

Министерство образования и науки Российской Федерации
САНКТ-ПЕТЕРБУРГСКИЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО
Высшая школа киберфизических систем и управления

О.А. Викторов

Система автоматического позиционирования
и управления циклом работы **IPOSplus**
учебное пособие

направление подготовки: 27.04.04. «Управление в технических системах»

квалификация (степень) выпускника: магистр

магистерская программа 27.04.04.06 «Системы управления электроприводами»

форма обучения: очная

Санкт-Петербург

2017г

УДК 621.3

Викторов О. А. Система автоматического позиционирования и управления циклом работы IPOsplus: учеб. пособие/ О. А. Викторов.- СПб., 2017.-59с.

Пособие соответствует ФГОС ВПО по направлению 27.04.04. «Управление в технических системах», магистерская программа 27.04.04.06 «Системы управления электроприводами», по дисциплине «автоматизированный электропривод».

Изложен материал необходимый для изучения принципов работы, организации, возможностей, а также получения навыков применения и программирования системы IPOsplus в преобразователях частоты MOVIDRIVE B. Предлагаются задания для самостоятельной работы, помогающие освоить материал.

Предназначено для студентов, аспирантов и слушателей курсов повышения квалификации.

Введение.

Цель данного пособия помочь студентам в выполнении курсового проекта по дисциплине «автоматизированный электропривод» по разработке на базе современных частотных электроприводов SEW EURODRIVE систем автоматизации машины, механизма или небольшого производственного участка.

В процессе курсового проектирования студенты должны получить и закрепить навыки:

- разработки структуры подобных систем,
- выбора элементов электропривода и связанных с ним датчиков и других устройств, применяемых при автоматизации,
- разработки принципиальных схем,
- выбора и настройки необходимых режимов работы частотного электропривода,
- написания программного обеспечения, обеспечивающего требуемый алгоритм работы.

Основная цель данного пособия – помочь освоить встроенную в преобразователь частоты **MOVIDRIVE B** систему автоматического позиционирования и управления циклом работы **IPOSplus**. При этом предполагается, что студенты уже имеют начальное представление об аппаратной части преобразователя, методах его настройки и ввода в эксплуатацию.

1. Система автоматического позиционирования и управления циклом работы IPOsplus.

1.1. Общее описание. Работа в редакторе ASSEMBLER.

Встроенная в MOVIDRIVE®B система автоматического позиционирования и управления циклом работы **IPOsplus** во многих случаях способна значительно разгрузить, а иногда и заменить программируемый логический контроллер (ПЛК) верхнего уровня. Это, иногда, позволяет при использовании оборудования SEW EURODRIVE заметно сэкономить на стоимости самого оборудования и затратах на монтаж.

Ниже перечислены основные возможности системы **IPOsplus**

1. При наличии обратной связи (датчика) может обеспечить управление позиционированием при перемещении из точки в точку. А также управление приводом независимо от наличия обратной связи (датчика), выбранных законов управления и режима регулирования привода (VFC, CFC, SERVO, n-control, torque-control, positioning).
2. Выполнение прикладной программы продолжается даже при неисправности преобразователя частоты (при этом возможна обработка ошибок в прикладной программе).
3. В одной прикладной программе может выполняться до 3-х задач параллельно и независимо друг от друга.
5. Развитая система прерываний.
6. Прикладные программы MOVIDRIVE®B могут содержать до 3000-4000 команд ассемблера.
7. Имеется 1024 32-битовых переменных, 128 из них сохраняются в энергонезависимой памяти.
8. Обработка двоичных и аналоговых входных/выходных сигналов.
9. Различные способы обмена данными:
 - системная шина (S-Bus);
 - RS-485;

– через модули сетевых интерфейсов.

10. Имеется возможность считывать и изменять все параметры преобразователя из программы IPOsplus (используя команды обмена данными).

Программы IPOsplus можно создавать с помощью ассемблера или компилятора С. Запуск их редакторов возможен из окна MT-MANAGER (Рис. 1).

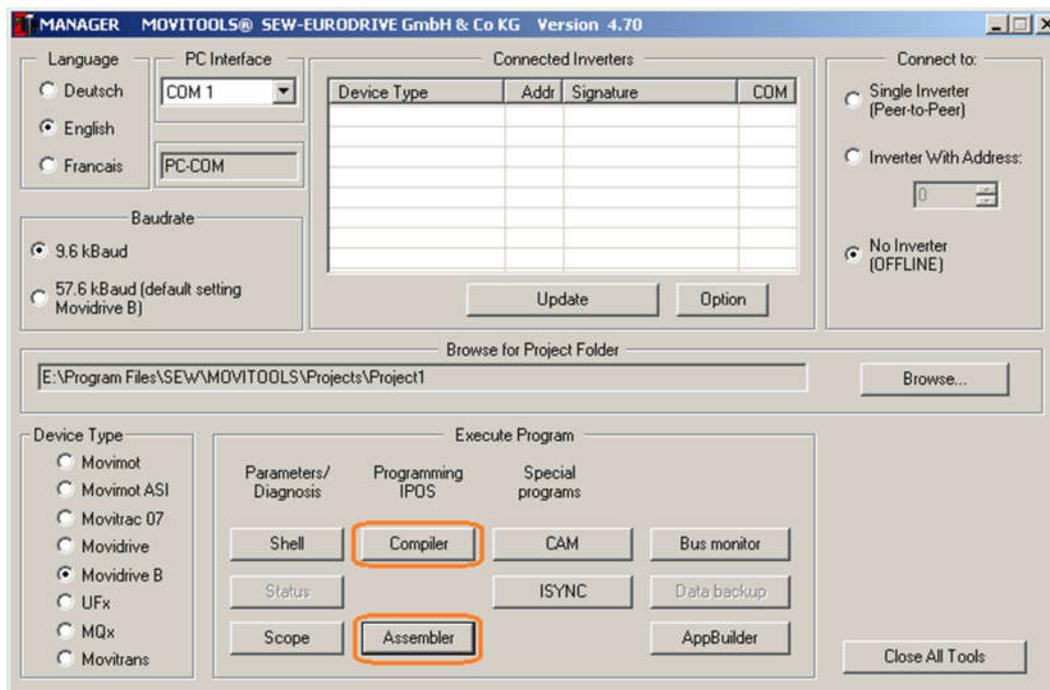
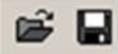


Рис. 1 окно MT-MANAGER

В данном пособии рассматривается только программирование на ассемблере (Рис. 2). Студенты, освоившие принципы программирования на языке ассемблер могут самостоятельно изучить особенности работы компилятора С для IPOsplus и выполнить курсовой проект с использованием этого языка [1].

В панели инструментов окна пользовательского интерфейса редактора ASSEMBLER (Рис. 2) первые три окошечка отображают статус выполнения задач. Они могут быть в четырех состояниях:

- START – задача выполняется;
- PSTOP – задача не запущена;
- BREAK – выполняется до отмеченной строки (при отладке);
- STEP – пошаговое выполнение (при отладке).

Кнопки  - позволяют загружать и сохранять файл созданной программы. А кнопки  - применяются соответственно для компиляции, компиляции и загрузки программы из окна в контроллер привода, выгрузки программы из контроллера и сравнения на идентичность программы в контроллере и окне пользовательского интерфейса. Те же функции доступны через пункт меню «Program» (Рис. 3). Запуск и остановка

