Управление этой осью осуществляется программированием функции «**S**». УЧПУ не выполняет контроль положения этой оси. Обычно такая ось применяется в двух случаях: 1) для управления высокоскоростным электрошпинделем, имеющим нижний порог скорости вращения. В этом случае при конфигурации данной оси в файле **AXCFIL** необходимо заявить инструкцию **MSS**; 2) для управления моторизованным инструментом. В этом случае при конфигурировании мотори-

зованного инструмента в файле **AXCFIL** он должен быть определён как дополнительная ось шпинделя без датчика и программироваться в УП трёхбуквенным кодом **USS**.

ось шпинделя с датчиком - ось шпинделя, контролируемая УЧПУ по скорости и положению. Управление этой осью осуществляется программированием функции «**S**». Применение такой оси позволяет выполнять следующие функции, управляемые от **SW** УЧПУ:

- нарезание резьбы;
- поддержание постоянной скорости резания;
- поддержание оборотной подачи по контуру;
- ориентация шпинделя в любую позицию по запросу от ПЛ.

диаметральная ось - координатная ось, перемещение которой программируется с коэффициентом масштабирования, равным 2. Типичным примером такой оси является ось **X** токарного станка, а также расточные/обточные головки фрезерных обрабатывающих центров;

ось с контрольной точкой - специальная координатная ось, определённая в интерполяторе скоординированных осей. При включении оси из ПЛ, она использует значение датчика шпинделя, как контрольную точку. Эта ось предназначена для управления шпинделем в скоординированных движениях с другими координатными осями. Такую ось можно точно позиционировать и/или вывести на микровыключатель абсолютного нуля. Обычно это ось **C** в токарных станках;

виртуальная ось - ось, имя которой используется при программировании интерполяции, но физически она на станке не существует. Её программирование вызывает движение, выполняемое другими осями, определёнными в виртуализации. При характеристике виртуальной оси следует учитывать следующее:

- виртуальная ось должна быть объявлена в том же интерполяторе, что и оси, движение которых она вызывает;
- для виртуальной оси должны быть объявлены следующие инструкции: **NAS, TRA, NTC**.

Обычно виртуальные оси в системе используются при программировании в УП трёхбуквенного кода **UAV** для следующих случаев:

- преобразование декартовых координат в полярную систему координат. В этом случае в файле **AXCFIL** необходимо определить две виртуальные оси;
- преобразование декартовых координат в цилиндрическую систему координат. В этом случае в файле **AXCFIL** необходимо определить одну виртуальную ось;

- поворот плоскости обработки в пространстве. В этом случае в файле **AXCFIL** необходимо определить три виртуальные оси;
- преобразование декартовых координат в косоугольную систему координат. В этом случае в файле **AXCFIL** необходимо определить две виртуальные оси;

Пример

NAS=U

TPA=101,

NTC=0,0

абсолютная ось - ось, которая не требует установки микровыключателя абсолютного нуля, т. к. имеет только один электрический нуль для всего хода. Примером могут служить оси с оптическими линейками или вращательными датчиками, где один оборот датчика соответствует одному обороту оси. Выход в абсолютный микроноль по этой оси выполняется поиском референтной метки, после чего она автоматически устанавливается в нуль;

ось с запросом выхода в «0» после её повторного включения - ось, которая может быть отключена из ПЛ, и при повторном её включении нет гарантии правильного чтения позиции с её датчика. Для оси, имеющей данный тип, при её выключении признак выхода в микроноль сбрасывается.

ось шпинделя с контролируемым разгоном - ось шпинделя с контролируемым от УЧПУ разгоном. Это свойство обеспечивает плавный контролируемый разгон после команды вращения шпинделя; если для оси устанавливается этот тип, то обязательно должна быть записана инструкция **TSM**.

Пример

TSM=4,18 ,

где:

4 - время, необходимое шпинделю для инверсии от максимальной скорости по часовой стрелке до максимальной скорости против часовой стрелки. Время выражается в секундах и/или долях секунды;

18 - напряжение инверсии в вольтах.

На рисунке 4.7 представлен график изменения напряжения на оси шпинделя на участке разгона.

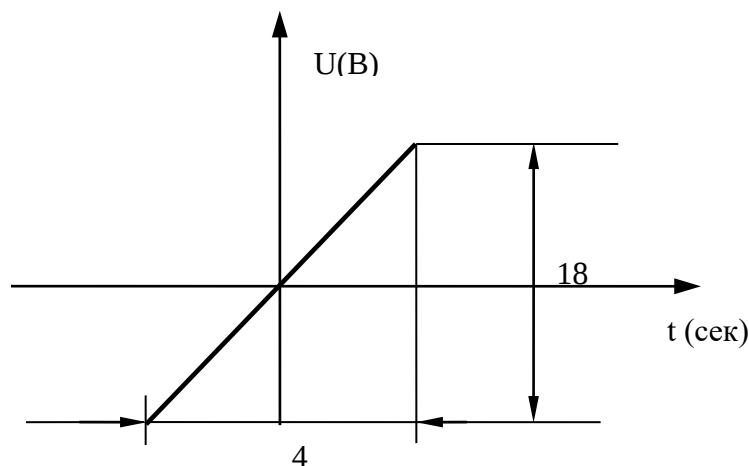


Рисунок 4.7 - График изменения напряжения для оси шпинделя на участке разгона

подчиненная параллельная ось - движение этой оси определяется движением главной оси. Главная и подчиненная ей ось - параллельны. Механические и электрические характеристики таких осей должны быть идентичны. Направление и скорость поиска микронуля, объявленные в инструкции **MCZ**, для подчиненной оси должны быть такими же, как и для главной оси. Этот тип устанавливается только для подчиненной оси, для которой объявляются следующие инструкции: **NAS, TPA, NTC, GAS, MCZ, MFC, SKW**;

Механический шаг главной и ей подчиненной оси могут иметь различный знак. Это различие определяется в инструкции **PAS** для этих осей.

Пример: **PAS=5000,1** (для главной оси) и **PAS=5000,-1** (для параллельной подчиненной оси);

ось вращения, позиционируемая от 0 до 359,999 градусов - ось вращения, которая позиционируется в пределах от 0 до 359,999 градусов. Эта характеристика используется для того, чтобы программирование перемещения для этой оси не превышало 360 градусов. В позиции 360 градусов координаты этой оси обнуляются;

ось шпинделя с двигателем переменного тока - ось шпинделя, управляемая двигателем переменного тока. Эта характеристика введена для того, чтобы при отработке циклов **G84, G86 SW** УЧПУ устанавливало бы для ПЛ, в интерфейсную часть пакета «К», бит запроса инверсии или же остановки шпинделя на дне отверстия;

ось со стартом от текущей позиции датчика - особенная координатная непрерывная ось, тип которой определен кодом **8000H**, например, **TPA=8001**. Всегда выполняет перемещения после её отключения (**RABI=0**) и последующего подключения (**RABI=1**) от текущего положения по датчику, даже после её перемещения каким-либо способом в отключённом состоянии. То есть, ось начнет движение без отработки накопленного рассогласования, полученного при её перемещении в отключённом состоянии,

в том числе, и при работе в режиме **«AUTO»** по управляющей программе.

Обычно это одна из механически связанных осей. Одна из этих осей (обычно ось **Z**) всегда находится во включённом состоянии (**RABI=1**). Вторая ось (обычно ось **W**) работает во включённом состоянии только в то время, когда первая ось не движется. Во время движения первой оси вторая ось выполняет движение за счёт механической связи между ними в отключённом состоянии, но с включённым ДОС. После того как перемещение второй оси в отключённом состоянии закончится, она переводится во включённое состояние и начинает работать от текущей координаты по своему ДОС без предварительной отработки накопленного рас- согласования.

Кадры УП, в которых есть перемещения по оси, тип которой определён кодом **8000H**, могут быть обработаны только тогда, когда активны функции **G29** и **G40**.

Программирование перемещений оси типа **8000H** в двух соседних кадрах УП, между которыми выполнялось её перемещение в выключенном состоянии, приведет к ошибке привода по этой оси (Сообщение __4 65 «Сбой привода»).

ось с датчиком без референтной метки - координатная непрерывная ось или ось «от точки к точке», ДОС которой не имеет референтной метки. Данная ось обычно объявляется для УЧПУ **NC-202**, **NC-220** и **NC-110** с модулем **ECDP**, работающих без ДОС, или для любых других УЧПУ, ДОС осей которых по какой-либо причине не имеют референтной метки, или она не может быть использована. Ось, объявленная с данным признаком, имеет особенности при выходе на микроноль.

Поиск микронуля оси выполняется так же, как для оси с ДОС, имеющим референтную метку, но на первом **CPU_tick** после съезда с микровыключателя абсолютного микронуля программное обеспечение само себе формирует признак нахождения микронуля. Значение **CPU_tick** определяется в файле **AXCFIL** в первом параметре инструкции **TIM**.

Особенности выхода в абсолютный микроноль для оси без сигнала референтной метки с ДОС:

- если в инструкции **MCZ** в первом параметре объявлен идентификатор сигнала **PLC** пакета **«A»** с нормально замкнутым контактом (НЗК), то **SW** контролирует съезд с микровыключателя каждый **CPU_tick**.

При скорости съезда оси с микровыключателя абсолютного микронуля $V=30$ мм/мин и **CPU_tick=2** мс ошибка поиска микронуля равна 0.001 мм ($V*2/60/1000=0.001$);

- если в инструкции **MCZ** в первом параметре объявлен идентификатор сигнала **PLC** пакета **«A»** с нормально разомкнутым контактом (НРК), то такой

сигнал необходимо инвертировать в быстрой части программы логики станка на свободный сигнал пакета «К». Идентификатор этого сигнала пакета «К» следует записать в первый параметр инструкции **MCZ**. В данном случае съезд оси с микровыключателя абсолютного микроула **SW** контролирует каждый **tick_logica** (инструкция **CLO** в файле характеристики **IOCFIL**).

При скорости съезда оси с микровыключателя абсолютного микроула $V=30$ мм/мин и **tick_logic**=10 мс ошибка поиска микроула оси равна 0.005 мм ($V \cdot 10 / 60 / 1000 = 0.005$).

ВНИМАНИЕ! ПОСЛЕ ЛЮБОГО СДВИГА МИКРОВЫКЛЮЧАТЕЛЯ АБСОЛЮТНОГО МИКРОУЛА ОСИ НЕОБХОДИМО ПОВТОРИТЬ ПРОЦЕДУРУ ОПРЕДЕЛЕНИЯ НУЛЯ ДЕТАЛИ ДЛЯ ВСЕХ ИСПОЛЪЗУЕМЫХ ИНСТРУМЕНТОВ.

ось с датчиком, имеющим координатно-кодированные референтные метки

- координатная непрерывная ось или ось «от точки к точке», линейный ДОС которой имеет координатно-кодированные референтные метки (**ККРМ**). Для выхода в абсолютный микроула по оси данного типа достаточно в режиме «**ВЫХОД В НОЛЬ**» («**НОМЕ**») выполнить перемещение, равное расстоянию между двумя соседними референтными метками (см. в файле **AXCFIL** инструкцию **MCZ** для ДОС с **ККРМ**);

ось с абсолютным ДОС, использующим код Грея - координатная непрерывная ось или ось «от точки к точке», использующая датчик с кодом Грея, который подключается к дискретным входам. Ось является абсолютной осью внутри одного оборота датчика и не требует выполнения процедуры «**ВЫХОД В НОЛЬ**». Для оси, включающей тип **80000H**, в секции 2 необходимо записать инструкцию **GRA**.

ось непрерывной правки шлифовального станка - координатная непрерывная ось, несущая правильный ролик для правки шлифовального круга. Если функция непрерывной правки активна, то на расстояние, пройденное осью непрерывной правки, система скорректирует радиус шлифовального круга и скорректирует позицию оси несущую шлифовальный круг, тем самым прижимая шлифовальный круг к обрабатываемой поверхности. Данный тип оси может быть определен в специальном **SW** для управления плоскошлифовальными станками.

Ось с абсолютным датчиком с интерфейсом SSI - ось с абсолютным однооборотным или многооборотным датчиком обратной связи.

4.2.2. Свойства секций файла **AXCFIL**

Секция 1 определяет конфигурирование параметров **CPU**, общее количество процессов, количество осей, их имена и тип управления объявленными осями для каждого процесса.

Секция 2 определяет конфигурирование параметров осей для каждого объявленного процесса.

Секция 3 определяет параметры компенсации погрешностей ходового винта и/или компенсацию позиции одной оси в зависимости от занимаемой позиции другой оси.

4.2.3. Секция 1

Секция 1 содержит инструкции, относящиеся к характеристике **CPU** и объявлению осей процессов. Эта секция состоит из следующих инструкций: **NBP, TIM, PRO, INx, CAS, COM, ACC**.

Инструкции **PRO, INx** и **CAS** должны быть записаны для каждого объявляемого процесса.

4.2.3.1. Инструкция NBP

Инструкция **NBP** предназначена для объявления количества процессов и подключения функций расширения **SW**, отмеченных в документации 16-теричным кодовым обозначением **ECDF** (метка **ECDF**).

Семантика:

NBP=proc_num, SW_extension .

Формат записи:

NBP=word, hexadecimal(4 цифры) ,

где:

proc_num - объявляется количество процессов, которые будут последовательно конфигурироваться в файлах **AXCFIL, PGCFIL** и **IOCFIL**; максимальное число процессов – 5 (только для версий **SW**, в номере которых присутствует буква «**М**», или версий, начиная с номеров 2.60, 3.60, и для всех версий 4.XX).

SW_extension - 16-теричный код, определяющий набор функций **SW**. Допустимы следующие коды:

- 0** - активизирует первый вид видеостраницы **#7**;
- 1** - активизирует второй вид видеостраницы **#7**;
- 2** - активизирует работу с модулем АЦП, который имеет маркировку **NC110-8АД 2000,12,28**.

Для модуля АЦП, имеющий маркировку **NC110-8А/Д 06-22-2005**, код **2** не устанавливается.

ECDF - 16-теричный код **ECDF** подключает функции расширения **SW**. Если данный код записан, то в файлах характеристики необходимо использо-

вать инструкции, отмеченные в данном документе меткой **ECDF**.

16-теричные коды можно суммировать для подключения нескольких функций одновременно.

Пример.

NBP=1, ECE2

где: **ECE2 = ECDF + 1 + 2.**

В этом примере 16-теричный код **ECE2** подключает следующие функции **SW**:

- функции расширения: код **ECDF**;
- второй вид видеостраницы **#7**: код **1**;
- модуль АЦП, имеющий маркировку **NC110-8АД 2000,12,28**: код **2**.

4.2.3.2. Инструкция TIM

Инструкция определяет тик **CPU** УЧПУ.

Семантика:

TIM=CPU_tick,,,, .

Формат записи

TIM=word,,,, ,

где:

CPU_tick - определяет временной интервал между двумя последовательными прерываниями таймера системы; **CPU_tick** может быть от 1 до 63; выражается в мс. Значение **CPU_tick** должно быть всегда первым значением слева. После числового значения **CPU_tick** необходимо записать четыре запятые.

ПРИМЕЧАНИЯ

1. Для УЧПУ **NC-110** и **NC-310** рекомендуемое значение **CPU_tick** равно 1 или 2 мс (**TIM=1,,,,**);
2. для УЧПУ **NC-200, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230** с циклическим опросом каналов ЦАП рекомендуемое значение **CPU_tick** равно 2 или 4 мс (**TIM = 2,,,,**).
3. Для УЧПУ серии **NC-XXX**, имеющих на передней панели пульта оператора логотип «**Balt-System**» рекомендуемое значение **CPU_tick** равно 1 или 2 мс (**TIM=1,,,,**);
4. Если в УЧПУ не встроен контроллер цифровой шины (CAN-Open или M2), то таймером системы обычно является элемент: **Intel 8254 Programmable Interval Timer**, при этом таймер будет использовать **IRQ-11**.
5. Если в УЧПУ встроен контроллер цифровой шины, то используется таймер этого контроллера. Номер **IRQ** для работы таймера тарируется переключками на контроллере и должен быть записан в инструкции, определяющей параметры связи ПрО с контроллером.
6. Номер **IRQ**, используемый в системе для работы таймера, должен быть освобожден в **BIOS**.

4.2.3.3. Инструкция PRO

Инструкция **PRO** предназначена для объявления номера текущего характеризруемого процесса.

Семантика:

PRO=proc_num .

Формат записи:

PRO=integer ,

где:

proc_num - объявляется номер процесса, являющийся текущим при характеристике (от 1 до 5); номер процесса не должен превышать число процессов, объявленных в инструкции **NBP**.

Примечание. При установке **proc_num** в файлах **AXCFIL**, **PGCFIL**, **IOCFIL** используйте п. 2.1 «Кодирование номеров версий **SW**» и номер **SW**, индицируемый в момент прохождения характеристики в строке «Версия программы **Z.XX**».

4.2.3.4. Инструкция COM

Инструкция **COM** используется для объявления осей (обычно ось шпинделя), являющихся общими для текущего характеризруемого и другого, ранее объявленного, процесса.

Семантика:

COM=proc_num,axis_name .

Формат записи:

COM=word,ASCII string ,

где:

proc_num - номер процесса, в котором были определены оси (обычно ось шпинделя), являющиеся общими для текущего характеризруемого процесса;

axis_name - объявляется список наименований осей (или одна ось), являющихся общими для текущего процесса и процесса, объявленного в **proc_num**.

ВНИМАНИЕ!

1. Объявление инструкции **COM** выполняется только в том случае, если существуют общие оси.
2. Описание общей оси в секции 2 выполняется только в том процессе, номер которого заявлен в первом параметре инструкции **COM**.

3. Оси, объявленные в инструкции **COM**, должны быть описаны в одном из предыдущих процессов, поэтому инструкция **COM** не записывается для первого процесса.
4. Общие оси не должны объявляться в дальнейшем в инструкции **CAS** текущего процесса.

Пример
COM=1,S

4.2.3.5. Инструкция INx

Инструкция **INx** содержит параметры характеристики интерполятора.

Семантика:

INx=cpu_num,axis_name,spindle_name,int_tick,block_num

Формат записи:

INx=word,ASCII string,ASCII string,word,word ,

где:

x - буквенно-цифровой символ, определяющий имя интерполятора. Координатные оси и шпиндель должны иметь один и тот же интерполятор. Имя интерполятора координатных осей объявляется также в секции 5 файла **PGCFIL** в инструкции **NIP**, относящейся к процессу, которому принадлежит определяемый интерполятор. Все оси «от точки к точке» определяются в интерполяторах с именами, отличными от имени интерполятора координатных осей. Если две оси «от точки к точке» в одном и том же процессе должны выполнять движение одновременно, то они должны быть записаны на разных интерполяторах. Имена интерполяторов в разных процессах должны быть разными;

cpu_num - значение **cpu_num** всегда должно быть равно **1**;

axis_name - наименование непрерывных осей, осей «от точки к точке», имеющих датчик, и виртуальных осей, за исключением подчинённой оси, конфигурируемой как параллельная ось, а также оси, определяемой в качестве штурвала; для наименования непрерывных осей используются символы: **A, B, C, D, X, Y, Z, U, V, W, P** или **Q**; для наименования оси «от точки к точке» можно использовать любую букву;

spindle_name - объявление имени оси шпинделя, если он имеет датчик. Для наименования оси шпинделя можно использовать любой алфавитный символ (обычно символ **S**);

int_tick - это временной интервал между двумя последовательными обращениями к программе интерполятора, выраженный в мс. Он должен быть равным или кратно больше значе-

ния **servo_tick** в инструкции **CAS** и значения **CPU_tick** в инструкции **TIM**;

block_num - это число кадров профиля, задающих перемещения, предварительно рассчитываемых для интерполятора. Может изменяться от 1 до 64. Если объявляется небольшое количество кадров, то существует риск остановки осей во время выполнения профиля, состоящего из кадров с малыми перемещениями. Если объявляется слишком большое число, то в этом случае имеется вероятность нехватки ОЗУ для других нужд. Каждый кадр занимает в ОЗУ 512 байтов. Для интерполятора осей «от точки к точке» значение **block_num** всегда равно 1, а для интерполятора координированных осей значение **block_num** может быть установлено от 2 до 64.

Пример

IN1 = 1,XYZB,S,1,32

X = 1-ая ось, Y = 2-ая ось, Z = 3-я ось, B = 4-ая ось, S = шпиндель с датчиком.

4.2.3.6. Инструкция CAS

Инструкция **CAS** содержит параметры характеристики управления приводами осей.

Семантика:

CAS=cpu_num,axis_name,servo_tick .

Формат записи:

CAS=word,ASCII string,word ,

где:

cpu_num - значение **cpu_num** всегда должно быть равно 1;

axis_name - объявляются наименования осей, имеющих датчик и/или ЦАП/ЦИП/Канал цифровой шины, а также подчинённых параллельных и виртуальных осей. Для наименования непрерывных осей допускаются следующие символы: **A, B, C, D, X, Y, Z, U, V, W, P, Q**. Для оси «от точки к точке» допускаются все алфавитные символы. «**S**» используется для наименования оси шпинделя. При объявлении осей их необходимо записывать в том порядке, в котором они объявлены в интерполяторах. Все оси, определённые в данной инструкции, требуют определения во второй секции соответствующего процесса данного файла;

servo_tick - определяет тик управления приводами осей, выраженный в мс. Должен быть равным или кратным значению **CPU_tick** в инструкции **TIM**.

Пример

CAS = 1,XYZSB,1

Эта запись объявляет наличие следующих компонентов:

- 1) непрерывные оси: X, Y, Z;
- 2) ось шпинделя: S;
- 3) ось «от точки к точке»: B;
- 4) тик управления приводами осей: 1 мс.

4.2.3.7. Инструкция ACC (ECDF)

Инструкция **ACC** предназначена для определения параметров **S**-образного и экспоненциального ускорения осей. Значения параметров в инструкции **ACC** действительны для всех интерполируемых осей во всех процессах, объявленных в инструкции **NBP**.

Выбор одного из 3-х вариантов изменения скорости может быть выполнен в любой момент времени изменением состояния интерфейсных сигналов **PLC U10N2 (ACC1)** и **U10N3 (ACC2)** (см. «Программирование интерфейса PLC»).

Семантика:

ACC=acc_linear_part_S, acc_nonlinear_part_S, acc_min_exp, acc_max_exp
.

Формат записи:

ACC=real, real, real, real ,

где:

acc_linear_part_S - определяет изменение абсолютного ускорения, заданного в инструкциях **RAP** и **MAN**, на линейном участке **S**-образного ускорения. Графическое изображение линейного ускорения, заданного в инструкциях **RAP** и **MAN**, представлено на рисунке 4.1. Графическое изображение зависимости линейного участка **S**-образного ускорения от значения **acc_linear_part_S** представлено на рисунке 4.2.

Значение **acc_linear_part_S** - безразмерная положительная величина, неравная нулю.

acc_nonlinear_part_S - определяет величину каждого участка линейного и нелинейного изменения скорости с **S**-образным ускорением. Части **S**-образного ускорения представлены на рисунке 4.3 и обозначены величинами **V1**, **V2** и **V3**.

Значение **acc_nonlinear_part_S** - безразмерная величина, которая должна находиться в области:

$$0 \leq \text{acc_nonlinear_part_S} \leq 0.5.$$

acc_min_exp - определяет ускорение в начале участка экспоненциального изменения скорости. Расчет изменения скорости выполняется на основании линейных ускорений,

заданных в инструкциях **RAP** и **MAN**. Графическое изображение начального участка экспоненциального ускорения представлено на рисунке 4.4.

Значение **acc_min_exp** – безразмерная положительная величина.

acc_max_exp

- определяет ускорение в конце участка экспоненциального изменения скорости. Расчет изменения скорости выполняется на основании линейных ускорений, заданных в инструкциях **RAP** и **MAN**. Графическое изображение начального участка экспоненциального ускорения представлено на рисунке 4.5.

Значение **acc_max_exp** – безразмерная положительная величина.



Рисунок 4.1

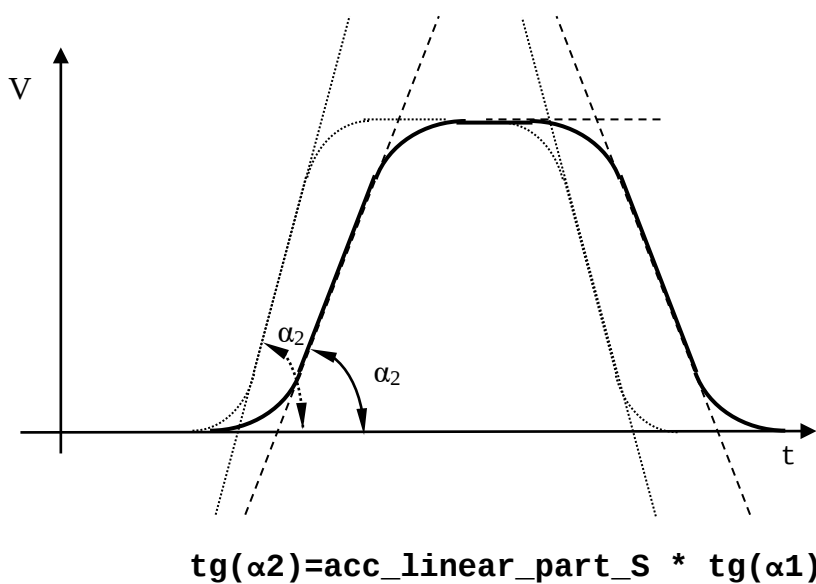
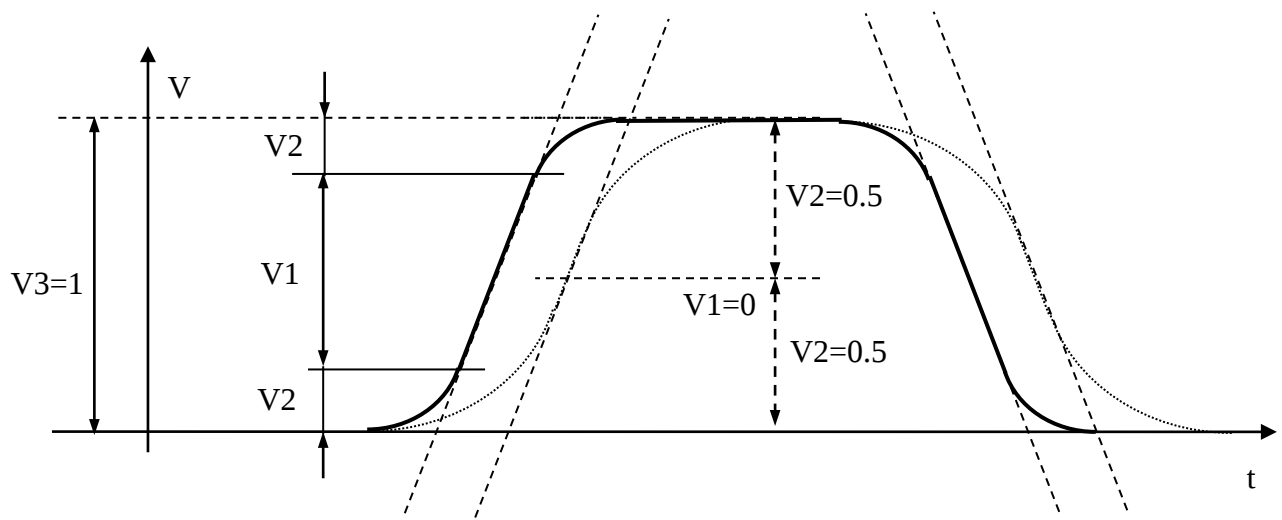


Рисунок 4.2



V_2 - часть участка V_3 , где ускорение нелинейное;
 V_1 - часть участка V_3 , где ускорение линейное.

Рисунок 4.3

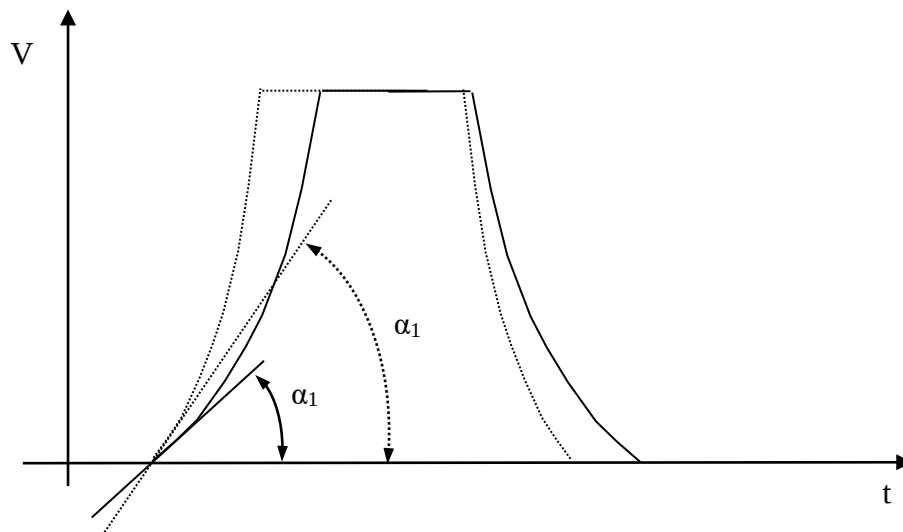


Рисунок 4.4

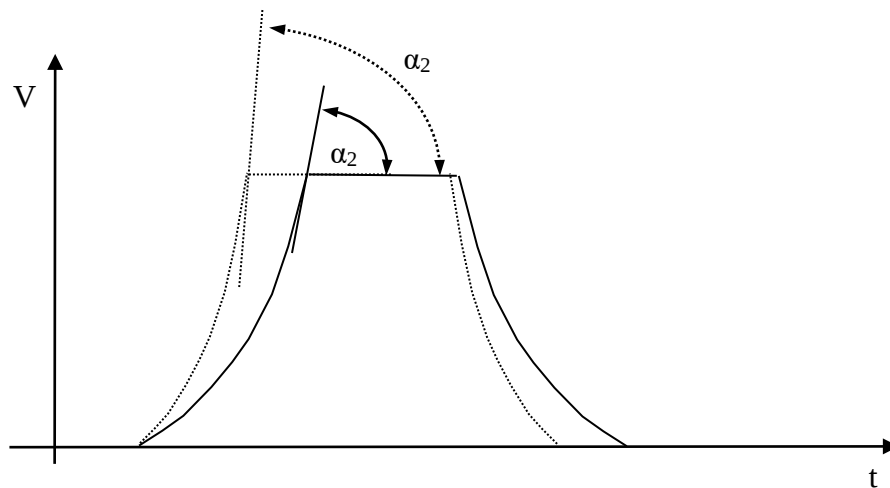


Рисунок 4.5

Пример

*1

.....
ACC = 2.0, 0.35, 1, 4

*2

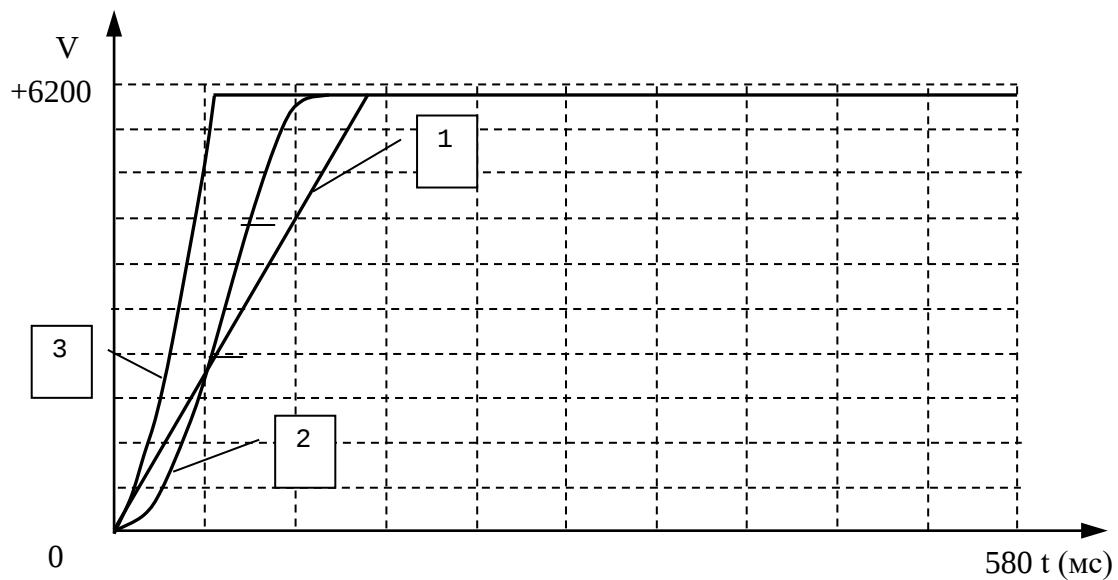
NAS = X

TPA = 1,

.....
RAP = 6000, 600

.....

На рисунке 4.6 представлены 3 графика скорости, которые были получены при выполнении кадра **G00G91X500** по трём законам разгона/торможения.



Изменение скорости:

1 - линейное;

2 - S-образное;

3 - экспоненциальное.

Рисунок 4.6

Пример

Запись секции 1 для двух процессов токарного станка, имеющего четыре оси и общий шпиндель с датчиком:

*1

NBP = 2, ECDF

TIM = 2, , , ,

PRO = 1

IN1 = 1, XZ, S, 2, 16

CAS = 1, XZS, 2

ACC = 2.0, 0.5, 1, 4

PRO = 2

COM = 1, S

IN2 = 1, XZ, S, 2, 16

CAS = 1, XZ, 2

4.2.4. Секция 2

Секция 2 включает в себя инструкции, содержащие параметры управления осями. Эта секция является оригинальной для каждого процесса, и поэтому должна быть объявлена в инструкции **PRO** для каждого процесса. Каждая ось, в том же порядке, как она объявлена в инструкции **CAS** в секции 1, должна быть объявлена в секции 2 в инструкции **NAS**, после которой должны быть записаны по порядку следующие инструкции:

TPA
NTC
RAP

После этих инструкций, если оси объявлены с датчиком, необходимо записать следующие инструкции:

PAS
GAS
SKW
POS
SRV

Эта секция может содержать и другие инструкции, наличие которых зависит от характеристик управляемого оборудования. К этим инструкциям относятся:

MCZ
MAN
GMnn
LOP
LOn
CUB
SWn
FRC
MFC
TSM
ASM
POM
ZNO
FBF
HWC
ATR
GRA
AFC
FOPP
FTP
ADG
LAS
FQF
VGAS

4.2.4.1. Инструкция PRO

Инструкция **PRO** предназначена для объявления номера текущего характеризруемого процесса.

Семантика:

PRO=proc_num .

Формат записи:

PRO=integer ,

где:

proc_num - объявляется номер процесса, являющийся текущим при характеристике (от 1 до 5); номер процесса не должен превышать число процессов, объявленных в инструкции **NBP** в файле **AXCFIL**.

4.2.4.2. Инструкция NAS

Инструкция **NAS** используется для определения текущей оси, выбранной для характеристики.

Семантика:

NAS=axis_name .

Формат записи:

NAS=ASCII string ,

где:

axis_name - объявляется наименование оси, выбранной для характеристики. Оно является одним из списка наименований осей, объявленных в инструкции **CAS** секции 1, и выбирается из списка в порядке их записи.

4.2.4.3. Инструкция TRA (общий случай)

Инструкция **TRA** предназначена для определения типа оси, выбранной для характеристики.

Семантика:

TRA=axis_type,switch_axis_name .

Формат записи:

TRA=hexadecimal (8 цифр),ASCII string ,

где:

axis_type - тип оси, шестнадцатеричный код которого устанавливается согласно данным, приведённым в таблице 4.1;

При пользовании таблицей 4.1 следует учесть, что шестнадцатеричный код типа оси получается путём суммирования соответствующих битов назначения, определяющих характеризующую ось;

Таблица 4.1 - Шестнадцатеричный код типа оси

Шестнадцатеричный код	Бит назначения	Характеризуемая ось
0001	0	Координатная ось
0002	1	Ось «от точки к точке»
0004	2	Ось вращения
0008	3	Переключаемая ось
0010	4	Ось шпинделя без датчика
0020	5	Ось шпинделя с датчиком
0040	6	Диаметральная ось
0080	7	Ось с контрольной точкой
0100	8	Виртуальная ось
0200	9	Абсолютная ось
0400	10	Ось с запросом выхода в «0» после её отключения
0800	11	Ось шпинделя с контролируемым разгоном
1000	12	Подчинённая параллельная ось
2000	13	Ось вращения, позиционируемая от 0 до 359,999 градусов
4000	14	Ось шпинделя с двигателем переменного тока
8000	15	Ось со стартом от текущей позиции датчика
10000	16	Ось с датчиком без референтной метки
20000	17	Не используется
40000	18	Ось, ДПС которой имеет координатно-кодированные референтные метки
80000	19	Ось, имеющая абсолютный ДПС с кодом Грея, подключенным к дискретным входам
100000	20	Непрерывная ось правки шлифовального станка
200000	21	Ось с абсолютным датчиком с интерфейсом SSI

switch_axis_name - имя оси, на которую переключается текущая характеризующая ось. Для наименования переключаемой оси допускается использовать следующие символы: **A, B, C, X, Y, Z, U, V, W, P, Q, D**. Если такой оси не существует, то после параметра **axis_type** необходимо записать только запятую.

Пример

Для объявления диаметральной координатной оси надо записать:

TPA=41, ;

для объявления переключаемой оси вращения надо записать:

TPA=0D, **switch_axis_name** .

4.2.4.4. Инструкция ТРА (попеременное управление осями от одного канала датчика). Версии SW с расширением Р-ПД

Функция попеременного управления двумя осями от одного канала датчика обеспечивается установкой в инструкции **ТРА** двух сигналов **PLC** для каждой оси. Для записи имён сигналов **PLC** в инструкцию **ТРА** введены дополнительно два параметра: **memory_signal** и **signal_recovery**.

Семантика:

ТРА=axis_type, switch_axis_name, memory_signal, signal_recovery .

Формат записи:

ТРА=hexadecimal (8 цифр), ASCII string, PLC variable, PLC variable ,

где:

axis_type - назначение **axis_type** (см. п/п 4.2.4.3 «Инструкция ТРА»);

switch_axis_name - наименование переключаемой оси (см. п/п 4.2.4.3 «Инструкция ТРА»);

memory_signal - сигнал пакета «А» или свободный сигнал пакетов «К» или «N», который предназначен для запоминания координаты позиции оси:

0 - не запоминать при отключении оси;

1 - запомнить при отключении оси.

signal_recovery - сигнал пакета «А» или свободный сигнал пакетов «К» или «N», который предназначен для восстановления запомненной координаты позиции оси:

0 - не восстанавливать при включении оси;

1 - восстановить при включении оси.

ВНИМАНИЕ !

1. **memory_signal** и **signal_recovery** для каждой оси должны быть индивидуальными.
2. Запоминание и восстановление координаты позиции оси должно выполняться только при зажатой оси.
3. Для правильного запоминания координаты позиции оси **memory_signal** должен быть равен «1» до выключения оси.
4. При восстановлении позиции оси **signal_recovery** должен быть равен «1» до включения оси.

Пример

Для объявления 1-ого канала датчика, который должен переключаться между координатными осями X и B, необходимо записать:

```
NAS=X
TRA=1,, I0K8, U10K8
NTC=1,1
...
NAS=B
```

TPA=5,, I0K11, U10K11
NTC=1, 2

4.2.4.5. Инструкция NTC

Инструкция **NTC** предназначена для определения номера ДОС и ЦАП/ЦИП для текущей оси.

Семантика:

NTC=trans, conv .

Формат записи:

NTC=integer, integer ,

где:

trans - определяет номер датчика в УЧПУ:

- 1) **NC-110**: максимальное число датчиков - 16;
NC-310: максимальное число датчиков - 8;

номер датчика рассчитывается по формуле:

$$n*4+p$$

где:

n - номер модуля **ECDA/RCDA/ECDP** (**n=0-3**); устанавливается перемычками на модуле, как указано в документе «Руководство по эксплуатации»;

p - номер датчика в модуле **ECDA/RCDA/ECDP** (**p=1-4**); соответствует номеру разъёма канала датчика на лицевой панели модуля:

а) модуль **ECDA**:

- **NC110-3/NC310-2**, **NC110-32** имеют 4 канала энкодера с маркировкой «**1**»-«**4**»;
- **NC110-31/NC310-21**, **NC110-33** имеют 2 канала энкодера с маркировкой «**1**»-«**2**»;

б) модуль **RCDA**:

- **NC110-35** имеет 4 канала датчика резольвер/индуктосин с маркировкой «**1**»-«**4**»;

- 2) **NC-201**, **NC-201M**, **NC-202**: максимальное число энкодеров - 3; номер датчика соответствует номеру разъёма канала датчика «**1**»-«**3**» на панели разъёмов УЧПУ;

- 3) **NC-200**, **NC-210**, **NC-220**: максимальное число энкодеров - 4; номер датчика соответствует номе-

ру разъёма канала датчика «1»-«4» на лицевой панели модуля **ECDA/ECDP**;

ВНИМАНИЕ! В УЧПУ **NC-202, NC-220, NC-302 и NC-110** с каналами ЦИП номер канала датчика для координатной оси в инструкции **NTC** записывается всегда, включая тот случай, когда управление осью выполняется без реального ДОС. Если управление координатной осью выполняется без реального ДОС, то ее номер канала датчика необходимо записать равным номеру канала ЦИП, который записан в инструкции **NTC** в параметре **conv**.

- 4) **NC-230**: максимальное число энкодеров – 5; номер датчика соответствует номеру разъёма канала датчика «1»-«5» на лицевой панели модуля **ECDA**;

conv - определяет номер и разрядность преобразователя ЦАП (DAC) или ЦИП (DPC) в УЧПУ:

- 1) **NC-110**: разрядность преобразователя - 14/16 разрядов, максимальное число каналов преобразователя – 16.
NC-310: разрядность преобразователя - 14 разрядов, максимальное число каналов преобразователя – 16.

Номер 14-ти разрядного преобразователя рассчитывается по формуле:

$$n \cdot 4 + q \quad (4.3)$$

Номер 16-ти разрядного преобразователя рассчитывается по формуле:

$$- (n \cdot 4 + q + 200) \quad (4.4)$$

В формулах (4.3) и (4.4):

n - номер модуля **ECDA/RCDA/ECDP** (**n=0-3**); устанавливается перемычками на модуле, как указано в документе «Руководство по эксплуатации»;

q - номер канала преобразователя в модуле **ECDA/RCDA/ECDP** (**q=1-4**);

а) модуль **ECDA**:

- **NC110-3/NC310-2** (14-ти разрядный ЦАП), **NC110-32** (16-ти разрядный ЦАП) имеют 4 канала ЦАП на разъёме с маркировкой «5»;
- **NC110-31/NC310-21** (14-ти разрядный ЦАП), **NC110-33** (16-ти разрядный ЦАП) имеют 2 канала ЦАП на разъёме с маркировкой «3»;

б) модуль **RCDA**:

- **NC110-35** (14-ти разрядный ЦАП) имеет 4 канала ЦАП на разъёме с маркировкой «5».

2) **NC-201, NC-201M:** максимальное число каналов 14-ти разрядного ЦАП – 4; номер ЦАП соответствует номеру канала ЦАП в разъёме «5» на панели разъёмов УЧПУ.

3) **NC-202:**

- максимальное число каналов 14/16-ти разрядного ЦАП – 3.

Номер 14-ти разрядного ЦАП соответствует номеру (**N**) канала ЦАП в разъёме «5» на панели разъёмов УЧПУ, например:

NTC=1, 1;

Номер 16-ти разрядного ЦАП соответствует номеру **-(N+200)** канала ЦАП в разъёме «5» на панели разъёмов УЧПУ, например:

NTC=1, -201

Примечание. Разрядность ЦАП (14/16) в УЧПУ **NC-202** устанавливается пользователем перемычками, см. «Руководство по эксплуатации».