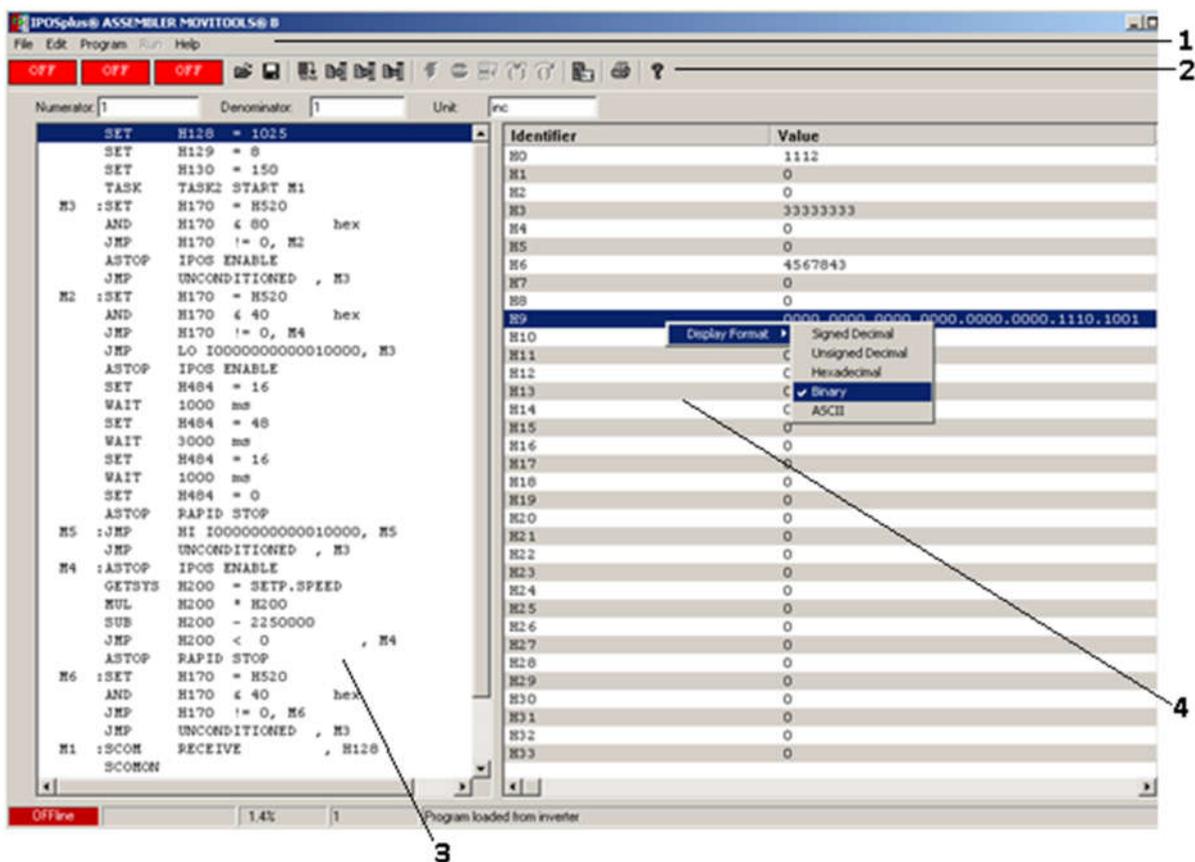


Рис. 2 окно пользовательского интерфейса редактора ASSEMBLER:

1 – строка меню, 2 – панель инструментов, 3 – окно программ, 4 – окно переменных.

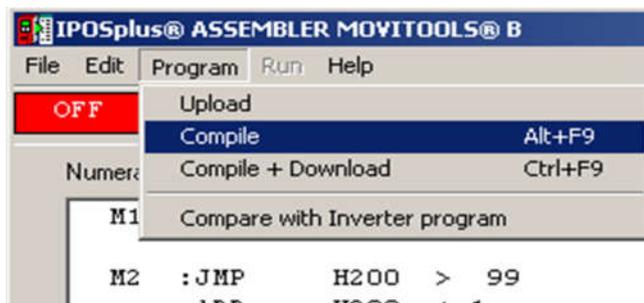


Рис. 3

выполнения программы осуществляется с помощью управляющих элементов



или через пункт меню «Run» (Рис. 4). Отладка фрагментов программы и пошаговая осуществляются с использованием клавиш



(эти же функции имеются в разделе меню «Run»). С помощью



кнопки - вызывается окно поддержки ввода, с помощью которого команды ассемблера вставляются в программу (Рис. 5).



Рис. 4

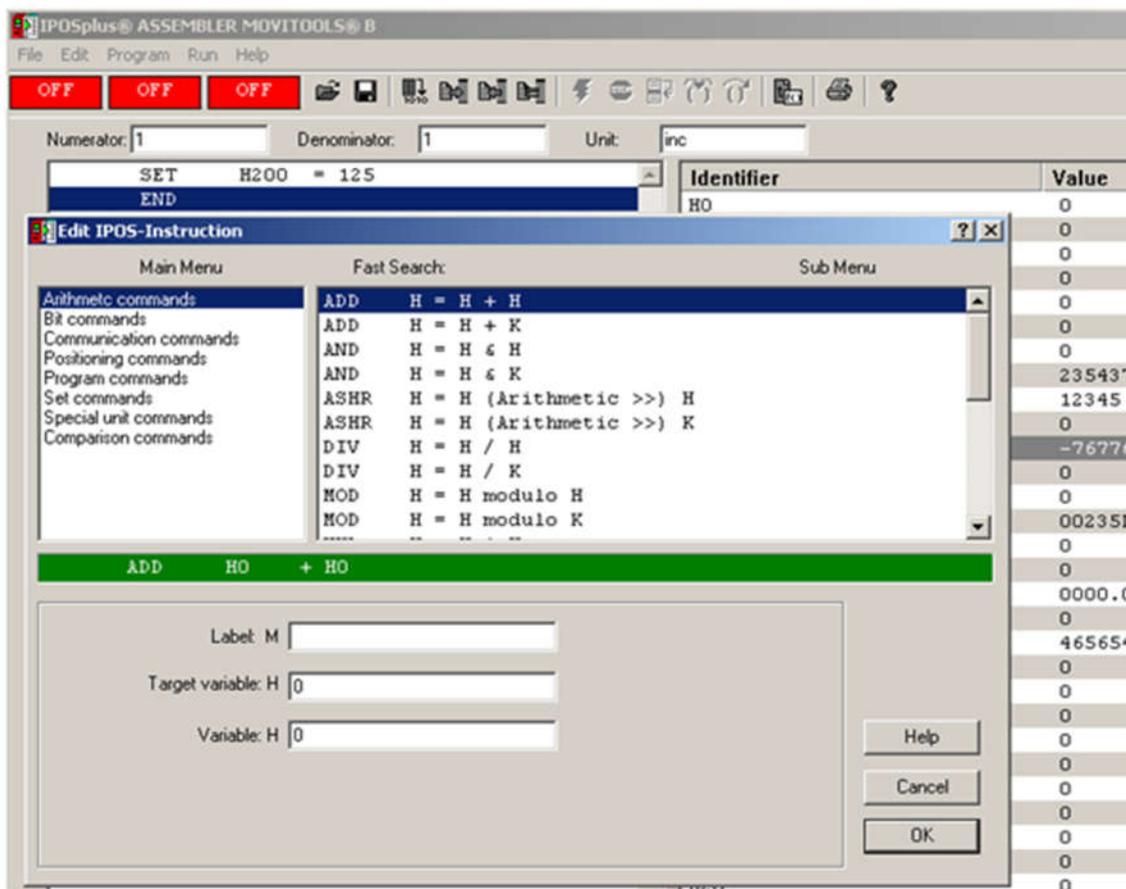


Рис. 5 окно поддержки ввода команд.

В окне поддержки ввода содержатся все возможные для использования команды, сгруппированные в 8 разделов (Main Menu). Выбрав раздел, в нем можно выделить нужную команду (Sub Menu), ввести аргументы для выбранной функции. После ввода всех аргументов и нажатия кнопки [OK], команда будет вставлена в программу над выделенной в ней строкой.

1.2. IPOS переменные.

Контроллер IPOSplus использует только глобальные переменные, общие для всех задач и прерываний. Локальных переменных, объявляемых в какой-либо одной задаче или функции, не предусмотрено. Переменные являются 32-битовыми и в операциях вычисления и сравнения обрабатываются как целые числа со знаком. В контроллере MOVIDRIVE®В пользователю доступно 1024 переменных (H0-H1023), из них 128 переменных (H0-H127) - сохраняются в энергонезависимой памяти. При создании программ необходимо помнить, что энергонезависимая память имеет ограниченный ресурс записи (200 000 раз). Поэтому при работе с переменными в энергонезависимой памяти **нельзя использовать циклическую запись.**

Переменные в окне редактора могут быть представлены в следующих формах: целые со знаком, целые без знака, шестнадцатеричные, двоичные, кодом ASCII. Выбор формы представления осуществляется по щелчку правой клавиши мыши (Рис. 6).

Identifier	Value
H0	0
H1	0
H2	0
H3	0
H4	0
H5	0
H6	0
H7	235437654
H8	12345
H9	0
H10	-767768
H11	0
H12	0
H13	00235D82 h
H14	0
H15	0
H16	0000.0011.1101.1001.0011.1010.0100.0100
H17	0
H18	465654
H19	0
H20	0
H21	0
H22	0
H23	0
H24	0

Display Format ▾

- Signed Decimal
- Unsigned Decimal
- Hexadecimal
- Binary
- ASCII

Рис. 6 выбор формы представления значений переменных.

Некоторым IPOS-переменным присвоены фиксированные внутренние функции, такие переменные называются системными. Для MOVIDRIVE® В системными являются IPOS-переменные **H453...H560**. Переменные указанного диапазона, не имеющие конкретного назначения, зарезервированы для внутренних функций и их нельзя использовать как свободные переменные для прикладной программы. В окне переменных возле каждой системной переменной справа от ее номера имеется сокращенное название, раскрывающее назначение этой переменной. Выделив интересующую системную переменную и нажав функциональную клавишу F1, можно получить краткое описание для этой переменной (Рис. 7). Для более подробного описания следует обратиться к руководству [1].