- максимальное число каналов 14-ти разрядного ЦАП – 1; номер 14-ти разрядного ЦАП равен 4.

4) **NC-200, NC-210:** максимальное число каналов 14-ти разрядного ЦАП – 5; номер ЦАП соответствует номеру канала ЦАП в разъёме «5» на лицевой панели модуля **ECDA**.

5) **NC-220:** максимальное число каналов 14/16-ти разрядного ЦАП – 4; Номер 14-ти разрядного ЦАП соответствует номеру (**N**) канала ЦАП в разъёме «5» на лицевой панели модуля **ECDP**, например:

NTC=1, 1;

Номер 16-ти разрядного ЦАП соответствует номеру **-(N+200)** канала ЦАП в разъёме «5» на лицевой панели модуля **ECDP**, например:

NTC=1, -201

Примечание. Разрядность ЦАП (14/16) в УЧПУ **NC-220** устанавливается пользователем перемычками модуля **ECDP**, см. «Руководство по эксплуатации».

- номер ЦАП соответствует номеру канала ЦАП в разъёме «5» на лицевой панели модуля **ECDP**;
- максимальное число каналов 12/14-ти разрядного ЦАП – 1:
 - а) номер 14-ти разрядного ЦАП равен **5**;
 - б) номер 12-ти разрядного ЦАП записывается в инструкции:

NTC = , -105 ,

где:

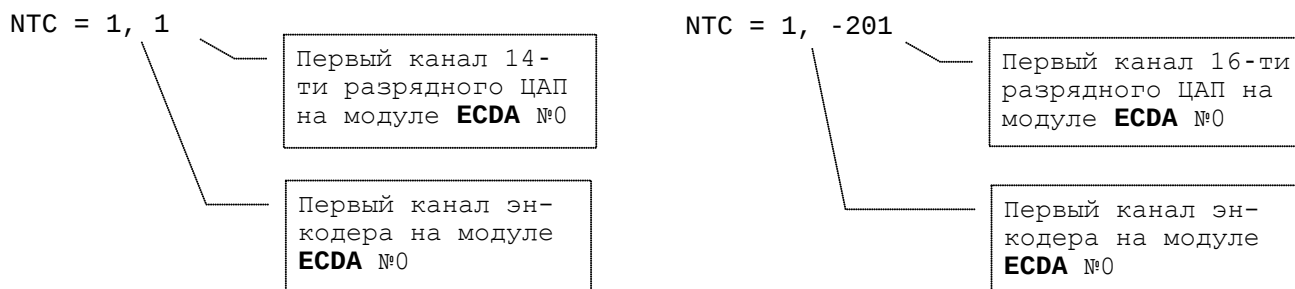
-100 – признак 12 разрядного ЦАП;

5 – номер канала ЦАП.

- 6) **NC-230**: максимальное число каналов 14-ти разрядного ЦАП – 6; номер ЦАП соответствует номеру канала ЦАП в разъёме «6» на лицевой панели модуля **ECDA**.

Примечание. Распределение каналов ЦАП/ЦИП по контактам разъёма ЦАП/ЦИП в модулях **ECDA/ECDP** УЧПУ указано в документе «Руководство по эксплуатации» для конкретного УЧПУ.

Примеры записи инструкции NTC для УЧПУ **NC-110** приведены на рисунке 4.8.



а) 14-ти разрядный ЦАП

б) 16-ти разрядный ЦАП

Рисунок 4.8 - Запись инструкции NTC для УЧПУ **NC-110**

4.2.4.6. Инструкция RAP

Инструкция используется для установки параметров скорости и ускорения для текущей оси.

Семантика:

RAP=fast_speed, fast_accel

Формат записи:

RAP=real, real ,

где:

fast_speed - представляет скорость быстрого хода оси, выраженной в следующих единицах:

- линейная ось: мм/мин;
- ось вращения: градус/мин;
- ось «от точки к точке»: позиция/мин

Для шпинделя этот параметр не записывается.

fast_accel - определяет ускорение быстрого хода, выраженное в следующих единицах:

- линейная ось: мм/с²;

- ось вращения: градус/с²;
- ось «от точки к точке»: позиция/с².

Для оптимизации ориентации шпинделя инструкция **RAP** может быть заявлена для шпинделя в случае, если шпиндель имеет датчик и может позиционироваться (например, шпиндель больших размеров). В этом случае для оси шпинделя вводится второй параметр инструкции **RAP** (ускорение, выраженное в об/с²).

Если этот параметр опущен или равен «0», шпиндель позиционируется только с контролем ошибки по положению. Если этот параметр больше «0», **SW** УЧПУ управляет замедлением в точке ориентации. Ускорение контролируется только в том случае, если шпиндель заявлен как шпиндель с контролируемым разгоном (**TPA=820**), в противном случае, величина ускорения не контролируется. Значение ускорения, введенное в инструкции **RAP** для оси шпинделя, не контролируется, но рекомендуется, чтобы оно не превышало максимального значения, рассчитываемого по формуле:

$$a = \frac{\text{диапазон}}{t \times 60} \times \frac{B}{KC} \quad (4.5),$$

где:

- a** - значение ускорения;
- диапазон** - скорость основного диапазона;
- B** - значение второго параметра инструкции **TSM**;
- t** - значение первого параметра инструкции **TSM**;
- KC** - величина **KC**.

Пример

Объявление оси шпинделя:

```
NAS = S
TPA = 820
...
GM1 = 3000, 7.5, 10
TSM = 5, 15
...
RAP = , 20
```

Расчёт ускорения:

$$a = \frac{3000}{5 \times 60} \times \frac{15}{7.5} = 20 \text{ (об/с}^2\text{)}$$

4.2.4.7. Инструкция GAS

Инструкция используется для определения действий текущей оси.

Семантика:

GAS=axis_bck1,dead_zone .

Формат записи:

GAS=real,real ,

где:

axis_bck1 - определяет величину зазора (люфта), который имеется при изменении направления движения оси в том случае, когда датчик измерения находится непосредственно на ходовом винте или на двигателе. Эта величина выражается в следующих единицах:

- линейная ось: мм или дюйм;
- ось вращения: градус;
- ось «от точки к точке»: позиция.

Значение люфта может быть положительным или отрицательным числом (для согласования направления отработки люфта при смене направления движения).

По умолчанию скорость выборки люфта устанавливается в системе на основании значения люфта **axis_bck1** и времени его выборки. Время выборки люфта зависит от коэффициента усиления (**KV**), заданного в инструкции **GMxx**. Скорость выборки люфта может быть изменена в инструкции **VGAS**;

dead_zone - определяет зону нечувствительности следящего привода (в пределах допуска позиционирования); в пределах этой зоны ось не контролируется **SW** УЧПУ; выражается в миллиметрах, дюймах, градусах, позициях.

Пример

GAS = 0.15, 0.005

4.2.4.8. Инструкция GAS (ECDF)

Семантика:

GAS=axis_bck1,dead_zone,zero_drift_read,zero_drift_cor .

Формат записи:

GAS=real,real,PLC variable,PLC variable ,

где:

axis_bck1 - определяет величину зазора (люфта), который имеется при изменении направления движения оси в том случае, когда датчик измерения находится непосредственно на ходовом винте или на двигателе. Эта величина выражается в следующих единицах:

- линейная ось: мм или дюйм;
- ось вращения: градус;
- ось «от точки к точке»: позиция.

Значение люфта может быть положительным или отрицательным числом (для согласования направления обработки люфта при смене направления движения).

По умолчанию скорость выборки люфта устанавливается в системе на основании значения люфта **axis_bck1** и времени его выборки. Время выборки люфта зависит от коэффициента усиления (**KV**), заданного в инструкции **GMXX**. Скорость выборки люфта может быть изменена в инструкции **VGAS**;

dead_zone - определяет зону нечувствительности следящего привода (в пределах допуска позиционирования); в пределах этой зоны ось не контролируется **SW** УЧПУ; выражается в миллиметрах, дюймах, градусах, позициях;

zero_drift_read - определяет идентификатор свободного сигнала **PLC**. Данный сигнал, установленный в программе логики станка в состоянии «1», является запросом чтения дрейфа нулевого уровня в канале управления оси;

zero_drift_cor - определяет идентификатор свободного сигнала **PLC**. Данный сигнал, установленный в ПЛ станка в состоянии «1», является запросом коррекции дрейфа нулевого уровня в канале управления оси.

Примечание:

- 1) **zero_drift_read** и **zero_drift_cor** для одной и той же оси должны быть определены разными сигналами **PLC**.
- 2) Управление сигналами **PLC** для чтения и коррекции дрейфа нулевого уровня приведено в «Руководстве по программированию интерфейса PLC»

Пример

```
NAS=X
GAS=-.05,,U220K0,U220K1
.....
NAS=Y
GAS=-.05,,U220K2,U220K3
.....
NAS=S
GAS=,,U220K3,U220K4
```

4.2.4.1. Инструкция VGAS

Инструкция предназначена для изменения скорости выборки люфта и может быть записана в подсекции текущей оси, если в инструкции **GAS** значение **axis_bck1** больше нуля. Если инструкция не записана, то скорость компенсации люфта устанавливается системой.

Семантика:

VGAS=bck1_axis_vel,bck1_mode_set,bck1_process .

bck1_axis_vel - этот параметр определяет значение скорости для выборки люфта **axis_bck1** при смене направления движения оси; мм/мин;

ВНИМАНИЕ! В случае ввода значения **bck1_axis_vel** для осей, участвующих в

совместном движении по контуру, например, круговой интерполяции, необходимо проверить точностные параметры обработанного контура.

bck1_mode_set - этот параметр определяет идентификатор свободного выходного сигнала на языке **PLC**, используемый в системе для установки скоростного режима выборки люфта:

***bck1_mode_set* = «0»** - включен режим выборки люфта со скоростью, рассчитанной системой по умолчанию;

bck1_mode_set* = «1»** - включен режим выборки люфта со скоростью, определенной значением ***bck1_axis_vel.

bck1_process - этот параметр определяет идентификатор свободного входного сигнала на языке **PLC**. Значение сигнала для ПЛ устанавливается в Про, если ***bck1_mode_set* = «1»**. ПЛ должна обеспечить формирование запроса приостановки подачи на время, пока ***bck1_process* = «1»**. Для этого в ПЛ рекомендуется записать логическое выражение, обеспечивающее установку сигнала **FOLD (U10K5)** в состояние «1», на время, пока состояние сигнала ***bck1_process* = «1»**.

Пример.

VGAS=3, U180K0, U180K1

Пример ПЛ:

```
;включение bck1_mode_set (U180K0) = «1», если процесс готов к
;обслуживанию осей CONP (I0K2) = «1»
U180K0=I0K2
;включение сигнала FOLD (U10K5) = «1» на время выборки
;люфта, пока bck1_process (U180K1) = «1»
U10K5=U180K1
$
```

4.2.4.2.

Инструкция PAS

Инструкция **PAS** устанавливает электрический и механический шаг текущей оси.

Семантика:

PAS=el_pitch, mec_pitch .

Формат записи:

PAS=real, real ,

где:

el_pitch - этот параметр должен быть только положительным; величина этого параметра зависит от типа датчика:

- если датчиком является инкрементальный энкодер, то величина ***el_pitch*** представляет собой количество импульсов за один оборот датчика, умноженное на 4;
- если датчиком является инкрементальный линейный энкодер (оптическая линейка), то величина ***el_pitch*** представляет собой количество импульсов умноженное на 4, соответствующее расстоянию 1 мм, указанному в параметре ***mec_pitch***;
- если датчиком является индуктосин, то инструкция должна быть: **PAS=65536,2;**
- если датчиком является резольвер, то инструкция должна быть: **PAS=65536,mec_pitch;**
- если датчиком является абсолютный энкодер, она представляет собой количество импульсов за один оборот датчика, рассчитанное по формуле:

$$2^{\text{resolution ST}},$$

где:

resolution ST - значение параметра **resolution ST** в инструкции **ATR**.

resolution ST, бит:		Значение <i>el_pitch</i>:
12	=	4096
13	=	8192
14	=	16384
15	=	32768
16	=	65536
17	=	131072
18	=	262144
19	=	524288
20	=	1048576
21	=	2097152
22	=	4194304
23	=	8388608
24	=	16777216
25	=	33554432
26	=	67108864
27	=	134217728
28	=	268435456
29	=	536870912
30	=	1073741824
31	=	2147483648

- если реального датчика оси нет, то значение электрического шага представляет количество шагов шагового двигателя на одном его обороте. Данный случай возможен только для УЧПУ **NC-202**, **NC-220** или **NC-110** и **NC-310** с модулем **ECDP**. Для расчета данного значения необходимо использовать «Руководство по эксплу-

атации» на конкретный привод и шаговый двигатель (ШД) .

mec_pitch

- 1) в случае, когда ось имеет датчик; это расстояние, пройденное осью, за количество импульсов, записанное в **el_pitch**. Значение **mec_pitch** может быть положительной или отрицательной величиной. Знак **mec_pitch** позволяет согласовать направление счета датчика с направлением движения оси:
 - для осей «от точки к точке» этот параметр представляет собой количество позиций, пройденных осью за один оборот датчика;
 - для оси шпинделя этот параметр представляет собой количество оборотов шпинделя, соответствующее одному обороту датчика.
- 2) в случае, когда ось с ШД без реального датчика; это расстояние, пройденное осью, за количество шагов ШД, записанное в **el_pitch**:
 - для осей «от точки к точке» этот параметр представляет собой количество позиций, пройденных осью за один оборот вала ШД;
 - для оси шпинделя ШД не используются.

Примеры

- 1) Для оси с датчиком оптическая линейка, имеющим 250 имп/мм:
PAS = 1000,1.
- 2) Для оси с датчиком энкодер, имеющим разрешающую способность 1250 имп/об, и механическим шагом 10 мм:
PAS = 5000,10.
- 3) Для оси с абсолютным линейным датчиком (**resolution_ST = 21 бит**) и длинной оси 1000 мм:
PAS = 2097152,1000 .

4.2.4.3. Инструкция PAS (ECDF)

Инструкция **PAS** устанавливает параметры измерительной системы текущей оси.

В этой инструкции для большинства осей обычно устанавливаются только два первых параметра (электрический и механический шаг), но для оси «от точки к точке», определяемой как «Магазин инструментов», необходимо дополнительно установить значение для третьего параметра, а после этого можно установить значение и для четвертого параметра.

Семантика:

PAS=el_pitch,mec_pitch,rev_num_motor,start_point_# .

Формат записи:

PAS=real,real,real,real ,

где:

el_pitch - описание параметра приведено в п. 4.2.4.2

mec_pitch

1) в случае, когда ось имеет датчик; это расстояние, пройденное осью, за количество импульсов, записанное в **el_pitch**. Значение **mec_pitch** может быть положительной или отрицательной величиной. Знак **mec_pitch** позволяет согласовать направление счета датчика с направлением движения оси:

- для осей «от точки к точке» этот параметр представляет собой количество позиций, пройденных осью за один оборот датчика;
- для осей «от точки к точке», определяемой как «Магазин инструментов», этот параметр представляет собой количество позиций на одном обороте оси;
- для оси шпинделя этот параметр представляет собой количество оборотов шпинделя, соответствующее одному обороту датчика.

2) в случае, когда ось с ШД без реального датчика; это величина отношения между расстоянием, пройденным осью, и числом оборотов вала соответствующего ШД, может быть положительной или отрицательной (для согласования направления движения и показания датчика):

- для осей «от точки к точке» этот параметр представляет собой количество позиций, пройденных осью за один оборот вала ШД;
- для осей «от точки к точке», определяемой как «Магазин инструментов», этот параметр представляет собой количество позиций на одном обороте оси;
- для оси шпинделя ШД не используются.

rev_num_motor - определяет число оборотов двигателя на один оборот оси «от точки к точке», определяемой как «Магазин инструментов».

start_point_# - номер, с которого начинается отсчет позиций оси «от точки к точке» после её выхода в абсолютный микроноль. Если **rev_num_motor** равен «0», то **SW** игнорирует установленное значение данного параметра.

Примеры

1) Для оси с датчиком энкодер, имеющим разрешающую способность 1250 имп/об, и механическим шагом 10 мм:

PAS = 5000,10.

2) Для оси «от точки к точке», определяемой как «Магазин инструментов» со следующими параметрами:

- количество позиций: 30;
- количество оборотов двигателя на один оборот оси: 6;

- датчик энкодер расположен на валу двигателя: 250 имп/об ($250 * 4 = 1000$);
- нумерация позиций оси после выхода в абсолютный микроноль должна быть начата с единицы;
 $PAS = 6000, 30, 6, 1$

4.2.4.4. Инструкция SKW

Инструкция **SKW** определяет главную ось, которая используется с текущей осью, являющейся подчиненной и параллельной этой главной оси.

Семантика:

SKW=master_axis, SKEW, KDS, SGAN

Формат записи:

SKW=ASCII string, real, real, real ,

где:

master_axis - это наименование главной оси, которая имеет в качестве подчиненной текущую ось; наименование главной оси имеет только одну букву; характеристики главной оси должны быть определены до объявления ее имени в этом параметре;

SKEW - определяет максимально допустимую асимметрию между параллельными осями; значение записывается:

- линейная ось: мм;
- ось вращения: градус;

KDS - определяет несовпадение между референтными метками датчиков главной оси и ей подчиненной оси; значение записывается в импульсах датчика.

Для определения значения **KDS** можно использовать следующий алгоритм:

1. записать значение параметра **KDS**, равное нулю;
2. перезагрузить УЧПУ;
3. вывести параллельные оси в абсолютный микроноль в режиме «HOME»;
4. зафиксировать значение со знаком из поля «D=» в видеостранице #1 или #7 и перевести его из миллиметров в количество импульсов датчика оси по формуле:
 $KDS = \text{электрический шаг} * D / \text{механический шаг};$
5. записать это значение количества импульсов как параметр **KDS** с тем знаком, который был зафиксирован в поле «D=»;
6. перезагрузить УЧПУ.

SGAN - определяет константу усиления для компенсации ошибки асимметрии осей. Рекомендуемый диапазон этой константы

от 0 до 2. При равенстве константы 0 компенсация не выполняется.

Расчёт ошибки асимметрии выполняется по формуле:

$$E_{\text{skew}} = \frac{E_{\text{гл.оси.}} - E_{\text{подч.оси.}}}{2} \quad (4.6)$$

В то же время константа усиления для обеих осей, главной и подчинённой, рассчитывается по формулам 4.7 и 4.8:

$$E_{\text{гл.оси.}}(n) = E_{\text{гл.оси.}}(n-1) + (E_{\text{skew}} * s_{\text{gan}}) \quad (4.7);$$

$$E_{\text{подч.оси.}}(n) = E_{\text{подч.оси.}}(n-1) + (E_{\text{skew}} * s_{\text{gan}}) \quad (4.8).$$

Чем больше константа усиления, тем в большей степени оси контролируются **SW** УЧПУ. Однако если значение константы очень большое, может возникнуть возбуждение; при очень маленьком значении константы симметрия (синхронность) будет наименьшей.

Примечание.

После выхода в абсолютный микроноль по главной и подчиненной осям **SW** следит за перекосом этих двух осей по их датчикам, даже если оси выключены в ПЛ, и при включении осей из ПЛ выполняет их выравнивание на значение **KDS**. Допустимый перекокс между осями, который может выравнивать **SW** после включения осей, не должен превышать значения **сервоошибка покоя**, установленного в инструкции **SRV** для главной оси. Если перекокс при включении осей из ПЛ будет больше допустимого значения, то **SW** формирует ошибку: «Сбой привода (ИМЯ ОСИ:НОМЕР ПРОЦЕССА)» и отключает станок.

Для повторного включения осей из ПЛ с произвольным перекосом между главной и подчиненной осями можно предложить следующие варианты:

1. Перезапуск УЧПУ. Перезапуск УЧПУ сбросит признак, что оси были выведены в абсолютный микроноль, и ошибка при включении осей не формируется.
2. Сброс признака «Ось выведена в ноль» после выключения оси из ПЛ. Для этого необходимо добавить в тип главной и подчиненной осей 16-тиричный код: **400H**. Таким образом, эти оси всегда будут включаться из ПЛ с уже сброшенным признаком их выхода в абсолютный микроноль.
3. Изменение значения сервоошибки покоя перед включением главной оси из ПЛ. Для этого перед включением осей из ПЛ, независимо от значения реального перекокса осей, в режиме **MDI** выполнить кадр, например **(PTA,X,1,10)**, по клавише **ПУСК**, изменяющий значение сервоошибки в покое по главной оси. Значение разрешённой ошибки (здесь 10 мм) должно быть больше значения реального перекокса. После включения осей и их выравнивания необходимо установить значение сервоошибки покоя равное значению, записанному в инструкции **SRV**, например, в режиме **MDI** выполнить кадр: **(PTA,X,1,0.5)** по клавише **ПУСК**.

Этот вариант выравнивания осей после их включения не разрешается использовать, если отключение станка произошло по ошибке датчика. Отключение осей по ошибке датчика не позволяет правильно выполнить их выравнивание. Если отключение станка было по причине ошибки датчика, то необходимо сделать невозможным дальнейшую работу без повторного выхода в ноль. Контроль исправности датчиков необходимо обеспечить в инструкции **FBF**.

4. Изменение значения сервоошибки покоя после включения главной оси из ПЛ. Для этого необходимо включить оси из ПЛ независимо от значения реального перекося осей. **SW**, если перекося оказался больше допустимого, формирует ошибку «Сбой привода» и снова отключает оси. После этого в режиме **MDI** необходимо выполнить кадр, например (**PTA,X,1,10**), по клавише **ПУСК**, изменяющий значение сервоошибки в покое перед включением осей. Значение разрешенной ошибки (здесь 10 мм) должно быть больше значения перекося. После включения осей вернуть обратно значение сервоошибки выполнением кадра (**PTA,X,1,1**) по клавише **ПУСК**.

Данный способ автоматически потребует повторного выхода осей в абсолютный микроноль.

Примечание.

Кадры с трехбуквенным кодом **PTA** можно выполнить в соответствующий момент из файла **FILMOV** по запросу ПЛ.

4.2.4.5. Инструкция **MCZ** (общий случай)

Инструкция **MCZ** используется для установки параметров выхода в позицию микровыключателя для текущей оси.

Семантика:

MCZ=zero_input,direction,speed .

Формат записи:

MCZ=PLC variable,boolean (2 цифры),real ,

где:

zero_input - этот параметр определяет идентификатор входного сигнала микровыключателя на языке **PLC**, используемый при поиске абсолютного микроноля оси. Например: **I0A6**.

Сигнал микровыключателя в исходном состоянии должен быть сигналом высокого уровня и переходить на низкий уровень при его включении (разрыве). Если сигнал пакета «**A**», определяющий вход микровыключателя, в исходном состоянии низкого уровня, то его необходимо инвертировать в ПЛ на свободный сигнал пакетов «**K**» или «**N**», и записать имя инверсного сигнала в инструкции **MCZ**.

Например: **U170K7=/I1A29** - логическое выражение в ПЛ.

MCZ=U170K7,0,200

ВНИМАНИЕ! Не рекомендуется выполнять инвертирование состояния сигнала микровыключателя, чтобы исключить увеличение времени реагирования системы при входе оси на микровыключатель, и исключить возможность потери состояния микровыключателя, если ПЛ остановлена.

Входом микровыключателя микроноля можно объявить один из входов, используемых в качестве микровыключателя ограничения перемещения в инструкции **MFC**.

direction - представляет собой направление поиска микровыключателя микроула. Если направление поиска микровыключателя положительное, параметр имеет значение 0, если отрицательное - 1;

speed - определяет скорость перемещения оси во время поиска микроула, выраженную в следующих единицах:

- линейная ось: мм/мин;
- ось вращения: градус/мин.

Скорость поиска должна быть достаточно низкой, чтобы избежать пропуска референтной метки с датчика на промежутке одного механического шага после съезда с микровыключателя микроула.

Пример

```
MCZ = I1A0, 0, 200
...
...
MFC = I1A0, I1A1;
```

Выход в микроула по выбранной оси с датчиком, имеющим референтную метку, в режиме работы «**ВЫХОД В НОЛЬ**» (HOME) можно условно разбить на три этапа:

Этап 1. Поиск позиции микровыключателя микроула. Для начала выполнения этого этапа состояние сигнала, записанного в параметре **zero_input** в инструкции **MCZ**, должно быть высокого уровня «1».

Способ выбора скорости поиска микровыключателя на этом этапе определяется состоянием переменной **RAP**, установленной из конфигурации в файле **PGCFIL** в инструкции **ASS** или оператором в процессе работы перед началом цикла «**ВЫХОД В НОЛЬ**»:

RAP=1 - переключатель корректора подач «**JOG**» в режиме работы «**ВЫХОД В НОЛЬ**» (HOME) отключён:

- скорость устанавливается **SW** равной максимальной скорости ручных перемещений, записанной в файле **AXCFIL** в инструкции **MAN** для выбранной оси;
- направление движения устанавливается **SW** на основе значения параметра **direction**, записанного в инструкции **MCZ** для выбранной оси;

RAP=0 - переключатель корректора подач «**JOG**» в режиме работы «**ВЫХОД В НОЛЬ**» (HOME) подключён:

- скорость устанавливается оператором выбором позиции переключателя корректора подач «**JOG**»;
- направление движения устанавливается оператором выбором позиции переключателя корректора подач «**JOG**» на основании значения параметра **direction**, записанного в инструкции **MCZ** для выбранной оси.

Примечание. Этап 1 **SW** не выполняет, если цикл «**ВЫХОД В НОЛЬ**» начинается при низком уровне состояния **zero_input**.

Этап 2. Вход в позицию и выход из позиции микровыключателя микроула. При входе в позицию микровыключателя микроула состояние сигнала, записанного в параметре

zero_input в инструкции **MCZ**, должно быть низкого уровня «0». На данном этапе **SW** выполняет автоматически следующие действия:

- 1) останавливает ось с контролируемым торможением;
- 2) устанавливает направление движения оси, противоположное заявленному в инструкции **MCZ** для выбранной оси;
- 3) разгоняет ось до скорости поиска микроула, заявленной в инструкции **MCZ**;
- 4) устанавливает ось в позицию микровыключателя микроула (низкий уровень его входа). Данное действие выполняется, если при останове оси длины микровыключателя не хватило, и ось проскочила его позицию;
- 5) ожидает выхода оси с позиции микровыключателя микроула на скорости поиска микроула.

Этап 3. Вход в позицию референтной метки датчика. При выходе из позиции микровыключателя микроула состояние сигнала, записанного в параметре **zero_input** в инструкции **MCZ**, должно быть высокого уровня «1». На данном этапе **SW** выполняет автоматически следующие действия:

- 1) выполняет поиск позиции референтной метки датчика на скорости поиска микроула. Позиция референтной метки датчика определяется модулем датчиков УЧПУ, который сообщает о моменте её появления в **SW**, поэтому важно, чтобы сигнал референтной метки датчика имел правильную фазировку по отношению к другим сигналам этого датчика см. «Руководство по эксплуатации».

Примечание.

1. Расстояние на оси между микровыключателем микроула и позицией ближайшей к нему референтной метки датчика, расположенной в направлении ее поиска, должно быть достаточным, чтобы зона срабатывания микровыключателя не влияла на поиск референтной метки.
2. Выход в микроула в режиме работы «ВЫХОД В НОЛЬ» (НОМЕ) по оси с кодом 10000H в инструкции **TPA**, которая не имеет датчика в случае использования шагового двигателя или имеет датчик без референтной метки, выполняется по этим же этапам. Позиция микроула для подобных осей устанавливается в **SW** на этапе 3 в момент выхода с микровыключателя микроула, который соответствует изменению входа микровыключателя с низкого уровня на высокий уровень.
- 2) После определения позиции референтной метки датчика позиция микроула оси может быть сдвинута на значение равное первому параметру в инструкции **ZNO**;
- 3) после вывода оси в заданную позицию **SW** активизирует для данной оси значение смещения нуля, записанное в нулевой начальной точке файла начальных точек.

4.2.4.6. Инструкция **MCZ** (**TPA=...+40000H**)

Если ось в инструкции **TPA** дополнительно определена кодом **40000H**, то инструкция **MCZ** для неё должна иметь вид, указанный ниже.

Семантика:

MCZ=zero_input, direction, speed, HDBRP, Δs .

Формат записи:

MCZ=PLC variable,boolean (2 цифры),real,real,real ,

где:

zero_input - этот параметр для датчиков с **ККРМ** должен отсутствовать.

direction - представляет собой направление поиска микронюля. Если направление поиска микронюля положительное, параметр имеет значение «0», если отрицательное – «1»;

speed - определяет скорость перемещения оси (**V**) во время поиска референтной метки, выраженную в мм/мин.
Максимальная скорость поиска референтной метки рассчитывается по формуле:

$$V \leq (\Delta s * 60 * 1000) / (2 * CPU_tick), \quad (4.9)$$

где:

V - максимальная скорость поиска референтной метки, мм/мин;

Δs - значение изменения расстояния между соседними референтными метками, мм;

CPU_tick - значение **CPU_tick** в инструкции **TIM**, мс

Пример

$$V \leq (0.02 * 60 * 1000) / (2 * 2) = 300 \text{ мм/мин};$$

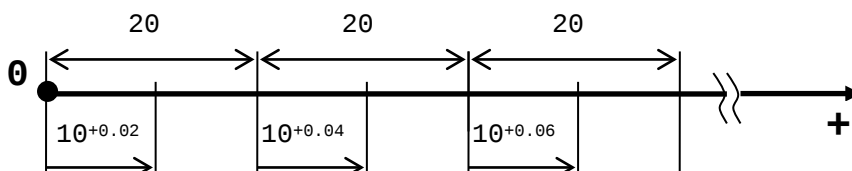
HDBRP - определяет половину расстояния между соседними базовыми референтными метками (по умолчанию 20 мм). Расстояние между соседними базовыми референтными метками указывается в паспорте датчика. Для линейных датчиков до 5000 мм значение **HDBRP** обычно равно 20 мм;

ΔS - определяет значение изменения расстояния между соседними референтными метками; указывается в паспорте линейки. Для линейных датчиков до 5000 мм значение **ΔS** равно 0.02 мм.

Знак «-» для значения этого параметра записывается в том случае, если направление счета датчика, установленного на станке, не совпадает с направлением отсчета его референтных меток.

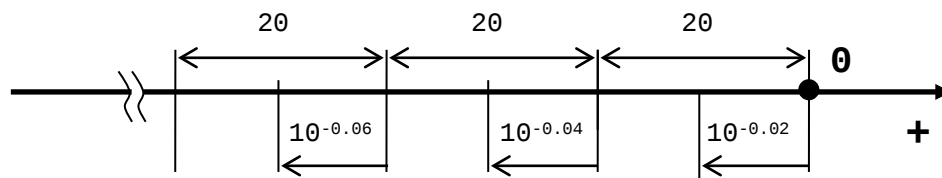
Примечание В случае применения датчика с ККРМ расстояние от нуля линейки до точки микронюля оси на станке определяется в параметре **null_offset** инструкции **ZNO**.

Пример 1. Направление изменения **ΔS** совпадает с направлением оси.



MCZ=, 0, 300, 10, 0.02

Пример 2. Направление изменения ΔS обратное к направлению оси.



MCZ=, 0, 300, 10, -0.02

4.2.4.7. Инструкция POS

Инструкция **POS** устанавливает параметры допуска позиционирования для текущей оси.

Семантика:

POS=*pos_allow*,*wait_pos* .

Формат записи:

POS=real,real ,

где:

pos_allow - представляет расстояние от теоретической точки позиционирования, в пределах которого движение оси будет считаться завершенным. Значение выражается в следующих единицах:

- линейная ось: мм или дюйм,
- ось шпинделя: оборот,
- ось вращения: градус,
- ось «от точки к точке»: позиция.

Если объявляемое значение допуска меньше минимальной разрешающей способности измерительной системы оси, определяемой отношением:

$$\frac{mec_pitch}{el_pitch},$$

то движение оси будет выполняться без контроля точность её позиционирования;

wait_pos - определяет максимальное время ожидания входа оси в допуск позиционирования, выраженное в секундах. По истечении этого времени, если ось не находится в допуске позиционирования, УЧПУ выключает станок и выдаёт сигнал об ошибке. Для оси шпинделя этот параметр не записывается.

4.2.4.8. Инструкция SRV

Инструкция **SRV** устанавливает значения ошибок (аномалии) привода для текущей оси.

Семантика:

SRV=servo_error_stand_by,servo_error_VFF,servo_error_no_VFF .

Формат записи:

SRV=real,real,real ,

где:

servo_error_stand_by - это максимально допустимая ошибка, когда ось находится в останове (в состоянии **STANDBY**); выражается в миллиметрах, дюймах, градусах, позициях;

servo_error_VFF - это максимально допустимое значение рассогласования, когда ось находится в движении при активной скоростной компенсации (с **VFF**); выражается в миллиметрах, дюймах, градусах, позициях;

servo_error_no_VFF - это максимально допустимое значение рассогласования, когда ось находится в движении без скоростной компенсации (без **VFF**); выражается в миллиметрах, дюймах, градусах, позициях.

ВНИМАНИЕ! - для оси шпинделя и осей «от точки к точке» эту инструкцию надо записывать: **SRV=0,0,0 .**

4.2.4.9. Инструкция MAN

Инструкция **MAN** используется для определения параметров скорости и ускорения в режиме «**РУЧНЫЕ ПЕРЕМЕЩЕНИЯ**» для текущей оси; для оси шпинделя не записывается.

Семантика:

MAN=max_man_speed,max_accel .

Формат записи:

MAN=real,real ,

где:

max_man_speed - определяет скорость быстрого хода в режиме «**РУЧНЫЕ ПЕРЕМЕЩЕНИЯ**», выраженную в следующих единицах:

- линейная ось: мм/мин;
- ось вращения: градус/мин;
- ось «от точки к точке»: позиция/мин;

max_accel - определяет величину ускорения быстрого хода в режиме «РУЧНЫЕ ПЕРЕМЕЩЕНИЯ» и ускорения в кадрах, выполняемых с одной из функций: **G01, G02, G03, G06**; величина ускорения выражена в следующих единицах:

- линейная ось: мм/с²;
- ось вращения: градус/с²;
- ось «от точки к точке»: позиция/с².

Примечание. Значения **max_man_speed** и **max_accel** не должны превышать соответствующие им значения **fast_speed** и **fast_accel**, объявленные в инструкции **RAP**.

4.2.4.10. Инструкция **MSS**

Инструкция **MSS** предназначена для определения минимальной скорости высокоскоростного электрошпинделя.

Семантика:

MSS=Vmin .

Формат записи:

MSS=real ,

где:

Vmin - определяет минимальную скорость вращения высокоскоростного электрошпинделя при напряжении в канале ЦАП, равном 0 В; значение скорости выражено в об/мин.

4.2.4.11. Инструкция **GMXX**

Инструкция **GMXX** определяет параметры скорости быстрого хода и коэффициент усиления для текущей оси.

Семантика:

GMxx=Vmax, KC, KV .

Формат записи:

GMxx=real, real, real ,

где:

XX - для оси шпинделя значение **XX** определяет номер конфигурируемого диапазона (от 1 до 99). Для остальных типов осей **XX** равно 0;

Для оси шпинделя следует объявлять столько инструкций **GMxx**, сколько существует диапазонов шпинделя.

Vmax 1) определяет максимально допустимую скорость, выраженную в следующих единицах:

- линейная ось: мм/мин;
- ось вращения: градус/мин;
- ось «от точки к точке»: позиция/мин

2) определяет максимальную скорость текущего диапазона шпинделя, выраженную в об/мин;

3) определяет максимальную скорость вращения высокоскоростного электрошпинделя, которая рассчитывается по формуле:

$$V_{max} = V - V_{min},$$

где:

V - значение максимальной скорости электрошпинделя по его паспорту;

V_{min} - значение минимальной скорости электрошпинделя по его паспорту, определённое в инструкции **MSS**.

KC

- определяет эталонное напряжение в вольтах (В), соответствующее максимальной скорости оси **V_{max}**; считается целесообразным объявлять это значение:

- для линейных осей: 7.5 В;
- для оси шпинделя: 8.5 В.

Этот параметр необходимо объявлять и для оси шпинделя с двигателем переменного тока; рекомендуемое значение: 7.5 В;

KV

- определяет константу усиления в сек⁻¹; обычно **KV = 20**.

KV шпинделя используется при его позиционировании (ориентации). Минимальное значение **KV**, необходимое для ориентации шпинделя, можно определить по следующей формуле:

$$KV = KC / (pos_allow * 360) ,$$

где:

KC - эталонное напряжение (В);

pos_allow - допуск ориентации шпинделя (оборот).

Для шпинделя без датчика параметр **KV** не используется **SW** УЧПУ, поэтому можно записать **KV=0**.

4.2.4.12. Инструкция GMXX (ECDF)

Инструкция **GMxx** определяет параметры скорости быстрого хода, коэффициент усиления и параметры контроля нулевой скорости для текущей оси.

Семантика:

GMxx=V_{max}, KC, KV, limit_speed, input_speed_limit

Формат записи:

GMxx=real,real,real,real,PLC variable ,

где:

xx - для оси шпинделя значение **xx** определяет номер конфигурируемого диапазона (от 1 до 99). Для остальных типов осей **xx** равно **0**;

Для оси шпинделя следует объявлять столько инструкций **GMxx**, сколько существует диапазонов шпинделя.

Vmax - см. описание **Vmax** в инструкции **GMxx** без метки **ECDF**;

KC - см. описание **KC** в инструкции **GMxx** без метки **ECDF**;

KV - см. описание **KV** в инструкции **GMxx** без метки **ECDF**;

limit_speed - определяет пороговое значение скорости оси, на основании которого **SW** для ПЛ формирует значение **input_speed_limit**. Пороговая скорость выражена в следующих единицах:

- линейная ось: мм/мин,
- ось вращения: градус/мин,
- ось шпинделя с датчиком: об/мин;
- ось шпинделя без датчика: Вольт.

Для оси шпинделя с датчиком значение **limit_speed** используется для контроля состояния «шпиндель остановлен», поэтому оно должно превышать скорость шпинделя при нулевом задании. Значение **limit_speed** может быть установлено индивидуально для каждого диапазона или только для первого диапазона. В последнем случае заданное значение для первого диапазона будет использоваться **SW** для контроля нулевой скорости всех заявленных диапазонов текущей оси шпинделя.

Для оси шпинделя без датчика пороговое напряжение для всех его диапазонов задаёт **SW** постоянным значением 39.04 мВ.

input_speed_limit - определяет идентификатор свободного сигнала **PLC** пакетов «K» или «N», назначаемого самим пользователем. Состояние сигнала устанавливает **SW** на основании значения, заданного в **limit_speed**. Если текущее значение скорости оси меньше, чем значение **limit_speed**, то состояние сигнала **PLC** будет установлено **SW** равным «1».

Для оси шпинделя без датчика **SW** динамически устанавливает состояние данного сигнала равным «1», если напряжение в её канале ЦАП меньше 39.04 мВ.

Для оси шпинделя с датчиком сигналы могут быть записаны в инструкции **GMxx** для каждого диапазона или только для первого диапазона. В последнем случае сигнал, записанный для первого диапазона, будет

использован для всех диапазонов, которые определены для текущей оси шпинделя.

4.2.4.13. Инструкция FRC (ECDF)

Инструкция **FRC** предназначена для определения параметров функции компенсации силы трения покоя на круговых контурах, а также для определения коэффициентов скоростной компенсации и интегральной составляющей контура управления координатной осью.

Семантика:

FRC=t_ffa, ΔVmax, ΔVmin, A1, A2, A3, kvff, Kvffi, input_ffa .

Формат записи:

FRC=word, real, real, real, real, real, real, real, PLC variable ,

где:

- t_ffa** - определяет время действия параметров функции компенсации силы трения покоя (**ΔVmax, ΔVmin, A1, A2, A3**), при смене направления движения оси на круговом контуре. Значение **t_ffa** (friction force alone) выражено в мс;
- ΔVmax** - определяет значение максимальной величины компенсации силы трения покоя: **ΔVmax**, (мм/мин). Значение **ΔVmax** может иметь знак;
- ΔVmin** - определяет значение минимальной величины компенсации силы трения покоя: **ΔVmin**, (мм/мин). Значение **ΔVmin** может иметь знак;
- A1** - определяет границу ускорения **A1**, (мм/с²);
- A2** - определяет границу ускорения **A2**, (мм/с²);
- A3** - определяет границу ускорения **A3**, (мм/с²);
- Kvff** - определяет значение коэффициента скоростной составляющей в контуре управления осью (по умолчанию равен «1»). Значение **Kvff** используется в **SW** только при выполнении движения, запрограммированного в кадре, и значении переменной **UEP** равной **0**;
- Kvffi** - определяет значение коэффициента интегральной составляющей ПИД-регулятора (по умолчанию равен «0»). Значение **Kvffi** используется в **SW** только при выполнении движения, запрограммированного в кадре, и значении переменной **UEP** равной **0**;
- input_ffa** - определяет идентификатор свободного сигнала **PLC** пакетов «К» или «N», назначаемого самим пользователем. Этот сигнал является запросом включения функции компенсации силы трения покоя. Функция компенсации

силы трения покоя включена, если сигнал запроса равен «1» и выключена, если он равен «0».

Каждая ось, содержащая инструкцию **FRC**, может иметь индивидуальный сигнал запроса включения функции компенсации трения покоя.

Примечание. Действие параметра **Kvff** и **Kvffi** не зависит от состояния сигнала PLC, объявленного в **input_ffa**.

Пример

NAS = X

.....

GM0 = ..., ..., ...

FRC = 10 , 50, 10, 50, 100, 300, 0.95, 0, U190K30

При переходе из квадранта в квадрант по круговому контуру оси имеют точки смены направления движения. Пример точек смены направления движения осей приведён на рисунке 4.9. В этих точках скорость движения оси равна нулю, и дальнейшее движение оси потребует преодоления силы трения покоя в механизмах станка (в редукторах, в направляющих осей), что приведёт в этот момент к увеличению ошибки кругового контура. Правильно подобранная компенсация силы трения покоя уменьшит эту ошибку и улучшит точность кругового контура. Подбор параметров для компенсации трения выполняется на осциллографе ошибки кругового контура при выполнении кадра круговой интерполяции. Это осуществляется в режиме осциллографа, который описан в документе «Руководство оператора».

Сила трения покоя в точках смены направления движения оси компенсируется кратковременным увеличением ее скорости. Компенсация силы трения покоя выполняется независимо от скорости, рассчитанной интерполятором для движения оси по контуру.

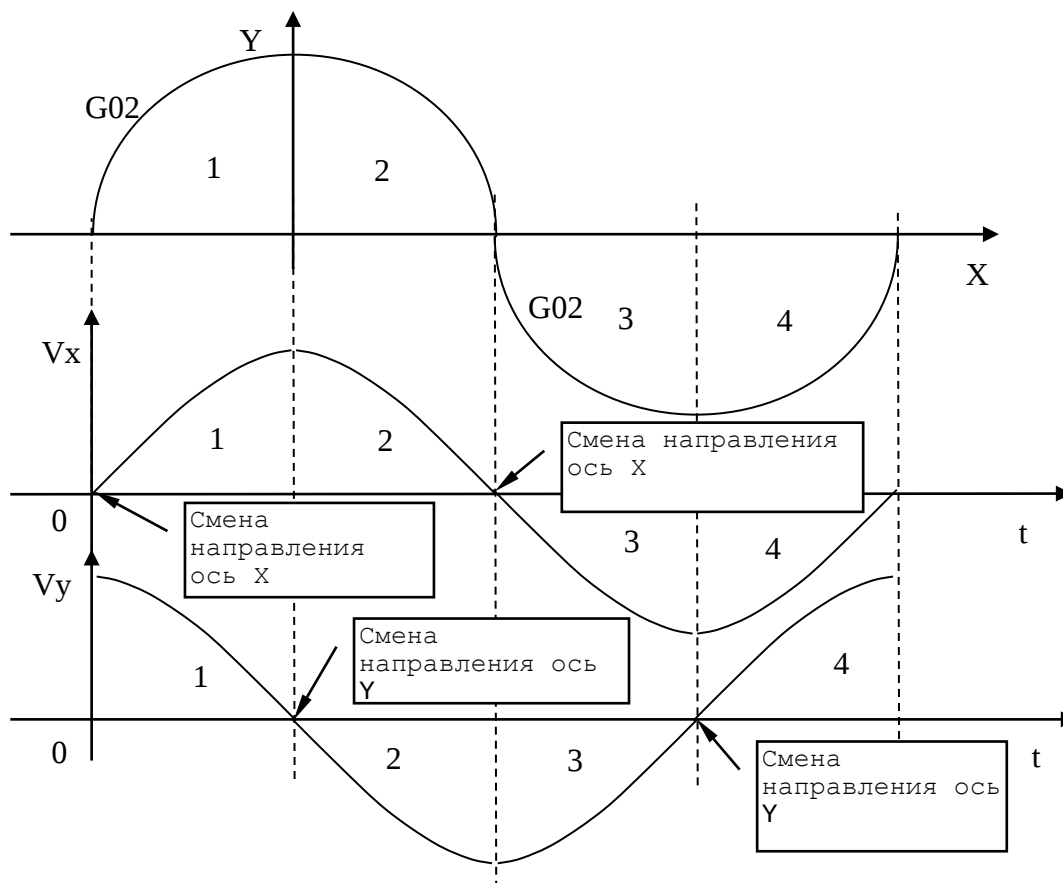


Рисунок 4.9

В простейшем случае для работы функции компенсации силы трения покоя достаточно задать значения для ее параметров **t_{ffa}** и **ΔV_{min}** (значения параметров **ΔV_{max}** , **$A1$** , **$A2$** , **$A3$** равны «0»).

В некоторых случаях значение величины компенсации трения покоя не постоянно на всём диапазоне ускорений, получаемых на разных скоростях и радиусах круга. Так, например, для большого значения ускорения требуется меньшая компенсация силы трения покоя. Для оптимизации коррекции силы трения покоя весь диапазон ускорений можно разбить максимально на 4 интервала, устанавливая граничные значения в **параметры** (**ΔV_{max}** , **ΔV_{min}** , **$A1$** , **$A2$** , **$A3$**).

Значение величины компенсации трения покоя для каждого интервала в **SW** рассчитываются по формулам, как показано на рисунке 4.10.

- 1: $\Delta V = \Delta V_{max} * A / A1$
- 2: $\Delta V = \Delta V_{max}$
- 3: $\Delta V = \Delta V_{max} * (1 - (A - A2) / (A3 - A2))$
- 4: $\Delta V = \Delta V_{min}$

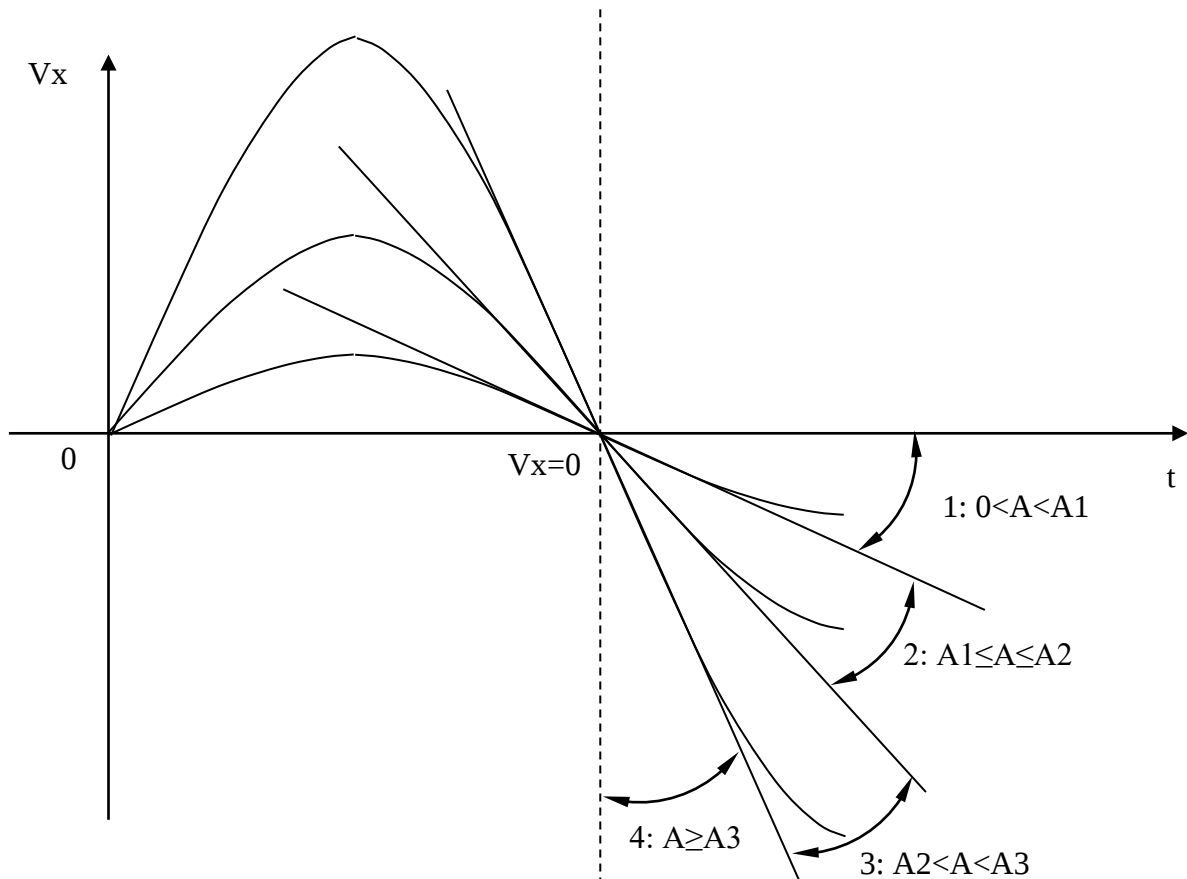


Рисунок 4.10

4.2.4.14.

Инструкция LOP

Инструкция **LOP** определяет границы оперативной зоны перемещений для текущей оси.

Семантика:

LOP=pos_op_limit,neg_op_limit

Формат записи:

LOP=real,real ,

где:

pos_op_limit - определяет собой границу оперативной зоны оси в положительном направлении относительно ее абсолютного микроула; значения выражаются в следующих единицах:

- линейная ось: мм;
- ось вращения: градус;

neg_op_limit - определяет собой границу оперативной зоны оси в отрицательном направлении относительно ее абсолютного микроула; значения выражаются в следующих единицах:

- линейная ось: мм;
- ось вращения: градус.

Если обе величины **pos_op_limit** и **neg_op_limit** равны 0, то контроль границ оперативной зоны перемещений для текущей оси не выполняется.

4.2.4.15. Инструкция LOn (ECDF)

Инструкция **L0x** предназначена для определения границ двух оперативных зон перемещений для текущей оси. Переключение контроля границ одной зоны на другую производится через интерфейсный выходной сигнал **PLC** пакета «N» (**U10N1**).

Семантика:

L0x=pos_op_limit,neg_op_limit,space

Формат:

L0x=real,real,real ,

где:

x - определяет номер оперативной зоны перемещений (**x** может принимать значения 1 или 2);

pos_op_limit - определяет границу оперативной зоны **x** в положительном направлении оси; значения выражаются в следующих единицах:

- линейная ось: мм,
- ось вращения: градус;

neg_op_limit - определяет границу оперативной зоны **X** в отрицательном направлении оси; значения выражаются в следующих единицах:

- линейная ось: мм,
- ось вращения: градус.

Если обе величины **pos_op_limit** и **neg_op_limit** равны **0**, то контроль границ оперативной зоны перемещений **X** не выполняется.

space - определяет расстояние, откладываемое от координаты границы зоны в направлении движения оси, на которое ей допускается выходить из зоны; значения выражаются в следующих единицах:

- линейная ось: мм,
- ось вращения: градус.

Инструкция **L01** обычно определяет границы основной зоны перемещений текущей оси. Контроль границ зоны **L01** активен, если сигнал PLC **U10N1** равен **0**.

Инструкция **L02** обычно определяет границы вспомогательной зоны перемещений текущей оси. Контроль границ зоны **L02** активен, если сигнал PLC **U10N1** равен **1**.

Пример.

Запись инструкций **L01** и **L02** для оси **X**.

NAS=X

.....

L01=+5, -1500, 20

L02=-1100, -1700, 12

Схема расположения границ оперативных зон **L01** и **L02**, записанных в примере, представлена на рисунке 4.11:

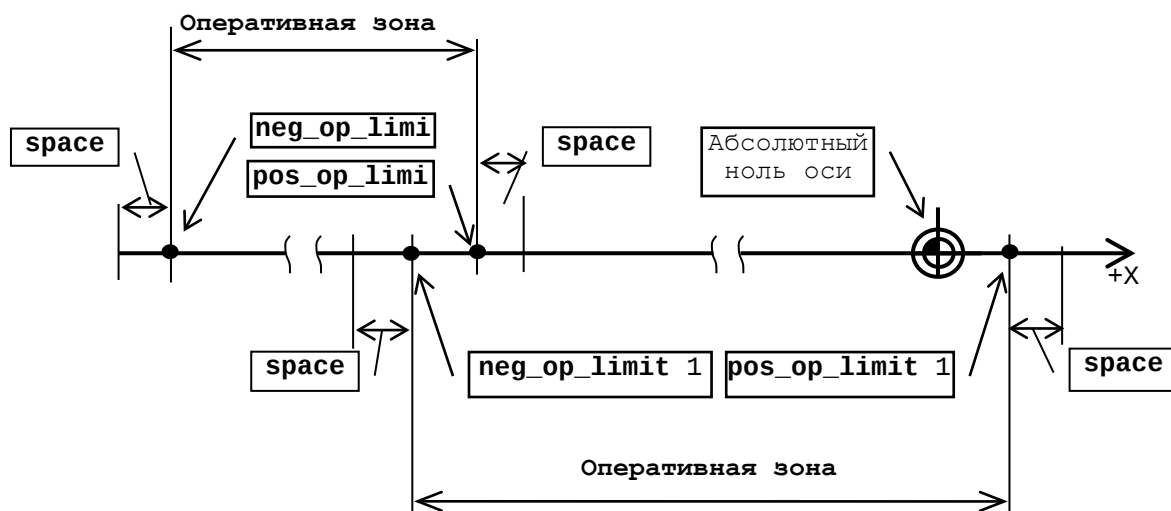


Рисунок 4.11

4.2.4.16.

Инструкция MFC

Инструкция **MFC** используется для определения входов физических ограничителей перемещений для текущей оси.

Семантика:

MFC=input_pos_op_limit,input_neg_op_limit .

Формат записи:

MFC=PLC variable,PLC variable ,

где:

input_pos_op_limit - определяет собой идентификатор входного сигнала PLC от физического ограничителя перемещения, расположенного в положительном направлении оси;

input_neg_op_limit - определяет собой идентификатор входного сигнала PLC от физического ограничителя перемещения, расположенного в отрицательном направлении оси.

Рекомендуется:

- устанавливать физический ограничитель с НЗК, т.е. в исходном состоянии его входной сигнал должен иметь высокий уровень и иметь низкий уровень при входе на физический ограничитель;
- устанавливать физический ограничитель такой длины, которая позволяла бы выполнять останов оси, оставаясь на нем;
- записывать для каждой характеризуемой оси инструкцию **AFC**, позволяющую при входе оси на физический ограничитель устанавливать аварийное значение ускорения.

Пример

MFC=I00A0,I00A1

4.2.4.17.

Инструкция AFC

Инструкция **AFC** используется для установки значения аварийного ускорения.

Семантика:

AFC=acc_MFC .

Формат записи:

AFC=real ,

где:

acc_MFC - определяет величину ускорения для останова оси при её входе на физические ограничители перемещений **input_pos_op_limit** или **input_neg_op_limit**, указан-

ные в инструкции **MFC** для текущей оси; величина ускорения выражена в следующих единицах:

- линейная ось: мм/с²;
- ось вращения: градус/с²;
- ось «от точки к точке»: позиция/с².

4.2.4.18. Инструкция CUB (ECDF)

Инструкция **CUB** предназначена для задания границ объёмной защищённой зоны «**CUB**». Инструкция **CUB** должна быть записана для 3-х линейных координатных осей станка.

Семантика:

CUB=pos_limit_cub,neg_limit_cub,space .

Формат записи:

CUB=real,real,real ,

где:

pos_limit_cub - определяет границу зоны «**CUB**» в положительном направлении относительно абсолютного микроула текущей оси, участвующей в определении этой зоны; значение выражается в мм или дюймах;

neg_limit_cub - определяет границу зоны «**CUB**» в отрицательном направлении относительно абсолютного микроула оси, участвующей в определении этой зоны; значение выражается в мм или дюймах;

space - определяет расстояние, отложенное внутрь защищённой зоны «**CUB**», по текущей оси, на которое ей разрешается заходить в зону «**CUB**»; мм или дюйм.

ПРИМЕЧАНИЯ

1. Инструкцию **CUB** можно не записывать для оси, если данная ось не участвует в определении объёмной защищённой зоны.
2. Подключение и отключение контроля защищённой зоны «**CUB**» выполняется интерфейсным сигналом PLC «**CUB**» пакета «**N**» (**U10N0**).
3. Имя оси, для которой задана инструкция **CUB**, должно быть записано в секции 5 файла **PGCFIL** в инструкции **CAH**.
4. Инструкция **CUB** не может быть записана для осей, указанных в таблице 4.2.

Таблица 4.2

16-тичный код	Бит назначения	Характеризуемая ось
0002	1	Ось «от точки к точке»
0004	2	Ось вращения
0010	4	Ось шпинделя без датчика
0020	5	Ось шпинделя с датчиком
0080	7	Ось с контрольной точкой
0100	8	Виртуальная ось
0800	11	Ось шпинделя с контролируемым разгоном

Продолжение таблицы 4.2

16-теричный код	Бит назначения	Характеризуемая ось
1000	12	Подчинённая параллельная ось
2000	13	Ось вращения, позиционируемая от 0 до 359,999 градусов.
4000	14	Ось шпинделя с двигателем переменного тока

Пример. Определение объемной защищенной зоны в осях X, Y, Z.
Файл AXCFIL секция 2:

```
*2
PR0=1
NAS=X
TPA=1,
CUB=-50, -100, 0.5
.....
NAS=Y
TPA=1,
CUB=-50, -80, 0.5
.....
NAS=Z
TPA=1,
CUB=-50, -120, 0.5
.....
```

Файл PGCFIL секция 5:

```
*5
PR0=1
.....
CAX=X, Y, Z
```

4.2.4.19. Инструкция SWX (ECDF)

Инструкция **SWX** предназначена для определения до восьми программных контролируемых зон (кулачков) для текущей оси.

Семантика:

SWx=pos_limit_SW, neg_limit_SW, input_SW .

Формат:

SWx=real, real, PLC variable ,

где:

x - номер программной контролируемой зоны ($x=1\div 8$);

pos_limit_SW - определяет границу контролируемой зоны в положительном направлении относительно абсолютного микро-нуля оси; значения выражаются в следующих единицах:

- линейная ось: мм,
- ось вращения: градус,
- ось «от точки к точке»: позиция,

- ось шпинделя: градус;

neg_limit_SW - определяет границу контролируемой зоны в отрицательном направлении относительно абсолютного микро-нуля оси; значения выражаются в следующих единицах:

- линейная ось: мм,
 - ось вращения: градус,
 - ось «от точки к точке»: позиция,
 - ось шпинделя: градус;

input_SW - определяет идентификатор свободного сигнала PLC пакетов «K» или «N», назначаемого самим пользователем. **input_SW** устанавливается из **SW** равным 1, если координата текущей оси находится в промежутке, ограниченном параметрами **pos_limit_SW** и **neg_limit_SW**, и сбрасывается в 0 при выходе из него.

Примечание. Инструкция **SWn** не программируется для осей, указанных в таблице 4.3.

Таблица 4.3

16-тичный код	Бит назначения	Характеризуемая ось
0010	4	Ось шпинделя без датчика
0100	8	Виртуальная ось
1000	12	Подчинённая параллельная ось

Пример

NAS=X

.....

SW1=-49.9999, -79.9999, I109K30

SW2=79.9999, 49.9999, I109K31

4.2.4.20. Инструкция TSM

Инструкция **TSM** предназначена для определения параметров инверсии текущей оси шпинделя.

Семантика:

TSM=max_inver_time,max_inver_ref .

Формат записи:

TSM=real,real ,

где:

max_inver_time - устанавливает максимальное время, необходимое шпинделю для инверсии от предельной скорости вращения по часовой стрелке до предельной скорости вращения против часовой стрелки; время выражается в секундах и/или в долях секунды;

max_inver_ref - устанавливает максимальный диапазон напряжения в процессе инверсии шпинделя; практически это - значение **КС** шпинделя (инструкция **GMxx**), умноженное на два.

При отключении вращения шпинделя с датчиком и ЦАП (**TPA=820,**) по командам **M05** и «**ОБЩИЙ СБРОС**» контролируемое торможение выполняется до скорости вращения, указанной в **limit_speed** инструкции **POM**, после чего на ЦАП шпинделя устанавливается **0 В**.

При отключении вращения шпинделя с ЦАП, но без датчика (**TPA=810,**) по командам **M05** и «**ОБЩИЙ СБРОС**» контролируемое торможение выполняется до скорости, соответствующей 39,06 мВ, после чего на ЦАП шпинделя устанавливается **0 В**.

ВНИМАНИЕ! параметры инструкции **TSM** используются для синхронизации инверсии вращения шпинделя при нарезании резьбы метчиком (**G84**) и остановки шпинделя в цикле **G86**, поэтому целесообразно очень внимательно измерять значение **max_inver_time** для ввода в инструкцию **TSM**.

Инструкция **TSM** должна быть определена и в случае объявления шпинделя с двигателем переменного тока (16-теричный код 4000H в инструкции **TPA**).

Пример
TSM = 3,17

4.2.4.21. Инструкция **ASM**

Инструкция **ASM** объявляет главную ось, которая с текущей осью, являющейся для неё подчинённой, используется при выполнении определённых технологических циклов.

Семантика:

ASM=master_axis .

Формат записи:

ASM=ASCII string ,

где:

master_axis - наименование главной оси, записанное строковым символом.

Эта инструкция записывается в следующих случаях:

- 1) в подсекции оси шпинделя для токарных станков. В этом случае главная ось объявляется для обеспечения постоянства скорости резания с функцией **G96**; обычно, **ASM=X**;
- 2) в подсекции осей с контрольной точкой (ось **C** токарных станков - см. инструкцию **TPA**), обычно записывается: **ASM=S**. В этом случае главная ось **S** объявляется для управления осью шпинделя в координатном режиме, обеспечивая фрезерный режим работы на токарном станке;

- 3) в подсекции оси вспомогательного шпинделя с ЦАП (поворотный стол) для фрезерных станков. В этом случае главная ось объявляется для обеспечения постоянства скорости резания с функцией **G96**; обычно, **ASM=Z**.

Пример

```

.....
NAS=C
TPA=85,
NTC=3,3
.....
.....
ASM=S
.....
NAS=S
TPA=820,
NTC=3,3
.....
.....
ASM=X

```

4.2.4.22. Инструкция POM

Инструкция **POM** используется для установки параметров позиционирования текущей оси шпинделя.

Семантика:

POM=spiaz, limit_speed .

Формат записи:

POM=real, real ,

где:

spiaz - определяет угол в оборотах между электрическим нулём датчика и положением ориентации шпинделя; ПЛ может запрашивать угол ориентации отличный от значения **spiaz**;

limit_speed - определяет пороговую скорость вращения шпинделя.
Эта скорость будет активизироваться алгоритм ориентации шпинделя, т.е. будет выдаваться эталонное напряжение; единица измерения – обороты в минуту (об/мин); обычно эта скорость устанавливается от 20 до 100 об/мин и является рабочей при ориентации шпинделя.

Значение **limit_speed** может быть использован в ПЛ для контроля состояния «шпиндель включен» **ABIM**, поэтому оно должно превышать скорость шпинделя при нулевом задании.

Примечание. Эта инструкция записывается только для шпинделя с датчиком.

4.2.4.23. Инструкция POM (ECDF)

Инструкция **POM** используется для установки параметров позиционирования оси шпинделя, а также параметров контролирования выхода на заданную скорость вращения.

Семантика:

POM=spiaz, limit_speed, max_deviation_speed, input_speed

Формат записи:

POM=real, real, real, PLC variable ,

где:

spiaz - определяет угол в оборотах между электрическим нулём датчика и положением ориентации шпинделя; ПЛ может запрашивать угол ориентации отличный от значения **spiaz**;

limit_speed - определяет пороговую скорость вращения шпинделя, при которой будет активизироваться алгоритм ориентации шпинделя, т.е. будет выдаваться эталонное напряжение; единица измерения – обороты в минуту (об/мин); обычно эта скорость устанавливается от 100 до 200 об/мин и является рабочей при ориентации шпинделя.

Значение пороговой скорости, которое используется для контроля останова шпинделя, записывается в 4-ом параметре инструкции **GM1**.

max_deviation_speed - определяет коэффициент допустимого отклонения заданной скорости вращения шпинделя с учётом положения корректора скорости вращения шпинделя; может принимать значения от **0** до **1**.

input_speed - определяет идентификатор свободного сигнала **PLC** пакетов «K» или «N», назначаемого самим пользователем.

input_speed = «1», если скорость шпинделя равна заданной скорости с учетом допустимого отклонения **max_deviation_speed**.

В ПЛ состояние **input_speed** должно анализироваться совместно с **input_speed_limit**, записанного в инструкции **GMxx**. Контроль сигнала **input_speed** в ПЛ должен быть отключен, если интерфейсный сигнал **I9K26(FG96)** имеет состояние «1».

Примечание Эта инструкция записывается только для шпинделя с датчиком.

Например, если допуск по скорости равен 0.2, скорость порога (**S**) равна 100 об/мин и значение корректора скорости вращения шпинделя равно 90%, то сигнал контроля скорости **U200K4** будет установлен в состояние «1» при скорости от 72 до 108 об/мин.

Пример

NAS = S

TPA = 820,

.....

GM1 = 1000, 7.5, 20, 3, U200K3

POM = 0.34, 20, 0.2, U200K4

4.2.4.24.**Инструкция ADP**

Инструкция **ADP** используется для установки параметров 3-х уровней нагрузки для оси шпинделя.

Семантика:

ADP=number_ADC, P_100, P_level_1, P_level_2 ,

Формат записи:

ADP=integer, real, real, real ,

где:

number_ADC - определяет номер канала АЦП в УЧПУ, к которому подключено внешнее устройство контроля нагрузки на шпинделе;

P_100 - определяет значение напряжения для 100% нагрузки на шпинделе; [Вольт]; может принимать значения от -10 до 10В.

P_level_1 - определяет предельное напряжение в канале АЦП для нормальной нагрузки на шпинделе; [Вольт]; может принимать значения от **0** до **|P_level_2|**. Если |напряжение| в канале АЦП меньше **P_level_1**, то индикация процента нагрузки на экране имеет зеленый цвет.

P_level_2 - определяет предельное напряжение в канале АЦП для повышенной нагрузки на шпинделе; [Вольт]; может принимать значения от **P_level_1** до **|P_100|**. Если |напряжение| в канале АЦП больше или равно **P_level_1**, но меньше **P_level_2**, то индикация процента нагрузки на экране имеет желтый цвет. **P_level_2** является критическим значением нагрузки на шпинделе, поэтому при превышении этого значения индикация процента нагрузки на экране имеет красный цвет.

Примечание .

- 1) Информация о текущем уровне нагрузки шпинделя устанавливается в ПЛ интерфейсными сигналами **SPWL1** и **SPWL2**.
- 2) индикация процента нагрузки на экране выводится на время, пока шпиндель включен.

4.2.4.25. Инструкция ZNO

Инструкция **ZNO** используется для записи параметров позиционирования текущей оси относительно позиции абсолютного микроула.

Семантика:

ZNO=null_offset, zero_shift, distance, input_distance, DBRP

Формат записи:

ZNO=real, real, real, PLC variable, real ,

где:

null_offset – определяет смещение нуля оси. Действие параметра **null_offset** в системе зависит от типа датчика оси.

Для инкрементального датчика без координатно-кодированных референтных меток **null_offset** это дистанция, на которую должна быть позиционирована ось в цикле выхода в абсолютную позицию микроула.

Для инкрементального датчика с координатно-кодированными референтными метками и абсолютного датчика с интерфейсом SSI **null_offset** это дистанция, на которую необходимо программно сместить ноль датчика, чтобы начало отчета датчика совпало с позицией физического нуля оси.

Значение **null_offset**, выражено в миллиметрах или дюймах;

zero_shift - не используется;

distance - определяет значение контролируемого расстояния, пройденного текущей осью; каждый раз после перемещения оси на заданное значение **distance SW** устанавливает значение **input_distance**, записанного в этой инструкции, в состояние «1»; контролируемое **SW** расстояние не зависит от направления движения оси; значение выражается в следующих единицах:

- линейная ось: мм или дюйм,
- ось «от точки к точке»: позиция,
- ось вращения: градус.

input_distance - определяет идентификатор свободного сигнала **PLC** пакетов «K» или «N», который будет устанавливаться в «1» каждый раз, когда осью будет пройден путь, равный значению **distance**. В случае необходимости циклической обработки данного сигнала в ПЛ следует выполнять сброс его в ноль каждый раз после прохождения осью заданного расстояния.

Пример

Для случая **ZNO=..., ..., 1500, U100K0, ...** в ПЛ для обнуления сигнала **U100K0** можно записать:

```
$
.....
```

P2=U100K0
U100K0=P2

DBRP - расстояние между референтными метками датчика; имеет два значения:

- 1) Для осей вращения с целым коэффициентом редукции и линейных осей **DBRP** - это расстояние в градусах или миллиметрах, соответствующее одному обороту датчика; в этом случае значение **DBRP** в инструкции не записывается.

Примечание. Если данный параметр не задан, алгоритм выхода в «ноль» выполняется с учётом **null_offset**.

- 2) Для оси вращения с нецелым коэффициентом редукции **DBRP** - это расстояние в градусах, на которое смещается референтная метка датчика при каждом следующем обороте оси. Расстояние определяется по формуле:

$$DBRP = 360 / k * (1 - \text{дробная часть } k) \quad (4.10)$$

Значение, рассчитанное по формуле 4.10, позволяет правильно выполнять повторный выход в микроноль с любого оборота оси вращения.

Пример

Коэффициент редукции $k = 4.645$.

Дробная часть коэффициента редукции равна 0.645.

Расстояние между референтными метками датчика:

$$360 / 4.645 * (1 - 0.645) = 27.513455 \text{ (градусов)}.$$

Запись инструкции:

NAS=X

.....

ZN0=-49.9999,,1500.0,U119K30,27.513455

4.2.4.26. Инструкция FBF

Инструкция **FBF** определяет параметры, используемые для сигнализации возможной ошибки датчика текущей оси.

Семантика:

FBF=space, time, input_break_wire, status_break_wire .

Формат записи:

FBF=real, real, PLC variable, boolean ,

где:

space - область, в которой не генерируется сигнал «Неисправность датчика»; измеряется в миллиметрах или дюймах;

time - время, в течение которого генерируется сигнал «Неисправность датчика»; измеряется в секундах (с);

input_break_wire - этот параметр определяет идентификатор сигнала на языке **PLC** для установки в системе состояния разрешения или запрета контролирования обрыва сигналов датчика (A, /A, B, /B) или его питания, например: **U230K0**;

status_break_wire - состояние параметра **input_break_wire** («0» или «1»), при котором в системе разрешено контролирование обрыва сигналов датчика или его питания.

Примеры:

FBF=,,U230K0,1 - контроль обрыва сигналов датчика будет выполняться при условии состояния сигнала PLC **U230K0** равно «1».

FBF=,,U230K0,0 - контроль обрыва сигналов датчика будет выполняться при условии состояния сигнала PLC **U230K0** равно «0».

ВНИМАНИЕ !

1. ошибка «**неисправность датчика**» (сообщение 4_68) будет сформирована при одновременном выполнении четырех условий:

- если инструкция **FBF** отсутствует в подсекции оси, или если в инструкции **FBF** записано значение любого параметра **space** и/или **time** отличное от нуля;
- ось должна быть включена и находиться в движении;
- при движении оси в УЧПУ не приходят сигналы с ее датчика по истечении времени, указанного в параметре **time**;
- при движении оси изменение ее расчетной позиции с момента последнего изменения показания ее датчика должно превысить значение, указанное в параметре **space**.

2. при отсутствии инструкции **FBF** или пропуске значения **input_break_wire** аппаратный контроль обрыва сигналов или питания датчика для УЧПУ **NC-200, NC-201, NC-201M, NC-210, NC-220, NC-230, NC-310** выполняется, а для УЧПУ **NC-110** не выполняется.

3. если инструкция **FBF** не записана, по умолчанию параметры принимают следующие значения:

- **space** = значение механического шага, записанное во втором параметре инструкции **PAS**;
- **time** = 400 тиков, указанных в инструкции **CAS**, например: **CAS=1,XYZS,2**. В этом примере **servo_tick** обслуживания привода равно 2 мс, поэтому значение параметра **time** равно 2*400=800 мс (0.8 секунд).

4. если в подсекции текущей оси в инструкции **FBF** оба параметра **space** и **time** равны нулю, то ошибка неисправности датчика для этой оси не генерируется.

Примечание. Состояние контролирования обрыва сигналов датчиков для всех координатных непрерывных осей рекомендуется разрешать на все время работы станка. Состояние контролирования обрыва сигналов датчиков для всех вспомогательных осей, включая ось шпинделя, рекомендуется разрешать только в то время, когда координатные оси не выполняют обработку детали.

4.2.4.27.**Инструкция HWC**

Инструкция **HWC**, объявленная для текущей оси, привязывает её к штурвалу коррекции, который позволяет изменять позицию данной оси на станке без изменения значения её расчётно-контролируемой координаты, индицируемой на экране УЧПУ по команде **UCV=0**. Установленную от штурвала коррекцию позиции оси и её знак можно определить внешним измерительным прибором и по индикации на экране УЧПУ после ввода команды **UCV=5**. Данная команда переключает поле индикации осей в режим «**Коррекция**».

Примечание. Информация о команде **UCV** приведена в документе «Руководство оператора» в подразделе «Отображение информации на дисплее» при описании видеостраницы **#1**.

Штурвал коррекции должен быть подключён к свободному входному разъёму датчика на модуле **ECDA** и описан как ось штурвала в текущем процессе. Штурвалом коррекции может быть штурвал уже предназначенный для движения осей, т.е. штурвал, имя оси которого записано в инструкции **ADV** (секция 3 файла **IOCFIL**). Рекомендуется в ПЛ исключить использование одного и того же штурвала в одно и тоже время для коррекции оси и для движения оси.

Семантика:

HWC=HW_axis_name, input_enable_HW

Формат записи:

HWC=ASCII string, PLC variable ,

где:

HW_axis_name - имя оси штурвала коррекции;

input_enable_HW - представляет собой идентификатор сигнала **PLC** пакета «**A**» или свободного сигнала **PLC** из пакетов «**K**» или «**N**» для вкл./выкл. штурвала коррекции оси, например: **U223K0**.
Если состояние **input_enable_HW** равно «**1**», то коррекция позиции оси от штурвала включена.
Если состояние **input_enable_HW** равно «**0**», то коррекция позиции оси от штурвала выключена.

Коррекция позиции оси от штурвала возможна после включения оси (ось замкнута по ДОС) в любом режиме работы и состоянии УЧПУ.

Автоматическое обнуление накопленной осью коррекции выполняется при её выходе в абсолютный микроноль в режиме «**ВЫХОД В НОЛЬ**» («**HOME**»).

Пример

- корректируемая ось: X,
- ось штурвала коррекции с датчиком 100 имп/об: A.

*1

.....
PRO=1

```

IN1=1,X,,2,16
CAS=1,XA,2
*2
PRO=1
NAS=X
TPA=1,
NTC=1,1
.....
PAS=10000,10
GM0=6000,7.5,16.666
.....
HWC=A,U140K0
.....
NAS=A
TPA=1,
NTC=2,
PAS=400,1
GAS=,
POS=,
SRV=,
*3
.....

```

При состоянии сигнала **PLC U140K0=1** полный оборот штурвала (400 имп/об) скомпенсирует позицию оси **X** на 1 мм (400 имп/об = 1 мм, т.к. **PAS=400,1**) в направлении его вращения.

4.2.4.28. Инструкция HWC (с версии X.86)

Инструкция **HWC** определяет имя оси коррекции и идентификатор сигнала **PLC** для включения/выключения коррекции для оси, в подсекции которой записана инструкция **HWC**.

Коррекция позиции текущей оси на станке выполняется за счет перемещения ее оси коррекции. Направление и величина коррекции оси зависит от направления и величины перемещения ее оси коррекции.

Коррекция позиции текущей оси на станке не изменяет ее текущую координату в системе, поэтому после коррекции оси на станке в видеостраницах режима «УПРАВЛЕНИЕ СТАНКОМ» в полях индикации координат осей: «ФАКТ», «Датчик абс.», «Датчик отн.» значение ее текущей координаты не изменится.

Выполненную коррекцию оси можно определить внешними измерительными приборами и по индикации на экране УЧПУ в поле «Коррекция». Вывод поля «Коррекция» на экран УЧПУ выполняется вводом команды **UCV=5** с клавишей **ENTER** в режиме «УПРАВЛЕНИЕ СТАНКОМ».

Примечание. Информация о команде **UCV** приведена в документе «Руководство оператора» в подразделе «Отображение информации на дисплее» при описании видеостраницы #1.

Объявленная в инструкции **HWC** ось коррекции обязательно должна иметь датчик. Датчик оси коррекции должен быть подключён к свободному входному разъёму датчика на модулях (**ECDA, RCDA, ECDP, RCDP**). Датчик оси коррекции не может быть подключен к каналу электронного штурвала. Расположение разъема для подключения электронного штурвала приведено в руководстве по эксплуатации для соответствующего УЧПУ. Штурвалы, расположенные на станочных пультах **NC301-6**, **СП NC302-6**, **NC310-7** и **NC110-7**, не могут быть осями коррекции.

Семантика:

HWC=HW_axis_name, input_enable_HW, input_correction_reset, input_select_HW_axis_mode

Формат записи:

HWC=ASCII string, PLC variable, PLC variable, PLC variable

где:

HW_axis_name - имя оси может быть выбрано из списка осей процесса, см. инструкцию **CAS**.

input_enable_HW - представляет собой идентификатор сигнала **PLC** пакета «**A**» или свободного сигнала **PLC** из пакетов «**K**» или «**N**» для вкл./выкл. режима «**Коррекция**» для текущей оси, например: **U223K0**.

Если состояние сигнала **input_enable_HW** равно «**1**», то режим «**Коррекция**» включен;

Если состояние сигнала **input_enable_HW** равно «**0**», то режим «**Коррекция**» выключен.

input_correction_reset - представляет собой идентификатор сигнала **PLC** пакета «**A**» или свободного сигнала **PLC** из пакетов «**K**» или «**N**» для управления накопленной величиной компенсации, например: **U223K1**.

Если состояние сигнала **input_correction_reset** равно «**1**», то при выключении режима «**Коррекция**» выполняется перенос накопленной величины компенсации в позицию текущей компенсируемой оси с учетом знака компенсации.

Если состояние сигнала **input_correction_reset** равно «**0**», то при выключении режима «**Коррекция**» перенос накопленной величины компенсации в позицию компенсируемой оси не выполняется.

input_select_HW_axis_mode - представляет собой идентификатор сигнала **PLC** пакета «**A**» или свободного сигнала **PLC** из пакетов «**K**» или «**N**» для определения режима работы компенсирующей оси, например: **U223K2**.

Если сигнал **input_select_HW_axis_mode** равен «**0**», то устанавливается режим, в котором компенсирующая ось может быть использована в следующих случаях:

- 1) компенсационный штурвал;
- 2) абсолютный компенсационный датчик.

В этом случае датчик устанавливается на неподвижной части станка (станине) и имеет с компенсируемой осью только электронную связь.

Если сигнал, заданный параметром **input_select_HW_axis_mode**, равен «**1**», то устанавливается режим, в котором компенсирующая ось может быть использована в качестве относительного компенсационного датчика. В этом случае датчик компенсирующей оси устанавливается на текущую компенсируе-

мую ось станка и имеет с ней механическую и электронную связь, например, датчик компенсирующей оси **Q** крепится к выдвигаемой компенсируемой оси **Z**.

ПРИМЕЧАНИЯ:

- 1) Если сигнал **input_select_HW_axis_mode** равен «1», то включение и выключение режима «**Коррекция**» в начале и в конце обрабатываемого контура должно выполняться в тот момент, когда компенсирующий датчик касается отслеживаемой поверхности.
- 2) Если сигналы **input_enable_HW** и **input_select_HW_axis_mode** равны «1», то для исключения больших колебательных движений компенсируемой оси разрешающая способность датчика компенсирующей оси должна быть на 2 порядка выше, чем требуемая точность обрабатываемой поверхности.
- 3) Включение режима «**Коррекция**» возможно после включения компенсируемой оси (ось замкнута по ДПС) в любом режиме работы и состоянии УЧПУ.
- 4) Обнуление величины компенсации, накопленной осью в режиме «**Коррекция**», может быть выполнено:
 - при отключении режима «**Коррекция**» (обнуление сигнала «**input_enable_HW**») с предустановленным в состоянии «1» сигналом, «**input_correction_reset**».
 - при выходе компенсируемой оси в абсолютный микроноль в режиме «**ВЫХОД В НОЛЬ**» («**HOME**»).

Характеризация оси коррекции, которая является для и описан как ось штурвала в текущем процессе, см. **Пример 1** и **Пример 2**. Осью коррекции может быть штурвал, имя оси которого записано в инструкции **ADV** (секция 3 файла **IOCFIL**).

Пример. Запись инструкции **HWC** для компенсирующего штурвала:

```
NAS=Q
TPA=1,
NTC=4,
PAS=100,1
GAS = ,
POS = ,
SRV = ,
....
NAS=Z
TPA=1,
NTC=3,3
....
HWC=Q,U223K0,,
.....
```

Пример. Запись инструкции **HWC** для компенсирующей оси:

```
NAS=Q
TPA=1,
NTC=4,
PAS=1000,1
GAS = ,
POS = ,
SRV = ,
....
NAS=Z
TPA=1,
NTC=3,3
```

.....
HWC=Q, U223K0, U223K1, U223K2

4.2.4.29. Инструкция ATR

Инструкция **ATR** определяет параметры и подключение абсолютного датчика с интерфейсом **SSI** для текущей оси. Инструкция может быть записана для оси, в инструкции **TPA** которой добавлен код 200000H.

Семантика:

ATR=resolution ST, resolution MT .

Формат записи:

ATR=word, word ,

где:

resolution ST - число разрядов абсолютного датчика на одном обороте:
 - для однооборотного абсолютного датчика угловых перемещений или абсолютного датчика линейных перемещений параметр **resolution ST** может иметь значения от 12 до 31 бит.
 - для многооборотного абсолютного датчика угловых перемещений значение параметра **resolution ST** может иметь значения от 12 до 19 бит, кроме этого значение **resolution ST** не должно превышать значения разрядности, установленной перемычками на модуле абсолютных датчиков.

Примечание. В паспорте датчика параметр **resolution ST** может быть обозначен: «resolution ST, bit» или «Разрешение в пределах одного оборота, бит (кол. позиций)».

Внимание! Значение **resolution ST** должно быть использовано для определения параметра **el_pitch** в инструкции **PAS**.

бит:		el_pitch:
12	=	4096
13	=	8192
14	=	16384
15	=	32768
16	=	65536
17	=	131072
18	=	262144
19	=	524288
20	=	1048576
21	=	2097152
22	=	4194304
23	=	8388608
24	=	16777216
25	=	33554432
26	=	67108864
27	=	134217728
28	=	268435456
29	=	536870912
30	=	1073741824
31	=	2147483648

resolution MT - число разрядов датчика для определения количества его оборотов:
 - для однооборотного абсолютного датчика угловых перемещений или абсолютного датчика линейных перемещений параметр **resolution MT** должен быть равен 0 бит.

- для многооборотного абсолютного датчика угловых перемещений значение параметра **resolution MT** должно быть равно паспортному значению «resolution MT, bit» («Максимальное разрешение количества оборотов, бит (кол. оборотов)»), но не превышать 12 бит.

4.2.4.30. Инструкция GRA

Инструкция **GRA** определяет параметры и подключение абсолютного датчика с кодом Грея для текущей оси. Инструкция может быть записана для оси, в инструкции **TPA** которой добавлен код 80000H.

Семантика:

GRA=input, N .

Формат записи:

GRA=PLC variable, word ,

где:

input_GRA - идентификатор сигнала **PLC** пакета «А», к которому подключён младший разряд датчика. Данный сигнал пакета «А» должен быть младшим в байте. Все разряды одного датчика, начиная с младшего, должны быть подключены к контактам модуля дискретных входов/выходов **I/O** последовательно и непрерывно по возрастающей. Данные сигналы используются системой для формирования абсолютной позиции оси.

N - число дискретных входов, используемое для подключения абсолютного датчика. Значение **N** должно быть равно разрядности датчика (до 16).

Например, для 12-разрядного датчика, подключённого к контактам модуля дискретных входов/выходов **I/O** и читаемого в системе 12-тью сигналами с **I0A8** по **I0A19** включительно, необходимо записать:

GRA=I0A8,12 ;

ВНИМАНИЕ! Если для оси в инструкции **TPA** записан код 80000h, а инструкция **GRA** отсутствует или значения ее параметров **input_GRA** и **N** равны нулю (**GRA=**, или **GRA=0,0**), то по умолчанию эти параметры принимают следующие значения:

input_GRA = I0A0
N = 12

4.2.4.31. Инструкция FOPP

Инструкция **FOPP** определяет возможность коррекции скорости оси «от точки к точке», в состоянии ее движения. Корректор **FOPP** корректирует скорость движения оси, заданного из ПЛ, а также заданного в кадре с 3-буквенным кодом **MOV**.

Семантика:

FOPP=input_FOPP .

Формат записи:

FOPP=PLC variable ,

где:

input_FOPP

- идентификатор сигнала **PLC**. Этот сигнал должен определять младший сигнал в слове **PLC**, в котором ПЛ записывает процент коррекции скорости для оси «от точки к точке». Процент коррекции скорости для оси «от точки к точке» может быть задан в слове **PLC** целым числом от 0 до 255.

input_FOPP не может принадлежать пакету «А» и должен принадлежать свободной части сигналов в пакетах «К» и «N».

Пример. Если в ПЛ для ввода значения коррекции скорости оси «от точки к точке» записано слово **W22K2**, то имя младшего сигнала в этом слове: **I22K16**, поэтому для оси «от точки к точке» должна быть записана строка следующего вида: **FOPP=I22K16**

Особенности применения корректора **FOPP**.

- 1) Если позиция оси «от точки к точке» для ее движения запрограммирована в ПЛ, то корректор скорости **FOPP** выполняет коррекцию скорости в движении оси от скорости **Vn**, рассчитанной до начала движения оси по формуле 4.11:

$$Vn = RAP * WFESSn \quad (4.11),$$

Где:

Vn - расчетная скорость оси до начала ее движения, управляемой на канале с номером **n** ($n=1 \div 2$).