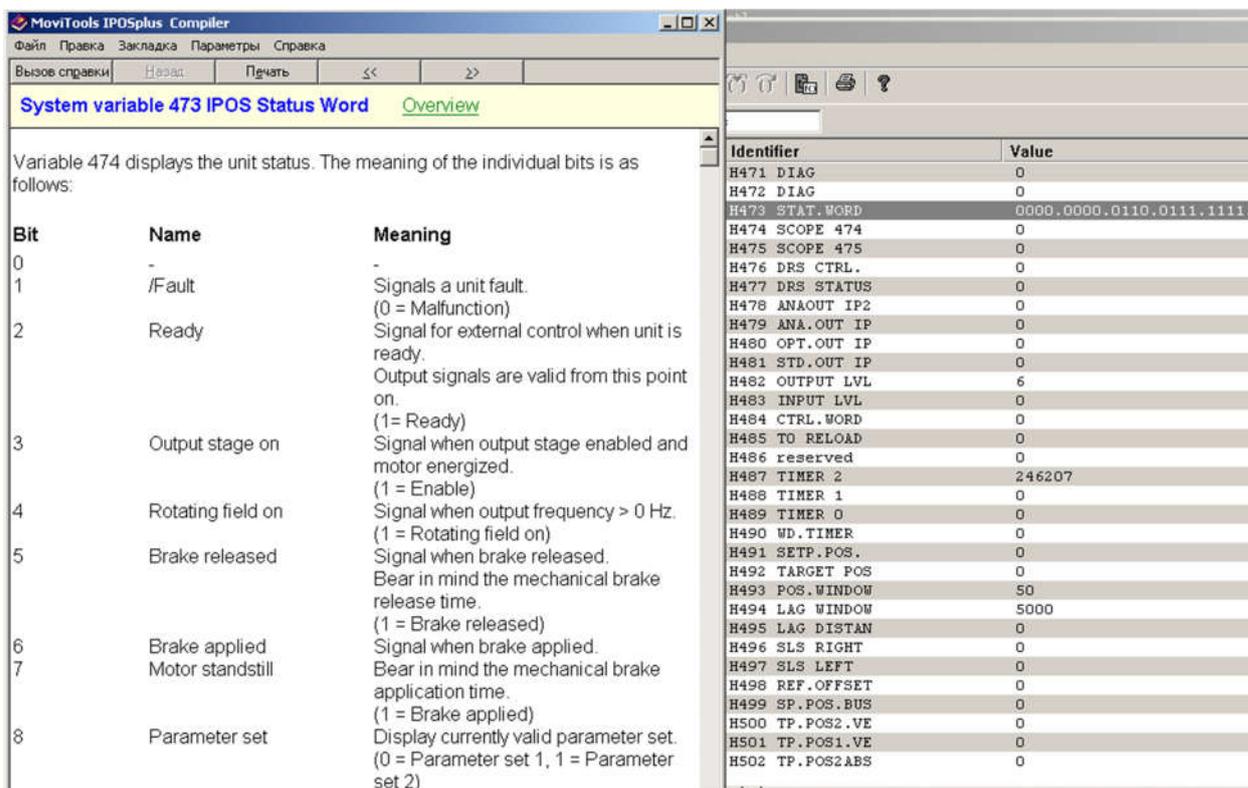


Рис. 7 Вызов справки для системных переменных.

Рассмотрим более подробно некоторые системные переменные. Начнем со слова состояния преобразователя частоты – переменной H473. Каждый бит этой переменной отображает один из параметров преобразователя описывающих его состояние (см. Таблица 1).

Таблица 1

№	Имя в компиляторе / ассемблере	Описание																																																								
473	StatusWord / STAT.WORD	Через это слово состояния возможен запрос режима работы преобразователя.																																																								
		<table border="1"> <thead> <tr> <th>Бит</th> <th>Функция при уровне "1"</th> <th>Бит</th> <th>Функция при уровне "1"</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Нет функции</td> <td>13</td> <td>Сигнал I макс (P442)</td> </tr> <tr> <td>1</td> <td>/Неисправность</td> <td>14</td> <td>/Степень использ.двигателя 1</td> </tr> <tr> <td>2</td> <td>Готов к работе</td> <td>15</td> <td>/Степень использ.двигателя 2</td> </tr> <tr> <td>3</td> <td>Вых. каскад ВКЛ</td> <td>16</td> <td>/DRS: предупреждение</td> </tr> <tr> <td>4</td> <td>Вращ. поле ВКЛ</td> <td>17</td> <td>/DRS: погрешность запаздывания</td> </tr> <tr> <td>5</td> <td>Тормоз отпущен</td> <td>18</td> <td>DRS: ведомый в позиции</td> </tr> <tr> <td>6</td> <td>Тормоз наложен</td> <td>19</td> <td>IPOS: кон.положение достигнуто (см. H493)</td> </tr> <tr> <td>7</td> <td>Двигатель остановлен (начиная с n < 20 об/мин)</td> <td>20</td> <td>IPOS: 0-позиция определена</td> </tr> <tr> <td>8</td> <td>Набор параметров</td> <td>21</td> <td>Резервный</td> </tr> <tr> <td>9</td> <td>Опорн.сигнал частоты вращ. (P40x)</td> <td>22</td> <td>/IPOS: неисправность</td> </tr> <tr> <td>10</td> <td>Сигнал о входе в част.окно (P41x)</td> <td>23..31</td> <td>Резервные</td> </tr> <tr> <td>11</td> <td>Сравнение уставки/действ.знач. (P42x)</td> <td></td> <td></td> </tr> <tr> <td>12</td> <td>Опорн.сигнал тока (P43x)</td> <td></td> <td></td> </tr> </tbody> </table>	Бит	Функция при уровне "1"	Бит	Функция при уровне "1"	0	Нет функции	13	Сигнал I макс (P442)	1	/Неисправность	14	/Степень использ.двигателя 1	2	Готов к работе	15	/Степень использ.двигателя 2	3	Вых. каскад ВКЛ	16	/DRS: предупреждение	4	Вращ. поле ВКЛ	17	/DRS: погрешность запаздывания	5	Тормоз отпущен	18	DRS: ведомый в позиции	6	Тормоз наложен	19	IPOS: кон.положение достигнуто (см. H493)	7	Двигатель остановлен (начиная с n < 20 об/мин)	20	IPOS: 0-позиция определена	8	Набор параметров	21	Резервный	9	Опорн.сигнал частоты вращ. (P40x)	22	/IPOS: неисправность	10	Сигнал о входе в част.окно (P41x)	23..31	Резервные	11	Сравнение уставки/действ.знач. (P42x)			12	Опорн.сигнал тока (P43x)		
Бит	Функция при уровне "1"	Бит	Функция при уровне "1"																																																							
0	Нет функции	13	Сигнал I макс (P442)																																																							
1	/Неисправность	14	/Степень использ.двигателя 1																																																							
2	Готов к работе	15	/Степень использ.двигателя 2																																																							
3	Вых. каскад ВКЛ	16	/DRS: предупреждение																																																							
4	Вращ. поле ВКЛ	17	/DRS: погрешность запаздывания																																																							
5	Тормоз отпущен	18	DRS: ведомый в позиции																																																							
6	Тормоз наложен	19	IPOS: кон.положение достигнуто (см. H493)																																																							
7	Двигатель остановлен (начиная с n < 20 об/мин)	20	IPOS: 0-позиция определена																																																							
8	Набор параметров	21	Резервный																																																							
9	Опорн.сигнал частоты вращ. (P40x)	22	/IPOS: неисправность																																																							
10	Сигнал о входе в част.окно (P41x)	23..31	Резервные																																																							
11	Сравнение уставки/действ.знач. (P42x)																																																									
12	Опорн.сигнал тока (P43x)																																																									
		 Выход IPOS In Position устанавливается на "1" также при отмене разрешения и при блокировке регулятора.																																																								

Переменная H484 представляет собой управляющее слово. Изменяя значения отдельных битов этой переменной можно управлять работой преобразователя частоты (Таблица 2).