RAP - скорость быстрого хода оси.

WFESSn - процент коррекции для расчета скорости оси в **n**-ом канале управления, где **n** - номер канала управления ($n=1 \div 2$).

Для 1-ого канала управления процент коррекции для расчета скорости оси ПЛ устанавливает в слове **W20K2 (WFESS1)**, а для 2-ого канала управления в слове **W20K3 (WFESS2)**.

Примечание. Если **WFESSn** равно 0, то **Vn=RAP**

Таким образом, в движении оси ее скорость рассчитывается по формуле 4.12:

$$Vmovn = Vn * FOPP \quad (4.12),$$

Где:

Vmovn - скорость оси в движении, управляемой в канале **n** ($n=1 \div 2$).

Vn - расчетная скорость оси до начала ее движения, управляемой на канале с номером **n** ($n=1 \div 2$).

FOPP - процент коррекции скорости **Vn**.

2) Если позиция оси для ее движения запрограммирована в кадре с 3-буквенным кодом **MOV**, то корректор **FOOP** выполняет коррекцию скорости, заданной в кадре с 3-буквенным кодом **MOV**.

4.2.4.32.

Инструкция FTP

Инструкция **FTP** записывается в подсекции оси «от точки к точке», имеющей неравномерные расстояния между ее соседними позициями.

Семантика:

FTP=file_name FTP/MPx .

Формат записи:

FTP = ASCII string/ASCII string ,

где:

file_name FTP - определяет имя файла коррекции позиций на оси «от точки к точке»;

MPx - логическое имя пути, где **x**=0÷6, к файлу коррекции позиций для оси «от точки к точке», имя которого записано в **file_name FTP**.

Структура и правила заполнения файла коррекции позиций на оси «от точки к точке».

Файл коррекции позиций на оси «от точки к точке» может быть создан в редакторе режима **«COMMAND»** или в редакторе **«ВП»**. Строки этого файла должны содержать 3 элемента, разделенных символом **«ПРОБЕЛ»**. Количество строк в файле должно быть равно значению параметра **мес_pitch**, записанному в инструкции **PAS** для текущей оси «от точки к точке» и равно количеству целых позиций на оси «от точки к точке».

Семантика строки файла:

Элемент №1	Элемент №2	Элемент №3
ser_num_point	coordinate_point	[comment]

Формат строки файла:

Элемент №1	Элемент №2	Элемент №3
integer	real	[ASCII string]

ser_num_point - определяет порядковый номер позиции оси «от точки к точке»; в этом столбце на каждой порядковой строке должен быть записан порядковый номер позиции оси. Если 16-тиричный код **ECDF** добавлен во втором параметре инструкции **NBP**, то номер позиции на 1-ой строке должен быть равен значению параметра **start_point_#**, записанному в инструкции **PAS** для те-

кушей оси. В противном случае номер позиции на 1-ой строке должен быть равен нулю.

coordinate_point - определяет реальную координату каждой позиции на оси «от точки к точке», выраженную в позициях от абсолютного микроула оси. Значение координаты для позиции, записанной в 1-ой строке, должно быть равно ее порядковому номеру. Значение ***coordinate_point*** для текущей порядковой позиции оси ***ser_num_point*** должно находиться в пределах от ***(ser_num_point-1)*** до ***(ser_num_point+1)***.

[comment] - определяет текстовый комментарий для текущей позиции.

Пример. Определение файла ТАВТТ1/МР0, содержащего таблицу коррекций для 4-х позиций на оси «от точки к точке».

```
NAS = B
TPA = 2,
...
PAS = 10000,4,1,0
...
FTP = ТАВТТ1/МР0
```

Пример. Определение значения координаты для каждой из 4-х позиций оси «от точки к точке» в файле ТАВТТ1/МР0. Графическое соответствие номеров позиций и их координат на оси представлено на рисунке 4.12.

	Элемент №1	Элемент №2	Элемент №3
Строка №1	0	0	позиция №0 с координатой 0 от микроула оси
Строка №2	1	1.5	позиция №1 с координатой 1.5 от микроула оси
Строка №3	2	2	позиция №2 с координатой 2 от микроула оси
Строка №4	3	3	позиция №3 с координатой 3 от микроула оси

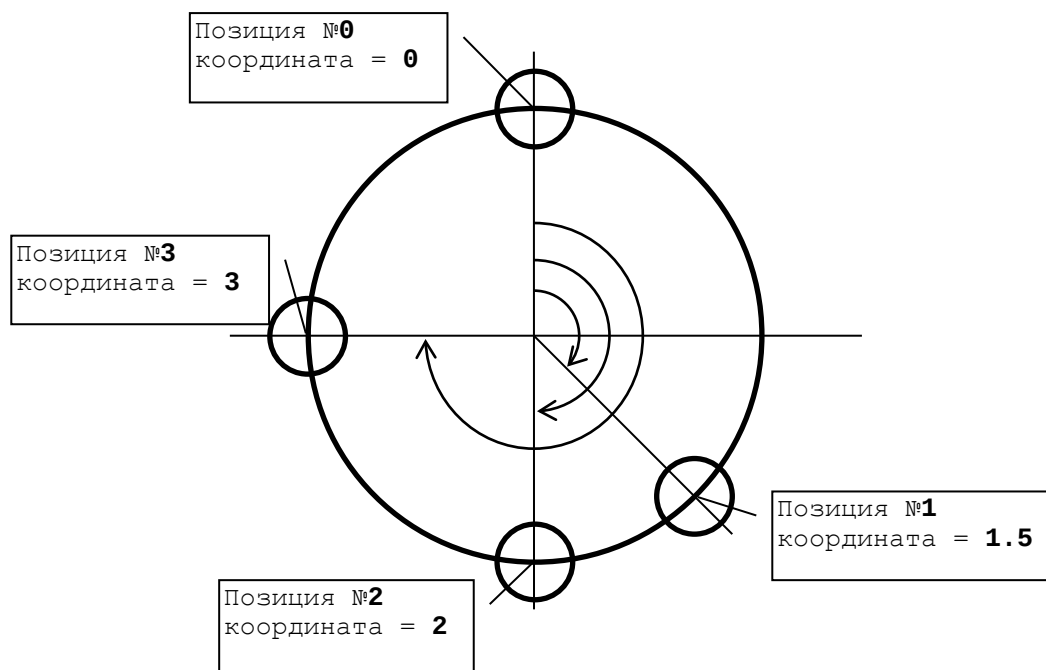


Рисунок 4.12

4.2.4.33. Инструкция ADG

Инструкция **ADG** записывается в подсекции осей, которые в системе могут переключаться из режима контроля по ДОС в режим контроля по напряжению на АЦП – далее режим «воздушный зазор». Режим «воздушный зазор» может быть использован для осей процесса, удовлетворяющих следующим условиям конструкции станка:

- 1) ось должна быть перпендикулярна к плоскости обработки, образуемой осями абсцисс и ординат;
- 2) ось должна нести инструмент (обычно лазерные, плазменные или газовые резак);
- 3) ось должна иметь устройство формирования напряжения в канале АЦП УЧПУ в зависимости от величины воздушного зазора между инструментом и заготовкой.

Для оси, поддерживающей режим «воздушный зазор», в инструкции **TPA** должен быть добавлен 16-тичный код 8000, например: **TPA=8001**,

В одном и том же процессе режим контроля по напряжению на АЦП одновременно могут поддерживать максимально до 6-ти осей.

Основной задачей режима «воздушный зазор» является поддержание заданного воздушного зазора (далее рабочий воздушный зазор) между инструментом и деталью при движении вдоль контура.

Режим «воздушный зазор» настраивается параметрами ***U_far*** и ***U_near*** в инструкции **ADG** и параметром ***U_working air gap*** (напряжение в рабочем воздушном зазоре между инструментом и деталью). Напряжение ***U_working air gap*** – величина переменная, зависит от применяемого инструмента и детали, поэтому определяется каждый раз при настройке инструмента в ручном режиме работы.

Напряжение касания инструмента с заготовкой (***U_touch***) в параметрах управления явно не вводится, но считается, что это напряжение должно быть больше, чем ***U_far*** и ***U_near***.

Схема шкалы «напряжение – скорость (величина + направление)» в воздушном зазоре между инструментом и заготовкой представлена на рисунке 4.13.

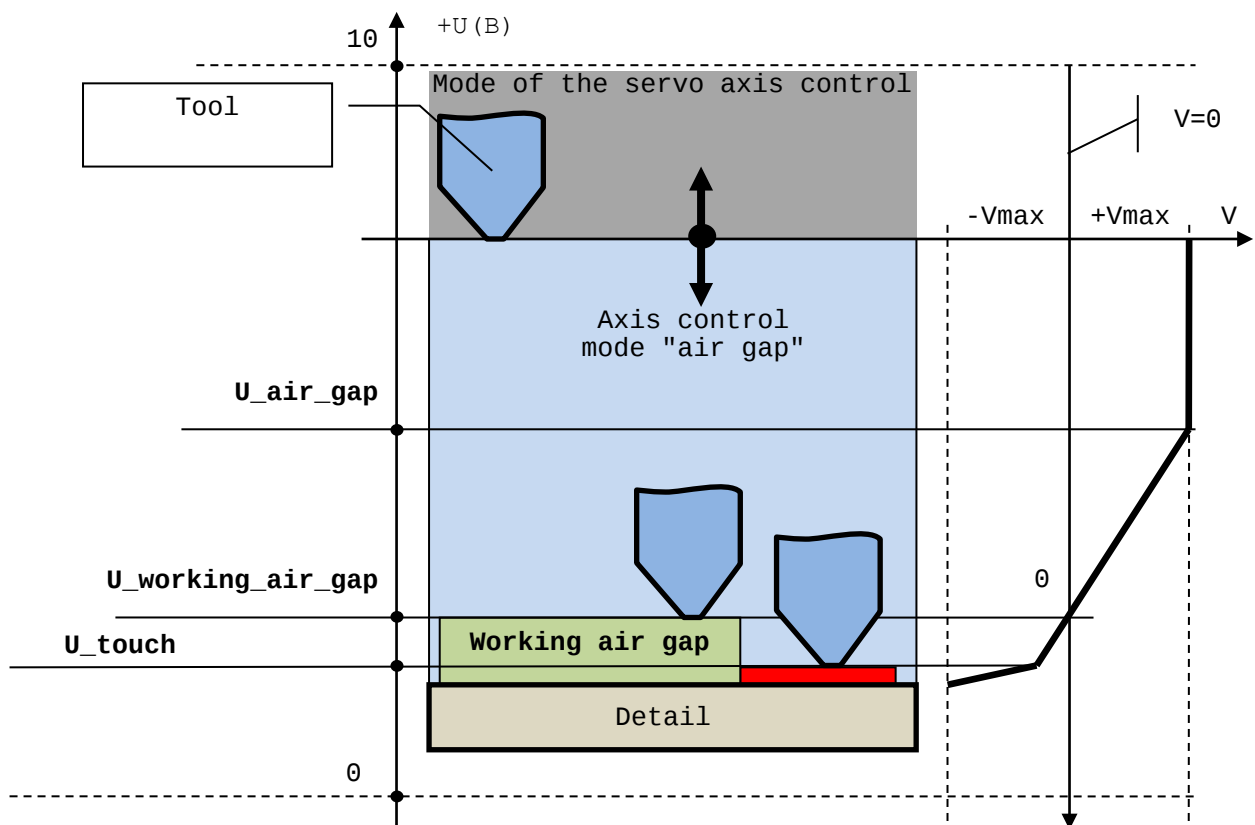


Рисунок 4.13

Семантика:

ADG=CHN_N_ADC, U_near, U_far, remember_air_gap, set_mode_air_gap, air_gap_status, indication_air_gap .

Формат записи:

ADG=integer, Real, Real, PLC variable, PLC variable, PLC variable, PLC variable ,

где:

CHN_N_ADC - номер канала АЦП, используемый для управления текущей осью в режиме «воздушный зазор».

U_near - граничное значение рабочего напряжения в канале АЦП, когда инструмент находится на минимально допустимом расстоянии от поверхности детали; может принимать только положительные значения от 0 до +10В.

U_far - граничное значение рабочего напряжения в канале АЦП, когда инструмент находится на максимальном удалении от поверхности детали; может принимать только положительные значения от 0 до +10В.

Параметры **U_{near}** и **U_{far}** позволяют настроить поддержку, как прямой, так и обратной зависимости величины напряжения (**U**) от величины воздушного зазора (**h**). Кроме этого **U_{touch}** может быть равно 0 или больше чем **U_{near}** и **U_{far}** . Таким образом поддерживает 4 варианта работы внешнего устройства контроля зазора:

- 1) $0 < U_{near} < U_{far} < U_{touch} < 10$ - прямая зависимость и $U_{touch} > 0$; рисунок 4.14.a
- 2) $0 < U_{far} < U_{near} < U_{touch} < 10$ - обратная зависимость и $U_{touch} > 0$; рисунок 4.14.b.
- 3) $0 < U_{near} < U_{far} < U_{touch} = 0$ - прямая зависимость и $U_{touch} = 0$; рисунок 4.14.c
- 4) $0 < U_{far} < U_{near} < U_{touch} = 0$ - обратная зависимость и $U_{touch} = 0$; рисунок 4.14.d.

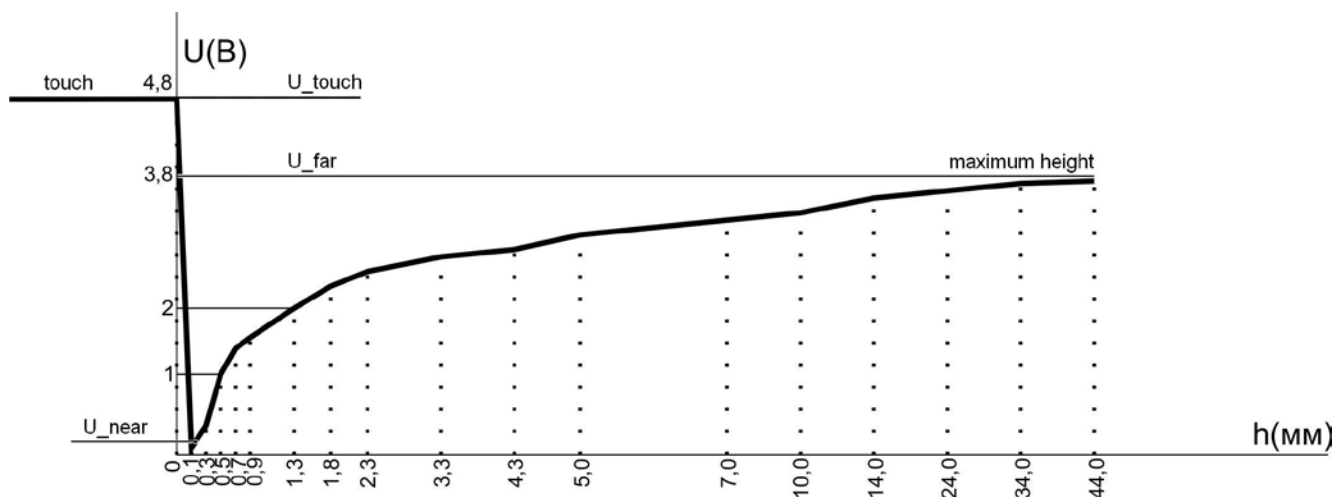


Рисунок 4.14 a)

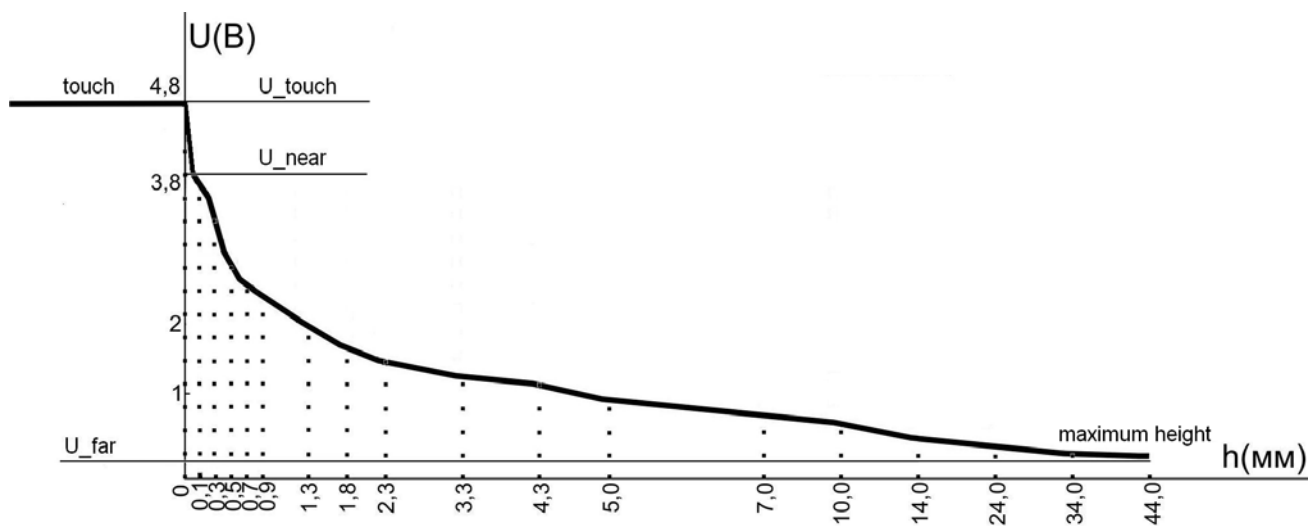


Рисунок 4.14 b)

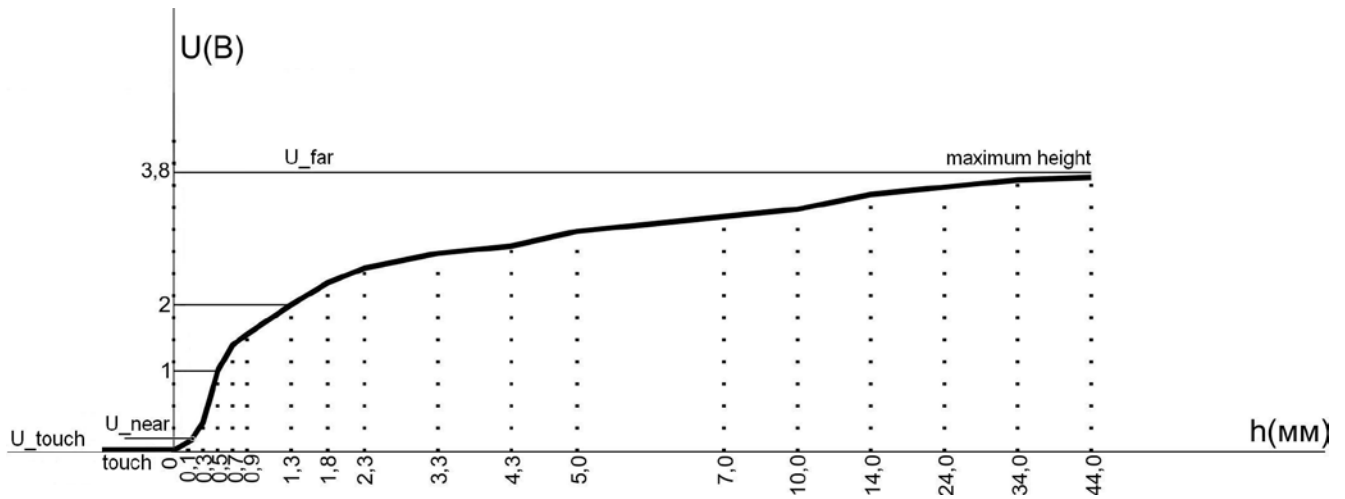


Рисунок 4.14 c)

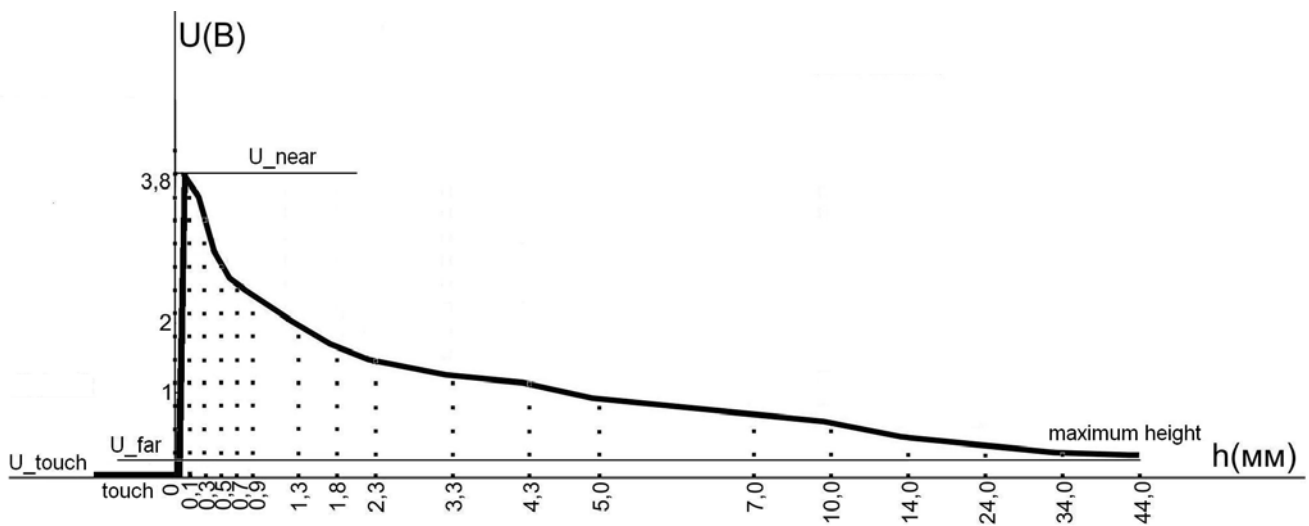


Рисунок 4.14 d)

remember_working_air_gap - идентификатор выходного сигнала **PLC** в зоне свободных сигналов пакетов «**K**» или «**N**». Кратковременный высокий уровень этого сигнала является запросом из ПЛ в **SW** для фиксации напряжения ***U_working_air_gap*** в канале АЦП для текущего воздушного зазора между инструментом и поверхностью заготовки.

В системе напряжение ***U_working_air_gap*** определяется автоматически по запросу оператора станка при настройке рабочего воздушного зазора, поддерживаемого текущей осью в режиме «воздушный зазор».

set_mode_air_gap - идентификатор выходного сигнала **PLC** в зоне свободных сигналов пакетов «**K**» или «**N**». Высокий постоянный уровень сигнала ***set_mode_air_gap***, установленный из ПЛ в **SW**, является запросом включения для текущей оси режима «воздушный зазор». Отмена режима «воздушный зазор» выполняется сбросом сигнала ***set_mode_air_gap***.

ВНИМАНИЕ! Установка сигнала **set_mode_air_gap** в состояние «1» требует автоматического выключения данной оси, которое выполняется установкой сигнала **RAB1** для данной оси в состояние «0» на все время, пока система работает в режиме «воздушный зазор».

air_gap_status идентификатор входного сигнала **PLC** в зоне свободных сигналов пакетов «K» или «N». Уровень сигнала **air_gap_status** формируется из **SW** в ПЛ только после включения для оси режима «воздушный зазор» (**set_mode_air_gap** = «1»).

Высокий уровень сигнала **air_gap_status** является признаком достижения рабочего воздушного зазора между инструментом и заготовкой.

Сброс сигнала **air_gap_status** выполняется в **SW** автоматически после отмены режима «воздушный зазор» (**set_mode_air_gap** = «0»).

indication_air_gap - идентификатор выходного сигнала **PLC** в зоне свободных сигналов пакетов «K» или «N». Высокий постоянный уровень сигнала **indication_air_gap**, установленный из ПЛ в **SW**, является запросом вывода на видеостраницы #1 и #7 текущего напряжения в канале АЦП и напряжения зафиксированного сигналом **remember_working_air_gap** для поддерживаемого воздушного зазора с символом «U». Эти величины напряжений выводятся на экран вместо строки, индицирующей текущую и запрограммированную скорость вращения шпинделя с символом «S». Возврат индикации скорости вращения шпинделя выполняется установкой сигнала **indication_air_gap** в «0».

Пример применения режима «воздушный зазор» для прямой зависимости величины напряжения (U) от величины воздушного зазора (h).

ADG=1,0.2,3.8,U150N3,U150N0,I150N1,U150N2

Пример части ПЛ для поддержки режима «воздушный зазор».

```
;Запрос прерывания подачи FOLD (U10K5=«1») после
;включения режима «воздушный зазор» на время пока не будет
;установлен признак air_gap_status - инструмент подведен
;к заготовке на расстояние равное «рабочий воздушный зазор»
U10K5=U150N0*/I150N1
```

```
;Запрос режима «воздушный зазор» set_mode_air_gap:
;- установка режима выполняется по функции M91
;- отмена режима выполняется по функции M90 и «Общий сброс»
```

$U150N0=[W3K0=91H]+U150N0*/[W3K0=90H]*/I0K1$

```
;Выключение 3-ей оси процесса (обычно ось Z) на время ее
;работы в режиме «воздушный зазор»
U10K10=I0K2*/U150N0
```

```
;запрос indication_air_gap (U150N2) для включения индикации
;напряжения
U150N2=/U250N0+U250N0
```

4.2.4.34.

Инструкция LAS

Инструкция **LAS** записывается в подсекции осей, несущих лазерный инструмент (макс. до 6 осей). Эта инструкция определяет:

1) параметры управления мощностью лазера в зависимости от скорости обработки профиля. Это управление выполняется формированием уровня напряжения в канале ЦАП или частоты в канале ЦАП на основании значений, введенных в следующие параметры инструкции **LAS: P_min, V_lim_min, P_max, V_lim_max**. Зависимость управляющего сигнала (напряжения **U**), используемого для установки мощности луча, от скорости на профиле (**V**) представлена на рисунке 4.15.

2) параметры управления мощностью лазера при управлении генератором импульсов с изменяемой частотой и скважностью в канале ЦАП/ЦАП на основании значений, введенных в следующие параметры инструкции **LAS: N_TIK_hig, U_hig, N_TIK_low, U_low**. Максимальная частота (**F**) генератора импульсов может быть рассчитана по формуле **$F=1000/(2*CPU_tick)$** , если минимальный **CPU_tick=1мс**, то **F_{max}= 500 Гц**.

Параметры генератора импульсов (**T** - период, **S** - скважность) представлены на рисунке 4.16.

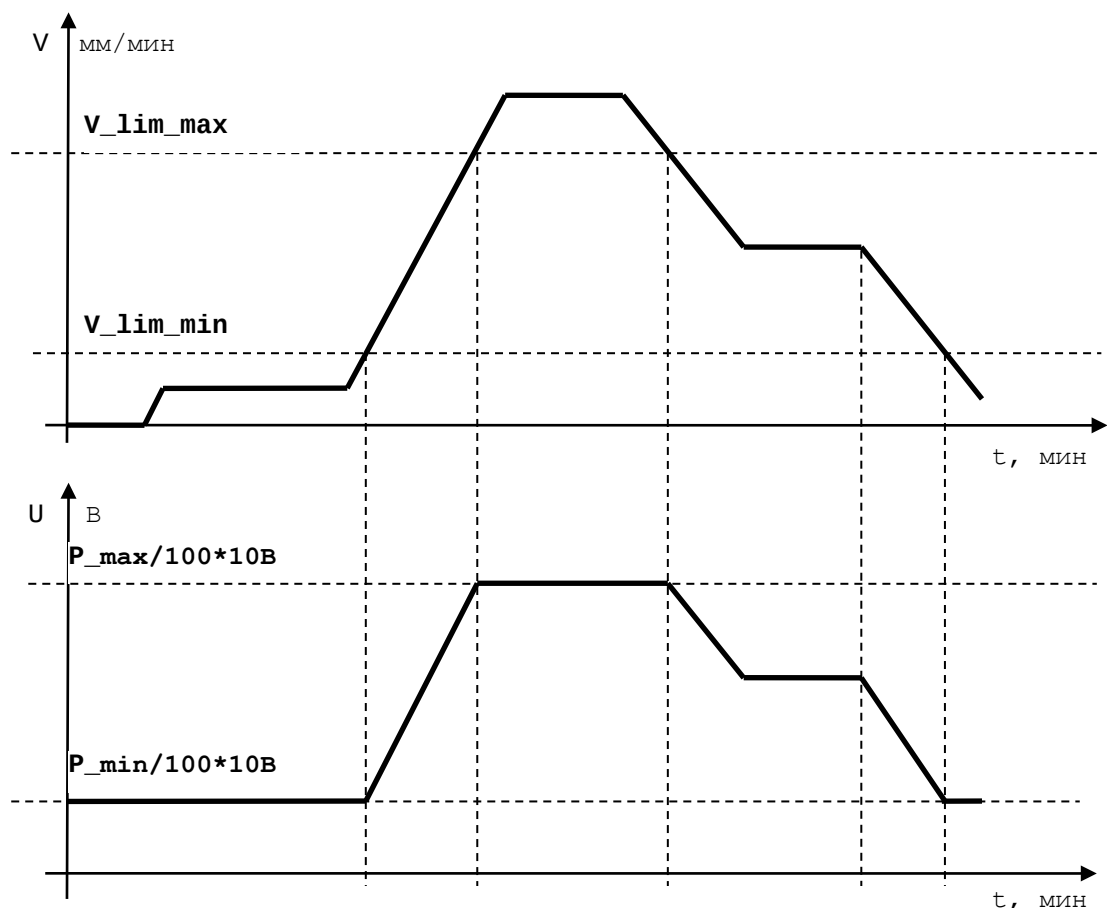


Рисунок 4.15.

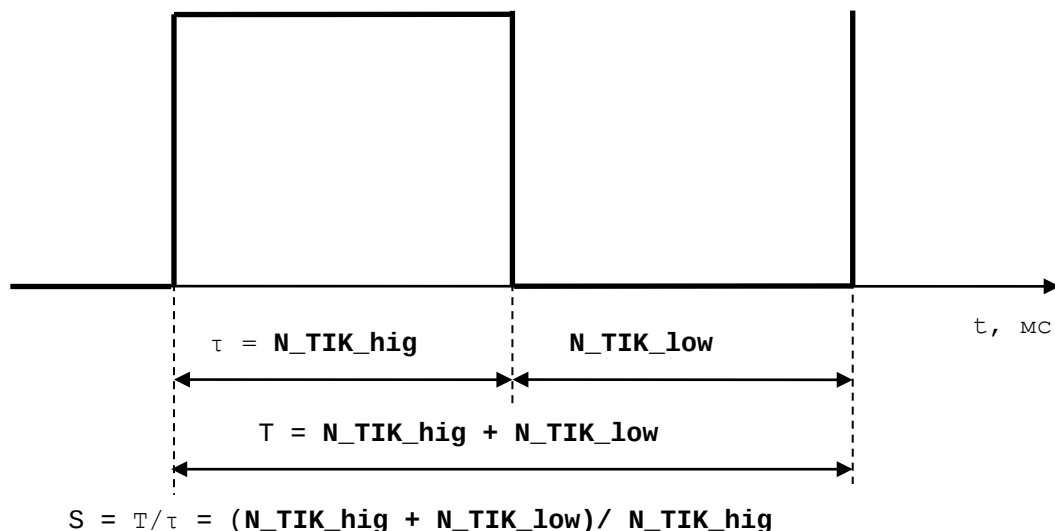


Рисунок 4.16

Семантика:

**LAS=CHN_N_PV,P_min,V_lim_min,P_max,V_lim_max,set_P_control,CHN_N_G
,N_TIK_hig,U_hig,N_TIK_low,U_low,set_G_control,P_0** .

Формат записи:

**LAS=Integer,Real,Real,Real,Real,PLC variable,Integer,Integer,Real,
Integer,Real,PLC variable,Real** ,

где:

CHN_N_PV - номер канала ЦАП (ЦИП), используемый для управления мощностью лазера в зависимости от контурной скорости движения вдоль обрабатываемого профиля. Формирование уровня управляющего сигнала в этом канале выполняется на основании значений параметров: **P_min, V_lim_min, P_max, V_lim_max**, если в ПЛ сигнал **set_P_control** установлен на состояние «1».

P_min - определяет значение процента от максимального значения напряжения ЦАП (10В), соответствующее минимальной мощности луча для скоростей меньших или равных **V_lim_min**.

V_lim_min - определяет нижнюю границу скорости [мм/мин] вдоль профиля для установки в канале **CHN_N_PV** минимальной мощности луча.

P_max - устанавливает значение процента от максимального значения напряжения ЦАП (10В), соответствующее максимальной мощности луча для скоростей равных или больше **V_lim_max**.

- V_lim_max** - определяет верхнюю границу скорости [мм/мин] вдоль профиля для установки в канале **CHN_N_PV** максимальной мощности луча.
- set_P_control** - идентификатор выходного сигнала **PLC** в зоне свободных сигналов пакетов «К» или «N». Высокий постоянный уровень сигнала **set_P_control**, установленный из ПЛ в **SW**, является запросом включения канала **CHN_N_PV**.
- CHN_N_G** - номер канала ЦАП (ЦИП), используемый для управления генератором импульсов. Формирование частоты импульсов и их скважности в этом канале выполняется на основании значений параметров: **N_TIK_hig**, **N_TIK_low**.
Если в **CHN_N_G** записан номер канала ЦАП, то для импульсов генератора дополнительно необходимо ввести напряжения **U_hig** для высокого уровня импульса и напряжение **U_low** для низкого уровня импульса.
Канал **CHN_N_G** включается, если в ПЛ сигнал **set_G_control** установлен на состояние «1».
- N_TIK_hig** - определяет длительность высокого уровня импульса генератора; выражается [количество **CPU_tick**].
- U_hig** - определяет величину напряжения для высокого уровня импульса генератора; выражается [В].
- N_TIK_low** - определяет длительность низкого уровня импульса генератора; выражается [количество **CPU_tick**].
- U_low** - определяет величину напряжения для низкого уровня импульса генератора; выражается [В].
- set_G_control** - идентификатор выходного сигнала **PLC** в зоне свободных сигналов пакетов «К» или «N». Высокий постоянный уровень сигнала **set_G_control**, установленный из ПЛ в **SW**, является запросом включения канала **CHN_N_G**.
- P_0** - корректор уровня управляющего сигнала в каналах **CHN_N_PV** и **CHN_N_G**; выражается [процент].

Пример.

LAS=0,10,100,100,1000,U200N30,5,1,+1,1,-1,U200N31,100

4.2.4.35. Инструкция FQF

Инструкция предназначена для установки параметра фильтра для контура положения.

Семантика:

FQF=frequency_filter_position .

Формат записи:

FQF=real ,

где:

frequency_filter_position - представляет значение полосы пропускания контура положения (**Wp**). Значение **frequency_filter_position** (единицы измерения, Гц) вычисляется при выполнении теста **KFKV**.

Если инструкция **FQF** отсутствует в подсекции оси, то значение **frequency_filter_position** равно 0 - фильтр контура положения выключен.

Особенности определения значения **frequency_filter_position** рассмотрены в приложении «А».

Примечание.

Значение **frequency_filter_position** используется в контуре управления оси, если ось выполняет движение со скоростной компенсацией (переменная **UEP=0**) в режимах работы «Ручной ввод кадра» (**MDI**), «Автоматический» (**AUTO**) или «Кадр» (**STEP**).

4.2.5. Секция 3

Секция 3 используется для установки параметров коррекции геометрических ошибок. **SW** УЧПУ предоставляет возможность выполнения компенсации геометрических ошибок для всех осей каждого процесса.

- 1) Максимальное количество точек компенсации для каждой оси - 1000.
- 2) Точки коррекции на одной оси должны быть эквидистанты (равноудалены друг от друга) и объявлены для полного перемещения линейной оси или для полного оборота оси вращения.
- 3) В случае оси вращения значение шага коррекции должно быть таким, чтобы оно укладывалось на одном обороте оси целое число раз.
- 4) Для каждой оси, имеющей геометрические ошибки, в секции 3 необходимо записать следующие инструкции: **NAS, PAS, Ennn, NM0** или **NAS, PAS, FEG, NM0**, если значения коррекции **Ennn** записаны в отдельном файле.
- 5) Для каждой оси, требующей коррекции своей позиции в зависимости от позиции другой оси, в секции 3 необходимо записать следующие инструкции: **NAC, PAS, Ennn, NM0** или **NAC, PAS, FEG, NM0**, если значения коррекции **Ennn** записаны в отдельном файле.
- 6) Значения соседних точек коррекции **E(n)** и **E(n-1)** могут быть равны друг другу. Если значения **E(n)** и **E(n-1)** в таблице отличаются друг от друга, то они должны удовлетворять условию, рассчитанному по формуле 4.13:

$$-32768 \leq \left| \frac{cp}{E(n) - E(n-1)} \right| \leq 32767 \quad (4.13),$$

где:

- ср** - шаг коррекции;
- E(n)** - значение коррекции в текущей точке;
- E(n-1)** - значение коррекции в предыдущей точке.

7) Каждая точка коррекции **E(n)** в таблице должна удовлетворять условию, рассчитанному по формуле 4.14:

$$-32768 \leq \left| \frac{e_p}{m_p} \times E(n) \right| \leq 32767 \quad (4.14),$$

Где:

- е_р** - электрический шаг;
- м_р** - механический шаг;
- E(n)** - значение коррекции в текущей точке;

8) Минимальный объявляемый шаг коррекции в инструкции **PAS** зависит от скорости быстрого хода и от тика управления приводом осей. Минимальный шаг рассчитывается по формуле 4.15:

$$\text{minimum pitch} = \frac{F_s \times \text{servo_tick}}{60} \times 2 \quad (4.15),$$

Где:

- F_s** - скорость быстрого хода (мм/мин);
- servo_tick** - тик CPU управления приводом осей (секунды).

Пример

Скорость быстрого хода равна 12 м/мин; тик управления приводом равен 2 мс. Рассчитаем минимальный шаг:

$$\text{minimum pitch} = \frac{12000 \times 0.002}{60} \times 2 = 0.8 \text{ (мм)}.$$

4.2.5.1. Инструкция PRO

Инструкция **PRO** предназначена для объявления номера текущего характеризуемого процесса.

Семантика:

PRO=proc_num .

Формат записи:

PRO=integer ,

где:

proc_num - объявляется номер процесса, выбранный для характеристики: от 1 до 5.

Номер процесса не должен превышать число процессов, объявленных в инструкции **NBP**. Инструкция **PRO** должна быть записана для каждого объявляемого процесса.

4.2.5.2. Инструкция NAS

Инструкция **NAS** объявляет наименование текущей оси, для которой должна быть составлена таблица коррекции геометрических ошибок.

Семантика:

NAS=axis_name .

Формат записи:

NAS=ASCII string ,

где:

axis_name - наименование оси текущего процесса, для которой требуется коррекция геометрических ошибок.

4.2.5.3. Инструкция PAS

Инструкция используется для установки шага коррекции текущей оси.

Семантика:

PAS=correct_pitch .

Формат записи:

PAS=real ,

где:

correct_pitch - определяет расстояние между двумя соседними точками коррекции геометрических ошибок. Значение выражается в единицах измерения оси. Значение **PAS** должно быть положительной константой.

Значение шага коррекции для оси вращения, должно быть кратно 360 градусам.

4.2.5.4. Инструкция EXXX

Инструкция **EXXX** определяет номер корректора и значение ошибки.

Семантика:

Exxx=error .

Формат записи:

Exxx=real ,

где:

XXX - номер корректора; должен быть возрастающим числом и содержать не более трех цифр;

error - ошибка позиции оси, определяемая разницей между позицией оси, считанной с ее ДОС и реальной позицией оси. Значение выражено в единицах измерения оси со знаком.

Примеры

- 1) E123 = .001
E124 = -.008
E125 = .005
- 2) E100 = .004
E110 = -.009
E120 = .006

4.2.5.5. Инструкция NM0

Инструкция **NM0** объявляет номер корректора, соответствующего опорной точке абсолютного микроула оси без учета параметра **null_offset**, установленного в инструкции **ZNO**.

Семантика:

NM0=correct_num .

Формат записи:

NM0=ASCII string ,

где:

correct_num - номер корректора, объявленный с **Ennn**, который соответствует позиции абсолютного микроула оси. Например, если корректор E002 = 0.11 совпадает с позицией абсолютного микроула оси, то следует объявить: NM0 = E002.

Пример 1. Коррекция оси X в секции 3, имеющей положительное значение механического шага в инструкции **PAS** в секции 2:

*1

```

...
PRO=1
IN1=1, X..., ..., ..., ...
...
*2
PRO=1
NAS=X
...
PAS=10000, 10
...
LOP=150, -150
*3
PRO=1
NAS=X
PAS=50
E001=-0.02
E002=0.00
E003=0.01
E004=0.01
E005=0.02
E006=0.04
E007=0.05
NM0=E004

```

Пример 2. Коррекция оси X в секции 3, приведенная в примере 1, для случая, когда эта ось имеет отрицательное значение механического шага в инструкции **PAS** в секции 2.

```

*1
...
PRO=1
IN1=1, X..., ..., ..., ...
...
*2
PRO=1
NAS=X
...
PAS=10000, -10
...
LOP=150, -150
*3
PRO=1
NAS=X
PAS=50
E001=0.05
E002=0.04
E003=0.02
E004=0.01
E005=0.01
E006=0.00
E007=-0.02
NM0=E002

```

1) Запись таблицы коррекции двух осей:

```

NAS = X
PAS = 10
E001 = 0.1
E002 = 0.11

```

```

.....
E00n = 0.05
NM0 = E002
;
NAS = Y
PAS = 10
E001 = 0.08
E002 = 0.12
.....
E00n = 0.07
NM0 = E002

```

4.2.5.6. Инструкция NM0 (ECDF)

Инструкция **NM0** объявляет номер корректора, соответствующего опорной точке на оси, учитывающей значение параметра **null_offset**, установленное в инструкции **ZN0**.

Семантика:

NM0=correct_num .

Формат записи:

NM0=ASCII string ,

где:

correct_num - номер корректора, объявленный с Ennn, который соответствует позиции смещенного абсолютного микроула оси на величину **null_offset** в инструкции **ZN0**. Например, если на одной и той же оси точка E002=0.12 совпадает с позицией ее абсолютного микроула, а точка E035=0.15 совпадает с позицией смещенного абсолютного микроула оси, то следует объявить: NM0=E035.

4.2.5.7. Инструкция NAC

Инструкция **NAC** используется для установки имён двух взаимосвязанных осей. Порядок записи имён осей в этой инструкции устанавливает взаимоотношение между ними.

Семантика:

NAC=slave_axis, master_axis .

Формат записи:

NAC=ASCII string, ASCII string ,

где:

slave_axis - наименование оси, для которой требуется коррекция геометрических ошибок в зависимости от позиции, которую занимает главная ось;

master_axis - наименование оси, позиция которой влияет на геометрическую ошибку корректируемой оси.

После инструкции **NAC** необходимо определить:

- значение шага коррекции в инструкции **PAS** для **master_axis**;
- значения дополнительной коррекции **Ennn** для **slave_axis**;
- номер коррекции **Ennn** в инструкции **NM0**, от которой с шагом, определённым в инструкции **PAS**, позиция главной оси дополнительно корректирует **slave_axis**.

Пример

<pre> NAS = X PAS = 10 E001 = 0.1 E002 = 0.11 E00n = 0.05 NM0 = E002 ; NAC = X, Y PAS = 5 E001 = 0.25 E002 = 0.12 E00n = 0.33 NM0 = E004 </pre>	{ {	конфигурирование коррекции погрешности геометрии для одной оси конфигурирование коррекции погрешности геометрии одной оси от позиции другой оси
---	--	--

4.2.5.8.

Инструкция FEG

Номера точек коррекций **Ennn** и значения коррекций в этих точках для одной оси можно записать в отдельном файле с указанием в инструкции **FEG** имени этого файла и его местоположения. Инструкция **FEG** записывается в секции 3 для каждой определяемой оси после инструкции **NAS** или **NAC**.

Формат записи:

FEG=file_name/MPx ,

где:

file_name - определяет имя файла, в котором записаны точки коррекции геометрических ошибок оси;

MPx - логическое имя пути (**x=0÷6**) к файлу, имя которого записано в параметре **file_name**.

Пример

```

NAS = X
PAS = 10
FEG = COMPX/MP2
NM0 = E002
;
NAC = X, Y
PAS = 5
FEG = COMPLY/MP2

```

NM0 = E004

E001 = 0.1	}	записи в файле COMPX/MP2
E002 = 0.11		
E00n = 0.05		
E001 = 0.25	}	записи в файле COMPHY/MP2
E002 = 0.12		
E00n = 0.33		

4.3. Примеры файла AXCFIL

4.3.1. Пример характеристизации управления двумя процессами

Пример характеристизации управления двумя процессами с функциями расширения:

1) процесс 1:

- 2 координатные оси с датчиками энкодер 1250 имп/об (X, Z);
- ось X с восемью точками коррекции геометрических ошибок;
- 1 ось «от точки к точке - магазин инструментов» с датчиком энкодер (T);
- шпиндель с датчиком (S).

2) процесс 2:

- 2 координатные оси с датчиками энкодер 1250 имп/об (X, Z);
- ось шпинделя, общая с процессом 1.

```
*1
NBP=2, ECDF
TIM=1, 0, 0, 0, 0
PRO=1
IN1=1, XZ, S, 1, 16
IN2=1, T, , 1, 1
CAS=1, XZTS, 1
ACC=1.5, 0.5, 1, 4
PRO=2
COM=1, S
IN3=1, XZ, , 1, 16
CAS=1, XZ, 1
*2
; процесс 1
PRO=1
NAS=X
TPA=41,
NTC=1, 1
RAP=12000, 800
GAS=, , U220K0, U220K8
PAS=5000, 1, ,
MCZ=I1A0, 0, 120
POS=.01, 2
SRV=.5, 2, 10
MAN=2000, 200
```

```

GM0=12000,7.5,20,,
LO1=18,-480,5
MFC=I1A0,I1A1
SW1=17,15,U220K16
NAS=Z
TPA=1,
NTC=2,2
RAP=12000,800
GAS=,,U220K1,U220K9
PAS=5000,1,,
MCZ=I1A4,0,120
POS=.01,2
SRV=.5,2,10
MAN=2000,200
GM0=12000,7.5,25,,
LO1=18,-330,5
MFC=I1A4,I1A5
SW1=17,15,U220K17
NAS=T
TPA=2,
NTC=6,6
GAS=,,U220K2,U220K10
PAS=5000,36,36,1
MCZ=U117K4,0,10
RAP=600,10
MAN=400,10
POS=.1,10
SRV=,,
GM0=6000,7.5,7,,
NAS=S
TPA=820,
NTC=4,4
PAS=5000,1,,
GAS=,,U220K3,U220K11
POS=0.01,
SRV=,,
GM1=5000,7.5,6,,
TSM=6,15
POM=0.5,100,0.1,U230K1
ASM=X
; процесс 2
PRO=2
NAS=X
TPA=41,
NTC=3,3
RAP=12000,800
GAS=,,U220K4,U220K12
PAS=5000,1,,
MCZ=I1A0,0,120
POS=.01,2
SRV=.5,2,10
MAN=2000,200
GM0=12000,7.5,20,,
LO1=18,-480,5
MFC=I1A0,I1A1
SW1=17,15,U220K18
NAS=Z
TPA=01,
NTC=5,5
RAP=12000,800
GAS=,,U220K5,U220K13
PAS=5000,1,,
MCZ=I1A4,0,120
POS=.01,2

```

```

SRV=.5,2,10
MAN=2000,200
GM0=12000,7.5,25,,
L01=18,-330,5
MFC=I1A4,I1A5
SW1=17,15,U220K19
*3
PRO=1
NAS=X
PAS=5
E010=.08
E020=-.05
E030=-.077
E040=.042
E050=-.006
E060=-.002
E070=.009
E080=.015
NM0=E030

```

4.3.2. Пример характеристики управления одним процессом

Пример характеристики управления одним процессом с подключёнными функциями расширения:

1) процесс 1:

- 3 координатные оси с датчиками энкодер 1250 имп/об (X, Y, Z);
- ось X с восемью точками коррекции геометрических ошибок;
- 1 ось «от точки к точке» - поворотный стол с датчиком энкодер (B);
- управляемый шпиндель с датчиком (S).

```

*1
NBP=2,ECDF
TIM=1,0,0,0,0
PRO=1
IN1=1,XYZ,S,1,16
IN2=1,B,,1,1
CAS=1,XYZBS,1
ACC=1.5,0.5,1,4
*2
; процесс 1
PRO=1
;
NAS=X
TPA=01,
NTC=1,1
RAP=12000,800
GAS=0,0,,
PAS=5000,1,,
MCZ=I1A0,0,120
POS=.01,2
SRV=.5,2,10
MAN=2000,400

```

GM0=12000, 7.5, 20, ,
 L01=18, -480,
 MFC=I1A4, I1A5
 CUB=50, 10, 2
 SW1=101, 100, U255K1
 SW2=201, 200, U255K2

;
 NAS=Y
 TPA=01,
 NTC=2, 2
 RAP=12000, 800
 GAS=0, 0, ,
 PAS=5000, 1, ,
 MCZ=I1A1, 0, 120
 POS=.01, 2
 SRV=.5, 2, 10
 MAN=2000, 400
 GM0=12000, 7.5, 20, ,
 L01=18, -330,
 MFC=I1A6, I1A7
 CUB=70, 10, 2
 SW1=201, 100, U255K3
 SW2=401, 200, U255K4

;
 NAS=Z
 TPA=01,
 NTC=3, 3
 RAP=12000, 800
 GAS=0, 0, ,
 PAS=5000, 1, ,
 MCZ=I1A2, 0, 120
 POS=.01, 2
 SRV=.5, 2, 10
 MAN=2000, 400
 GM0=12000, 7.5, 20, ,
 L01=18, -330,
 MFC=I1A8, I1A9
 CUB=100, 10, 2
 SW1=201, 100, U255K3
 SW2=401, 200, U255K4

;
 NAS=B
 TPA=2,
 NTC=4, 4
 GAS=0, 0, ,
 PAS=5000, 72, 72, 0
 MCZ=I1A3, 0, 20
 RAP=72, 4
 MAN=72, 2
 POS=.05, 10
 SRV=0, 0, 0
 GM0=72, 7.5, 20, ,

;
 NAS=S
 TPA=820,
 NTC=5, 5
 PAS=5000, 1, ,

```
GAS=0,0,,  
POS=0.05,0  
SRV=0,0,0  
GM1=5000,7.5,8,,  
TSM=5,15  
POM=0.3,100,0.15,U255K0  
RAP=,10  
ASM=Z  
;  
*3  
PRO=1  
NAS=X  
PAS=5  
E010=.08  
E020=-.05  
E030=-.077  
E040=.042  
E050=-.006  
NM0=E030
```

5. ХАРАКТЕРИЗАЦИЯ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА. ФАЙЛ PGCFIL

5.1. Технологический процесс

Технологический процесс содержит все характеристики числового управления механической обработкой на станке. Управление технологическим процессом выполняется в двух направлениях:

- программное;
- от станочной консоли.

Программное управление заключается в выполнении станочных циклов из управляющей программы. Управляющая программа на специальном языке запрашивает операции, выполняемые на станке.

Управление от станочной консоли предоставляет возможность оператору вмешиваться в ход выполнения на станке циклов, заданных из управляющей программы. Это вмешательство выполняется изменением позиции переключателей «F%», «S%», «JOG», «ВЫБОР РЕЖИМА РАБОТЫ», расположенных на станочной консоли.

5.2. Особенности форматированных файлов

Управление технологическим процессом обеспечивается доступом к ряду файлов, используемых для обслуживания инструментов и определения начальных точек осей.

К таким файлам относятся:

- файлы начальных точек;
- файлы корректоров инструментов;
- файлы срока службы инструментов;
- файлы свободного расположения инструментов в магазине инструментов.

Имена этих файлов должны быть объявлены в файле **PGCFIL** в инструкции **FIL** для каждого объявленного процесса.

Для каждого типа файлов рекомендуем использовать следующие имена:

- для файлов начальных точек: **FI1EOR, FI2EOR, FI3EOR, FI4EOR, FI5EOR**;
- для файлов корректоров инструментов: **FI1COR, FI2COR, FI3COR, FI4COR, FI5COR**;
- для файлов срока службы инструментов: **GE1TOL, GE2TOL, GE3TOL, GE4TOL, GE5TOL**;
- для файлов свободного расположения инструментов в магазине инструментов: **FI1RAN, FI2RAN, FI3RAN, FI4RAN, FI5RAN**.

Если УЧПУ управляет только одним процессом, то для этих файлов рекомендуется использовать следующие имена: **FI1EOR, FI1COR, GE1TOL, FI1RAN**.

Для создания форматированных файлов начальных точек, корректоров и срока службы инструментов должна быть использована команда **FOR** с соответствующими параметрами (см. «Руководство оператора»). Эту команду можно сформировать и выполнить двумя способами. Преимущество имеет способ создания файлов с использованием опции меню «**FOR**» из меню режима «**COMMAND**»: **F2 (MODIF) -> F2 (FOR)**. В этом случае создание форматированных файлов оператор может выполнить в

диалоговом режиме работы, что исключит возможные ошибки ввода строки формата в файле **FORMAT**. Создание файлов начальных точек, корректоров и срока службы инструментов подробно представлено в «Руководстве оператора».

Файл свободного расположения инструментов в магазине инструментов не является форматированным файлом и создается командой **EDI,FI1RAN/MP0**. Этот файл представляет собой таблицу инструментов, используемую в системе для поиска и смены инструментов. Работа с этим файлом описана в «Руководстве по программированию интерфейса PLC» в разделе «Функция **T**».

5.3. Загрузка файла **PGCFIL**

Файл **PGCFIL** – файл характеристики процесса. Первой инструкцией файла устанавливается слово **NEW** или **OLD**.

Если установлено слово **NEW**, то при инициализации УЧПУ сбрасывается вся память, накопленная ранее для объявленных в системе процессов:

- выбранная командой **SPG** управляющая программа;
- функция **T** (номер инструмента, установленного в шпинделе) и т. п.

Если установлена инструкция **OLD**, при запуске программного обеспечения УЧПУ восстанавливается вся память, накопленная ранее для объявленных в системе процессов:

- выбранная командой **SPG** управляющая программа;
- функция **T** (номер инструмента, установленного в шпинделе);
- значения сигналов пакета «**T**», присвоенных из ПЛ;
- возможность работы по команде **RCM** после перезапуска программного обеспечения или перезапуска УЧПУ.

Примечание. Значения переменных **E** и **SYVAR** не сохраняются.

ВНИМАНИЕ! - УСЛОВИЕМ ДЛЯ СОХРАНЕНИЯ НАКОПЛЕННОЙ ИНФОРМАЦИИ О ПРОЦЕССЕ ПЕРЕД ВЫКЛЮЧЕНИЕМ УЧПУ ПО ИНСТРУКЦИИ **OLD** ЯВЛЯЕТСЯ СОСТОЯНИЕ СИГНАЛА **U10K0 (MUSPE)**, РАВНОЕ «1». ДАЖЕ ПРИ ОДНОКРАТНОМ НЕВЫПОЛНЕНИИ ДАННОГО УСЛОВИЯ СУЩЕСТВУЕТ ВЕРОЯТНОСТЬ НАРУШЕНИЯ ТАБЛИЦЫ РАЗМЕЩЕНИЯ ФАЙЛОВ НА ДИСКЕ. ДЛЯ ВОССТАНОВЛЕНИЯ ДИСКА ПОЛЬЗУЙТЕСЬ СТАНДАРТНЫМИ СРЕДСТВАМИ **DOS (SCANDISK)** ИЛИ РЕЖИМОМ ВОССТАНОВЛЕНИЯ С ДИСКЕТЫ, СОДЕРЖАЩЕЙ АРХИВНЫЙ ФАЙЛ **FLASH** ДИСКА, КОТОРЫЙ БЫЛ СДЕЛАН ВАМИ РАНЕЕ.

5.4. Содержание файла **PGCFIL**

Файл **PGCFIL** состоит из шести секций. Если УЧПУ управляет несколькими процессами, то секции 2, 4, 5 и 6 должны быть записаны для каждого процесса. Секции 1 и 3 являются общими для всех процессов.

Создать или открыть на редактирование уже существующий файл **PGCFIL** можно в режиме «**COMMAND**» командой **EDI,PGCFIL/MP0**, а также используя меню режима «**COMMAND**»: **F2 (MODIF) -> F1 (EDI) -> F7**

(MP0) -> установить в списке файлов курсор на имя файла **PGCFIL** или набрать в строке редактирования имя **PGCFIL** и завершить ввод команды по клавише **ENTER**.

ПРИМЕЧАНИЯ:

1. Имя файла **PGCFIL** в конкретном УЧПУ необходимо уточнить в секции 2 файла **FCRSYS/MP0**.

2. Порядок набора команды, используя меню режима **«COMMAND»**, здесь записан от верхнего уровня меню, на который можно выйти, нажимая несколько раз клавишу **ESC**.

Секция 1 является необязательной (факультативной); позволяет персонализировать трёхбуквенные коды, применяемые для управления процессом. Если эта секция отсутствует, то по умолчанию принимается таблица трёхбуквенных кодов, содержащаяся в памяти УЧПУ. Коды, записанные в секции 1, являются общими для всех конфигурируемых процессов.

Секция 2 факультативная; позволяет персонализировать переменные **SW** УЧПУ. Каждая инструкция секции определяется наименованием переменной **SW** УЧПУ или ключевым словом **«NEW»**. Секция 2 является специфичной для каждого процесса.

Секция 3 факультативная; позволяет персонализировать трёхбуквенные коды **JCL**. Если эта секция отсутствует, то по умолчанию принимается таблица трёхбуквенных кодов **JCL**, содержащаяся в памяти УЧПУ. Инструкции секции 3 являются общими для всех конфигурируемых процессов.

Секция 4 обязательная; позволяет персонализировать: библиотеку технологических программ; файлы (начальных точек, корректоров инструмента, срока службы инструмента, произвольного размещения инструментов в магазине инструментов (**TOOL RANDOM**)); параметры языка программирования **ASSET**. Секция 4 является специфичной для каждого процесса.

Секция 5 обязательная, персонализирует управляемое оборудование (станок). Секция 5 является специфичной для каждого процесса.

Секция 6 факультативная; позволяет персонализировать оси для перемещения и корректоры, расположенные на консоли пульта оператора или станочного пульта. Секция 5 является специфичной для каждого процесса.

5.5. Секция 1

Секция 1 является необязательной; позволяет персонализировать трёхбуквенные коды, применяемые для управления процессом. Если эта секция отсутствует, то по умолчанию принимается таблица трёхбуквенных кодов, содержащаяся в памяти УЧПУ. Коды, записанные в секции 1, являются общими для всех конфигурируемых процессов.

5.5.1. Инструкция TRI

Инструкция **TRI** используется для изменения наименований трёхбуквенных кодов УП или их удаления.

Семантика:

TRI=old_name, new_name, synchro_code .

Формат записи:

TRI=ASCII string, ASCII string, hexadecimal (2 цифры) ,

где:

old_name - наименование трёхбуквенного кода, которое изменяется или удаляется;

new_name - наименование трёхбуквенного кода, которое изменяется, или буква **D**, если осуществляется удаление трёхбуквенного кода;

synchro_code - коды синхронизации для трёхбуквенных кодов могут принимать следующие значения:

- 1** - с запросом синхронизации и без переключателя;
- 2** - с запросом синхронизации, с переключателем;
- 3** - без запроса синхронизации, с переключателем;
- 4** - без запроса синхронизации и без переключателя.

Если код синхронизации не будет объявлен, то он примет значение по умолчанию, указанное в таблице 5.1.

Если код синхронизации для трёхбуквенного кода установлен значением 1, то в УП кадр с этим трёхбуквенным кодом будет выполнен только после окончания расчета и выполнения предыдущего кадра. Если код установлен значением 4, то выполнение кадра с этим трёхбуквенным кодом будет начато без ожидания окончания расчета и выполнения предыдущего кадра.

Если код синхронизации для трёхбуквенного кода установлен значениями 2 или 3, то эту установку можно менять непосредственно в УП символами синхронизации «#» и «&». Для этого в кадре УП перед трёхбуквенным кодом записывается один из символов синхронизации «#» и «&».

- **#** - с запросом синхронизации;
- **&** - без запроса синхронизации.

Если код синхронизации для трёхбуквенного кода установлен значениями 1 или 4, то символы «#» и «&» игнорируются.

Если эта секция отсутствует, используется таблица трёхбуквенных кодов, которая находится в памяти УЧПУ. Значения её представлены в таблице 5.1.

Таблица 5.1 - Трёхбуквенные коды, находящиеся в памяти УЧПУ

Трёхбуквенный код	Код синхронизации по умолчанию	Трёхбуквенный код	Код синхронизации по умолчанию	Трёхбуквенный код	Код синхронизации по умолчанию
CLS	3	DPT	3	RED	3
BNC	3	DIS	3	SCR	3
BGT	3	TOF	3	WRT	3
BLT	3	RQU	3	EXE	2
BEQ	3	RQP	3	SND	2
BNE	3	UCG	3	WAI	1
BGE	3	CLG	3	REL	1
BLE	3	DCG	3	GTA	2
EPP	3	UAV	1	PTA	2
RPT	3	USS	1	DEF	3
ERP	3	FIL	3	EAX	2
UAO	3	TGL	3	DAX	2
UOT	3	DFP	3	MOV	1
UIO	3	EPF	3	WTA	1
MIR	3	SPA	3	DAW	2
URT	3	SPF	3	ADR	2
SCF	1	SPP	3	TRD	2
RQO	1	CLP	3	MPT	1
DPI	1	DLY	3	MDV	1
DTL	1	CAN	3	RLG	3
DLO	3	CLO	3	BSP	3
DSA	3	CRE	3	ESP	3
ASC	3	DER	3	-	-
DSC	3	INP	3	-	-
CTL	3	OPN	3	-	-
DAM	3	OUT	3	-	-

Примеры

1) TRI = CLS, D, - удаляет трёхбуквенный код CLS.

2) TRI = DIS, VIS, - заменяет наименование трёхбуквенного кода DIS на VIS.

3) TRI=EXE, RUN,

Эта запись заменяет имя **EXE** именем **RUN**. Код синхронизации для трёхбуквенного кода **RUN** по умолчанию (2) «с запросом синхронизации, с переключателем». Если символ "&" вставляется в кадре перед записью **RUN**, например:

```
...
<блок кадров>
&(RUN, PPRG/MP1, 2)
...
```

То запрос синхронизации трёхбуквенного кода **RUN** игнорируется, и кадр с **RUN** будет выполнен до окончания выполнения уже рассчитанного блока кадров.

5.6. Секция 2

Секция 2 является необязательной и предназначена для характеристики системных переменных, содержащихся в символьной таблице, значения которых представлены в таблице 5.2. Секция 2 записывается для каждого процесса.

Возможны следующие типы системных переменных:

- 1) boolean;
- 2) byte;
- 3) integer;
- 4) loon integer;
- 5) real;
- 6) loong real;
- 7) ASCII string.

Таблица 5.2 – Системные переменные символьной таблицы

Наименование переменной (<i>var_name</i>) (1)	Максимальное количество по умолчанию (<i>max_num</i>) (2)	Тип по умолчанию (<i>def_type</i>) (3)	Допустимый тип (<i>amm_type</i>) (4)	Код синхрони- зации (<i>synchro_code</i>) (5)	Параметр модифика- ции
E	40	6	62	3	2
o	5	0	0	3	2
p	15	0	0	3	2
l	15	0	0	3	2
c	15	0	0	3	2
TMR	1	6	32	3	1
UOV	1	6	32	3	1
JOG	1	6	32	3	1
RTA	1	6	8	3	1
RT0	1	6	8	3	1
ERF	1	6	32	3	1
MCD	1	6	32	3	1
USB	1	1	1	3	1
UVR	1	1	1	3	1
US0	1	1	1	3	1
URL	1	1	1	3	1
UCV	1	2	2	3	1
RAP	1	1	1	3	1
MDF	1	1	1	3	1
UAS	1	1	1	3	1
RMS	1	2	2	3	1
UEP	1	1	1	3	1
SA	1024	1	127	2	1
SK	1024	1	127	2	1
SN	1024	1	127	2	1
SYVAR	200	2	127	3	1
TIM	7	4	16	3	1
TOT	7	4	16	3	1
SSL	1	6	32	4	1
ACP	1	6	32	4	1
VOL	1	1	1	3	1
ERR	1	1	1	3	1
IOSTA	1	2	2	3	1
MBR	1	1	1	3	1
RTR	1	1	1		1
SRT	1	6	32		1
VRT	1	6	32		1
PRT	1	6	32		1

ПРИМЕЧАНИЯ:

1. Значение параметра модификации представляет собой номер колонки таблицы 5.2 (номера колонок записаны в заголовке таблицы 5.2), наименование которой соответствует параметру, который можно модифицировать.
2. Допустимое количество для каждой переменной **o**, **p**, **l**, **c** и **E** равно 65535.

5.6.1. Инструкция PRO

Инструкция **PRO** предназначена для объявления номера текущего характеризуемого процесса.

Семантика:

PRO=proc_num .

Формат записи:

PRO=integer ,

где:

proc_num - объявляется номер процесса, являющегося текущим при характеристике (от 1 до 5). Номер процесса не должен превышать число процессов, объявленных в инструкции **NBP**. Инструкция **PRO** должна быть записана для каждого объявляемого процесса.

5.6.2. Инструкция SIM

Инструкция **SIM** используется для изменения атрибутов характериземой переменной или определения новых переменных. Инструкция должна быть записана для каждого процесса. В каждом процессе может быть объявлено максимально 20 новых переменных символьной таблицы 5.2.

Семантика:

SIM=var_name, var_new, max_num, def_type, amm_type, synchro_code .

Формат записи:

SIM=ASCII string, ASCII string, word, hexadecimal (2 цифры), word, hexadecimal (2 цифры) ,

где:

var_name - наименование переменной, один из атрибутов которой необходимо заменить, или **NEW**, если вводится новая переменная;

var_new - новое наименование переменной, которое заменяет старое, указанное в параметре **var_name**, или наименование новой вводимой переменной, если в параметре **var_name** записано слово **NEW**;

max_num - максимальное количество элементов, допустимое для переменной любого типа равно 65535;

def_type - код типа переменной по умолчанию (шестнадцатеричный); при определении параметра пользуйтесь таблицей 5.4; для переменных **o**, **p**, **l**, **c** параметр **def_type** не существует, поэтому в таблице 5.2 записано значение «0»;

amm_type - число, соответствующее возможному формату объявляемой переменной. Если переменная может быть разных форматов, это число представляет собой сумму числовых значений различных форматов. При определении параметра пользуйтесь таблицей 5.5.

Пример

amm_type = 54.

Число 54 интерпретируется как 32+16+4+2, т.о. возможными форматами переменной могут быть: **real**, **long integer**, **integer**, **byte**.

synchro_code - код синхронизации объявляемой переменной (аналогично инструкции **TRI**).

Соответствие кодов типам переменных представлено в таблице 5.4.

Таблица 5.4 - Код типа переменной

Тип переменной	Код типа переменной
boolean	1
byte	2
integer	3
long integer	4
real	5
long real	6
ASCII	7

Соответствие цифрового значения формата типам переменных представлено в таблице 5.5.

Таблица 5.5 - Цифровое значение формата переменной

Тип переменной	Цифровое значение формата
boolean	1
byte	2
integer	4
long integer	8
real	16
long real	32
ASCII	64

Пример

SIM = E,,200,,, - инструкция меняет количество переменных E с 40 (по умолчанию) до 200.

5.7. Секция 3

Секция 3 является необязательной; позволяет персонализировать трёхбуквенные коды **JCL**. Если эта секция отсутствует, то по умолчанию принимается таблица трёхбуквенных кодов **JCL**, содержащаяся в памяти УЧПУ. Инструкции секции 3 являются общими для всех конфигурируемых процессов.

5.7.1. Инструкция JCL

Инструкция **JCL** меняет наименование трёхбуквенных кодов **JCL**. Трёхбуквенные коды **JCL** используются в режиме «УПРАВЛЕНИЕ СТАНКОМ». Команды с кодами **JCL** могут быть набраны оператором УЧПУ в строке ввода/редактирования и выполнены по клавише «**ENTER**», или записаны в файле **FILCMD**, и выполнены по запросу ПЛ.

Семантика:

JCL=old_name, new_name, synchro_code .

Формат записи:

JCL=ASCII string, ASCII string, hexadecimal (2 цифры),

где:

old_name - наименование трёхбуквенного кода, которое необходимо изменить;

new_name - может быть наименованием трёхбуквенного кода, на которое меняется старое наименование, или буквой **D**, если выполняется удаление трёхбуквенного кода, указанного в параметре **old_name**;

synchro_code - код синхронизации (значение присваивается аналогично тому, как в инструкции **TRI**). Если код не объявляется, то по умолчанию принимается код из таблицы 5.6. Код синхронизации не может быть изменён, т.к. коды **JCL** выполняются с клавишей **ENTER**.

Таблица 5.6 - Код синхронизации по умолчанию

Трёхбуквенные коды JCL	Код синхронизации по умолчанию	Трёхбуквенные коды JCL	Код синхронизации по умолчанию
CAC	4	RIF	4
CAO	4	RLG	4
CTU	4	VLG	4
ORA	4	MTO	4
VOA	4	DBT	4
UCA	4	GSE	4
VTU	4	UCG	4
DIS	4	CLG	4
DPT	4	DCG	4
EVA	4	URP	4
RCM	4	PTM	4
ERM	4	VIC	4
ESE	4		
SPG	4		
REL	4		

Примеры

- 1) **JCL = CAO, D**, - инструкция удаляет трёхбуквенный код CAO.
- 2) **JCL = VTU, UTV**, - инструкция меняет наименование трёхбуквенного кода VTU на код UTV. Код синхронизации по умолчанию - 4.

5.8. Секция 4

Секция 4 используется для характеристики библиотеки УП, файлов начальных точек, корректоров, произвольного инструмента (**RANDOM TOOL**), а также параметров языка **ASSET**. Секция 4 записывается для каждого процесса и содержит следующие инструкции: **PRO**, **ASS**, **NPL**, **NDD**, **PRF**, **FIL**, **FLC**, **STR**, **CHN**, **SCR**.

5.8.1. Инструкция PRO

Инструкция **PRO** предназначена для объявления номера текущего характеризуемого процесса.

Семантика:

PRO=proc_num .

Формат записи:

PRO=integer ,

где:

proc_num - объявляется номер процесса, являющийся текущим при характеристике (от 1 до 5). Номер процесса не должен превышать число процессов, объявленных в инструкции **NBP**. Инструкция **PRO** должна быть записана для каждого объявляемого процесса.

5.8.2. Инструкция ASS

Инструкция **ASS** используется для присвоения значения вещественного типа системным переменным. Значение вещественного типа, присвоенное системной переменной, становится форматом переменной по умолчанию. Если переменная - двоичного типа, то существует два способа присвоения ей значения вещественного типа:

- 1) для значения больше нуля ей присваивается значение, равное **1**;
- 2) для значения, равного нулю, присваивается значение **0**.

Семантика:

ASS=var_name, value .

Формат записи:

ASS=ASCII string, real ,

где:

var_name - наименование системной переменной, которой присваивается значение;

value - значение вещественного типа, которое присваивается системной переменной.

- ASS=TMR, value** (формат **long real**) - определяет временной интервал при **G04**.
- ASS=UOV, value** (формат **long real**) - определяет величину припуска.
- ASS=JOG, value** (формат **long real**) - определяет величину перемещения в режиме «**РУЧНЫЕ ФИКСИРОВАННЫЕ ПЕРЕМЕЩЕНИЯ**».
- ASS=RTA, value** (формат **long real**) - определяет величину переквалификации измерительного щупа для оси абсцисс.
- ASS=RT0, value** (формат **long real**) - определяет величину переквалификации измерительного щупа для оси ординат.
- ASS=ERF, value** (формат **long real**) - определяет максимальную динамическую ошибку формы.
- ASS=MCD, value** (формат **long real**) - определяет максимальную величину отклонения направляющих косинусов между двумя соседними элементами. Значение лежит в пределах от 0 до 2; если объявляемое значение превышено, то выполняется останов. Значение, объявленное в **MCD**, используется, пока включён режим динамики движения **G27**.

Пример

ASS=MCD, 1 - объявленное отклонение равно 90 градусам. Если это значение превышено, то выполняется останов.

- ASS=USB, value** (формат **boolean**) - может принимать следующие значения:
- 1** - разрешает выполнение кадров с символом «/» (пропуск);
 - 0** - запрещает выполнение кадров с символом «/» (пропуск).
- ASS=UVR, value** (формат **boolean**) - может принимать следующие значения:
- 1** - разрешает выполнение кадров УП со скоростью быстрого хода взамен запрограммированной скорости;
 - 0** - отменяет вышеописанную функцию.
- ASS=US0, value** (формат **boolean**) - может принимать следующие значения:
- 1** - разрешает выполнение останова по **M01**;
 - 0** - запрещает выполнение останова по **M01**.
- ASS=URL, value** (формат **boolean**) - может принимать следующие значения:
- 1** - разрешает коррекцию скорости быстрого хода корректором ручной подачи «**JOG**»;
 - 0** - отменяет вышеописанную функцию.
- ASS=UCV, value** (формат **byte**) - может принимать следующие значения:
- 0** - активизирует индикацию расчетных значений координат;
 - 1** - активизирует индикацию абсолютных значений координат, считанных с датчиков;

- 2** - активизирует индикацию ошибки позиции осей;
- 3** - активизирует индикацию остатка пути.
- 4** - активизирует индикацию значений координат, считанных с датчика, относительно активной начальной точки;
- 5** - активизирует индикацию коррекции позиции осей, накопленную при использовании компенсационного штурвала.

ASS=RAP, value (формат **boolean**) - может принимать следующие значения:

- 1** - разрешает автоматический возврат на профиль после останова (**HOLD**) и/или автоматический выход в позицию абсолютного микропуля;
- 0** - отменяет вышеописанную функцию.

ASS=UAS, value (формат **boolean**) - может принимать следующие значения:

- 1** - отключает оси для испытания программ;
- 0** - подключает оси.

ASS=RMS, value (формат **byte**) - определяет процент изменения скорости при выводе инструмента из отверстия в цикле нарезания резьбы метчиком.

Пример

- ASS=RMS, 110** - увеличивает на 10% скорость возврата.
- ASS=RMS, 10** - уменьшает на 90% скорость возврата.

ASS=UEP, value (формат **boolean**) - может принимать следующие значения:

- 0** - разрешает скоростную компенсацию (с **VFF**);
- 1** - отменяет вышеописанную функцию (без **VFF**).

ASS=VOL, value (формат **boolean**) - может принимать следующие значения:

- 1** - разрешает штурвал;
- 0** - запрещает штурвал.

ASS=SSL, value (формат **long real**) - определяет максимальное количество оборотов шпинделя.

ASS=SK1023, value (формат **byte**) - определяет величину 1024-го байта пакета «К», соответствующего **W255K3**.

ASS=MDF, value (формат **long real**) - коэффициент уменьшения скорости движения по эквидистанте при выполнении круговой интерполяции с внутренней стороны. Коэффициент может принимать значения от 0 до 1. Подробное описание **MDF** в «Руководстве программиста MC».

ASS=SRT, value (формат **long real**) - определяет шаг дробления стружки.

Значение шага дробления стружки выражено в миллиметрах или дюймах для движения, заданного по линейным осям или совместного движения линейных осей и осей вращения. Для движения, заданного только по координатным осям вращения, значение шага дробления стружки выражено в градусах.

Пример

ASS=SRT, 10

ASS=VRT, value (формат **long real**) - определяет скорость дробления стружки.

Значение скорости дробления выражено коэффициентом в относительных единицах. Для определения скорости дробления в абсолютных единицах значение коэффициента необходимо умножить на текущую подачу по оси.

Пример

ASS=VRT, 0.2

ASS=PRT, value (формат **long real**) - определяет выдержку времени дробления стружки, в секундах в режиме **G94** или в оборотах в режиме **G95**, выполняемую после очередного шага дробления стружки.

Пример

ASS=PRT, 2

ASS=E25, value (формат **real**).

ASS=E30, value (формат **long real**).

Примечание. Если превышены объявленные максимальные значения, ошибка не визуализируется.

5.8.3. Инструкция NPL

Инструкция **NPL** определяет максимальное количество подпрограмм и меток, вызываемых в ходе выполнения последовательности обработки детали.

Семантика:

NPL=prog_num, label_num .

Формат записи:

NPL=integer, integer ,

где:

prog_num - количество подпрограмм, которые вызываются командой **CLS** (максимум 255) в ходе выполнения последовательности обработки детали; если этот параметр опущен, УЧПУ по умолчанию принимает значение 10;

label_num - количество меток (максимум 255), вызываемых в ходе выполнения последовательности обработки детали; если

этот параметр опущен, УЧПУ по умолчанию принимает значение **10**.

Примечание. Следует помнить, что для каждой метки отводится 23 байта, а для каждой программы – 12 байтов памяти. Эта инструкция необходима для определения области памяти, занимаемой адресами меток и программ.

5.8.4. Инструкция NDD

Инструкция **NDD** используется для объявления устройства памяти **MPx**, используемого для хранения управляющих программ.

Семантика:

NDD=MPx .

Формат записи:

NDD=ASCII string ,

где:

MPx - символьное имя ($x=0\div6$) устройства памяти для хранения УП, используемого по умолчанию.

Если инструкция **NDD** не записана, по умолчанию используется устройство памяти **MP1**.

5.8.5. Инструкция PRF

Инструкция **PRF** устанавливает количество профилей и кадров УП черновой токарной обработки.

Семантика:

PRF=prof_num, block_num .

Формат записи:

PRF=integer, integer ,

где:

prof_num - максимальное количество профилей, которые могут быть определены трёхбуквенным кодом **DFP**. Если этот параметр не определён, УЧПУ по умолчанию принимает значение, равное 8;

block_num - максимальное количество кадров в профиле, определённом кодом **DFP**; если этот параметр не определён, УЧПУ по умолчанию принимает значение, равное 16;

Значения параметров этой инструкции определяют объём памяти в **ОЗУ**, необходимый для их размещения.

Инструкция используется только для токарных станков.

Для активизации параметров этой инструкции, принимаемых по умолчанию, необходимо записать: **PRF=**,

Для использования токарных циклов визуального программирования, описанных в документе «Руководство оператора. Визуальное программирование», необходимо записать: **PRF=255,255** .

5.8.6. Инструкция FIL

Инструкция **FIL** определяет данные файлов технологического процесса.

Семантика:

FIL=name1/MPx,name2/MPx,name3/MPx,name4/MPx,name5/MPx .

Формат записи:

FIL=ASCII string,ASCII string,ASCII string,ASCII string,ASCII string ,

где:

name1 - наименование файла начальных точек; обычно – **FILEOR**;

name2 - наименование файла корректоров; обычно – **FILCOR**;

name3 - наименование файла срока службы инструмента;

name4 - наименование файла произвольного размещения инструмента в магазине инструментов (**RANDOM TOOL**), обычно – **FILRAN**;

name5 - не используется;

MPx - устройство памяти, где хранится файл.

5.8.7. Инструкция FLC

Инструкция **FLC** определяет данные файлов технологического процесса.

Семантика:

FLC=name1/MPx,name2/MPx,name3/MPx,name4/MPx .

Формат записи:

FLC=ASCII string,ASCII string,ASCII string,ASCII string,ASCII string ,

где:

name1 - не используется;

name2 - не используется;

- name3** – наименование файла сообщений от ПЛ текущего процесса **FInMSG**, где **n** - номер процесса;
- name4** - не используется;
- MPx** – устройство памяти (**x=0÷6**), где хранится файл сообщений от ПЛ текущего процесса;

Если инструкция **FLC** отсутствует, то имя файла сообщений от ПЛ должно быть указано в секции 2 файла **FCSYS/MP0** с логическим именем **FILMS5**. В этом случае файл сообщений от ПЛ является общим для всех объявленных процессов.

5.8.8. Инструкция STR

Инструкция **STR** определяет количество структур форматированных файлов, к которым может быть осуществлено обращение из УП при использовании языка программирования **ASSET**.

Семантика:

STR=struct_num .

Формат записи:

STR=word ,

где:

struct_num - количество структур форматированных файлов, которое может быть использовано для их создания и доступа к данным, размещённым в их строках. Максимальное количество структур равно 255.

Структура записей форматированного файла должна быть определена перед его созданием.

Структура определяет количество ячеек в каждой записи и формат каждой ячейки форматированного файла.

Примечание. Подробная информация о форматированных файлах и методах доступа к ним из УП в языке **ASSET** приведена в документе «Руководство программиста».

5.8.9. Инструкция SER

Инструкция **SER** определяет память, используемую для передачи данных по последовательному каналу **RS232**.

Семантика:

SER=byte_num .

Формат записи:

SER=hexadecimal (2 цифры) ,
где:

byte_num - количество байтов, используемых для обработки пользователем последовательного порта RS232.

Пример

Если определена следующая инструкция:

SER=0 ,

то для обработки последовательного канала отводится стандартное количество байтов: 850 байтов.

Если определено другое значение, отличное от нуля (**n**), то будет зарезервировано 850+**n** байтов, но не более чем 64 Кб.

5.8.10. Инструкция CHN

Инструкция **CHN** определяет количество каналов доступа к форматированным файлам при использовании языка программирования **ASSET**.

Семантика:

CHN=chan_num .

Формат записи:

CHN=word ,

где:

chan_num - количество логических каналов для доступа к форматированным файлам при использовании языка программирования **ASSET**. Если параметр равен 1, доступ может быть осуществлён только к одному файлу. При установке значения, большего 1, необходимо помнить, что для работы с каждым каналом требуется около 750 байтов пользовательской памяти.

Примечание. Подробно язык **ASSET** описан в документе «Руководство программиста».

5.8.11. Инструкция SCR

Инструкция **SCR** определяет элементы данных визуализации при использовании языка **ASSET**.

Семантика:

SCR=dim .

Формат записи:

SCR=word ,

где:

dim - размер памяти (количество байтов) для дисплея пользователя на языке **ASSET**. Минимальный размер элемента данных - 6 байтов, распределенных следующим образом:

- 1 байт для строки;
- 1 байт для колонки;
- 2 байта (слово) для длины элемента данных;
- 1 байт для каждого символа **ASCII** элемента данных;
- 1 байт для атрибута.

Размер экрана, доступный для программирования визуализации на языке **ASSET**, составляет 19 строк и 78 колонок.

Примечание. Подробные сведения об использовании языка **ASSET** для визуализации приведены в документе «Руководство программиста».

5.9. Секция 5

Секция 5 предназначена для характеристики управляемого оборудования и должна быть записана для каждого процесса. Секция 5 состоит из следующих инструкций: **PRO**, **NIP**, **DPM**, **SMC**, **TOF**, **GXX**, **PRC**, **CWP**, **NAM**, **NPD**, **G70**, **MBR**, **TAS**, **INU**, **CAX** (**ECDF**).

5.9.1. Инструкция PRO

Семантика:

PRO=proc_num .

Формат записи:

PRO=integer ,

где:

proc_num - объявляется номер процесса, являющийся текущим при характеристике (от 1 до 5); номер процесса не должен превышать число процессов, объявленных в инструкции **NBP**. Инструкция **PRO** должна быть записана для каждого объявляемого процесса.

5.9.2. Инструкция NIP

Инструкция **NIP** определяет имя интерполятора координатных осей для текущего процесса.

Семантика:

NIP=int_name .

Формат записи:

NIP=ASCII string ,

где:

int_name - алфавитно-цифровой символ, определяющий имя интерполятора координатных осей, объявленного в **AXCFIL** в инструкции **INX**.

ВНИМАНИЕ! эта инструкция должна быть введена в обязательном порядке после объявления номера процесса.

Пример

NIP=3 .

5.9.3. Инструкция DPM

Инструкция **DPM** определяет рабочие параметры измерения щупом.

Семантика:

DPM=appro_quote, safe_quote, mis_speed .

Формат записи:

DPM=real, real, real ,

где:

appro_quote - определяет расстояние в миллиметрах, которое ось проходит с объявленной (медленной) скоростью измерения для получения точного момента касания; до координат этого расстояния ось движется со скоростью быстрого хода;

safe_quote - размер безопасности щупа в миллиметрах;

mis_speed - скорость измерения (касания), выражается в мм/мин.

Минимальная разрешающая способность измерения щупом зависит от тика интерполятора и от скорости измерения щупом в соответствии с формулой:

$$\text{разрешающая способность} = \frac{\text{mis_speed} \times \text{int_tick}}{1000 \times 60} \quad (5.1).$$

Пример

int_tick = 1 мс;

mis_speed = 100 мм/мин;

$$\text{разрешающая способность} = \frac{100 \times 1}{1000 \times 60} = \frac{100}{60000} = 0.00166 \text{ (мм)} .$$

5.9.4. Инструкция SMC

Инструкция **SMC** определяет максимальную величину коррекции размера инструмента.

Семантика:

SMC=*max_correct* .

Формат записи:

SMC=real ,

где:

max_correct - определяет максимальное значение модификации корректоров инструментов. Параметр используется при переквалификации инструмента после выполнения цикла измерения, т. е. тогда, когда необходимо модифицировать корректор при износе инструмента. Если износ инструмента больше этой величины, то модификация корректора не осуществляется, а выдаётся сигнал об ошибке.

5.9.5. Инструкция TOF

Инструкция **TOF** определяет тип управляемого оборудования от УЧПУ.

Семантика:

TOF=*mach_type* .

Формат записи:

TOF=hexadecimal (2 цифры) ,

где:

mach_type - определяет код типа управляемого оборудования (станка). Значение кода обязательно должно содержать один из кодов типа управляемого оборудования (01; 02; 05 или 06) и может быть дополнено вспомогательными кодами 10 и/или 20; код может принимать следующие значения:

- 01** - только фрезерный станок;
- 02** - только токарный станок;
- 05** - фрезерный станок, переключаемый в токарный режим;
- 06** - токарный станок, переключаемый во фрезерный режим;

Дополнительные коды:

- 10** - дополнительный код, разрешающий в строке ввода и редактирования вводить значения коррекции на длину инструмента по диаметральной оси в диаметральных размерах. Ввод значений коррекции выполняется по

кнопке «**F4**» («Ввод корректора») в видеостраницах **#1** и **#7** режима «**УПРАВЛЕНИЕ СТАНКОМ**». По умолчанию ввод коррекции для диаметральной оси выполняется на радиус;

- 20** - дополнительный код, который инвертирует направление оси ординат. Обычно этот код дополнительно устанавливается для токарных станков, суппорт которого расположен справа от оси вращения шпинделя и его поперечная ось определена в системе осью ординат.

Пример

TOF=12 (02+10 - определяет управление токарным станком (02); с вводом длины инструмента по диаметральной оси в диаметрах (10)).

Если при характеристике **TOF=5**, и активизируется трёхбуквенный код (**CTL,T**), предоставляется возможность использовать на фрезерном станке технологию работы, характерную для токарного станка, а именно:

- выполнять циклы черновой обработки;
- выполнять циклы нарезания резьбы;
- поддерживать постоянную скорость резания.

Если при характеристике **TOF=6**, и используется трёхбуквенный код (**CTL,F**), предоставляется возможность применять на токарном станке технологию работы, характерную для фрезерного станка, т. е. использовать параметры корректоров как корректор длины (**Z**) и корректор диаметра инструмента.