№	Имя в компиляторе / ассемблере	Описание																																																																				
484	ControlWord / CTRL.WORD	<p>Управляющее слово IPOS^{plus}® (СЧИТЫВАНИЕ и УПРАВЛЕНИЕ функциями преобразователя). Управляющее слово IPOS^{plus}® можно использовать всегда, независимо от режима работы, источника управляющего сигнала и источника уставки. В устройстве это управляющее слово IPOS^{plus}® связано с функциями клемм и управляющим словом шины (сетевой, RS-485/RS-232 и SBus) через логическую операцию "ИЛИ".</p> <table border="1"> <thead> <tr> <th>Бит</th> <th>Функция при уровне "1"</th> <th>Бит</th> <th>Функция при уровне "1"</th> </tr> </thead> <tbody> <tr><td>0</td><td>Нет функции</td><td>16</td><td>Резервные</td></tr> <tr><td>1</td><td>Нет разрешения</td><td>17</td><td>Кулачковый датчик 0-поз.</td></tr> <tr><td>2</td><td>Направо</td><td>18</td><td>Запуск выхода в 0-поз.</td></tr> <tr><td>3</td><td>Налево</td><td>19</td><td>Ведомый автономен</td></tr> <tr><td>4</td><td>n11/n21 (фиксир.уставка 1)</td><td>20</td><td>Блокировка уставки</td></tr> <tr><td>5</td><td>n12/n22 (фиксир.уставка 2)</td><td>21</td><td>Резервный</td></tr> <tr><td>6</td><td>Выбор набора фикс.уставок</td><td>22</td><td>DRS: установка в начало отсчета</td></tr> <tr><td>7</td><td>Выбор набора парам. (набор парам.2)</td><td>23</td><td>DRS: запуск ведомого</td></tr> <tr><td>8</td><td>Выбор набора ген.темпа (набор ген.темпа 2)</td><td>24</td><td>DRS: обучение</td></tr> <tr><td>9</td><td>Внутр.задатчик: разгон</td><td>25</td><td>Резервный</td></tr> <tr><td>10</td><td>Внутр.задатчик: замедление</td><td>26</td><td>Резервный</td></tr> <tr><td>11</td><td>Внешняя ошибка</td><td>27</td><td>Резервный</td></tr> <tr><td>12</td><td>Сброс сообщения об ошибке</td><td>28</td><td>Резервный</td></tr> <tr><td>13</td><td>Управление удержанием</td><td>29</td><td>Резервный</td></tr> <tr><td>14</td><td>Правый KB</td><td>30</td><td>Блокировка регулятора</td></tr> <tr><td>15</td><td>Левый KB</td><td>31</td><td>Резервный</td></tr> </tbody> </table>	Бит	Функция при уровне "1"	Бит	Функция при уровне "1"	0	Нет функции	16	Резервные	1	Нет разрешения	17	Кулачковый датчик 0-поз.	2	Направо	18	Запуск выхода в 0-поз.	3	Налево	19	Ведомый автономен	4	n11/n21 (фиксир.уставка 1)	20	Блокировка уставки	5	n12/n22 (фиксир.уставка 2)	21	Резервный	6	Выбор набора фикс.уставок	22	DRS: установка в начало отсчета	7	Выбор набора парам. (набор парам.2)	23	DRS: запуск ведомого	8	Выбор набора ген.темпа (набор ген.темпа 2)	24	DRS: обучение	9	Внутр.задатчик: разгон	25	Резервный	10	Внутр.задатчик: замедление	26	Резервный	11	Внешняя ошибка	27	Резервный	12	Сброс сообщения об ошибке	28	Резервный	13	Управление удержанием	29	Резервный	14	Правый KB	30	Блокировка регулятора	15	Левый KB	31	Резервный
Бит	Функция при уровне "1"	Бит	Функция при уровне "1"																																																																			
0	Нет функции	16	Резервные																																																																			
1	Нет разрешения	17	Кулачковый датчик 0-поз.																																																																			
2	Направо	18	Запуск выхода в 0-поз.																																																																			
3	Налево	19	Ведомый автономен																																																																			
4	n11/n21 (фиксир.уставка 1)	20	Блокировка уставки																																																																			
5	n12/n22 (фиксир.уставка 2)	21	Резервный																																																																			
6	Выбор набора фикс.уставок	22	DRS: установка в начало отсчета																																																																			
7	Выбор набора парам. (набор парам.2)	23	DRS: запуск ведомого																																																																			
8	Выбор набора ген.темпа (набор ген.темпа 2)	24	DRS: обучение																																																																			
9	Внутр.задатчик: разгон	25	Резервный																																																																			
10	Внутр.задатчик: замедление	26	Резервный																																																																			
11	Внешняя ошибка	27	Резервный																																																																			
12	Сброс сообщения об ошибке	28	Резервный																																																																			
13	Управление удержанием	29	Резервный																																																																			
14	Правый KB	30	Блокировка регулятора																																																																			
15	Левый KB	31	Резервный																																																																			

Используя переменную H520 можно считывать уровни сигналов поступающих на двоичные управляющие входы преобразователя (Таблица 3). Каждому входу соответствует свой бит этой переменной. Кроме того, через контроллер IPOS^{plus} можно управлять состоянием двоичных выходов. Делается это через установку/сброс битов переменных H480 (выходы платы расширения) и H481 (выходы базового блока) – см. Таблица 4. Управление двоичными выходами из программы IPOS^{plus} возможно только в том случае,

№	Имя в компиляторе / ассемблере	Описание							
520	InpLevelB / INPUTLVLB MOVIDRIVE® B (MOVIDRIVE® A: H483)	Только СЧИТЫВАНИЕ уровня сигналов двоичных входов.							
		Бит	IPOS-имя	DIO (+ DIO)	DIO + DIP	DIO + сетевая шина	DIP	DIP + сетевая шина	Сетевая шина P870 = Control word 2
		0	DI00	В зависимости от базового блока, например X13:1					
		1	DI01	X13:2					
		2	DI02	X13:3					
		3	DI03	X13:4					
		4	DI04	X13:5					
		5	DI05	X13:6					
		6	DI06	X16:1					
		7	DI07	X16:2					
		8	DI10	X22:1	X22:1	X22:1	X60:1	X60:1	Бит 8
		9	DI11	X22:2	X22:2	X22:2	X60:2	X60:2	Бит 9
		10	DI12	X22:3	X22:3	X22:3	X60:3	X60:3	Бит 10
		11	DI13	X22:4	X22:4	X22:4	X60:4	X60:4	Бит 11
		12	DI14	X22:5	X22:5	X22:5	X60:5	X60:5	Бит 12
		13	DI15	X22:6	X22:6	X22:6	X60:6	X60:6	Бит 13
		14	DI16	X22:7	X22:7	X22:7	X60:7	X60:7	Бит 14
		15	DI17	X22:8	X22:8	X22:8	X60:8	X60:8	Бит 15
		16					X60:1		
		17		(X22:1)	X60:2				
		18		...	X60:3				
		19		(X22:8)	X60:4				
		20			X60:5				
		21			X60:6				
		22			X60:7				
		23			X60:8				

если с помощью программы SHELL на нужные выходы установлена функция IPOS OUTPUT (Рис. 8).

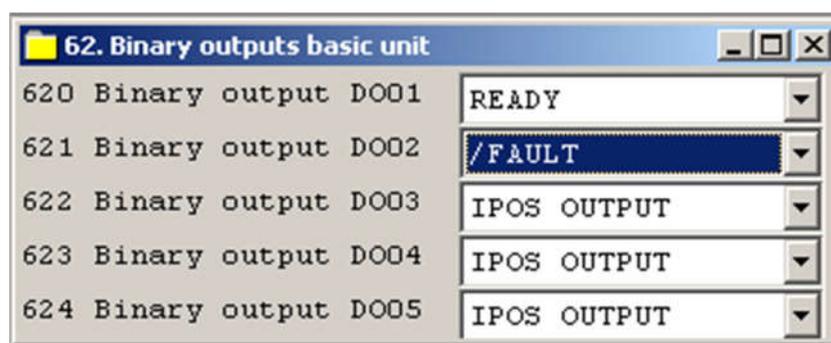


Рис. 8 Установка функции IPOS OUTPUT на дискретные выходы

При написании программы можно использовать три таймера, доступ к текущим значениям которых осуществляется через переменные H487-H489. Переменная H485 позволяет организовать циклическую работу таймера TIMERO с заданным периодом (Таблица 5).

Таблица 4

№	Имя в компиляторе / ассемблере	Описание																																																																																																																																																
480	OptOutpIPOS / OPT.OUT IP	<p>Только УПРАВЛЕНИЕ двоичными выходами устройства расширения входов-выходов DIO11 / DIP11. СЧИТЫВАНИЕ возможно: на MOVIDRIVE® A – через H482; на MOVIDRIVE® B – через H521. Если ни одна из опций DIO11 и DIP11 не установлена, то с помощью этой переменной можно использовать виртуальные клеммы сетевой шины в слове состояния 2 (если, например, P873 = STATUS WORD 2). Биты переменной H480 отображаются на двоичные выходы базового блока, если соответствующие клеммы запрограммированы на функцию IPOS OUTPUT. Если в программе IPOS переключается какой-либо двоичный выход, то соответствующий физический выход (клемма, запрограммированная как IPOS-выход) переключается с задержкой в 1 мс.</p> <table border="1"> <thead> <tr> <th>Бит</th> <th>IPOS-имя</th> <th>DIO (+ DIO)</th> <th>DIO + DIP</th> <th>DIO + сетевая шина</th> <th>DIP</th> <th>DIP + сетевая шина</th> <th>Сетевая шина</th> </tr> </thead> <tbody> <tr> <td colspan="2"></td> <td colspan="3">P63x воздействует на DIO</td> <td colspan="2">P63x воздействует на DIP</td> <td>P873 = Status word 2</td> </tr> <tr> <td>0</td> <td>DO10</td> <td>X23:1</td> <td>X23:1</td> <td>X23:1</td> <td>X61:1</td> <td>X61:1</td> <td>Бит 8</td> </tr> <tr> <td>1</td> <td>DO11</td> <td>X23:2</td> <td>X23:2</td> <td>X23:2</td> <td>X61:2</td> <td>X61:2</td> <td>Бит 9</td> </tr> <tr> <td>2</td> <td>DO12</td> <td>X23:3</td> <td>X23:3</td> <td>X23:3</td> <td>X61:3</td> <td>X61:3</td> <td>Бит 10</td> </tr> <tr> <td>3</td> <td>DO13</td> <td>X23:4</td> <td>X23:4</td> <td>X23:4</td> <td>X61:4</td> <td>X61:4</td> <td>Бит 11</td> </tr> <tr> <td>4</td> <td>DO14</td> <td>X23:5</td> <td>X23:5</td> <td>X23:5</td> <td>X61:5</td> <td>X61:5</td> <td>Бит 12</td> </tr> <tr> <td>5</td> <td>DO15</td> <td>X23:6</td> <td>X23:6</td> <td>X23:6</td> <td>X61:6</td> <td>X61:6</td> <td>Бит 13</td> </tr> <tr> <td>6</td> <td>DO16</td> <td>X23:7</td> <td>X23:7</td> <td>X23:7</td> <td>X61:7</td> <td>X61:7</td> <td>Бит 14</td> </tr> <tr> <td>7</td> <td>DO17</td> <td>X23:8</td> <td>X23:8</td> <td>X23:8</td> <td>X61:8</td> <td>X61:8</td> <td>Бит 15</td> </tr> <tr> <td>8</td> <td></td> <td></td> <td></td> <td>X61:1</td> <td></td> <td></td> <td></td> </tr> <tr> <td>9</td> <td></td> <td>(X23:1)</td> <td></td> <td>X61:2</td> <td></td> <td></td> <td></td> </tr> <tr> <td>10</td> <td></td> <td>...</td> <td></td> <td>X61:3</td> <td></td> <td></td> <td></td> </tr> <tr> <td>11</td> <td></td> <td>(X23:8)</td> <td></td> <td>X61:4</td> <td></td> <td></td> <td></td> </tr> <tr> <td>12</td> <td></td> <td></td> <td></td> <td>X61:5</td> <td></td> <td></td> <td></td> </tr> <tr> <td>13</td> <td></td> <td></td> <td></td> <td>X61:6</td> <td></td> <td></td> <td></td> </tr> <tr> <td>14</td> <td></td> <td></td> <td></td> <td>X61:7</td> <td></td> <td></td> <td></td> </tr> <tr> <td>15</td> <td></td> <td></td> <td></td> <td>X61:8</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Бит	IPOS-имя	DIO (+ DIO)	DIO + DIP	DIO + сетевая шина	DIP	DIP + сетевая шина	Сетевая шина			P63x воздействует на DIO			P63x воздействует на DIP		P873 = Status word 2	0	DO10	X23:1	X23:1	X23:1	X61:1	X61:1	Бит 8	1	DO11	X23:2	X23:2	X23:2	X61:2	X61:2	Бит 9	2	DO12	X23:3	X23:3	X23:3	X61:3	X61:3	Бит 10	3	DO13	X23:4	X23:4	X23:4	X61:4	X61:4	Бит 11	4	DO14	X23:5	X23:5	X23:5	X61:5	X61:5	Бит 12	5	DO15	X23:6	X23:6	X23:6	X61:6	X61:6	Бит 13	6	DO16	X23:7	X23:7	X23:7	X61:7	X61:7	Бит 14	7	DO17	X23:8	X23:8	X23:8	X61:8	X61:8	Бит 15	8				X61:1				9		(X23:1)		X61:2				10		...		X61:3				11		(X23:8)		X61:4				12				X61:5				13				X61:6				14				X61:7				15				X61:8			
Бит	IPOS-имя	DIO (+ DIO)	DIO + DIP	DIO + сетевая шина	DIP	DIP + сетевая шина	Сетевая шина																																																																																																																																											
		P63x воздействует на DIO			P63x воздействует на DIP		P873 = Status word 2																																																																																																																																											
0	DO10	X23:1	X23:1	X23:1	X61:1	X61:1	Бит 8																																																																																																																																											
1	DO11	X23:2	X23:2	X23:2	X61:2	X61:2	Бит 9																																																																																																																																											
2	DO12	X23:3	X23:3	X23:3	X61:3	X61:3	Бит 10																																																																																																																																											
3	DO13	X23:4	X23:4	X23:4	X61:4	X61:4	Бит 11																																																																																																																																											
4	DO14	X23:5	X23:5	X23:5	X61:5	X61:5	Бит 12																																																																																																																																											
5	DO15	X23:6	X23:6	X23:6	X61:6	X61:6	Бит 13																																																																																																																																											
6	DO16	X23:7	X23:7	X23:7	X61:7	X61:7	Бит 14																																																																																																																																											
7	DO17	X23:8	X23:8	X23:8	X61:8	X61:8	Бит 15																																																																																																																																											
8				X61:1																																																																																																																																														
9		(X23:1)		X61:2																																																																																																																																														
10		...		X61:3																																																																																																																																														
11		(X23:8)		X61:4																																																																																																																																														
12				X61:5																																																																																																																																														
13				X61:6																																																																																																																																														
14				X61:7																																																																																																																																														
15				X61:8																																																																																																																																														
481	StdOutpIPOS / STD.OUT IP	<p>Только УПРАВЛЕНИЕ двоичными выходами базового блока. Если в программе IPOS переключается какой-либо двоичный выход, то соответствующий физический выход (клемма, запрограммированная как IPOS-выход) переключается с задержкой в 1 мс.</p> <table border="1"> <thead> <tr> <th>Бит</th> <th>IPOS-имя</th> <th></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>DB00</td> <td>непрограммируемый выход, фиксированное назначение: "Тормоз"</td> </tr> <tr> <td>1</td> <td>DO01</td> <td>если P620 = IPOS OUTPUT</td> </tr> <tr> <td>2</td> <td>DO02</td> <td>если P621 = IPOS OUTPUT</td> </tr> <tr> <td>3</td> <td>DO03</td> <td>только на MOVIDRIVE® B, если P622 = IPOS OUTPUT</td> </tr> <tr> <td>4</td> <td>DO04</td> <td>только на MOVIDRIVE® B, если P623 = IPOS OUTPUT</td> </tr> <tr> <td>5</td> <td>DO05</td> <td>только на MOVIDRIVE® B, если P624 = IPOS OUTPUT</td> </tr> </tbody> </table>	Бит	IPOS-имя		0	DB00	непрограммируемый выход, фиксированное назначение: "Тормоз"	1	DO01	если P620 = IPOS OUTPUT	2	DO02	если P621 = IPOS OUTPUT	3	DO03	только на MOVIDRIVE® B, если P622 = IPOS OUTPUT	4	DO04	только на MOVIDRIVE® B, если P623 = IPOS OUTPUT	5	DO05	только на MOVIDRIVE® B, если P624 = IPOS OUTPUT																																																																																																																											
Бит	IPOS-имя																																																																																																																																																	
0	DB00	непрограммируемый выход, фиксированное назначение: "Тормоз"																																																																																																																																																
1	DO01	если P620 = IPOS OUTPUT																																																																																																																																																
2	DO02	если P621 = IPOS OUTPUT																																																																																																																																																
3	DO03	только на MOVIDRIVE® B, если P622 = IPOS OUTPUT																																																																																																																																																
4	DO04	только на MOVIDRIVE® B, если P623 = IPOS OUTPUT																																																																																																																																																
5	DO05	только на MOVIDRIVE® B, если P624 = IPOS OUTPUT																																																																																																																																																

Таблица 5

№	Имя в компиляторе / ассемблере	Описание
485	T0_Reload / T0 RELOAD	СЧИТЫВАНИЕ и УСТАНОВКА времени цикла для пользовательского таймера 0. Если с командой SET INTERRUPT (SETINT) пользовательский таймер (TIMER0 (H489)) нужно использовать циклически, то в H485 можно указать время цикла. Каждый раз, когда таймер 0 заканчивает отсчет (H489 = 0), в него снова загружается значение времени из переменной H485. Диапазон значений: $0...2^{31} - 1$ мс.
487	Timer_2 / TIMER 2	СЧИТЫВАНИЕ и УСТАНОВКА времени для пользовательского таймера 2. Пользовательский таймер 2 ведет прямой отсчет. Диапазон значений: $0...2^{31} - 1$ мс.
488	Timer_1 / TIMER 1	СЧИТЫВАНИЕ и УСТАНОВКА времени для пользовательского таймера 1. Пользовательский таймер 1 ведет обратный отсчет до 0. Диапазон значений: $0...2^{31} - 1$ мс.
489	Timer_0 / TIMER 0	СЧИТЫВАНИЕ и УСТАНОВКА времени для пользовательского таймера 0. Пользовательский таймер 0 ведет обратный отсчет до 0. При использовании команды SET INTERRUPT (SETINT) в момент достижения этим таймером значения = 0 выполняется переход по прерыванию. Если с командой SET INTERRUPT (SETINT) пользовательский таймер нужно использовать циклически, то в переменной T0 RELOAD (H485) можно указать время цикла. См. гл. "Управление задачами и прерывания". Диапазон значений: $0...2^{31} - 1$ мс.
490	WdogTimer / WD.TIMER	СЧИТЫВАНИЕ и УСТАНОВКА времени для пользовательского контрольного таймера. Этот контрольный таймер ведет обратный отсчет до 0. Команда WATCHDOG ON (WDON) активирует этот таймер и задает время цикла. Диапазон значений: $0...2^{31} - 1$ мс.