Если УЧПУ при включении имеет конфигурацию для фрезерного станка, то по умолчанию функции **G** будут следующие:

G00 - G80 - G20 - G40 - G27 - G90 - G70 - G17 - G94 - G97.

Если УЧПУ при включении имеет конфигурацию для токарного станка, то по умолчанию функции **G** будут следующие:

G00 - G80 - G20 - G40 - G27 - G90 - G70 - G17 - G95 - G96.

Форматы записи файла корректоров для токарных и фрезерных станков отличаются друг от друга. Однако файл корректоров для токарных станков может содержать информацию, подобную информации для фрезерных станков. В этом случае записи файла корректоров для токарных станков могут быть сконфигурированы с тем же значением, что и для фрезерных станков.

При создании файла корректоров текущего процесса, используя меню режима «**COMMAND**», система использует значение *mach_type*, записанное для этого процесса.

5.9.6. Инструкция GXX

Инструкция **GXX** определяет **G**-функции, инициализируемые при включении УЧПУ.

Семантика:

GXX=class1,class2,class3,class4,class5,class6 .

Формат записи:

GXX=decimal (2 цифры), decimal (2 цифры), decimal (2 цифры), decimal (2 цифры), decimal (2 цифры), decimal (2 цифры) ,

где:

class1-6 – код **G**-функции, инициализируемый по включению УЧПУ. Эта инструкция объявляется в том случае, если пользователь желает инициализировать УЧПУ с конфигурацией, отличной от принятой по умолчанию и определённой инструкцией **TOF**.

ВНИМАНИЕ !

1. инструкция **GXX** должна быть объявлена после инструкции **TOF**.
2. количество параметров в инструкции **GXX** фиксировано и равно 6. Если значение параметра отсутствует, то необходимо записать символ "\", например отсутствуют значения для **class3** и **class6**: **GXX=00,27,,70,94,**
3. если в каких-либо параметрах **class1-6** значения кодов **G**-функций будут отсутствовать, то при включении УЧПУ этим параметрам в системе будут установлены значения, определённые значением типа станка в инструкции **TOF**.
4. можно объявить только одну **G**-функцию для каждого класса. Допустимые коды указаны в таблице 5.7.

Таблица 5.7

Класс	Код G -функции
1	00 – 01 – 02 – 03
2	27 – 28 – 29
3	90 – 91
4	70 – 71
5	93 – 94 – 95
6	96 – 97

Пример

GXX=01,,,70,95,97

5.9.7. Инструкция PRC

Инструкция **PRC** используется для определения точности вычислений.

Семантика:

PRC=so_pre .

Формат записи:

PRC=real ,

где:

so_pre – определяет точность выполнения вычислений, выраженную в мм; если эта инструкция опущена, по умолчанию принимается значение 0.01 мм.

5.9.7.1. Особенности расчёта значения **so_pre**

Начиная с версий **SW** 2.28P, 3.28P, и во всех версиях 4.XXP значение, определённое в инструкции **PRC**, учитывается:

- 1) при выполнении круговой интерполяции. Изменение радиуса дуги в её начальной и конечной точке должно находиться в пределах значения **so_pre**, как указано на рисунке 5.1. Для сопряжения начальной и конечной точек дуги окружности, заданной в пределах значения константы **so_pre**, производится отклонение координаты начала центра окружности от заданной в кадре программы с круговой интерполяцией;

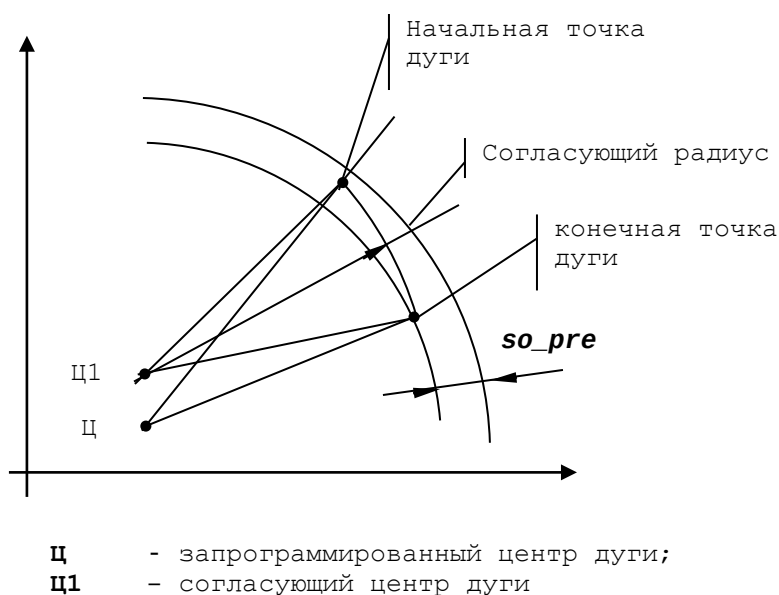
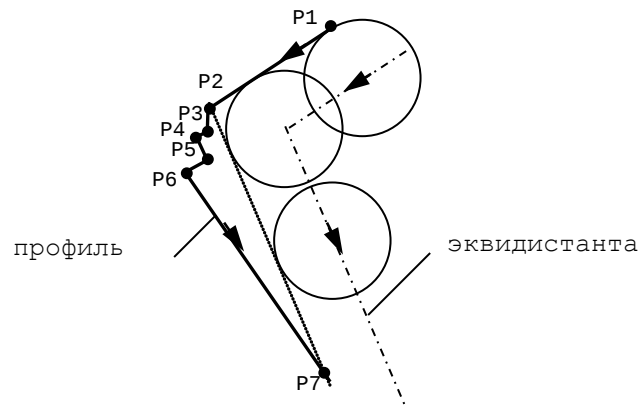


Рисунок 5.1 – Учёт значения **so_pre** при круговой интерполяции

- 2) при выполнении сопряжения геометрических элементов (линии, окружности) в программе на языке **GTL**;
- 3) при расчёте эквидистанты; если линейные перемещения при расчёте эквидистанты меньше, чем значение в **PRC**, такие перемещения будут пропущены.

Таким образом, если эквидистанта содержит много линейных перемещений меньше 0.01 мм (по умолчанию **so_pre=0.01**), и все они должны быть выполнены, то значение **so_pre** необходимо установить меньше минимального перемещения. Но это обязывает более жёстко (в пределах значения **so_pre** рассчитывать координаты начала и конца дуги.

Пропуск кадров с величиной перемещения меньше **so_pre** будет выполнен по схеме, как указано на рисунке 5.2.



Перемещения между точками профиля P1-P2 и P6-P7 больше значения **so_pre**, а между точками P2-P3, P3-P4, P4-P5 и P5-P6 - меньше.

Рисунок 5.2 - Схема пропуска кадров с величиной перемещения меньше **so_pre**

5.9.8. Инструкция CWP

Инструкция **CWP** определяет для различных процессов используемые клавиши и корректора.

Семантика:

CWP=cw_console,offset .

Формат записи:

CWP=hexadecimal (4 цифры),boolean (2 цифры) ,

где:

cw_console - 16-теричный код, определяющий:

- 1) программную поддержку переключателя или клавиш выбора режимов работы (**MDI, AUTO, ..., RESET**);
- 2) должны ли клавиши и корректора, активизируемые для одного процесса, действовать одновременно и на другие процессы;
- 3) запрещена ли видеостраница «ПРОЦЕСС n»;
- 4) возможно ли исполнение кадров УП в соответствии с круговым приоритетом; при установке этого бита последовательность выполнения кадров УП, вызываемых из различных процессов, следующая:
 - 1-й кадр УП - 1-й процесс;
 - 1-й кадр УП - 2-й процесс;
 - 1-й кадр УП - 3-й процесс;
 - 1-й кадр УП - 4-й процесс;
 - 1-й кадр УП - 5-й процесс;
 - 2-й кадр УП - 1-й процесс.

Этот параметр устанавливается в соответствии с данными, приведёнными в таблице 5.8. Значение параметра представляет собой бит или совокупность битов, равных «1» и имеющих разное назначение.

Таблица 5.8 – Назначение битов в параметре **sw_console**

16-теричный код	Бит	Значение	Назначение
01	0	1	Не используется
02	1	1	Клавиша «СТОП»
04	2	1	Корректор скорости шпинделя
08	3	1	Корректор рабочей подачи
10	4	1	Корректор ручной подачи
20	5	1	Переключатель режимов
40	6	1	Клавиша «ПУСК»
80	7	1	Запрещение экрана «ПРОЦЕСС n»
100	8	1	Круговой приоритет
200	9	1	Подключение щупа (датчика касания) в УЧПУ NC-230
400	10	1	Не используется
800	11	1	Не используется
1000	12	1	Не используется
2000	13	1	Не используется
16-теричный код	Бит	Значение	Назначение
4000	14	1	Устанавливает вывод числа в поле значений осей в формате 5.3 «XXXXX.XXX» (по умолчанию формат 5.4 «XXXXX.XXXX»)
8000	15	1	1) NC-200 : код 8000H не устанавливается. 2) NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230 : код 8000H устанавливается для <u>подключения</u> переключателя режимов работы, установленного на станочной консоли пульта оператора. 3) NC-110, NC-310 : код 8000H устанавливается для <u>отключения</u> кнопок переключения режимов работы, установленных на станочном пульте (код 8000H устанавливается при использовании УЧПУ NC-110, NC-310 без станочного пульта).

Примечание. 14-ый бит (таблица 5.8), включённый в суммарный код параметра **sw_console**, устанавливает вывод на экран в поле значений осей, выбираемых по командам UCV=0, UCV=1, ..., UCV=5, в формате 5.3 (три знака после десятичной точки).

offset - параметр может принимать следующие значения:

- 0** - при разработке УП используются: коррекция на длину, радиус инструмента и язык технологического программирования **GTL**;
- 1** - при разработке УП не используются: коррекция на длину, радиус инструмента и язык технологического программирования **GTL**.

Пример 1

Запись инструкции для УЧПУ **NC-210** с индикацией позиции осей, имеющей 3 цифры после десятичной точки: **CWP=C000,0** ,
где: **C000=8000+4000**

Пример 2

Запись инструкции для УЧПУ **NC-230** с индикацией позиции осей, имеющей 3 цифры после десятичной точки: **CWP=C200,0** ,

где: $C200=8000+4000+200$

5.9.9. Инструкция NAM

Инструкция **NAM** определяет наименование оси, параллельной оси шпинделя.

Семантика:

NAM=paral_spind .

Формат записи:

NAM=ASCII string ,

где:

paral_spind - определяет наименование оси, параллельной оси шпинделя, заявленной по умолчанию.

5.9.10. Инструкция NPD

Инструкция **NPD** определяет ось абсцисс и ось ординат, используемые в текущем процессе по умолчанию.

Семантика:

NPD=abs_axis, ord_axis .

Формат записи:

NPD=ASCII string, ASCII string ,

где:

abs_axis - определяет по умолчанию ось абсциссы плоскости станка;

ord_axis - определяет по умолчанию ось ординаты плоскости станка.

ВНИМАНИЕ! Если процесс содержит только одну ось, следует определить в обоих параметрах инструкции **NPD** одно и то же имя оси.

5.9.11. Инструкция G70

Инструкция **G70** определяет единицы измерения, в которых задаются параметры характеристики.

Семантика:

G70=value .

Формат записи:

G70=hexadecimal (2 цифры) ,

где:

value - код, определяющий единицы значений параметров в файлах характеристики:

0 – значения параметров заданы в миллиметрах;

1 – значения параметров заданы в дюймах.

Если инструкция **G70** опускается, по умолчанию за единицу измерения принимается миллиметры.

5.9.12. Инструкция MBR

Инструкция **MBR** определяет количество кадров УП, которое можно обрабатывать в режиме движения по профилю в обратном направлении.

Семантика:

MBR=retrace .

Формат записи:

MBR=word ,

где:

retrace - определяет максимальное число кадров УП, выполняемых по профилю в обратном направлении в режимах «АВТОМАТИЧЕСКИЙ» («**AUTO**») и «КАДР» («**STEP**»). Значение может изменяться от 1 до 255.

5.9.13. Инструкция TAS

Инструкция **TAS** определяет функциональные параметры измерительного щупа, используемые в циклах **G72** и/или **G73**.

Для подключения измерительного щупа (датчика касания) установлены два способа.

Способ 1: подключение измерительного щупа через специальный канал – канал датчика касания (см. «Руководство по эксплуатации»);

Способ 2: подключение измерительного щупа через сигнал **PLC** .

Семантика:

TAS=probe_input,probe_status,path,type,mes_input,mes_status .

Формат:

TAS=PLC variable, hexadecimal (2 цифры), ASCII string, ASCII string, PLC variable, hexadecimal (2 цифры) ,

где:

probe_input - идентификатор входного сигнала **PLC**, используемого для считывания состояния измерительного щупа. Изменение состояния сигнала **probe_input** обеспечивает аварийный возврат щупа в исходную координату измерительного цикла на быстром ходу. Этот сигнал используется в **SW** только обеспечения безопасности измерительного щупа в цикле измерения и не предназначен для определения измеряемых координат; параметр устанавливается при подключении щупа и способом 1 и способом 2.

probe_status - определяет исходное значение входного сигнала **probe_input**; **probe_status** может иметь два значения:

- «1» - если в исходном состоянии измерительный щуп имеет нормально замкнутый контакт;
- «0» - если в исходном состоянии измерительный щуп имеет нормально разомкнутый контакт;

path - указывает преимущественную позицию приближения измерительного щупа для осей, формат следующий:

наименование оси (**X, Y, Z, U, V, W**), значение **path** может иметь знак «+» или «-»;

type - определяет тип измерительного щупа:

S - ориентируемый,
N - неориентируемый;

mes_input - идентификатор входного сигнала **PLC**, используемого для считывания состояния измерительного щупа. Изменение состояния сигнала **mes_input** обеспечивает измерение координаты в измерительных циклах **G72, G73** и **G74**, а также возврат щупа в исходную координату измерительного цикла на быстром ходу. Этот сигнал используется в **SW** только для измерения и не должен иметь имя, записанное в **probe_input**.

Значение **mes_input** записывается только при подключении щупа способом 2.

mes_status - определяет значение входного сигнала **PLC mes_input**, получаемого от измерительного щупа:

«1» - нормально замкнутый контакт,
«0» - нормально разомкнутый контакт.

Примечание. Поля **path** и **type** в настоящий момент не используются.

При подключении щупа первым способом измерительный канал щупа не требует какой-либо характеристики (параметры **mes_input** и **mes_status** в инструкции **TAS** отсутствуют).

При установке значений **mes_input** и **mes_status** в инструкции **TAS** первый способ подключения канала измерения щупа игнорируется, и измерение выполняется по способу 2.

5.9.14. Инструкция INU

Инструкция **INU** определяет функциональные параметры измерительного щупа, используемые в цикле **G74**.

Семантика:

INU=probe_input,probe_status .

Формат записи:

INU=PLC variable,hexadecimal (2 цифры) ,

где:

probe_input - идентификатор входного сигнала PLC, используемого для считывания состояния измерительного щупа в цикле **G74**. Изменение состояния сигнала **probe_input** обеспечивает аварийный возврат щупа в исходную координату измерительного цикла на быстром ходу. Этот сигнал используется в **SW** только обеспечения безопасности измерительного щупа в цикле измерения и не предназначен для определения измеряемых координат; параметр устанавливается при подключении щупа и способом 1 и способом 2; имя сигнала **probe_input** может совпадать с именем сигнала, записанным в **probe_input** в инструкции **TAS**;

probe_status - определяет исходное значение входного сигнала, записанного в **probe_input**; **probe_status** может иметь два значения:

- «1» - если в исходном состоянии измерительный щуп имеет нормально замкнутый контакт;
- «0» - если в исходном состоянии измерительный щуп имеет нормально разомкнутый контакт.

5.9.15. Инструкция CAX (ECDF)

Инструкция **CAX** предназначена для указания имён трёх осей текущего процесса, используемых в определении объёмной защищённой зоны «**CUB**».

Семантика:

CAX=axis_name_1,axis_name_2,axis_name_3 .

Формат записи:

CAX=ASCII string,ASCII string,ASCII string ,

где:

axis_name(_1,_2,_3) - последовательность из трёх имён осей, которые определяют объёмную защищённую зону «**CUB**».

5.9.16. Инструкция QFD

Инструкция **QFD** предназначена для определения значения альтернативной подачи, к подаче заданной в кадре или блоке кадров, выполняемых в режимах **«MDI»**, **«AUTO»** и **«STEP»**.

Семантика:

QFD=value .

Формат записи:

QFD=real ,

где:

value – значение альтернативной подачи, к подаче заданной для кадра (блока кадров) с адресом **F** или заданной подготавливающей функцией **G00** (скорость быстрого хода). Значение **QFD** выражено в мм/мин.

Запрос перехода на альтернативную подачу устанавливается в ПЛ интерфейсным сигналом **PLC QCKFD** на то время, пока этот сигнал установлен в состояние 1.

Переход на альтернативную подачу будет выполнен, если текущая подача с адресом **F** запрограммирована с активной **G94**.

Если инструкция **QFD** опущена, то запрос замены текущей подачи, заданной в кадре, на альтернативную подачу игнорируется.

5.9.17. Инструкция TGA

Инструкция **TGA** предназначена для определения имени оси вращения, используемой в режиме поддержания инструмента тангенциально к профилю (режим **TGM**). Применение режима **TGM** рассмотрено в документе «Руководстве программиста MC».

Семантика:

TGA=axis_name, angle .

Формат записи:

TGA=ASCII string, real ,

где:

axis_name – определяет имя непрерывной координатной оси вращения, управляемой в режиме **TGM**.

angle – угол между положительным направлением оси ординат и направлением абсолютного микропуля оси вращения. В случае если ось вращения невозможно вывести в ноль, то **angle** должен быть равен 0.

5.10. Секция 6

Секция 6 предназначена для определения осей, выбираемых для ручных перемещений клавишами станочной консоли, а также значений её корректоров. Секция 6 записывается для каждого процесса и содержит следующие инструкции: **PRO**, **MAS**, **FR0**, **SS0**, **FM0**.

5.10.1. Инструкция PRO

Инструкция **PRO** предназначена для объявления номера текущего характеризуемого процесса.

Семантика:

PRO=proc_num .

Формат записи:

PRO=integer ,

где:

proc_num - объявляется номер процесса, являющийся текущим при характеристике (от 1 до 5). Номер процесса не должен превышать число процессов, объявленных в инструкции **NBP**. Инструкция **PRO** должна быть записана для каждого объявляемого процесса.

5.10.2. Инструкция MAS

Инструкция **MAS** предназначена для объявления наименования осей, участвующих в движении в режиме «РУЧНЫЕ ПЕРЕМЕЩЕНИЯ».

Семантика:

MAS=axis_name .

Формат записи:

MAS=ASCII string ,

где:

axis_name - список наименований осей, которые могут участвовать в режиме «РУЧНЫЕ ПЕРЕМЕЩЕНИЯ». Координаты этих осей визуализируются на видеоэкране УЧПУ. Максимально может быть объявлено 7 осей для УЧПУ **NC-110** и **NC-310**, 5 осей - для **NC-230**, 4 оси - для **NC-200**, **NC-210**, **NC-220** и 3 оси - для **NC-201**, **NC-201M**, **NC-202**. Наименования осей не разделяются запятыми.

Пример

MAS = XYZ

5.10.3. Инструкция FRO

Инструкция **FRO** предназначена для объявления значений корректора подачи «F».

5.10.3.1. Инструкция FRO для УЧПУ NC-301 и NC-302

Семантика:

FRO=feed_1, feed_2, ..., feed_17 .

Формат записи:

FRO=real, real, ..., real ,

где:

feed_1, feed_2, ..., feed_17 - эти параметры определяют значения, которые соответствуют каждой из 17 позиций корректора подачи «F». Должны быть написаны все 17 значений, даже если изменяется только одно из них. Значения 16-ой и 17-ой позиций должны быть равны. Если эта инструкция не записывается, в УЧПУ принимаются по умолчанию значения, приведённые в таблице 5.9.

Таблица 5.9 – Соответствие позиций корректора «F» значениям коррекции для УЧПУ NC-301 и NC-302

Позиция корректора «F»	Значение коррекции
1	0.000
2	0.100
3	0.200
4	0.300
5	0.400
6	0.500
7	0.600
8	0.700
9	0.800
10	0.900
11	1.000
12	1.100
13	1.200
14	1.300
15	1.400
16	1.500
17	1.500

5.10.3.2. Инструкция FRO для УЧПУ серии NC-XXX

Семантика:

FRO=feed_1, feed_2, ..., feed_12 .

Формат записи:

FRO=real, real, ..., real ,

где:

feed_1, feed_2, ..., feed_12 - эти параметры определяют значения, которые соответствуют каждой из 12 позиций корректора подачи «F». Должны быть написаны все 12 значений, даже если изменяется только одно из них. Значения 11-й и 12-й позиций должны быть равны. Если эта инструкция не записывается, в УЧПУ принимаются по умолчанию значения, приведённые в таблице 5.10.

Таблица 5.10 – Соответствие позиций корректора «F» значениям коррекции

Позиция корректора «F»	Значение коррекции
1	0.000
2	0.125
3	0.250
4	0.375
5	0.500
6	0.625
7	0.750
8	0.875
9	1.000
10	1.125
11	1.250
12	1.250

5.10.4. Инструкция SSO

Инструкция **SSO** предназначена для объявления значений корректора скорости вращения шпинделя «S».

Семантика:

SSO=*speed_1, speed_2, ..., speed_12* .

Формат записи:

SSO=*real, real, ..., real* ,

где:

speed_1, speed_2, ..., speed_12 - эти параметры определяют значения, которые соответствуют каждой из 12 позиций корректора скорости шпинделя «S». Должны быть записаны все 12 значений, даже если изменяется только одно из них.

Для УЧПУ **NC-301** и **NC-302** значения от 8-ой до 12-ой позиций должны быть равны друг другу. Для других УЧПУ серии **NC-XXX** значения только 11-ой и 12-ой позиций должны быть равны друг другу. Если эта инструкция не записывается в УЧПУ, то для УЧПУ **NC-301** и **NC-302** значения по умолчанию приведены в таблице 5.11, а для других УЧПУ серии **NC-XXX** значения по умолчанию приведены в таблице 5.12.

Таблица 5.11 - Соответствие позиций корректора «S» значениям коррекции для УЧПУ NC-301 и NC-302

Позиция корректора «S»	Значение коррекции
1	0.50
2	0.60
3	0.70
4	0.80
5	0.90
6	1.00
7	1.10
8	1.20
9	1.20
10	1.20
11	1.20
12	1.20

Таблица 5.12 - Соответствие позиций корректора «S» значениям коррекции

Позиция корректора «S»	Значение коррекции
1	0.75
2	0.80
3	0.85
4	0.90
5	0.95
6	1.00
7	1.05
8	1.10
9	1.15
10	1.20
11	1.25
12	1.25

5.10.5. Инструкция FMO

Инструкция **FMO** предназначена для объявления значений корректора ручной подачи «JOG».

Семантика:

FMO=manual_1, manual_2, ..., manual_12 .

Формат записи

FMO=real, real, ..., real ,

где:

manual_1, manual_2, ..., manual_12 - эти параметры определяют значения, которые соответствуют каждой из 12 позиций корректора ручной подачи «JOG». Должны быть записаны все 12 значений, даже если изменяется только одно из них.

Значения 11-й и 12-й позиций должны быть равны друг другу.

Если эта инструкция не записывается в УЧПУ, то для УЧПУ **NC-301** и **NC-302** значения по умолчанию приведены в таблице 5.13, а для других УЧПУ серии **NC-XXX** значения по умолчанию приведены в таблице 5.14.

Таблица 5.13 - Соответствие позиций корректора «JOG» значениям коррекции для УЧПУ NC-301 и NC-302

Позиция корректора JOG	Значение коррекции
1	0.00
2	0.10
3	0.20
4	0.30
5	0.40
6	0.50
7	0.60
8	0.70
9	0.80
10	0.90
11	1.00
12	1.00

Таблица 5.14 - Соответствие позиций корректора «JOG» значениям коррекции

Позиция корректора JOG	Значение коррекции
1	-1.00
2	-0.50
3	-0.20
4	-0.05
5	-0.01
6	0.00
7	0.01
8	0.05
9	0.20
10	0.50
11	1.00
12	1.00

5.11. Пример файла PGCFIL для УЧПУ NC-110

```

NEW
*1
*2
SIM=E,,60,,,
SIM=p,,100,,,
*3
*4
PR0=1
ASS=US0,1
NPL=20,20
PRF=255,255
NDD=MP1
FIL=FILE0R/MP0,FILCOR/MP0,,,
PR0=2
ASS=US0,1
NPL=15,15
PRF=255,255
NDD=MP1
FIL=FILE0R/MP0,FILCOR/MP0,,,
*5
PR0=1
NIP=1

```

TOF=6
NAM=Z
NPD=Z, X
PRO=2
NIP=3
TOF=6
NAM=Z
NPD=Z, Z
*6
PRO=1
MAS=XZ
PRO=2
MAS=Z

6. ХАРАКТЕРИЗАЦИЯ ЛОГИКИ. ФАЙЛ **IOCFIL**

Файл **IOCFIL** характеризует параметры ПЛ, позволяющие персонализировать УЧПУ для конкретного применения. ПЛ осуществляет связь между вспомогательными механизмами и осями станка с **SW**. Файл **IOCFIL** состоит из четырёх секций.

Создать или открыть на редактирование уже существующий файл **IOCFIL** можно в режиме «**COMMAND**» командой **EDI,IOCFIL/MP0**, а также используя меню режима «**COMMAND**»: **F2 (MODIF) -> F1 (EDI) -> F7 (MP0) ->** установить в списке файлов курсор на имя файла **IOCFIL** или набрать в строке редактирования имя **IOCFIL** и завершить ввод команды по клавише **ENTER**.

ПРИМЕЧАНИЯ:

1. Имя файла **IOCFIL** в конкретном УЧПУ необходимо уточнить в секции 2 файла **FCRSYS/MP0**.
2. Порядок набора команды в меню режима «**COMMAND**» здесь записан от верхнего уровня меню, на который можно выйти, нажимая несколько раз клавишу **ESC**.

6.1. Секция 1

Секция 1 предназначена для определения модулей дискретных входов/выходов, используемых в ПЛ, параметров синхронизации выполнения ПЛ и объявления количества символьных имён, присваиваемых сигналам ПЛ. Секция состоит из следующих инструкций: **ALM**, **INx**, **OUn**, **CLO**, **SPL**. Инструкции этой секции являются общими для всех процессов.

6.1.1. Инструкция **ALM**

Инструкция **ALM** определяет состояние **SW** в отношении к ПЛ, устанавливаемое при включении УЧПУ.

Семантика:

ALM=address .

Формат записи:

ALM=hexadecimal (4 цифры) ,

где:

address - любое число в заданном формате, (обычно **address=0**).

SW в отношении к ПЛ может быть в двух режимах:

1. «разработка/отладка ПЛ»
 2. «работа ПЛ»
1. Режим «разработка/отладка ПЛ» устанавливается комментированием инструкции **ALM** с последующим перезапуском УЧПУ.

Данный режим должен быть установлен разработчиком ПЛ постоянно на весь период разработки и отладки ПЛ или только на время исправления ошибок, выявленных в режиме «**работа ПЛ**».

В этом режиме **SW** при включении УЧПУ не выполняет загрузку ПЛ и для начала работы на станке необходимо выполнить процедуру компиляции ПЛ (см. «Руководство программирования интерфейса **PLC**»). Процедура компиляции должна быть выполнена разработчиком ПЛ после любого изменения текста ПЛ.

2. Режим «**работа ПЛ**» устанавливается записью инструкции **ALM** без знака комментария.

Данный режим устанавливается разработчиком ПЛ сразу после завершения периода разработки и отладки ПЛ.

В этом режиме **SW** при включении УЧПУ автоматически выполняет загрузку ПЛ и разрешает начало работы на станке и не выполняет запросы оператора на компиляцию ПЛ.

Пример

Режим «**разработка/отладка ПЛ**»:

;ALM=0 – при включении УЧПУ загрузка ПЛ не выполняется;

Режим «**работа ПЛ**»:

ALM=0 – при включении УЧПУ загрузка и работа ПЛ выполняется автоматически.

6.1.2. Инструкция **INX**

Инструкция **INx** определяет входные разъемы модулей **I/O** и **SSB-I/O**, которые должны быть доступны из ПЛ.

Семантика:

INx=XX, XX, XX, XX, XX, XX, XX, XX .

Формат записи:

INx=word, word, word, word, word, word, word, word ,

где:

x – цифровой код, который идентифицирует номер инструкции
x = (0÷3);

XX – определяет номер разъема в пакете «**A**», предназначенный для входных сигналов модулей **I/O** и **SSB-I/O**: 0-3, 8-11, 16-19 24-27. Для каждого типа УЧПУ с учетом его конфигурации номера разъемов в пакете «**A**», соответствующие сигналам входа, указаны в документе «Руководство по эксплуатации».

Пример

Для двух модулей **I/O**, присутствующих в УЧПУ **NC-110**:

IN0=0,1,2,3,,,

Для модуля **I/O** УЧПУ **NC-210**, **NC220** и **NC-230**:

IN0=0,1,,,,,

6.1.3. Инструкция OUX

Инструкция **OUX** определяет выходные разъемы модулей **I/O** и **SSB-I/O**, которые должны быть доступны из ПЛ.

Семантика:

OUX=XX,XX,XX,XX,XX,XX,XX,XX .

Формат записи:

OUX=word,word,word,word,word,word,word,word ,

где:

X - цифровой код, который идентифицирует номер инструкции
x=(0÷3);

XX - определяет номер разъема в пакете «А», предназначенный для выходных сигналов модулей **I/O** и **SSB-I/O**: 4-7, 12-15, 20-23, 28-31. Для каждого типа УЧПУ с учетом его конфигурации номера разъемов в пакете «А», соответствующие сигналам выхода, указаны в документе «Руководство по эксплуатации».

Пример

Для двух модулей **I/O**, присутствующих в УЧПУ **NC-110**:

OU0=4,5,,,,,

Для модуля **I/O** УЧПУ **NC-210**, **NC220** и **NC-230**:

OU0=4,5,,,,,

6.1.4. Инструкция CLO

Инструкция **CLO** объявляет время опроса программы логики.

Семантика:

CLO=tick_logic,time_slow .

Формат записи:

CLO=word,word ,

где:

tick_logic - тик логики; это временной интервал между двумя последовательными выполнениями быстрой и медленной части ПЛ; он должен быть кратно больше **CPU_tick**, записанного в инструкции **TIM** файла **AXCFIL**; значение выражается в мс;

time_slow - время, затрачиваемое на выполнение цикла медленной логики внутри тика логики; должно быть меньше или равно 50% значения, установленного в параметре **tick_logic**, но не меньше величины, установленной в инструкции **TIM** файла **AXCFIL**; значение выражается в мс.

Период обчёта таймеров ПЛ равен значению, установленному в параметре **tick_logic**.

6.1.5. Инструкция SPL

Инструкция **SPL** предназначена для определения количества символьных имён, назначаемых для сигналов ПЛ и имеет следующий вид:

Семантика:

SPL=num_name .

Формат:

SPL=word ,

где:

num_name - максимальное число символьных имён для сигналов ПЛ равно 5000, при отсутствии инструкции **SPL** по умолчанию их количество равно 1; каждое символьное имя занимает 13 байтов ОЗУ. Правила записи символьных имён в ПЛ рассмотрены в «Руководстве программирования интерфейса PLC».

ВНИМАНИЕ!

1. Не устанавливайте инструкцию **SPL**, если символьные имена в ПЛ не используются.
2. После отладки ПЛ, имеющей символьные имена, инструкцию **SPL** можно закомментировать для возможности увеличения количества тех параметров в файлах характеристики, которое было уменьшено в связи с использованием символьных имён в ПЛ. например:
 - количество точек в инструкции **INx** файла **AXCFIL**;
 - количество переменных в инструкциях **SIM**, **PRF**, **NPL**, **STR**, **SER**, **CHN**, **SCR** файла **PGCFIL**.

Пример

Запись инструкций в секции 1:

```
*1
ALM=0
IN0=0,1,2,3,,,,
OU0=4,5,,,,,
CLO=10,2
SPL=400
```

6.2. Секция 2

Секция 2 используется для объявления характеристик и режимов исполнения вспомогательных функций. Секция записывается для каждого объявляемого процесса и содержит следующие инструкции: **PRO**, **Mxx**, **GPS**.

6.2.1. Инструкция PRO

Инструкция **PRO** предназначена для объявления номера текущего характеризуемого процесса.

Семантика:

PRO=proc_num .

Формат записи:

PRO=integer ,

где:

proc_num - объявляется номер процесса, являющийся текущим при характеристизации (от 1 до 5). Номер процесса не должен превышать число процессов, объявленных в инструкции **NBP**. Инструкция **PRO** должна быть записана для каждого объявляемого процесса.

6.2.2. Инструкция Mxx

Инструкция **Mxx** объявляет код **M** функций и их характеристики.

Семантика:

Mxx=m_type,m_request,class .

Формат записи:

**Mxx=hexadecimal (2 цифры),hexadecimal (2 цифры),
hexadecimal (2 цифры) ,**

где:

xx - код функции **M** (от 0 до 99);

m_type - 16-теричный код, который идентифицирует тип объявляемой функции **М**; байт устанавливается в соответствии с данными, приведенными в таблице 6.1. Значение байта представляет собой бит или совокупность битов, равных «1» и имеющих разное назначение;

Таблица 6.1 – Код типа **М**-функции

16-теричный код	Бит	Назначение
0001	0	Функция выполняется до перемещения
0002	1	Функция выполняется после перемещения
0004	2	Функция выполняется в СТОПе
0008	3	Функция не визуализируется
0010	4	Функция немедленного действия
0020	5	Резерв
0040	6	Модальная функция
0080	7	Функция, визуализируемая после сброса

m_request - 16-теричный код, который идентифицирует режим исполнения объявляемой функции **М**; байт устанавливается в соответствии с данными, приведенными в таблице 6.2. Значение байта представляет собой бит или совокупность битов, равных «1» и имеющих разное назначение;

Таблица 6.2 – Код режима исполнения **М**-функции

16-теричный код	Бит	Назначение
0001	0	Резерв
0002	1	Останов в конце кадра при подтверждении
0004	2	функция с блокировкой расчета
0008	3	Останов в конце кадра
0010	4	функция запроса на коррекцию инструмента
0020	5	Запрос сброса в конце исполнения
0040	6	Резерв
0080	7	Конец обработки детали

class - 16-теричный код, который идентифицирует 16 классов визуализации (от 0 до F) и 16 классов запомненного поиска (от 0 до F) объявляемой функции **М**; байт устанавливается в соответствии с данными, приведенными в таблице 6.3. Значение байта состоит из двух частей. Класс визуализации - правая 16-теричная цифра байта (от 0 до F), определяется его младшими 4-мя битами (от 0 до 3). Класс поиска с запоминанием - левая 16-теричная цифра байта (от 0 до F), определяется его старшими 4-мя битами (от 4 до 7).

Таблица 6.3 – Код класса визуализации и поиска **М**-функции

Бит	Назначение
От 0 до 3	16-теричный код класса визуализации - определяет позицию М-функции в поле индикации М-функций на видеостранице #1
От 4 до 7	16-теричный код класса поиска с запоминанием

При объявлении параметров инструкции **Мхх** руководствуйтесь следующей информацией:

- для функций **М** с запросом смены коррекции и с запросом сброса после исполнения должен быть установлен бит блокировки расчётов;
- класс **0** запомненного поиска и класс **0** визуализации означает - никакого запоминания и никакой визуализации;
- из функций **М** одного класса визуализации визуализируется только последняя из запрограммированных в кадре функций **М**;
- из функций **М** одного класса запомненного поиска отрабатывается последняя, записанная в кадре или найденная командой **RCM**.

По завершении запомненного поиска функции **М** отрабатываются следующим образом:

- первыми отрабатываются функции, выполняемые до перемещения, а затем - функции, выполняемые после перемещения;
- первыми отрабатываются функции, которые объявляются с низшим классом поиска с запоминанием.

Функции **М** немедленного действия имеют следующие характеристики:

- они могут быть запрограммированы при непрерывном режиме движения (**G27, G28**);
- они могут быть запрограммированы только в тех кадрах, где задаются перемещения по непрерывным координатным осям;
- код **BCD** запрограммированной в кадре функции **М** передаётся в пакет «**К**» (**W03K1**) одновременно с началом движения и сохраняется все время активизации непрерывного режима движения. По окончании непрерывного режима код сбрасывается.

Функции **М** с объявлением «останов в конце кадра при подтверждении», выполняют останов при установке команды: **US0=1**. При **US0=0** эти функции не выполняют останов в конце кадра. Обычно такой признак устанавливают для функции **M01 (M01=02, 02, 10)**.

Типичной функцией **М** с объявлением «останов в конце кадра» является функция **M00 (M00=02, 08, 10)**.

Типичной функцией **М** с объявлением «запрос сброса в конце исполнения» является функция **M30 (M30=02, 24, 10)**.

Функции **М** с кодом режима исполнения 80 используются только в задаче **ARMD** для установки переменной **PartFinish**, рассмотренной в руководстве по эксплуатации **ARMD**.

Пример

M03=45, 00, 11 - Функция **M03** выполняется до перемещения, отрабатывается в состоянии **СТОП**, функция модальная, класс визуализации и класс запомненного поиска равен 1.

6.2.3. Инструкция GPS

Инструкция **GPS** используется для объявления типа отработки вспомогательных функций.

Семантика:

GPS=manage_code .

Формат записи:

GPS=hexadecimal (2 цифры) ,

где:

manage_code - код отработки вспомогательных функций **S**, **T**, **M**, который означает следующее:

GPS=0 - последовательная отработка;

GPS=1 - параллельная отработка.

Под параллельной обработкой понимается одновременная отработка функций **S**, **T** и первой функции **M**, запрограммированных в одном кадре и выполняемых до или после перемещения осей.

При последовательном типе обработки функции выполняются в следующем порядке: сначала **S**, **T**, затем **M**, выполняемые до перемещения, после чего - функция индексной оси. Функция индексной оси обрабатывается всегда после других вспомогательных функций.

Если инструкция **GPS** отсутствует, по умолчанию принимается последовательный тип обработки.

6.3. Секция 3

Секция 3 используется для объявления управляемого оборудования, которое должно обрабатываться в ПЛ. Секция 3 состоит из следующих инструкций: **PRO**, **ASM**, **TAX**, **ASx**, **UCDA**, **ADV**, **CWD**, **ADC**, **DAC**. Инструкции этой секции должны быть записаны для каждого объявляемого процесса.

6.3.1. Инструкция PRO

Инструкция **PRO** предназначена для объявления номера текущего характеризуемого процесса.

Семантика:

PRO=proc_num .

Формат записи:

PRO=integer ,

где:

proc_num - объявляется номер процесса, являющийся текущим при характеристике (от 1 до 5). Номер процесса не должен превышать число процессов, объявленных в инструкции **NBP**. Инструкция **PRO** должна быть записана для каждого объявляемого процесса.

6.3.2. Инструкция ASM

Инструкция **ASM** определяет наименование оси шпинделя текущего процесса.

Семантика:

ASM=spindle_name .

Формат записи:

ASM=ASCII string ,

где:

spindle_name - имя оси шпинделя.

Пример

ASM=S

6.3.3. Инструкция TAX

Инструкция **TAX** определяет характеристики оси «от точки к точке».

Семантика:

TAX=axis_name,axis_type,phase_disp, axis_pos,limit_1,limit_2 .

Формат записи:

TAX=ASCII string,hexadecimal (2 цифры),real,real,real,real ,

где:

X - номер оси «от точки к точке» (от 1 до 8). Значение номера оси «от точки к точке» используется в ПЛ для её активизации;

axis_name - наименование оси, объявленной в инструкции **INX** файла **AXCFIL**;

axis_type - 16-теричный код, который идентифицирует тип оси «от точки к точке»; соответствие между типом оси и его идентифицирующим 16-теричным кодом представлено в таблице 6.4;

Таблица 6.4 – Назначение битов в коде **axis_type**

Бит	Назначение
0	0 - для оси «от точки к точке» без ЦАП/ЦИП; 1 - для оси «от точки к точке» с ЦАП/ЦИП
1	0 - для линейных осей «от точки к точке»; 1 - для вращательных осей «от точки к точке»
3, 2	00 - для неабсолютных осей с запросом выхода в позицию абсолютного микроула; 10 - для абсолютных осей с резольвером 11 - для абсолютных осей с энкодером;
с 4 по 7	Незначимые

phase_disp - определяет смещение фаз между электрическим и механическим нулем оси «от точки к точке»; выражается в позициях; это значение должно быть меньше, чем величина механического шага, установленного в инструкции **PAS** файла **AXCFIL**;

axis_pos - определяет количество позиций оси «от точки к точке».

limit_1 и **limit_2** устанавливаются по-разному для разных типов осей «от точки к точке»:

1) для осей «от точки к точке» без ЦАП/ЦИП:

limit_1 - представляет порог первого замедления оси «от точки к точке», в позиции которого в ПЛ передается команда первого замедления; значение выражается в количестве позиций;

limit_2 - представляет порог второго замедления оси «от точки к точке», в позиции которого в ПЛ передается команда второго замедления; значение выражается в количестве позиций;

2) для абсолютных линейных осей с энкодерным датчиком и ЦАП/ЦИП:

limit_1 - параметр не устанавливается;

limit_2 - этот параметр определяет количество позиций смещения абсолютной оси после её отключения и последующего подключения к управлению;

axis_pos

limit_2 = -----
2 * n° фактических позиций оси «от точки к точке»

Особенности управления:

- Включение абсолютной оси необходимо выполнять в той её позиции, с которой начинается отсчет всех её позиций.
- После включения такой оси для неё сразу можно программировать требуемую позицию.
- Если абсолютная ось объявлена с резольвером, то она представляет линейную ось, вал датчика которой при перемещении в последнюю позицию оси максимально может выполнить вращение от 0 до 359.999 градусов. При выполнении перемещения в обратном направлении, из последней позиции оси в начальную её

позицию, вал датчика выполнит вращение в обратном направлении от 359.999 до 0 градусов.

- Если абсолютная ось объявлена с энкодером, то её перемещение в запрограммированную позицию автоматически выполняется по кратчайшему пути.

3) для осей, отличающихся от тех, которые описаны в перечислении 1) - 2):

limit_1 - параметр не устанавливается;

limit_2 - параметр не устанавливается.

Абсолютной осью «от точки к точке» называется ось с передаточным отношением между датчиком и оборотом оси, равным 1:1. Для этой оси не требуется выход в позицию абсолютного микроула.

6.3.4. Инструкция ASX

Инструкция **ASX** объявляет характеристики индексной оси.

Семантика:

ASx=axis_name,mode,axis_type,pos_num .

Формат записи:

ASx=ASCII string,hexadecimal (2 цифры),hexadecimal (2 цифры),word
,

где:

x - номер индексной оси (**x=1÷3**);

axis_name - наименование индексной оси, используемое при программировании;

mode - определяет режим управления движением индексной оси; параметр может принимать следующие значения:

0 - абсолютный режим, при котором задание для перемещения представляет собой координаты позиции, в которую должна будет позиционирована ось;

1 - инкрементальный режим, при котором задание для перемещения представляет собой количество позиций, которое ось должна выполнить;

axis_type - определяет тип индексной оси:

0 - вращательная ось;

1 - линейная ось;

pos_num - определяет количество позиций индексной оси.

6.3.5. Инструкция UCDA

Инструкция **UCDA** определяет наименование координатных осей или осей «от точки к точке», управление которыми выполняется от ПЛ.

Семантика:

UCDA=axis_name .