В режимах регулирования скорости и момента управление (а также при использовании технологического ПИД-регулятора) уставку задания на привод можно подавать через H524 (Таблица 6).

Таблица 6

№	Имя в компиляторе / ассемблере	Описание
524	IPOS_Setp / IPOS_SETP	IPOS-уставка, управляющее воздействие ПИД-регулятора при H540 = 1. Если H540 = 0 или 2, возможна запись значения прямо из прикладной программы. H524 можно использовать как уставку момента или частоты вращения, если задано P100 "Источник уставки" = IPOS, а P700 "Режим работы 1" = xxx&M-control или CFC или SERVO. В этом случае 1 инкременту в H524 соответствуют 0,2 об/мин уставки частоты вращения или 0,01 % I _{ном} уставки момента.

Для этого параметр P100 Setpoint source должен быть настроен на значение IPOS SETPOINT (Рис. 9). Подробное описание назначения параметров можно найти в [2].

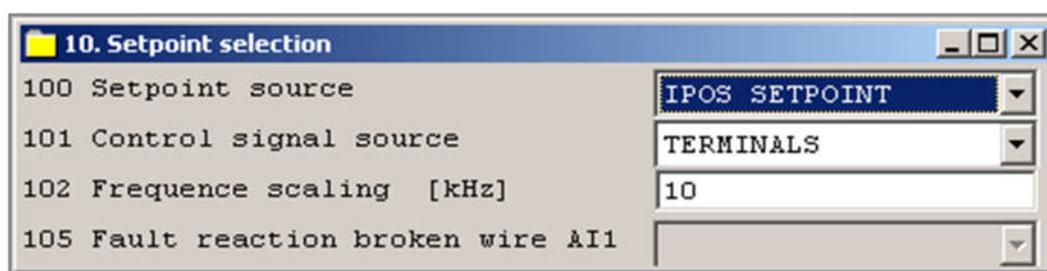


Рис. 9

В данном разделе дано описание не всех системных переменных. Другие системные переменные будут рассматриваться по мере дальнейшего изложения материала.

1.3. Базовые команды IPOSplus.

Начнем с команд работы с битами собранными в группе Bit commands (Рис. 10 - Рис. 12) в окне поддержки ввода команд. Имеются команды установки **BSET** (Рис. 10) и сброса **BCLR** (Рис. 11) определенного бита у требуемой переменной.

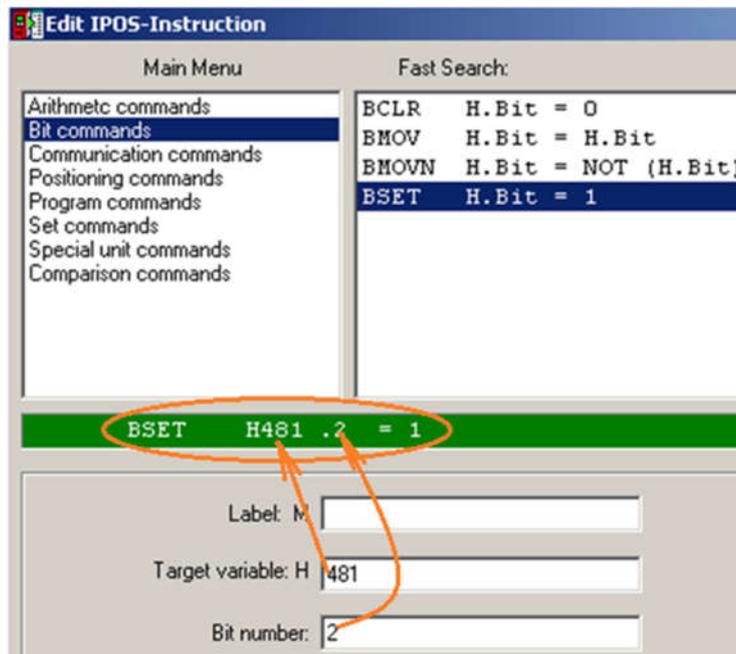


Рис. 10 команда установки бита

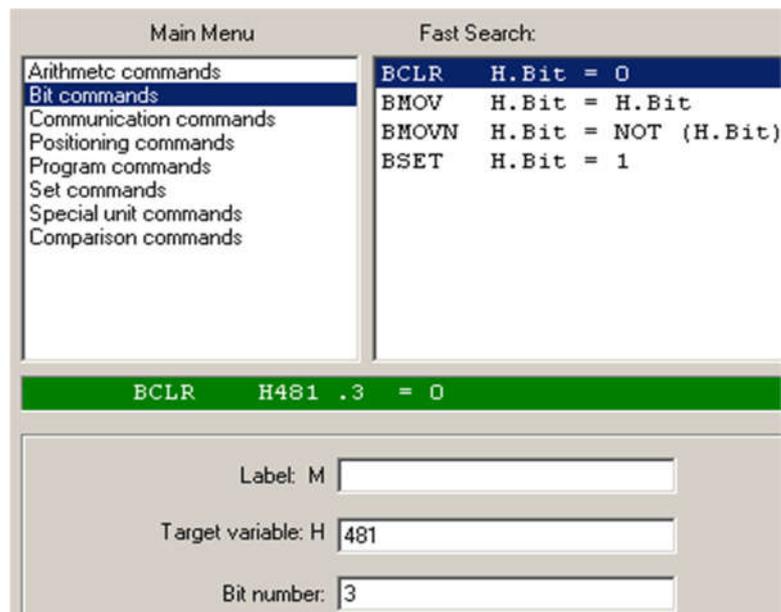


Рис. 11 команда сброса бита

Команда **BMOV** присваивает значение произвольного бита одной переменной биту другой переменной (Рис. 12). А команда **BMOVN** выполняет присвоение проинвертированного значения бита исходной переменной.