Формат записи:

UCDA=ASCII string ,

где:

axis_name - определяет наименование координатных осей или осей «от точки к точке», для которых аналоговое напряжение на ЦАП задаётся от ПЛ; может быть определено максимально три оси.

6.3.6. Инструкция ADV

Инструкция **ADV** определяет характеристики управления штурвалами в зависимости от их количества (1 или 2 штурвала) и способа подключения каждого штурвала.

6.3.6.1. Управление штурвалом/штурвалами

Подключить штурвал к УЧПУ можно двумя способами:

- 1) через канал электронного штурвала;
- 2) через канал энкодера.

В системе можно сформировать только один штурвал, управляемый от **SW** или макс. 2 штурвала, управляемых от ПЛ.

Способ управления штурвалом от **SW** называется «**внутреннее управление штурвалом**». Данный способ управления штурвалом не требует его конфигурирования в файлах характеристики и в ПЛ.

Способ управления одним или двумя штурвалами от ПЛ называется «**внешнее управление штурвалом**». Данный способ управления штурвалами требует их конфигурирования в файлах характеристики и в ПЛ.

Управление штурвалом, подключённым через канал энкодера, может быть выполнено только способом внешнего управления.

ПРИМЕЧАНИЯ:

1. Разъёмы для подключения штурвала к УЧПУ указаны в документе «Руководство по эксплуатации».
2. Разрешающая способность штурвала - 100 импульсов на оборот (имп/об) .

1. Внутреннее управление штурвалом

1.1. Внутреннее управление штурвалом, подключённым через канал электронного штурвала, не требует характеристики.

1.2 Внутреннее управление штурвалом выполняется **SW** и активируется командой **VOL=1** в режиме «**БЕЗРАЗМЕРНЫЕ РУЧНЫЕ ПЕРЕМЕЩЕНИЯ**» или «**ФИКСИРОВАННЫЕ РУЧНЫЕ ПЕРЕМЕЩЕНИЯ**».

Шкала деления выбирается установкой одного из режимов работы:

- режим «**БЕЗРАЗМЕРНЫЕ РУЧНЫЕ ПЕРЕМЕЩЕНИЯ**»: 1 оборот штурвала – 1 мм;
- режим «**ФИКСИРОВАННЫЕ РУЧНЫЕ ПЕРЕМЕЩЕНИЯ**»: 1 оборот штурвала – 0.1 мм.

Движение оси происходит при выполнении следующих требований:

- 1) ось должна быть включена, и переменная **UAS=0**;
- 2) ось должна находиться в состоянии покоя (сигнал PLC **STAND-BY=«1»**);
- 3) ось должна быть выбрана, т. е. выделена зелёной полосой -курсором в поле «**ФАКТ**» на видеостраницах **#1, #7**;
- 4) **ПРОЦЕСС** должен быть в состоянии «**IDLE**»;
- 5) интерфейсный сигнал PLC: **COMU** должен быть высокого логического уровня (**COMU=«1»**);
- 6) интерфейсный сигнал PLC: **FOLD** должен быть низкого логического уровня (**FOLD=«0»**).
- 7) выбранная ось должна быть определена в инструкции **MAS** (секция 6 файла **PGCFIL**) и не должна являться:
 - виртуальной осью;
 - осью «от точки к точке»;
 - осью шпинделя.

Когда штурвал активен (**VOL=1**), выбранная ось, управляемая от штурвала, движется, если:

- 1) выбранная ось определена в таблице осей процесса;
- 2) не является:
 - виртуальной осью;
 - осью «от точки к точке»;
 - шпинделем.

Примечание - Если все условия выполнены, то в видеостраницах **#1, #6, #7** выводится сообщение «Штурвал готов к работе»

2. Внешнее управление двумя штурвалами

2.1 Внешнее управление двумя штурвалами требует характеристики. Инструкция **ADV** записывается в соответствии с п. 6.3.6.2.

В этом случае разрешение работы от штурвалов выполняется запросами от ПЛ (Версии **SW 2.33P(РИБ), 3.33P(РИБ)** и последующие, а также все версии 4.XXR).

Внешнее управление штурвалами выполняется **SW** и активизируется ПЛ в любом режиме работы. Запрет штурвалов в каком-либо режиме работы выполняется от ПЛ. Управление штурвалами выполняется по двум независимым каналам. Распределение штурвалов между каналами управления выполняется в инструкции **ADV**.

Движение оси от штурвала выполняется при следующих условиях:

- 1) ось должна быть включена, и переменная **UAS=0**;

- 2) оси **ПРОЦЕССА** должны находиться в состоянии покоя (сигнал **STAND-BY=«1»**);
- 3) состояние **ПРОЦЕССА** должно быть **«IDLE»**;
- 4) интерфейсный сигнал PLC: **COMU** должен быть высокого логического уровня (**COMU=«1»**);
- 5) интерфейсный сигнал PLC: **FOLD** должен быть низкого логического уровня (**FOLD=«0»**).
- 6) выбранная ось должна быть определена в инструкции **MAS** (секция 6 файла **PGCFIL**) и не должна являться:
 - виртуальной осью;
 - осью «от точки к точке»;
 - осью шпинделя.
- 7) штурвалу, до его активизирования на своем канале управления, должна быть назначена шкала в байте **W15N2** для этого канала;
- 8) оси для движения от штурвалов должны быть выбраны в соответствующих каналах управления (слова **W15N0** и **W15N1**);
- 9) штурвалы должны быть активизированы сигналами **U15N24** или **U15N25** после выполнения условий п.1) – п.7).
- 10) если все условия выполнены, то в видеостраницах **#1**, **#6**, **#7** выводится сообщение **«Штурвал готов к работе: Имя оси в канале1/Имя оси в канале2»**.

6.3.6.2. Управление двумя штурвалами

В случае внешнего управления штурвалами инструкция **ADV** определяет характеристики для управления каждым штурвалом в зависимости от способа его подключения.

Семантика:

ADV=axis_name_HW, axis_name_HW .

Формат записи:

ADV=ASCII string, ASCII string ,

где:

axis_name_HW - это значение объявляет штурвал и способ его подключения к УЧПУ на первом канале управления;

axis_name_HW - это значение объявляет штурвал и способ его подключения к УЧПУ на втором канале управления.

axis_name_HW может принимать следующие значения:

- 1) имя оси «штурвала», если штурвал подключён через канал датчика энкодер. Имя оси «штурвала» должно быть определено в файле **AXCFIL** в соответствии с примером.
- 2) цифра **«0»**, если штурвал подключён к штатному разъёму канала электронного штурвала в УЧПУ **NC-200**, **NC-201**, **NC-201M**, **NC-202**, **NC-210**,

NC-220, NC-230, либо штурвал установлен на станочном пульте УЧПУ **NC-110** или **NC-310**;

Пример

Характеризация двух штурвалов:

- на первом канале управления определяется штурвал, подключённый к четвертому каналу энкодера. Для подключения штурвала к каналу энкодера в файле **AXCFIL** конфигурируется ось «штурвала» с именем «**P**»;

- на втором канале управления определяется штурвал, подключённый к штатному каналу электронного штурвала:

```
;Файл AXCFIL
```

```
*1
```

```
.....
```

```
PRO=1
```

```
IN1=1,XZ,S,2,64
```

```
CAS=1,XZSP,2
```

```
*2
```

```
PRO=1
```

```
NAS=X
```

```
.....
```

```
NAS=Z
```

```
.....
```

```
NAS=S
```

```
.....
```

```
NAS=P
```

```
TPA=1,
```

```
NTC=4,
```

```
PAS=400,1
```

```
POS=,
```

```
GAS=,
```

```
SRV=,,
```

(**FBF=,,,**) - всегда отключайте аппаратный контроль обрыва сигналов датчика штурвала, если штурвал не имеет инверсных сигналов синусов и косинусов.

```
.....
```

```
;Файл IOCFIL:
```

```
*1
```

```
.....
```

```
*2
```

```
.....
```

```
*3
```

```
.....
```

```
ADV=P,0
```

```
*4
```

Особенности определения штурвалов в файлах характеристики:

- для штурвала, который подключён через канал электронного штурвала, в инструкции **ADV** необходимо записать цифру «**0**»;
- для штурвала, который подключён через канал энкодера в модуле **ECDA**, в инструкции **ADV** необходимо записать имя оси «штурвала», определённое в файле **AXCFIL**;

- при необходимости одновременного движения двух осей от одного штурвала:
 - в инструкции **ADV** записываются оба параметра цифрой «0», если штурвал подключён к УЧПУ через канал электронного штурвала;
 - в инструкции **ADV** записываются оба параметра именем оси «штурвала», если штурвал подключён к УЧПУ через канал энкодера.

6.3.7. Инструкция CWD

Инструкция **CWD** отключает обработку задач из ПЛ.

Семантика:

CWD=ctrl_word .

Формат записи:

CWD=hexadecimal (4 цифры) ,

где:

ctrl_word - 16-тиричный код, отключающий управление в ПЛ для неиспользуемых задач. Применяется для того, чтобы сделать работу УЧПУ более быстрой.

Ниже приводится соответствие между режимом и битом **CWD**, его определяющим. Установкой бита в «1» можно исключить из управления задачи, не используемые в конкретном применении УЧПУ:

- 1) 0001H - управление шпинделем;
- 2) 0002H - управление штурвалом;
- 3) 0004H - управление выводом на экран сообщений файла **RUMES5** от ПЛ (**W17K3**, разъём 21K, разъём 22K);
- 4) 0008H - управление консолью (15-ый и 16-ый разъёмы пакета «К»);
- 5) 0010H - управление осью «от точки к точке» на первом канале (**W20K0**);
- 6) 0020H - управление осью «от точки к точке» на втором канале (**W20K1**);
- 7) 0040H - управление балансировкой осей (**W9K1**);
- 8) 0080H - управление командами от логики (**W17K0**, **U10K23**);
- 9) 0100H - управление ЦАП (**W12K0**, **W13K2**, **W13K3**);
- 10) 0200H - свободно;
- 11) 0400H - управление активизацией осей (**W10K1**, **W17K1**);
- 12) 0800H - управление дезактивизацией ограничения перемещений (**W12K2**, **W12K3**); переключение первой и второй оперативной зоны сигналом **U10N1** в версии с расширением **PIB**;
- 13) 1000H - управление активизацией и дезактивизацией номера инструмента и корректора (**U10K16**, **U10K21**);

- 14) 2000H - управление сигналами (**U10K0, I0K2, U10K5, U10K7**);
- 15) 4000H - управление перемещением осей от ПЛ (**W12K1, I0K24**);
- 16) 8000H - свободно.

Примечание - Назначение сигналов, байтов и разъемов пакетов «К» и «N», записанных в круглых скобках, указано в «Руководстве по программированию интерфейса PLC».

6.3.8. Инструкция ADC

Инструкция **ADC** предназначена для определения номеров каналов АЦП (максимум 6 номеров), доступных для чтения.

Семантика:

ADC=conv1, conv2, conv3, conv4, conv5, conv6, conv7, conv8 .

Формат:

ADC=integer, integer, integer, integer, integer, integer ,

где:

conv1-8 - номера каналов АЦП (1-8) в УЧПУ.

Номера каналов АЦП, определённые в инструкции **ADC**, могут быть использованы в команде **ADR**. Команда **ADR** может быть записана в кадре управляющей программы или выполнена в режиме «MDI» («РУЧНОЙ ВВОД КАДРА») по кнопке «ПУСК». Значение напряжения, прочитанное из канала АЦП, сохраняется в переменных типа **real** (вещественные).

Формат:

(ADR, ADC_num, variable) ,

где:

ADC_num - номер канала АЦП (1-8) должен соответствовать номеру канала АЦП, объявленному в инструкции **ADC**;

variable - **Е-параметр** с номером, соответствующим **Е-параметрам** типа **real** (вещественные) **E25-E29**.

Примечание - Для передачи данных из одного процесса в другой используйте переменные **SYVAR** того же типа, что и **Е-параметр**, как указано в «Руководстве программиста».

6.3.9. Инструкция DAC

Инструкция **DAC** предназначена для определения номеров каналов ЦАП (ЦИП), доступных для записи в них напряжения.

Семантика:

DAC=conv1, conv2, conv3, conv4, conv5, conv6, conv7, conv8 .

Формат:

DAC=integer, integer, integer, integer, integer, integer ,

где:

conv1-8 - номера канала преобразователя (1-16).

Пример

DAC=1, 2, 8, 14, 16, , ,

Если преобразователь 16 разрядов, то к номеру преобразователя должен быть добавлен код (-200), например:

Пример

DAC=-201, -202, -208, -214, -216, , ,

conv1-8, определённые в инструкции **DAC**, могут быть использованы в команде **DAW**. Команда **DAW** может быть записана в кадре управляющей программы или выполнена в режиме «**MDI**» («РУЧНОЙ ВВОД КАДРА») по кнопке «ПУСК».

Формат:

(DAW, conv_num, value) ,

где:

conv_num - номер канала ЦАП (1-16) должен соответствовать номеру канала ЦАП, объявленному при характеристике;

value - значение напряжения в пределах ±10В; может быть задано в явном виде или через **Е-параметр** с номером, соответствующим **Е-параметрам** типа **real** (вещественные) **E25-E29**.

ВНИМАНИЕ !

До программирования кода **DAW** для оси эта ось должна быть выключена в ПЛ (разомкнут контроль оси по датчику обратной связи и отключена подача напряжения от ПЛ).

6.4. Секция 4

Секция 4 определяет переменные пакета «**T**» и содержит только инструкцию **TXX**.

6.4.1. Инструкция Txx

Инструкция **Txx** устанавливает значения переменным пакета «Т».

Семантика:

Txx=value

Формат записи:

Txx = hexadecimal (2 цифры) ,

где:

xx - номер записи (от 1 до 64);

value - значение от **00** до **FF**; значения, записанные в инструкции **Txx**, могут использоваться как:

- 1) байтовая константа, которая в ПЛ используется только для чтения;
- 2) начальное значение байтовой переменной после каждого перезапуска УЧПУ, которое может быть изменено в ПЛ по каким-либо условиям.

Номер записи «**xx**» в инструкции **Txx** определяет байт пакета «Т», в который будет передано значение в соответствии со следующей схемой:

```

T01 = значение → W0T0
T02 = значение → W0T1
T03 = значение → W0T2
T04 = значение → W0T3
T05 = значение → W1T0
.....
.....
.....
T63 = значение → W15T2
T64 = значение → W15T3

```

Подробные сведения о назначении и использовании пакета «Т» приведены в документе «Программирование интерфейса PLC».

6.5. Пример файла IOSFIL

```

*1
ALM = 0
IN1 = 0, 1, 2, 3,,,
OU0 = 4, 5,,,,,
CLO = 10, 3
*2
PRO = 1
M00 = 06, 0C, 00
M01 = 02, 04, 00
M02 = 02, 24, 20

```

M03 = 45, 00, 21
 M05 = 06, 00, 21
 M06 = 02, 14, 62
 M07 = 45, 00, 44
 M09 = 06, 00, 44
 M10 = 45, 00, 77
 M11 = 06, 00, 77
 M12 = 45, 00, 77
 M13 = 45, 00, 21
 M14 = 45, 00, 21
 M19 = 45, 00, 21
 M30 = 02, 24, 00
 M60 = 02, 04, 03
 PRO = 2
 M00 = 6, C, 0
 M01 = 2, 4, 0
 M02 = 2, 24, 20
 M03 = 45, 0, 21
 M05 = 6, 0, 21
 M06 = 2, 14, 62
 M07 = 45, 0, 44
 M09 = 6, 0, 44
 M10 = 45, 0, 77
 M11 = 6, 0, 77
 M12 = 45, 0, 77
 M13 = 45, 0, 21
 M14 = 45, 0, 21
 M19 = 45, 0, 21
 M30 = 2, 24, 0
 M60 = 2, 4, 3

*3

PRO = 1
 ASM = S
 TA1 = V, E, 0, 32, .7, .3
 AS1 = B, 0, 0, 360
 UCDA = B
 CWD = 0040
 ADV = A, 0
 PRO = 2
 ASM = S
 CWD = 0040

*4

T01 = 10
 T02 = 0
 T03 = 10
 T04 = 0
 T05 = 15
 T06 = 0
 T07 = 2
 T08 = 0
 T09 = 9
 T10 = 0
 T11 = 2
 T12 = 0

7. ИНДИКАЦИЯ ОШИБОК ХАРАКТЕРИЗАЦИИ ФУНКЦИОНАЛЬНОГО УРОВНЯ

7.1. Видеокадр диагностики файлов характе- ризации функционального уровня

УЧПУ при наличии в его памяти файлов характеристики после диагностики аппаратных модулей выполняет инициализацию этих файлов.

По окончании инициализации файла **FCRSYS** (системный уровень), если она прошла без ошибок, выполняется инициализация файлов функционального уровня: **AXCFIL**, **PGCFIL**, **IOCFIL**. Результаты диагностики визуализируются в видеокадре, который представлен на рисунке 7.1.

ИНДЕКС	NC-110
ОШИБКА	
СЕКЦИЯ: X	ПРОЦЕСС: Y
ОШИБКА В СТРОКЕ: Содержание строки с ошибкой	
НОМЕР ОШИБКИ: xxx ОСЬ: n	

Рисунок 7.1 – Видеокадр диагностики файлов характеристики AXCFIL, PGCFIL, IOCFIL

Параметры видеокадра:

- **ИНДЕКС** - спецификатор типа файла, содержащего ошибку; может принимать следующие значения:

ОСИ : **AXCFIL**;
ПРОЦЕСС: **PGCFIL**;
ЛОГИКА : **IOCFIL**;

- **X** - номер секции файла, содержащей ошибку;
- **Y** - номер процесса, в котором обнаружена ошибка;
- **xxx** - код ошибки от 0 до 209; каждому коду ошибки соответствует сообщение об этой ошибке из списка сообщений, представленных в пп. 7.2.1-7.2.5;
- **n** - наименование оси в подсекции, в которой обнаружена ошибка при диагностике секции 2 файла **AXCFIL**.

7.2. Ошибки инициализации файлов характеристики функционального уровня

Ошибки, выявляемые при инициализации трёх файлов характеристики функционального уровня **AXCFIL**, **PGCFIL** и **IOCFIL**, подразделяются на четыре группы:

- 1) синтаксические ошибки и ошибки формата, общие для трёх файлов характеристики функционального уровня; имеют номера от 1 до 100;
- 2) ошибки, выявляемые во время анализа файла характеристики осей **AXCFIL**; имеют номера от 101;
- 3) ошибки, выявляемые во время анализа файла характеристики процесса **PGCFIL**; имеют номера от 101;
- 4) ошибки, выявляемые во время анализа файла характеристики логики **IOCFIL**; имеют номера от 101.

Ошибки, имеющие номер от 101, относятся к ошибкам одного из файлов характеристики. Номера ошибок, визуализируемых в видеокадре, и объяснение причин, их вызывающих, приведены в пп. 7.2.1-7.2.5.

При обнаружении таких ошибок вверху слева на экране визуализируется:

- «ОСИ», если ошибка обнаружена при инициализации файла **AXCFIL**;
- «ПРОЦЕСС», если ошибка обнаружена при инициализации файла **PGCFIL**;
- «ЛОГИКА», если ошибка обнаружена при инициализации файла **IOCFIL**.

7.2.1. Синтаксические и семантические ошибки

Ошибки семантики и формата записи представлены в таблице 7.1.

Таблица 7.1 - Ошибки семантики и формата

Номер ошибки	Объяснение	Примечание
1	Не хватает секций в файле характеристики	В файле характеристики отсутствует секция
2	Каждая строка файлов должна начинаться с символа «*», с буквы или с символа «;»	
3	Секция не определена	Файл характеристики начинается без обозначения «Секция 1» («*1»)
4	Несоответствие между номером текущей секции и ожидаемым номером секции	Например, секция *3 объявляется сразу после секции *1
5	Не хватает «CR» после номера секции	Идентификатор секции должен быть на одной из линий текста (т. е. строка должна заканчиваться CR, LF)
6	Неразрешённый оператор	Имя оператора не разрешено в данной секции или в данной конфигурации
7	Слишком длинный оператор	Максимальное количество символов в операторе равно 5
8	После идентификатора оператора должен быть знак «=»	

Продолжение таблицы 7.1

Номер ошибки	Объяснение	Примечание
9	Недопустимый символ в идентификаторе оператора	Название оператора должно быть записано заглавными или строчными буквами без пропусков, после него может быть пропуск, а затем знак «=»
10	Не хватает параметров в операторе	Количество параметров для оператора фиксировано и должно соблюдаться. Если параметр равен нулю, нужно вместо него записать «,»
11	Слишком много параметров в одном операторе	Количество параметров в каждом операторе фиксировано и должно соблюдаться
13	Ошибка в параметре «ЦЕЛОЕ ЧИСЛО»	Ошибка выдаётся в том случае, если в параметре «ЦЕЛОЕ ЧИСЛО» записана не цифра
14	Переполнение целого числа	Предельные значения «ЦЕЛОГО ЧИСЛА» от -32768 до +32767
15	Ошибка в параметре «СЛОВО»	В параметре «СЛОВО» присутствуют нецифровые знаки
16	Переполнение параметра «СЛОВО»	Предельные значения параметра «СЛОВО» заключены между 0 и 65535
17	Первым символом в идентификаторе сигнала PLC должен быть «I» или «U»	Значение должно быть записано идентификатором сигнала PLC
18	Ошибка значения номера разъёма или контакта в идентификаторе сигнала пакета PLC	Ошибка индицируется в следующих случаях: 1) если номер разъёма - для пакета «K» и «N» больше 255 - для пакета «A» больше 31 - для пакета «T» больше 15 2) если номер контакта больше 31 Например: U256K20, U200K32, I32A1, I32A32 - неправильные имена идентификаторов
19	Буква, идентифицирующая пакет сигналов PLC, должна быть: K - для пакета «K», N - для пакета «N», A - для пакета «A», T - для пакета «T»	
20	Ошибка в параметре «real»	В параметре «real» присутствуют нецифровые символы
21	Ошибка в 16-теричном параметре	В 16-теричном параметре присутствуют нецифровые символы, за исключением символов A, B, C, D, E, F
22	16-теричное число имеет более двух цифр	16-теричные числа могут состоять максимально из двух цифр
23	Параметр из символов ASCII слишком длинный	В параметре из символов ASCII может быть максимально 10 знаков, кроме знаков разделителя («,») и комментария («;»)
24	В файле слишком много секций	Определено слишком много секций в файле характеристики
25	Файл конфигурации процесса PGCFIL должен начинаться с NEW или OLD	

7.2.2. Ошибки характеристики осей - файл AXCFIL

Ошибки файла характеристики осей **AXCFIL** представлены в таблице 7.2.

Таблица 7.2 - Ошибки файла характеристики осей **AXCFIL**

Номер ошибки	Объяснение	Примечание
101	Тик слишком большой для CPU	Значение, установленное в инструкции TIM, больше 63
102	Номер CPU в инструкции INx или CAS не равен «1»	
103	Ошибочное определение номера CPU	CPU управления осями нумеруется «1». Выдается ошибка, если в инструкции CAS номер CPU не равен «1»
104	Тик управления приводом слишком большой либо равен нулю	Тик управления приводом выражается в мс, изменяется от 1 до 63. Выдается ошибка, если в инструкции CAS тик CPU не лежит в этих пределах
105	Ошибочное определение номера CPU интерполятора	CPU интерполяторов нумеруется 1. Выдается ошибка, если в инструкции INx номер CPU не 1
106	Слишком много элементов	Количество элементов интерполяции в буфере интерполятора может изменяться от 1 до 64. Выдается ошибка, когда в инструкции INx не соблюдены эти пределы
107	Тик интерполятора слишком большой или равен нулю	Тик интерполятора выражается в мс и может изменяться от 1 до 63. Выдается ошибка, если в инструкции INx не соблюдены эти пределы
108	Один интерполятор может управлять только одним шпинделем	Поле «Наименование оси шпинделя» инструкции INx должно содержать только один символ
109	Интерполятор не может быть объявлен без осей	
110	Таблица интерполяторов переполнена	В одном процессе допускается одновременное наличие не более 9 интерполяторов
111	Дублированный интерполятор	Имя интерполятора, объявленное в текущей инструкции, уже было использовано для определения предыдущего интерполятора
112	В инструкции CAS должна быть определена по крайней мере одна ось	
113	Дублированные оси в последовательности определения осей.	В инструкции CAS имеются две оси с одинаковыми именами
114	Таблица осей переполнена	Выдается ошибка в следующих случаях: 1) для NC-110, NC-310 объявлено - больше 17 осей (любых); - больше 8 непрерывных координатных осей, исключая виртуальные оси; 2) для NC-200, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230 объявлено - больше 17 осей (любых); - больше 6 непрерывных координатных осей, исключая виртуальные оси.

Продолжение таблицы 7.2

Номер ошибки	Объяснение	Примечание
115	Переполнена таблица осей одного интерполятора	В интерполяторе непрерывных скоординированных осей для NC-110, NC-200, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230, NC-310 допускается одновременное объявление не более 8 непрерывных координатных осей, включая виртуальные оси. Ошибка выдаётся, если в инструкции INx объявленное количество осей превышает 8.
116	Линейная ось используется несколькими интерполяторами	Одна ось может управляться только одним интерполятором
118	Ось объявлена в нескольких инструкциях CAS для одного процесса или для различных процессов в случае общей оси	
119	Ось объявлена, но не определена	Эта ошибка выдаётся, когда ось, объявленная в инструкции INx, не определена в инструкции CAS
120	Тик интерполятора не кратен тикку управления приводом	Выдаётся ошибка, если тик интерполятора больше и не кратен тикку управления приводом
121	Не хватает инструкции TIM в первой секции AXCFIL	
124	На CPU с тиком, равным нулю, не может быть управления приводом и интерполяторов	Если тик модуля CPU, определённый инструкцией TIM, равен нулю, это значит, что модуль не присутствует, и, следовательно, на нем не может быть ни управления приводом, ни интерполяторов
126	Управление приводом разных осей на одном и том же CPU должно иметь одинаковый тик	
127	Тик управления приводом не является кратным тикку CPU	
128	Тик одного из интерполяторов не является кратным тикку CPU	
131	Дублированная инструкция	Текущая инструкция повторена в подсекции данной NAS
132	Не хватает инструкции TRA	
133	Не хватает инструкции NTC	Инструкция NTC должна всегда быть объявленной

Продолжение таблицы 7.2

Номер ошибки	Объяснение	Примечание
134	Не хватает инструкции GAS, а имеется датчик	Инструкция должна быть определена для текущей подсекции NAS в том случае, если присутствует датчик, соединенный с данной осью, т. е. если поле «номер датчика» инструкции NTC отлично от нуля
135	Не хватает инструкции PAS, а имеется датчик	см. ошибку 134
137	Не хватает инструкции RAP	Если объявляется датчик и преобразователь (инструкция NTC), и ось - не шпиндель, то RAP должен присутствовать
138	Не хватает инструкции MAN, а имеется датчик	См. ошибку 137
139	Ручная скорость больше скорости быстрого хода	
140	Ускорение ручных перемещений больше ускорения быстрого хода	
141	Не хватает инструкции POS, а имеется датчик	Если имеется датчик, то должен объявляться допуск позиционирования POS
142	Не хватает инструкции SRV, а имеется датчик	
147	Не должна присутствовать GM0, если ось - шпиндель	
148	Должна быть определена GM1, если ось - шпиндель	
149	Ось, определённая инструкцией NAS, не объявлена	Эта ошибка выдаётся в том случае, когда определённая в NAS ось не объявлена в Секции 1
150	Если ось - не шпиндель, следует объявить только GM0	Если ось - не шпиндель, следует объявить только инструкцию GM0
151	Инструкция в секции 2 AXCFIL - без предшествующей NAS	
152	Нет инструкций GMnn, RAP, MAN, POS, если нет интерполятора	Необходимо наличие интерполятора, если есть GMnn, RAP, MAN, POS
153	Электрический шаг должен быть отличен от нуля	
154	Механический шаг должен быть отличным от нуля	
155	Шпиндель должен быть заявлен в секции 2, и в секции 1	Эта ошибка выдаётся, когда в секции 1 ось определена как ось шпинделя (инструкции INx), а в секции 2 нет оси, для которой в TRA записан код оси шпинделя.
157	Координатная ось не может быть осью «от точки к точке»	
158	Наименование коммутированной оси должно состоять только из одного знака	Это условие опрашивается для проверки совпадения наименований осей с их определением, осуществленным в секции 1
159	Не существует переключённой оси, определённой TRA	Ось, определённая как переключённая, должна быть объявлена в секции 1

Продолжение таблицы 7.2

Номер ошибки	Объяснение	Примечание
160	Ошибочный номер датчика	Номер датчика изменяется в пределах от 1 до 16
161	Ошибочный номер преобразователя (канала ЦАП)	Номер преобразователя изменяется в пределах от 1 до 16 (или от -201 до -216)
163	Направление поиска микро-нуля должно быть равно 0 или 1	
166	Наименование главной оси должно состоять только из одного символа	Это условие опрашивается для проверки совпадения с определением наименования осей, выполненным в секции 1
167	Не существует оси, определённой как главная, в ASM	Ось, определённая как главная, должна быть объявлена в секции 1
168	Несовпадение в объявлении переключаемых осей	Выявлено, что ось, объявленная переключаемой в инструкции TRA, на самом деле не является переключаемой (в TRA данной оси). Например: ось X - координатная и переключаемая на Y; тогда для X TRA=9,Y, а для Y - TRA=9,X, т. е. X переключается на Y, а Y - на X. Если не соблюдается это условие, выдаётся ошибка
169	Не хватает определения оси в секции 2	Все оси, определённые в секции 1, должны иметь соответствующую подсекцию NAS
172	Максимальное значение XX в инструкции GMnn может быть равно 99	
173	Не хватает инструкции NAS или PAS	Эта ошибка возникает, когда записывается инструкция Ennn до ввода инструкций NAS и PAS, или когда записывается инструкция PAS, и ей не предшествует соответствующая инструкция NAS, или когда записывается инструкция NM0, и ей не предшествует NAS и PAS
174	Заявление (номер) Ennn - не в возрастающем порядке	
175	В таблице значений компенсации должно присутствовать не менее 2 значений	
176	Синтаксическая ошибка в NM0	
177	Номер точки компенсации для позиции абсолютного микро-нуля не существует, т.е. в NM0 заявлена точка, не определённая под Ennn = величина	

Продолжение таблицы 7.2

Номер ошибки	Объяснение	Примечание
178	В ТРА заявлена ось с преобразователем, а номер преобразователя не заявлен в соответствующей инструкции NTC (или наоборот)	
179	В ТРА ось шпинделя заявлена как ось с датчиком и без него одновременно	Например: ТРА=30
180	В TSM один или два параметра опущены или равны 0	
181	Заявление TSM должно быть только в подсекции для оси шпинделя (ТРА=20 или ТРА=10)	
182	Инструкции GMnn, TSM в подсекции оси записаны до инструкции NTC	
183	Превышено максимальное количество процессов в инструкции NBP	
184	В инструкции PRO объявлен номер процесса, превышающий количество, установленное в инструкции NBP	В инструкции PRO объявлен номер процесса, превышающий максимальное число процессов, определённое в инструкции NBP, или который ранее не подвергался конфигурации
185	CAS и INx заявлены без предварительного объявления NBP	Не допускаются инструкции CAS и INx, если не объявлена инструкция NBP
186	Переполнение памяти при характеристике	Ошибка выдаётся, если переполнена память, объём которой установлен во 2-ом параметре инструкции NBP
187	В инструкции PRO не определён номер процесса	
188	Введена инструкция POM для оси, не являющейся шпинделем	Если ось не является шпинделем, то инструкция POM не вводится. Необходимо ввести вначале инструкции ТРА и PAS, а затем инструкцию POM
189	Не хватает инструкции POM	Для оси шпинделя с датчиком необходимо ввести инструкцию POM
190	Номер преобразователя в ADP или ADG превышает допустимое значение	
191	В инструкции ADP второй параметр должен быть отличным от нуля	
193	SKW объявлено не для подчинённой параллельной оси	
194	Наименование главной оси содержит больше одного знака	
195	Главная ось не объявлена	В инструкции SKW не сконфигурированная ось объявляется как главная
196	Ось, обозначенная как главная, уже объявлена как главная для другой оси	В поле «наименование главной оси» в инструкции SKW следует указать ось, которая ещё не объявлена главной в другой инструкции SKW
197	Подчинённая ось уже объявлена в инструкции SKW в другой подсекции NAS	

Продолжение таблицы 7.2

Номер ошибки	Объяснение	Примечание
198	Ошибка в объявлении подчинённой оси	Подчинённая ось объявляется только со следующими инструкциями: NAS, TPA, NTC, MCZ, MFC, SKW
199	Ось объявлена подчинённой самой себе	
200	Подчинённая ось не может быть интерполирована	В инструкции INx подчинённая ось объявлена как координатная
201	Подчинённая ось не может быть объявлена в секции 2 раньше главной	Подчинённая ось объявляется после главной оси
202	Подчинённая ось должна объявляться на том же CPU, что и главная	
203	Если ось объявлена как ось с контрольной точкой, то для нее необходимо записать инструкцию ASM с именем соответствующей ей главной оси	
204	Инструкции ADG, CUB и Swн не допустимы для данного типа оси	Для записи в подсекции оси инструкции ADG в инструкции TPA для нее должен быть добавлен код 8000H.
205	В инструкции Swн использование сигналов PLC пакета «А» недопустимо	Сигнал PLC должен быть в свободной части сигналов пакета «К»
206	Недопустимые значения коэффициентов ускорений в инструкции ACC	Значения коэффициентов 1, 3, 4 должны быть больше нуля. Значение коэффициента 2 должно быть в пределах от 0 до 0.5.
207	Количество осей, объявленных в инструкции CUB, больше трёх	
208	В NTC объявлен номер датчика или канала ЦАП, не соответствующий тарировке модуля энкодер-ЦАП	
209	Объявление инструкций MCZ, FBF, ZNO и FTP необходимо выполнять после объявления инструкции PAS	
210	Номер ЦАП/ЦИП больше 200	Недопустимый номер ЦАП/ЦИП.
211	Тип оси в инструкции TPA не равен числу 2	1. Если в инструкции PAS параметр QRM не равен нулю, то значение типа оси в инструкции TPA должно быть записано только цифрой 2. 2. Инструкция FTP может быть записана только для «от точки к точке»
212	Инструкция FRC должна быть объявлена для оси после инструкции GM0	

Продолжение таблицы 7.2

Номер ошибки	Объяснение	Примечание
213	Файл, объявленный в инструкции FTP для оси «от точки к точке», не корректный	1) количество записей (строк) должно быть равно количеству позиций оси «от точки к точке», заявленному во втором параметре инструкции PAS; 2) в 1-ой записи номер позиции должен быть равен значению 4-ого параметра инструкции PAS для оси «от точки к точке»; 3) нумерация позиций в 1-ом столбце файла должна быть последовательной и непрерывной; 4) Признак конца файла (EOF) должен быть на следующей строке после строки, определяющей последнюю позицию оси «от точки к точке»
214	Файл, объявленный в инструкции FTP для оси «от точки к точке», не существует.	Создайте файл в редакторе режима «COMMAND» или редакторе «ВП»
215	В инструкции ZNO значение «параметр3» не равно «0», а «параметр4» не задан	Установите в «параметре4» сигнал пакета «K» или «N» или удалите значение в «параметре3»
216	Ошибка в инструкции GRA	Сигнала PLC должен быть самым младшим сигналом в слове. Сигнала PLC должен принадлежать только пакету «A» .
217	Ошибка в инструкции GRA	Разрядность датчика с кодом Грея больше 16
218	Ошибка в инструкции GRA	Заявлена инструкция GRA , а код оси не 80000H
219	Неправильное имя оси в инструкции HWC	
220	Ошибка в инструкции CANn	Значение n должно быть равно 1,2,3 или 4.
221	Значение параметра zero_input в инструкции MCZ должно быть пустым, если для оси установлен ДПС с ККРМ	
222	Значение параметра KC в инструкции GMnn не может быть больше 10В	
223	Ошибка в инструкции FOPP	Сигнала PLC должен быть самым младшим сигналом в слове. Сигнала PLC не должен принадлежать пакету «A» .
224	Ошибка в инструкции FOPP	Инструкция FOPP может быть объявлена только для оси «от точки к точке».
225	Ошибка в инструкции ADG	В инструкции ADG не определены имена сигналов PLC в параметрах: 4, 5, 6 и 7.

Продолжение таблицы 7.2

Номер ошибки	Объяснение	Примечание
226	Ошибка в инструкции ТРА	Для оси, управляемой по CANopen-шине, в инструкции ТРА добавлены следующие 16-теричные коды: 8, 200, 1000, 10000, 80000.
227	Инструкция АТР может быть записана только для оси с абсолютным датчиком с интерфейсом SSI	Инструкция АТР может быть записана для оси, если в параметре <i>axis_type</i> инструкции ТРА этой оси добавлен 16-теричный код: 200000.
228	Для оси с абсолютным датчиком с интерфейсом SSI не записана инструкция АТР	Для оси в параметре <i>axis_type</i> инструкции ТРА добавлен 16-теричный код 200000, а инструкция АТР отсутствует.

7.2.3. Ошибки при управлении осями

Процесс управления осями диагностируется как при инициализации, так и во время работы системы. При обнаружении ошибки на дисплей выдаётся сообщение.

Существуют три вида ошибок, один из которых проявляется при инициализации, два других могут возникнуть в любой момент работы металлорежущего станка.

Ошибки управления осями:

1) ошибка инициализации энкодера; возникает при инициализации модуля энкодеров, если модуль не работает или работает неправильно;

2) ошибка сервоцикла: «Имя задачи»; возникает, если тик CPU, установленный в инструкции TIM, либо тик для сервоконтроля в инструкции CAS, либо тик для интерполятора в инструкции INX недостаточен для задачи, имя которой указано в тексте ошибки;

3) ошибка УЧПУ; возникает во время обычного функционирования станка из-за ошибки в вычислениях сопроцессора.

7.2.4. Ошибки характеризации процесса - файл PGCFIL

Ошибки файла характеризации процесса PGCFIL представлены в таблице 7.3.

Таблица 7.3 - Ошибки файла PGCFIL

Номер ошибки	Объяснение	Примечание
101	Инструкция дублируется	
102	Трёхбуквенный код в секции 1 не допустим	
103	Недопустимый трёхбуквенный код в секции 1, ожидаемый символ - D	
104	Код синхронизации должен быть в пределах от 1 до 4	
105	Индекс переменной символьной таблицы переполнен	Индекс, присвоенный переменной, не допустим

Продолжение таблицы 7.3

Номер ошибки	Объяснение	Примечание
106	Тип по умолчанию должен быть в пределах от 1 до 7	
107	Код, определяющий типы, должен быть меньше, чем 127	
108	Код синхронизации по умолчанию должен быть от 1 до 4	
109	Имя новой переменной или уже существует в символьной таблице, или написано некорректно	
110	Количество программ не может превышать 255	
112	Имя устройства, заявленное в NDD, написано некорректно, его длина не должна быть более трёх знаков	
113	Имя интерполятора должно состоять из одного знака	
114	Заявленный в NIP интерполятор не существует	
115	В NAM, NPD или TAG наименование оси должно состоять из одного знака	
116	Оси в NAM или NPD заявлены, но не определены в AXCFIL	
117	В NAM и NPD могут быть объявлены только координатные непрерывные оси	Имя оси шпинделя не может быть записано в NAM и NPD
118	Заявления NIP, NAM, NPD, TOF не присутствуют в PGCFIL	
119	Символьная таблица переменных переполнена (в секции 2 количество новых переменных не может быть более 20)	
120	Состояние измерительного щупа должно быть 0 или 1	
121		
122	3-й параметр в TAS должен состоять из двух символов ASCII: первый символ - имя оси, второй символ - «+» или «-».	
123	Тип щупа может быть S или N.	
124	Дублирование осей в строке определения оси	
125	В PGCFIL заявления NAM и NPD не могут быть записаны, если предварительно не было записано заявление NIP	
126	Максимальное число определяемых осей: NC-110, NC-310 - 8 осей; NC-200, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230 - 6 осей.	
128	Имя переменной в SIM или ASS отсутствует	
129	Переменная символьной таблицы не найдена	
131	В инструкции NIP неправильно объявлен интерполятор (например, если X и A - переключаемые оси, то они должны объявляться на одном и том же интерполяторе)	
132	Неправильное наименование оси	
133	Неправильное значение в TOF	Правая цифра в числе, записанном в инструкции TOF, может быть равна 1, 2, 5 или 6
134	Ошибка синтаксиса в инструкции FIL	
135	В инструкции STR объявлено слишком много структур	
136	В инструкции SCR объявлено слишком много видеостраниц	

Продолжение таблицы 7.3

Номер ошибки	Объяснение	Примечание
137	В инструкции CHN объявлено слишком много каналов	
139	Инструкция GXX должна быть записана после инструкции TOF	
140	Инструкция GXX содержит одну из перечисленных ошибок: - число состоит более чем из 2-х знаков; - число содержит символы букв; - неизвестный номер G	
146	Ось, объявленная в инструкции TGA, должна быть записана в интерполяторе непрерывных координатных осей.	
147	В инструкции TGA угол отклонения может иметь значение от -360 до 360 градусов	
148	Ось, объявленная в инструкции TGA, не ось вращения.	
184	В инструкции PRO объявлен номер процесса, объявленный ранее или не сконфигурированный	
186	Переполнение памяти при характеристизации	

7.2.5. Ошибки характеристизации логики - файл IOCFIL

Ошибки файла характеристизации логики **IOCFIL** представлены в таблице 7.4.

Таблица 7.4 - Ошибки файла **IOCFIL**

Номер ошибки	Объяснение	Примечание
101	Дублированная инструкция	
103	Тик «быстрой логики» слишком большой или равен 0	Тик быстрой логики выражен в мс; должен быть не более 255 мс, не должен быть равен нулю
104	Тик CPU не должен быть равен нулю	Пересмотреть инструкцию TIM в AXCFIL
105	Тик быстрой логики не кратен тiku CPU	Пересмотреть инструкцию CLO или инструкцию TIM в AXCFIL
106	Время для медленной логики объявлено неправильно	
107	В инструкции объявлен параметр с недопустимым номером.	В данной инструкции цифровое значение параметра не должно превышать максимально допустимый номер
108	Номер входного разъёма пакета «А» уже заявлен ранее в инструкции INx	
109	Номер входного разъёма пакета «А» уже заявлен ранее в инструкции OUn	
110	Номер выходного разъёма пакета «А» уже заявлен ранее в инструкции OUn	

Продолжение таблицы 7.4

Номер ошибки	Объяснение	Примечание
111	Номер разъёма пакета «А», заявленный в инструкции INx, объявлен в инструкции OUn	
113	Первый параметр инструкции Mnn (первый байт) должен быть отличным от нуля	
114	Тип оси должен быть 0 (абсолютный) или 1 (инкрементальный)	В инструкции ASn параметр типа оси должен быть 0 или 1
115	Наименование оси должно состоять из одного символа	
116	Не хватает инструкции CLO	
117	Определённая ось не существует	В данной инструкции объявлено наименование несуществующей оси
118	Определённая ось не является шпинделем.	В инструкции ASM объявляется осью шпинделя ось, которая не является шпинделем.
120	Для оси «от точки к точке» без преобразователя порог первого замедления должен быть больше нуля	Когда ось от «точки к точке» не имеет ЦАП, порог первого замедления должен иметь значение, отличное от нуля (предел1 в инструкции TAn)
121	Не хватает инструкции NPX или она равна 0	
124	Количество осей, определённых в инструкции UCDA, не может превышать 4.	
126	Объявленный в INx входной модуль не присутствует в УЧПУ	
127	Наименование оси не является допустимым	
129	Для индексных осей в инструкции ASn должно быть определено число позиций (четвертый параметр инструкции)	
130	Значение параметра в инструкции GPS может быть 0 или 1	
131	Количество позиций для оси от точки к точке в инструкции TAX должно быть равно количеству позиций в инструкции PAS для этой же оси в файле AXCFIL.	
184	В инструкции PRO записан номер процесса, превышающий максимальное число процессов, определённое в инструкции NBP, или же выбран процесс, ранее не сконфигурированный	
186	Переполнение памяти при конфигурации	

8. Настройка сети

8.1. Особенности настройки сети

Для работы УЧПУ в локальной сети необходимо, в каталоге C:\CNC32\MP0 создать следующие файлы:

- 1) **netcfg.ini** - файл содержит параметры локальной сети и имя сетевого адаптер;
- 2) **admin.ini** - файл содержит имена ресурсов УЧПУ и уровни доступа к ним;
- 3) **user.ini** - файл содержит имена пользователей сети и их уровень доступа к ресурсам УЧПУ;
- 4) **servers.ini** - файл содержит имена удаленных сетевых ресурсов, доступных для пользователя УЧПУ;
- 5) **[gatecfg.ini]** - файл содержит параметры шлюзов и параметры других сетей, которые должны иметь доступ к УЧПУ. Файл **gatecfg.ini** можно не создавать, если все компьютеры и УЧПУ в сети принадлежат к одной рабочей группе.

8.2. Настройка параметров сети в файле netcfg.ini

Файл **netcfg.ini** содержит следующие инструкции: **ip, mask, user, password, lancard, namecnc, group, syspass.**

Для выбора инструкции и ввода значений их параметров установите режим «КОМАНДА» и выполните в меню движение по следующему пути: «F2»(МОДИФ) -> «F7»(SNT) -> «F1»(Netcfg). Далее для выбора инструкции и ввода значения ее параметра используйте управляющие клавиши: «ПЕРЕВОД НА СТРОКУ ВПЕРЕД», «ПЕРЕВОД НА СТРОКУ НАЗАД» и «ENTER».

Для сохранения параметров сети в файле **netcfg.ini** в главном меню **SNT** нажмите клавишу «F7»(Применить настройки) в противном случае нажмите клавишу «F8»(Выход).

Пример текста файла **netcfg.ini**:

```
ip:192.168.0.3
mask: 255.255.255.000
user: NIK126
password: INIK
namecnc: CNC3
lancard: r8139
group: WGROUP1
syspass: Vg
```

8.2.1. Инструкция ip

Семантика:

ip:A1.A2.A3.A4 .

Формат:

ip:byte.byte.byte.byte ,

где:

A1.A2.A3.A4 - адрес УЧПУ (узла) в сети. В протоколе IP 4-ой версии (IPv4) адрес узла состоит из 4-х байтов.

Пример

ip:192.168.0.3

8.2.2. Инструкция mask

Семантика:

mask:M1.M2.M3.M4 .

Формат:

mask:byte.byte.byte.byte ,

где:

M1.M2.M3.M4 - маска подсети; состоит из 4-х байтов. Маска подсети определяет какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети.

Пример

mask:255.255.255.0

8.2.3. Инструкция user

Семантика:

user:user_name .

Формат:

user:ASCII string ,

где:

user_name - имя пользователя УЧПУ. ***user_name*** может содержать максимум 8 алфавитно-цифровых (латинских) символов и может быть использовано для настройки доступа пользователю УЧПУ к другим узлам сети.

Пример

user:NIK126

8.2.4. Инструкция password

Семантика:

password:*user_password* .

Формат:

password:ASCII string ,

где:

user_password - пароль пользователя в УЧПУ. Пароль пользователя УЧПУ может содержать максимально 11 алфавитно-цифровых (латинских) символов и использован для настройки доступа пользователю УЧПУ **user_name** к другим узлам сети.

Пример

password:INIK

8.2.5. Инструкция lancard

Семантика:

lancard:*lancard_name* .

Формат:

lancard:ASCII string ,

где:

lancard_name - имя сетевой карты, интегрированной в материнской плате УЧПУ. Для выбора **lancard_name** в колонке списка производителей «Производитель:» должен быть выбран производитель сетевой карты с клавишей «ENTER» и далее в колонке списка драйверов «Драйвер:» должен быть выбран драйвер для сетевой карты, имя которой выводится в колонке «Сетевая карта:». Выбор сетевой карты должен быть сделан с клавишей «ENTER», в противном случае нажмите клавишу «Esc».

Пример выбора сетевой карты 8139 представлен на рисунке 8.1.

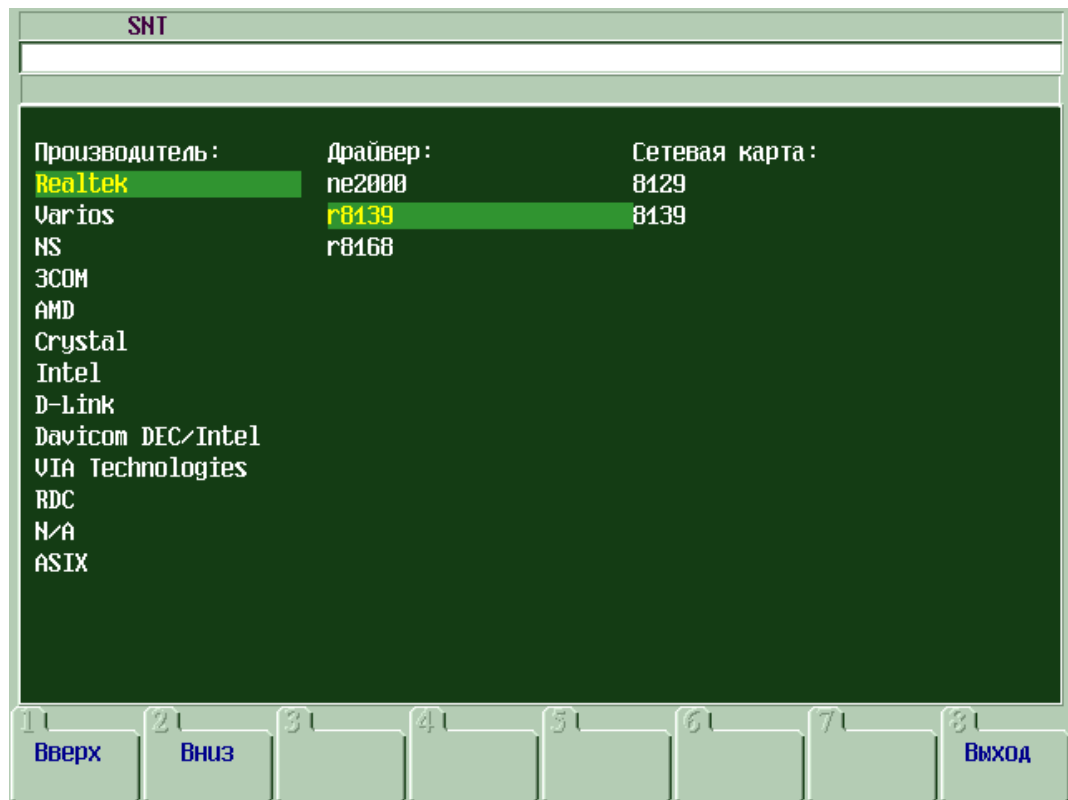


Рисунок 8.1.

Пример

lancard:r8139

8.2.6. Инструкция namesnc

Семантика:

namesnc:lan_name_cnc .

Формат:

namesnc:ASCII string ,

где:

lan_name_cnc - сетевое имя УЧПУ. **lan_name_cnc** может содержать максимально 12 алфавитно-цифровых (латинских) символов.

Пример. namesnc:CNC3

8.2.7. Инструкция group

Семантика:

group:name_workgroup .

Формат:

group:ASCII string ,

где:

name_workgroup - имя рабочей группы, в которую входит УЧПУ, **name_workgroup** может содержать максимально 12 алфавитно-цифровых (латинских) символов.

Пример

group:WGROUP1

8.2.8. Инструкция syspass

Семантика:

syspass:code .

Формат:

syspass:ASCII string ,

где:

code - пароль доступа для изменения данных в файле **netcfg.ini**. По умолчанию значение **code** соответствует паролю: BS .
Если текст файла **netcfg.ini** открыт в редакторе на ПК, то текст пароля зашифрован и поэтому не может быть изменен.

Примечание - Если файл **netcfg.ini** в системе не существует, то действует пароль: BS .

8.3. Настройка параметров сети в файле user.ini

Записи файла **user.ini** определяют имена пользователей сети, их пароли и их уровни доступа к ресурсам УЧПУ.

Для ввода записей в файле **user.ini** установите режим «КОМАНДА» и выполните движение по следующему пути: «F2» (МОДИФ) -> «F7» (SNT) -> «F1» (Настройка пользователей). Далее для выбора инструкции и ввода значения ее параметра используйте управляющие клавиши: «ПЕРЕВОД НА СТРОКУ ВПЕРЕД», «ПЕРЕВОД НА СТРОКУ НАЗАД», «ПЕРЕВОД КУРСОРА ВЛЕВО», «ПЕРЕВОД КУРСОРА ВПРАВО» и «ENTER».

Меню режима «Настройка пользователей» представлена на рисунке 8.2:

Добавить пользователя - Для добавления нового пользователя нажмите функциональную клавишу «F3» (Добавить пользователя). Далее введите

имя пользователя, пароль пользователя и его группу. Ввод каждого значения выполните клавишей «ENTER».

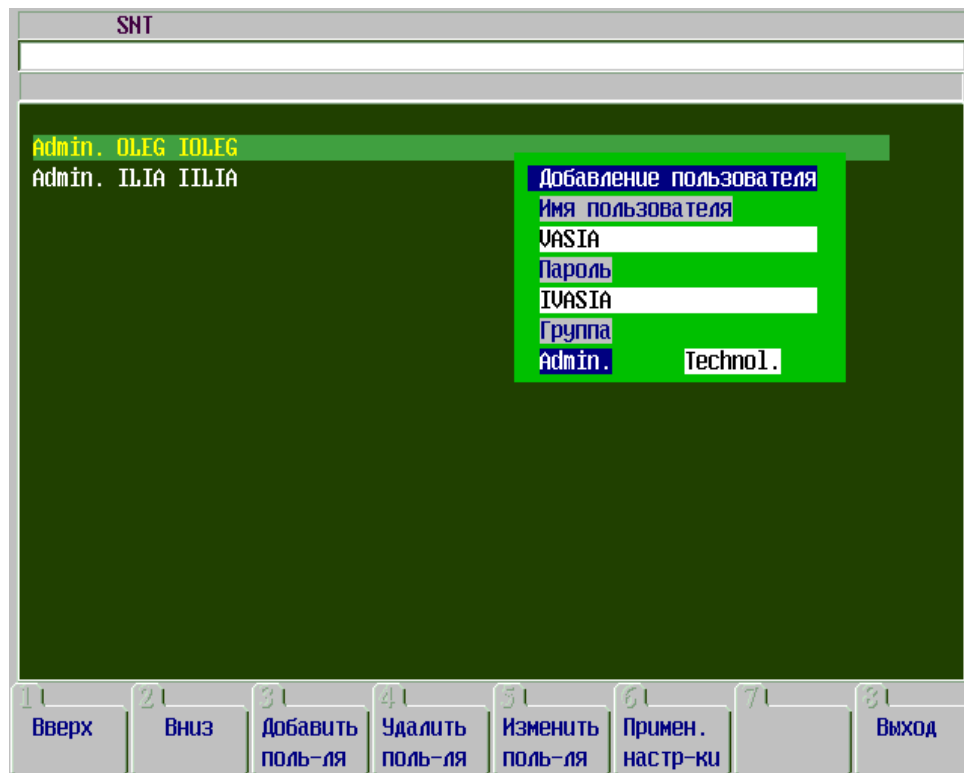


Рисунок 8.2.

Примечание - Группа «Admin.» имеет права выполнять редактирование файлов, расположенных в УЧПУ, с удаленного узла локальной сети.

Удалить пользователя - для удаления пользователя нажмите функциональную клавишу «F4» (**Удалить пользователя**). Удаление пользователя выполнится без запроса подтверждения.

Для сохранения измененных данных о пользователях, имеющих доступ к сетевым ресурсам УЧПУ, в файле **user.ini** в меню (**Настройка пользователей**) нажмите клавишу «F6» (**Применить настройки**) в противном случае нужно нажать клавишу «F8» (**Выход**).

Пример файла **user.ini**

```
Admin. OLEG IOLEG
Admin. ILIA IILIA
```

8.4. Настройка сетевых ресурсов в файле **admin.ini**

Записи файла **admin.ini** определяют имена сетевых ресурсов, расположенных в УЧПУ, а также уровень доступа к ним.

Для ввода записей в файле установите режим «КОМАНДА» и выполните движение по следующему пути: «F2» (МОДИФ) -> «F6» (SMP) -> «F7» (Назначить пользователя). Далее для выбора диска и каталога, к которому должен быть разрешен доступ

извне, используйте клавиши управления курсором: «ПЕРЕВОД НА СТРОКУ ВПЕРЕД», «ПЕРЕВОД НА СТРОКУ НАЗАД» и клавишу выбора: «ENTER».

Установите курсор в список файлов внутри каталога, к которому должен быть разрешен сетевой доступ. В строке ввода и редактирования введите имя сетевого ресурса с клавишей «ENTER», и в меню нажмите клавишу выбора уровня доступа к этому ресурсу:

«F1» (Admin)

«F2» (Technol)

Пример файла **user.ini**

C:\CNC32\MP1 Admin MP1CNC3

- 1) **servers.ini** - записи файла определяют удаленные сетевые ресурсы, к которым будет иметь пользователь, имя и пароль которого определены в **netcfg.ini**;
- 2) **gatecfg.ini** - записи этого файла определяют параметры шлюзов и параметры других сетей, которые должны иметь доступ к УЧПУ. Файл можно не создавать, если все компьютеры и УЧПУ в сети принадлежат к одной и той же рабочей группе.

8.5. Настройка внешних сетевых ресурсов в файле **servers.ini**

Записи файла **servers.ini** определяют удаленные сетевые ресурсы, к которым будет иметь доступ пользователь, имя и пароль которого определены в **netcfg.ini**. Записи файла делаются во внешнем редакторе.

Формат строки:

PC_name\Resource[:ip]] ,

где:

PC_name - сетевое имя компьютера максимально 12 алфавитно-цифровых (латинских) символов.

Resource - имя сетевого ресурса, доступ к которому разрешен для пользователя УЧПУ.

:ip - ip-адрес ПК, сетевое имя которого записано в параметре **PC_name**.

ВНИМАНИЕ! Если на ПК установлена ОС WINOWS 7, то требуется обязательно ввод значений для **PC_name** и **Resource**.

Пример записей в файле **servers.ini**

192.168.2.19\TMP1

ADMINPC\TRANSMIT
BS-VASIA\D\0:192.168.1.23

8.6. Настройка доступа к другой сети в файле **gatecfg.ini**

Записи файла **gatecfg.ini** определяют удаленные сетевые ресурсы в другой сети, к которым будет иметь доступ пользователь, имя и пароль которого определены в **netcfg.ini**. Записи файла делаются во внешнем редакторе.

Для организации сетевого доступа к УЧПУ и компьютерам, расположенным в разных рабочих группах, и соединенных друг с другом маршрутизатором (шлюз) в каталоге C:\CNC32\MP0 должен быть создан файл **gatecfg.ini**, в который должны быть введены записи со следующим форматом:

Name_gate:IP_ADDRESS_GATE_current_LAN:IP_ADDRESS_other_LAN:MASKA_other_LAN ,

где:

Name_gate - мнемоническое имя шлюза. Имя шлюза является текстовой маркировкой текущей записи.

IP_ADDRESS_GATE_current_LAN - IP-адрес шлюза, в котором расположена текущая рабочая группа с УЧПУ, которая должна получить доступ к ресурсам, расположенным в другой сети.

IP_ADDRESS_other_LAN - IP-адрес другой сети, к ресурсам которой могут получить доступ УЧПУ текущей сети.

MASKA_other_LAN - маска другой сети, к ресурсам которой могут получить доступ УЧПУ текущей сети.

": " - символ предназначен для разделения параметров в одной и той же записи.

Пример записи в файле **gatecfg.ini** (соответствует рисунку В.1):

A_to_B:192.168.5.1:10.2.12.0:255.255.255.0

ПРИМЕЧАНИЯ:

- 1) Запишите в файле **servers.ini** текущего УЧПУ на отдельной строке IP-адрес УЧПУ или компьютера\имя ресурса, расположенного в другой сети, ресурс которого открыт для доступа.
- 2) Запишите в файле **servers.ini** текущего УЧПУ на отдельной строке: **IP-адрес компьютера\имя ресурса**, расположенного в этой же рабочей группе, если на компьютере установлена ОС **WINDOWS 7.0**, ресурс которого открыт для доступа.

Пример записи в файле **servers.ini** в УЧПУ, расположенном в сети 192.168.5.0, для получения доступа к ресурсу 10.2.12.4/USER,

расположенному в другой сети 10.2.12.0 (соответствует рисунку 8.3): **10.2.12.4/USER**

Подключение ЧПУ к компьютерной сети.

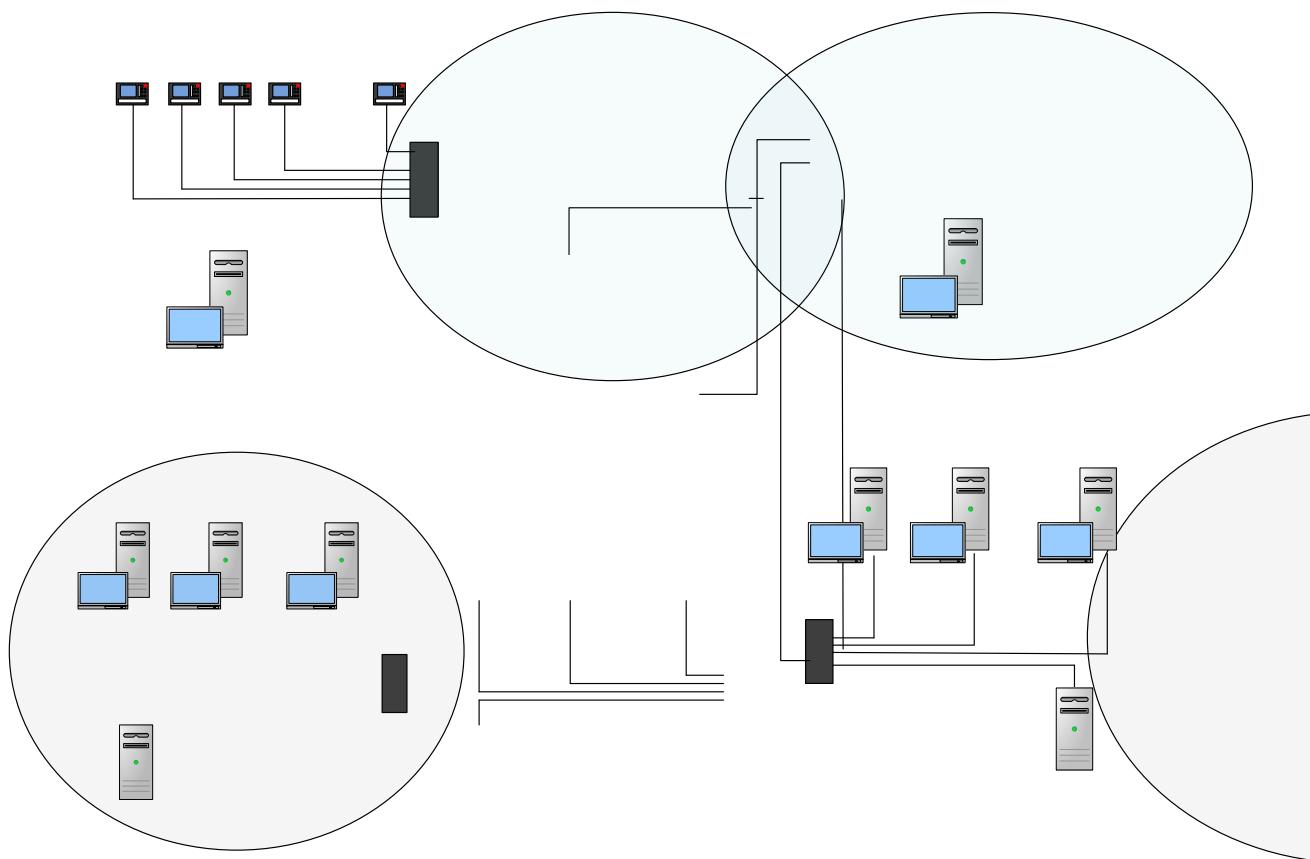


Рисунок 8.3

Пример записей в файле **gatecfg.ini**

aaa:10.0.0.1:192.168.2.0:255.255.255.0

9. КОДЫ СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ РТА И ГТА

9.1. Назначение кодов РТА и ГТА

Коды **РТА** и **ГТА** описаны в данном документе, т. к. имеют системное назначение.

Код **РТА** обычно выполняется при настройке оси из строки ввода и редактирования в режиме «**РУЧНОЙ ВВОД**» («**MDI**») по кнопке «**ПУСК**», реже в режимах «**АВТОМАТИЧЕСКИЙ**» («**AUTO**») или «**КАДР**» («**STEP**»), или по запросу от ПЛ для обеспечения адаптации управления к изменяющимся условиям работы. Последнее значение для каждого параметра, заданное с кодом **РТА**, сохраняется до перезапуска или выключения УЧПУ.

Код **ГТА** обычно выполняется из текущего или параллельного процесса, а также по запросу от ПЛ для дополнительного контроля технологии процесса обработки.

9.1.1. Код РТА

Код **РТА** изменяет параметры, характеризующие объявленную ось. Формат:

(**РТА, имя оси, номер параметра, величина**) ,

где:

имя оси - наименование оси, параметр которой подлежит изменению; должно соответствовать имени оси, объявленному при характеристике, и быть представлено символом **ASCII**;

номер параметра - номер изменяемого параметра оси должен быть представлен числом от 1 до 18. Значения параметров, заданные для номеров с 9 по 13, учитываются в системе только при использовании версий **SW** с буквами «**PIB**» или с 16-тиричным кодом **ECDF**. Каждому изменяемому параметру оси соответствует свой номер:

- | | |
|-----------|--|
| 1 | - сервоошибка в останове, мм; |
| 2 | - сервоошибка со скоростной компенсацией, мм; |
| 3 | - сервоошибка без скоростной компенсации, мм; |
| 4 | - время ожидания входа оси в допуск, с; |
| 5 | - ускорение ручной подачи и при G01 , мм/с ² ; |
| 6 | - ускорение быстрого хода, мм/с ² ; |
| 7 | - скорость быстрого хода, мм/мин; |
| 8 | - скорость ручной подачи, мм/мин; |
| 9 | - константа усиления KV , с ⁻¹ ; |
| 10 | - время действия компенсации трения, мс; |

- 11** - значение минимальной величины компенсации трения ΔV_{min} , мм/мин;
- 12** - значение коэффициента скоростной составляющей ПИД-регулятора (**Kvff**);
- 13** - значение коэффициента интегральной составляющей ПИД-регулятора (**Kvffi**);
- 14** - значение максимального времени инверсии шпинделя; (инструкция **TSM**, 1-ый параметр);
- 15** - значение максимального напряжения инверсии шпинделя (инструкция **TSM**, 2-ой параметр);
- 16** - значение максимального изменения скорости на профиле (инструкция **FAV**, 1-ый параметр);
- 17** - значение максимального изменения ускорения на профиле (инструкция **FAV**, 2-ой параметр);
- 18** - значение максимально допустимой скорости оси при установке эталонного напряжения, указанного во втором параметре инструкции **GMO** данной оси;
- 19** - действительный номер позиции оси «от точки к точке». Выполнение трехбуквенного кода **PTA** с номером параметра **19** установит для заданной оси признак «Ось выведена в ноль» без ее вывода в ноль;
- 20** - значение параметра **acc_linear_part_S** в инструкции **ACC**;
- 21** - значение параметра **acc_nonlinear_part_S** в инструкции **ACC**;
- 22** - значение параметра ΔV_{max} в инструкции **FRC**;
- 23** - значение полосы пропускания фильтра для контура положения.

величина - новое значение параметра; может быть представлено константой или **E**-параметром в формате **real**.

Пример

(**PTA**, **Y**, 8, 700) - изменяет для оси **Y** параметр 8 (скорость ручной подачи) на значение, равное 700 мм/мин.

9.1.2. Код **GTA**

Код **GTA** записывает значение объявленного параметра оси в указанную системную переменную.

Формат:

(**GTA**, имя оси, **NM**, переменная) ,

где:

ИМЯ ОСИ - наименование оси, параметр которой подлежит записи в системную переменную; должно соответствовать имени оси, объявленному при характеристике, и быть представлено символом **ASCII**;

NM - цифра **N** определяет номер процесса (1-5), из которого должно быть прочитано значение параметра, определённого цифрой **M**. Цифра **M** может принимать следующие значения:

- 1** - текущая позиция оси относительно микроула в текущем процессе;
- 2** - текущая позиция оси относительно текущей исходной точки (с учётом длины инструмента) в текущем процессе;
- 3** - результат измерения оси относительно микроула в текущем процессе;
- 4** - номер текущей абсолютной исходной точки в текущем процессе.
- 9** - текущая позиция с датчика оси относительно микроула в текущем процессе.

переменная - имя системной переменной (**E, SA, SK, SYVAR**), куда должен быть записан объявленный параметр оси.

Запись первого и второго параметров возможна как для интерполированных осей, так и для осей «от точки к точке». Третий и четвёртый параметры могут быть записаны только для интерполированных осей.

Пример

(GTA, X,11,E30) - нажать клавишу «ПУСК» (режимы «MDI»/«AUTO»/«STEP») - записывает для оси **X** первого процесса параметр «текущая позиция оси относительно микроула» в системную переменную **E30** текущего процесса.

(GTA, X,21,E30) - нажать клавишу «ПУСК» (режимы «MDI»/«AUTO»/«STEP») - записывает для оси **X** второго процесса параметр «текущая позиция оси относительно микроула» в системную переменную **E30** текущего процесса.

ПРИЛОЖЕНИЕ А

Особенности управления осями

Приложение к инструкции FQF

При настройке контура положения оси важными параметрами, влияющими на точность и устойчивость отработки, являются следующие параметры:

- 1) **KV** - коэффициент усиления;
- 2) **KFF** - скоростная компенсация;
- 3) **Wp** - полоса пропускания контура положения;
- 4) **Ws** - полоса пропускания контура скорости;
- 5) **Ki** - интегральный коэффициент.

Примечание - Поскольку ввод в контур управления значения **Ki** ухудшает устойчивость системы, поэтому настройка **Ki** не рассматривается.

Система автоматического управления включает в себя: УЧПУ (регулятор положения), электропривод (ЭП), электродвигатель (ЭД), датчик обратной связи (ДП) и нагрузку.

Общие схемы тестируемых систем автоматического управления представлены на рисунках А.1 и А.2, где:

- θ_z - заданная позиция.
- θ_d - позиция с ДП.
- Δe - значение сервоошибки ($\theta_z - \theta_m$).
- ω_m - заданная скорость вращения ЭД.

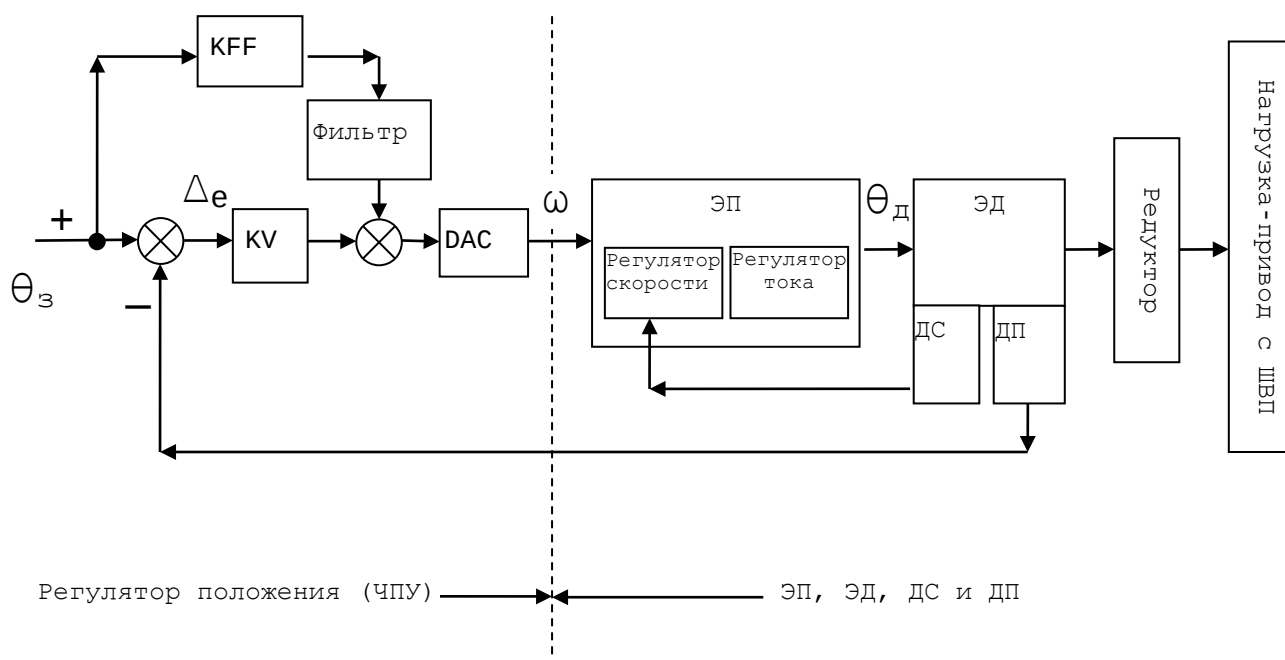
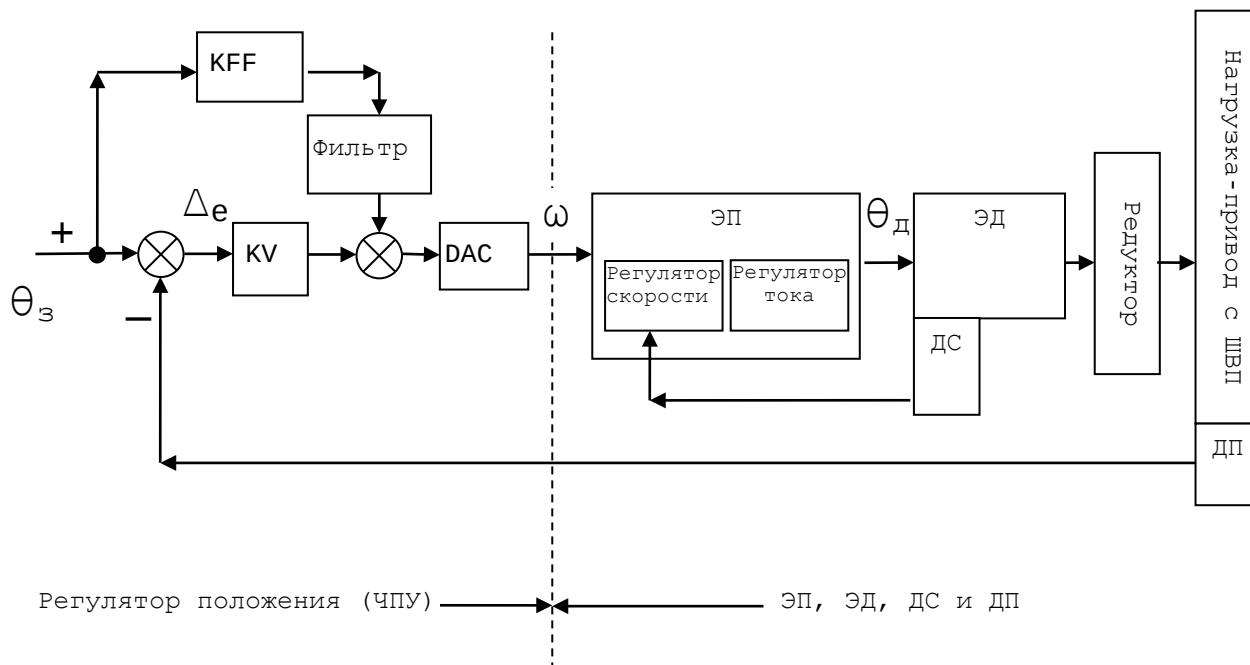


Рисунок А.1



Автоматическая настройка контура положения оси включает в себя автоматический поиск оптимальных значений для параметров **KV**, **KFF** и **Wp** оси, имеющей ДОС, в любом УЧПУ серии **NC-XXX**.

При настройке контура положения важными параметрами, влияющими на точность и устойчивость обработки, являются полоса пропускания контура положения **Wp**, скоростная компенсация **KFF** и коэффициент усиления **KV**.

Существует несколько различных методов для подбора оптимальных значений параметров **KV** и **KFF**, но все они мало подходят для тестирования контура управления в автоматическом режиме. Нами используется метод, позволяющий выполнить автоматический поиск оптимальных значений для **Wp**, **KFF** и **KV**.

Приоритетным показателем, определяющим точность и устойчивость обработки, нами выбрана полоса пропускания контура положения.

При этом под полосой пропускания подразумевается частота, при которой амплитуда синусоиды на выходе измеряемой системы уменьшается на 0.71 от первоначальной (например, при 0,5 Гц) амплитуды синусоиды.

Полоса пропускания контура положения должна быть достаточно широкой, чтобы обеспечить необходимую динамическую точность. Однако увеличивать полосу пропускания контура положения больше необходимой не следует, т.к. шум на выходе увеличивается пропорционально ширине полосы пропускания, что приводит к уменьшению точности. Поэтому существует оптимальная полоса пропускания контура положения. Для современных приводов с шарико-винтовой передачей таким общепризнанным значением полосы пропускания контура положения является 30 Гц. Кроме того, оптимальная полоса пропускания контура положения находится в прямой зависимости от полосы пропускания контура скорости.

Для исключения нежелательных колебаний необходимо, чтобы полоса пропускания контура скорости была не менее чем в 2 раза выше, чем полоса пропускания контура положения. Поэтому, если полоса

пропускания контура скорости равна 70 Гц, то оптимальной полосой пропускания контура положения выбирается 30 Гц.

Однако, если полоса пропускания контура скорости равна 40 Гц, то получаем для оптимальной полосы значение равное 20 Гц, (т.к. $40/2=20$).

При этом увеличение или уменьшение полосы пропускания до оптимального значения происходит в тесте при помощи **KFF** (или **KV**).

Несмотря на то, что скоростная компенсация **KFF** сильно влияет на полосу пропускания и во многих случаях улучшает точность обработки, она имеет и существенные недостатки.

Прежде всего, необходимо отменить генерируемые при помощи **KFF** высокочастотные шумовые компоненты. Кроме того, **KFF** из-за своей не замкнутости является очень чувствительным к изменению параметров устройств (по насыщению ЦАПа, по ограничению некоторых параметров привода).

Общепринятым методом удаления вышеназванных недостатков является установка фильтра низкой частоты после **KFF**. При этом, как правило, полоса этого фильтра выбирается равной оптимальной полосе пропускания контура положения.

При таком выборе полосы фильтра значения **KFF** очень близки к единице.

Окончательная проверка оптимальности значений **KFF** (или **KV**) происходит при помощи коэффициента колебательности **M**, который характеризует склонность системы к колебаниям. На практике $M=1.1$ обеспечивает хороший запас устойчивости (по фазе более 60 градусов). Значения $M > 1.1$ могут вызвать в системе колебания, а значения < 1.1 являются признаком инерционности системы, т.к. уменьшение значения **M** связано с уменьшением полосы пропускания. Поэтому значения параметров **KV**, **KFF** и **Fпп**, полученные в конце автоматического теста, являются оптимальными.

Поскольку тестирование оси выполняется в достаточно жестких условиях, без контролируемого разгона/торможения и без **S**-образного закона для разгона/торможения, то конечные результаты теста могут быть скорректированы опытными пользователями после ввода этих параметров. Результаты тестирования осей, участвующих в совместной интерполяции, должны быть выровнены по результатам тестирования худшей оси следующим образом:

- 1) значения сервоошибок всех осей в случае **UEP=1** (**KFF** выключено) были равны друг другу;
- 2) значения сервоошибок всех осей в случае **UEP=0** (**KF** включено) в автоматическом режиме работы были равны друг другу.

Значения **KV**, **KFF**, **Wp** и **Ws** определяются при выполнении тестовой программы **KVKF**. Программа **KVKFF** расположена в устройстве памяти **MP0**.

До запуска программы **KVKFF** необходимо выполнить следующие настройки для тестируемой оси:

- 1) Скомпенсировать смещение нуля.
- 2) Установить в инструкции **GMnn** шкалу «скорость-напряжение», соответствующую установленной скорости на приводе.
- 3) Проверить точность выполнения перемещений.

Перед запуском программы необходимо подготовить следующие данные о тестируемой оси для последующего их ввода после запуска программы **KVKFF**:

- Вв. макс. скор. в (мм/мин) в **GM0** - например 9000
- Вв. макс. напр. в (В) в **GM0** - например 7.5 или -7.5
- Введ. знач. соотн. W_s/W_p **K** - рекомендуется 2, при этом полоса контура положения в 2 раза меньше полосы пропускания канала скорости.
- Введите призн. измен. **KFF** - например 1, при этом тест определит оптимальный **KFF**, при 0 тест определит оптимальный **KV**

Значения по умолчанию для остальных параметров тестирования оси представлены в таблице **A.1**.

Таблица A.1

Номер E-параметра	Значение по умолчанию	Обозначение (размерность)	Назначение
E100	20	W_{sn} (Гц)	Начальное значение полосы пропускания в контуре управления по скорости (W_s)
E101	160	W_{sm} (Гц)	Максимальное значение полосы пропускания в контуре управления по скорости (W_s)
E102	20	dW_s (Гц)	Шаг увеличения полосы пропускания в контуре управления по скорости
E103	10	D_s (%)	Точность расчетов W_s
E116	16.67	KV_0	Начальный коэффициент усиления KV в контуре управления по положению
E117	0.7	KFF_0	Начальный коэффициент скоростной компенсации KFF
E104	1	dKV	Шаг изменения KV
E105	0.1	$dKFF$	Шаг изменения KFF
E106	1	D_p (%)	Точность расчетов полосы пропускания в контуре управления по положению (W_p)
E107	1.1	M	Оптимальный коэффициент колебательности
E108	2	K	Значение отношения W_s/W_p
E109	5	dW_p (Гц)	Шаг уменьшения W_p при поиске оптимальной колебательности в контуре управления по положению
E27	9221	V (мм/мин)	Максимальная скорость в GM0
E29	7.5(-7.5)	KC (В)	Максимальное напряжение в GM0
E115	1(0)		1 Kff 0 KV

Примечание - поскольку в УЧПУ можно подключить фильтр с помощью параметра E31, в котором задается полоса равная оптимальной полосе канала положения. Фильтр позволяет исключить сигналы скоростного канала с частотой находящейся за пределами полосы пропускания контура положения.

При E31=0 фильтр отключается.

ПРИЛОЖЕНИЕ В

Особенности управления осей

Для точного управления приводом оси необходимо:

- 1) скомпенсировать смещение нуля в контуре управления.
- 2) правильно согласовать значения параметров в инструкции

GMnn с соответствующими настройками этого привода (**скорость оси** при эталонном напряжении **KC** и коэффициент усиления **KV**). Правильность согласования скорости **Vmax** (инструкция **GMnn** для конфигурируемой оси) с действительной скоростью оси при эталонном напряжении можно проверить, если выполнить сравнение значения расчетной сервоошибки оси для любой заданной скорости оси и значения действительной сервоошибки при её движении на этой же скорости. Расчетная и действительная сервоошибки на заданной скорости в идеальном случае должны быть равны с погрешностью, равной классу точности станка.

Значение расчетной сервоошибки вычисляется по формуле:

$$E = V / (60 * KV) ,$$

где:

V - скорость оси при её движении в режиме «**Безразмерные ручные перемещения**» (**MANU**) из поля индикации **F**;

KV - значение коэффициента усиления, записанное в инструкции **GMnn** для тестируемой оси.

Значение действительной сервоошибки этой же оси необходимо определить с экрана УЧПУ из поля индикации осей при её движении в режиме «**Безразмерные ручные перемещения**» (**MANU**) на постоянной скорости **V**. Режим индикации осей для вывода сервоошибки устанавливается командой **UCV=2**, выполненной по клавише **ENTER**.

ПРИМЕЧАНИЯ:

Сравнивать расчетную и действительную сервоошибки при выполнении перемещения, заданного в кадре, можно, если:

- 5) скоростная компенсация сервоошибки отключена. Отключение скоростной компенсации сервоошибки в программных режимах работы устанавливается командой **UEP=1**, выполненной по клавише **ENTER**.
- 6) скомпенсировано смещение нуля привода.

Внимание!

При настройке привода необходимо стремиться к тому, чтобы скорость **Vmax** в инструкции **GMnn** при **эталонном напряжении 7.5В** и максимальная паспортная скорость, записанная в инструкции **RAP**, для каждой оси не отличались более чем на 25% ($7.5В + 0.25 * 10В = 10В$). В противном случае зона нечувствительности привода увеличивается.

После выполнения согласования скорости **Vmax** и действительной скорости оси при эталонном напряжении необходимо согласовать значение **KV** в инструкции **GMnn** с коэффициентом усиления привода. Практически, устанавливая значение **KV**, пользователь указывает системе максимально допустимое рассогласование по оси. Значение напряжения для одного импульса датчика оси зависит от её электромеханического

шага (количество импульсов в одном мм) и параметров инструкции **GMnn** (**Vmax** и **эталонное напряжение**).

Внимание!

Оси, одновременно участвующие в интерполяции, например круговой, должны иметь одинаковые сервоошибки при одинаковой скорости движения. Выравнивание сервоошибок осей пользователь должен обеспечить подбором значений **KV** в инструкциях **GMnn** этих осей.

3) Установить для каждой координатной оси значение низкочастотного фильтра в контуре положения (инструкция **FQF**), используя тестовую программу **KVKF** рассмотренную в **ПРИЛОЖЕНИИ А**.

Выполнить выравнивание значений сервоошибок при движении осей на одной и той же скорости со скоростной компенсацией по худшей из этих осей, используя значения **Kvff** в инструкции **FRC**.

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

ДДС (transducer)	- датчик обратной связи;
ОЗУ (RAM)	- оперативное запоминающее устройство;
ПЛ (logica)	- программа логики станка;
Про (SW)	- программное обеспечение;
УП (program)	- управляющая программа (для обработки детали);
УЧПУ (CNC)	- устройство числового программного управления;
ЦП (CPU)	- центральный процессор;
ЦАП (DAC)	- цифро-аналоговый преобразователь;
ЦИП (DPC)	- цифро-импульсный преобразователь;
ШД (SM)	- шаговый двигатель;
ВП (VP)	- визуальное программирование;
ffa	- сила трения в покое (англ. friction force alone);
НЗК (N.C.)	- нормально замкнутый контакт
HW	- штурвал (англ. handwheel)
SSI	- синхронно-последовательный интерфейс (англ. Synchronous Serial Interface)

ПЕРЕЧЕНЬ ТЕРМИНОВ

Инициализация	- установка начальных данных для функционирования SW УЧПУ в соответствии со значениями, объявленными при характеристике, и их диагностики.
ПЛ	- программа логики станка разрабатывается на языке PLC (см. документ «Программирование интерфейса PLC »).
УП	- управляющая программа, определяет технологию обработки детали.
Характеризация	- запись параметров и характеристик управляемого оборудования, а также аппаратных и программных модулей УЧПУ, в соответствии с требованиями, предъявляемыми к составлению файлов FCSYS , AXCFIL , IOCFIL и PGCFIL , описанных в данном документе.