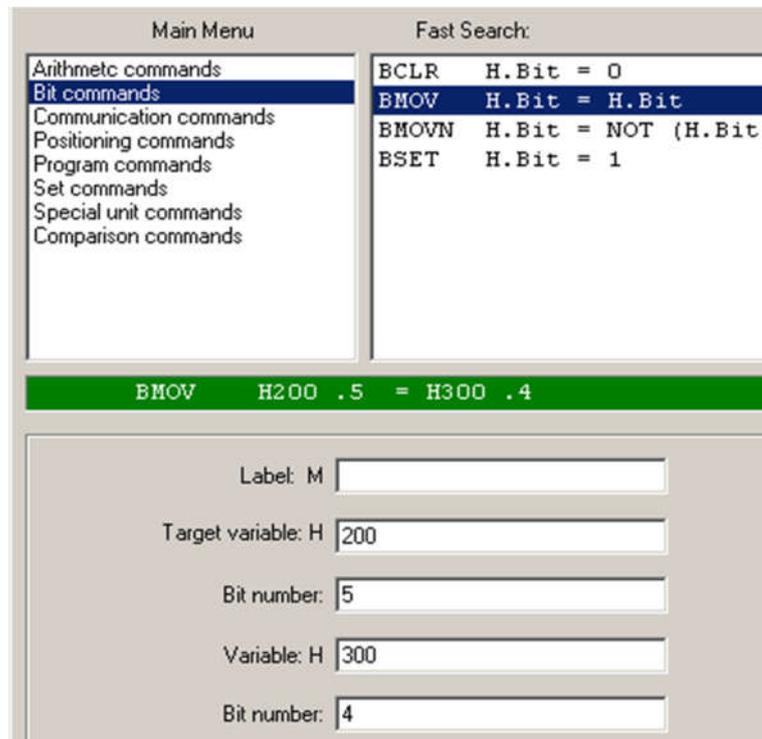
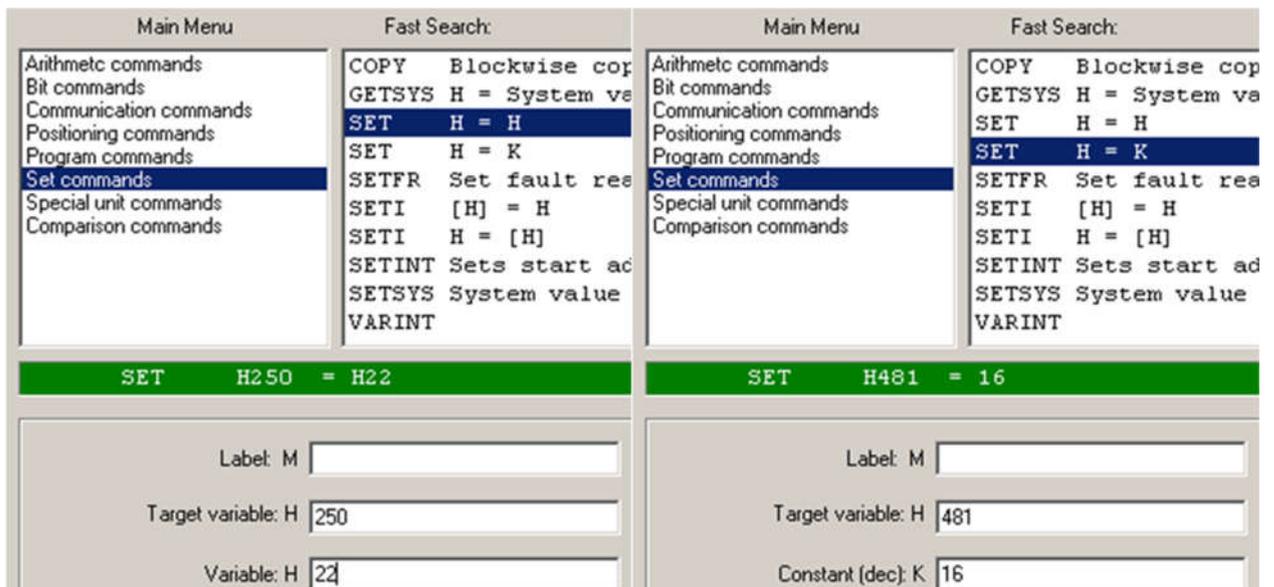


Рис. 12 команда копирования значения бита

В группе Set commands имеются команды **SET** присвоения переменной значения другой переменной или некоторой константы. С помощью этой команды в переменную H (Target variable) загружается содержимое аргумента (другой переменной H Рис. 13а) или константы К (Рис. 13б)). Аргумент после выполнения операции всегда остается неизменным.



а).

б).

Рис. 13 команды присваивания SET

В эту же группу включены команды присвоения с использованием косвенной адресации **SETI [H]=H** (Рис. 14а) и **SETI H = [H]** (Рис. 14б). В первом случае значение исходной переменной присваивается переменной, адрес которой хранится в переменной [H]. Во втором случае в переменную H загружается значение исходной переменной, адрес которой находится в переменной[H].

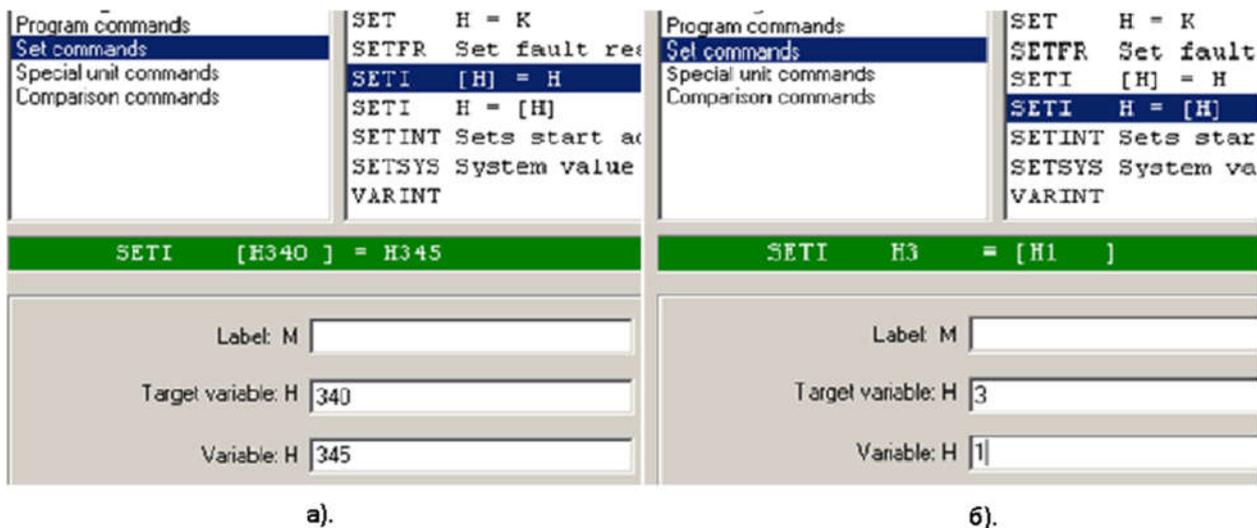


Рис. 14 команды присваивания с косвенной адресацией

Среди Set commands имеются команды для работы с системными переменными **GETSYS** и **SETSYS**. Первая команда позволяет считать значения системных переменных в другие переменные, а вторая – наоборот, записать в системную переменную. Так с помощью команды **GETSYS** можно считывать:

- состояние двоичных (вместо обращения к H520) и аналоговых входов,
 - двоичных (вместо обращения к H521) и аналоговых выходов,
 - текущих значений таймеров (H488-H489),
 - актуальных величин скорости и тока
- и некоторых других о которых будем говорить далее.

В группу Program Commands включены 5 разновидностей команды перехода **JMP**. Общим для них является то, что при их выполнении производится проверка условия и если оно выполняется, то происходит переход на метку указанную в команде. Если условие не выполняется, то выполняется следующая за **JMP** команда. В простых случаях, условие

формулируется как результат сравнения (>, >=, <, <=, ==, !=) значения переменной (Target variable) с другой переменной, константой или нулем. Так на Рис. 15 дан пример, в котором проверяется: будет ли переменная H232 больше переменной H233, при выполнении этого условия осуществляется переход на метку M12.

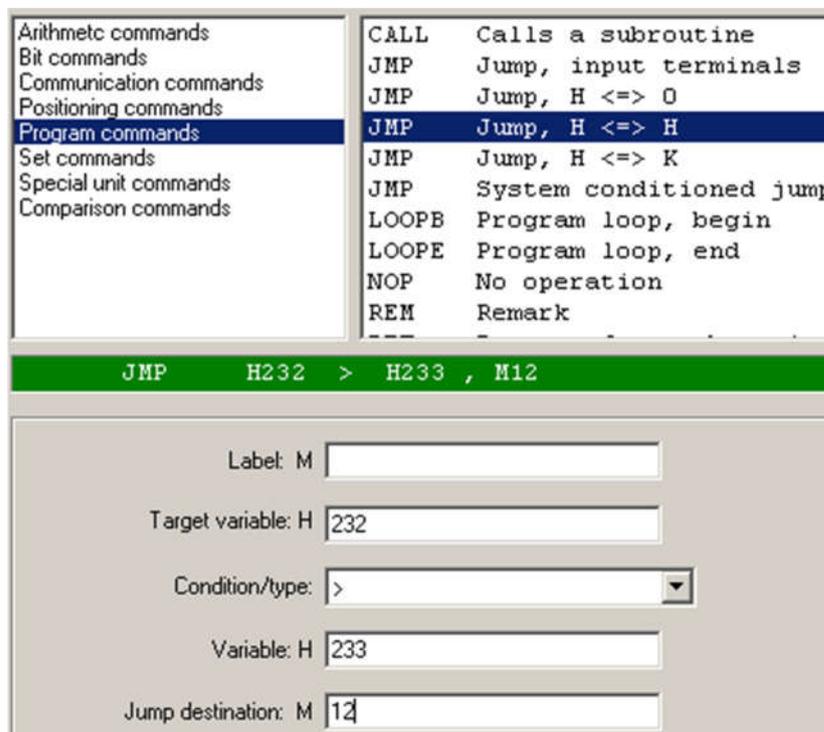


Рис. 15 команда перехода по результату сравнения значений двух переменных

Существует разновидность команды перехода, в которой условие задается в виде маски выбора входных клемм. При выполнении этой команды уровень на выбранных входах проверяется на совпадение с высоким (HI) или низким уровнем (LO), в зависимости от аргумента команды Condition/type (Рис. 16).

Еще одна разновидность команды перехода работает по результату выполнения системного условия (Рис. 17). В качестве таких условий можно выбрать:

- **N == 0** - переход, если частота вращения равна нулю;

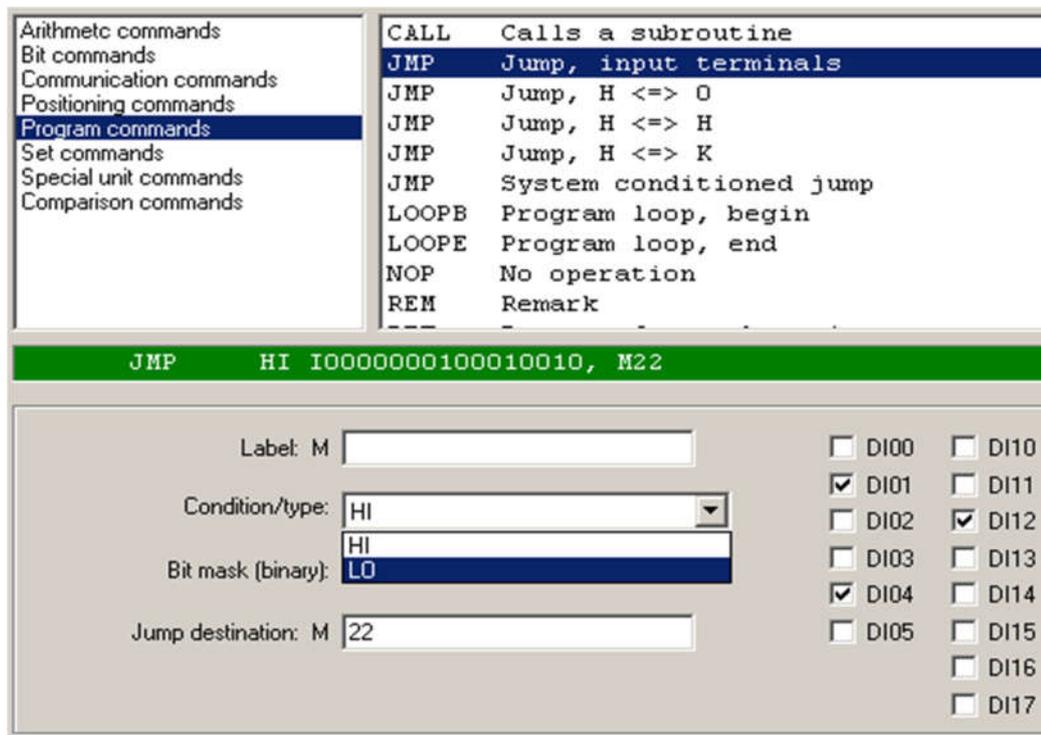


Рис. 16 команда перехода по результату сравнения маски входных клемм с аргументом

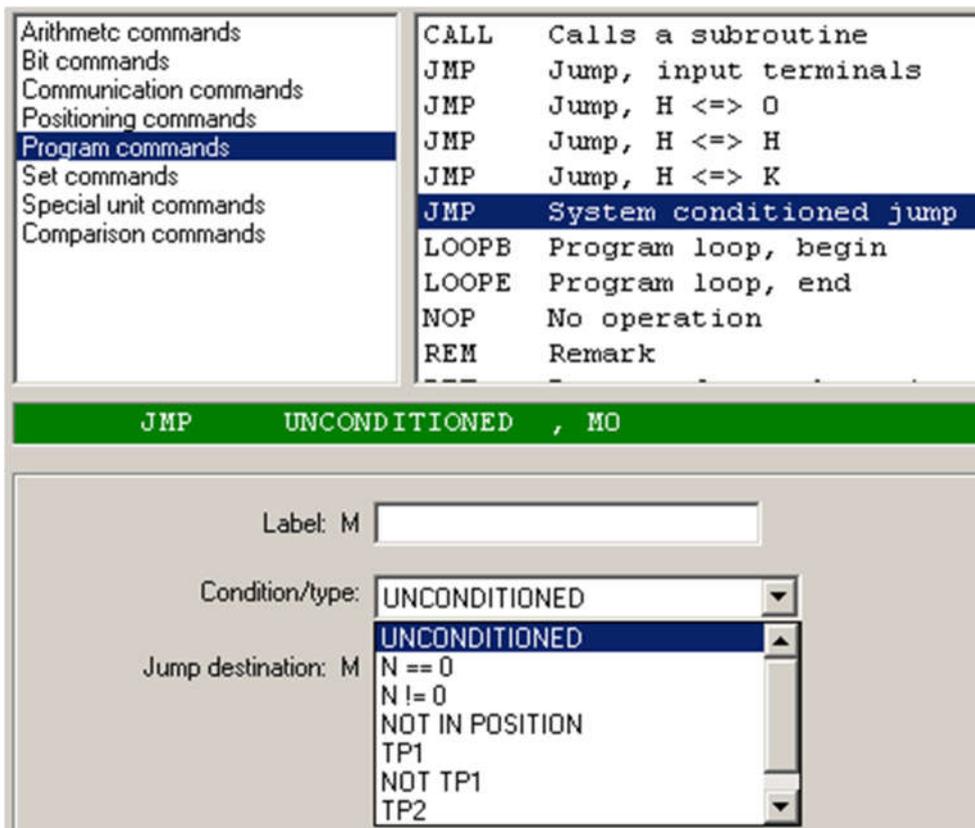


Рис. 17

- **N != 0** - переход, если частота вращения не равна нулю;

- **NOT IN POSITION** - переход, если привод не в конечном положении (см. стр.51);
- **TP1** - переход, если изменяется уровень сигнала на входе обучения DI02;
- **NOT TP1** - переход, если уровень сигнала на входе обучения DI02 не изменяется;
- **TP2** - переход, если изменяется уровень сигнала на входе обучения DI03;
- **NOT TP2** - переход, если уровень сигнала на входе обучения DI03 не изменяется;
- **UNCONDITIONED** - безусловный переход.

В группе Program Commands имеются команды организации цикла: **LOOPB** и **LOOPE** и команды перехода к подпрограмме и возврата из нее: **CALL** и **RET**.

Команда **WAIT** задерживает выполнение программы на время, указанное в аргументе в мс. По истечении этого времени выполнение программы продолжается (Рис. 18).

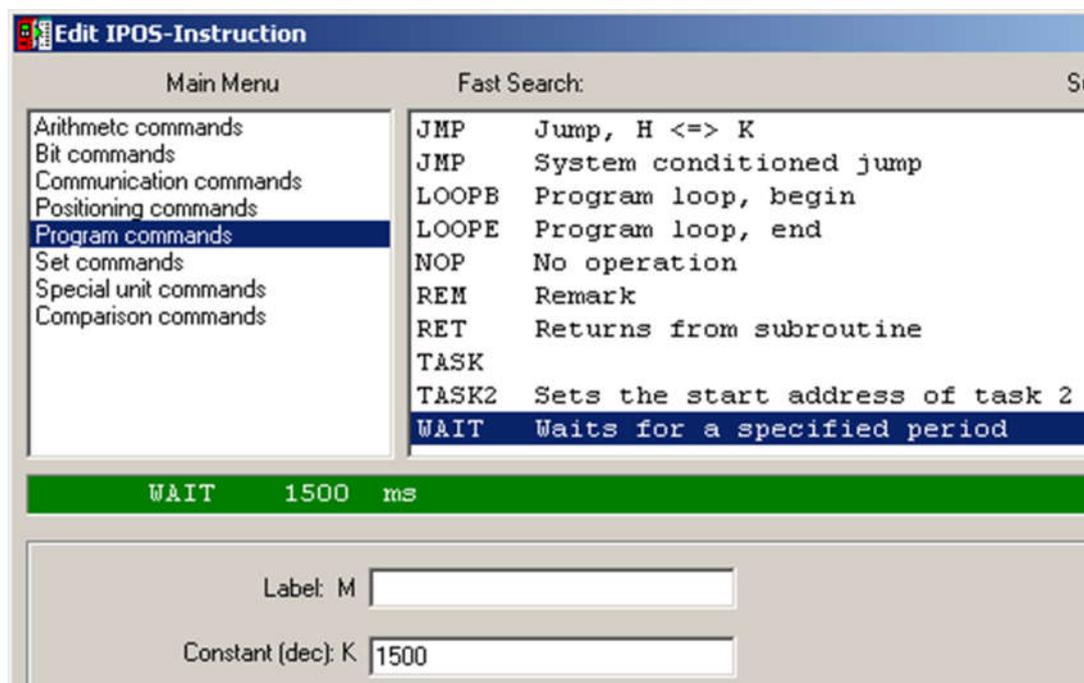


Рис. 18 команда задержки

На Рис. 19 приведено два варианта программ управления двоичными выходами с поочередным их включением, для получения эффекта «бегущих огней», иллюстрирующие работу команды **WAIT**.

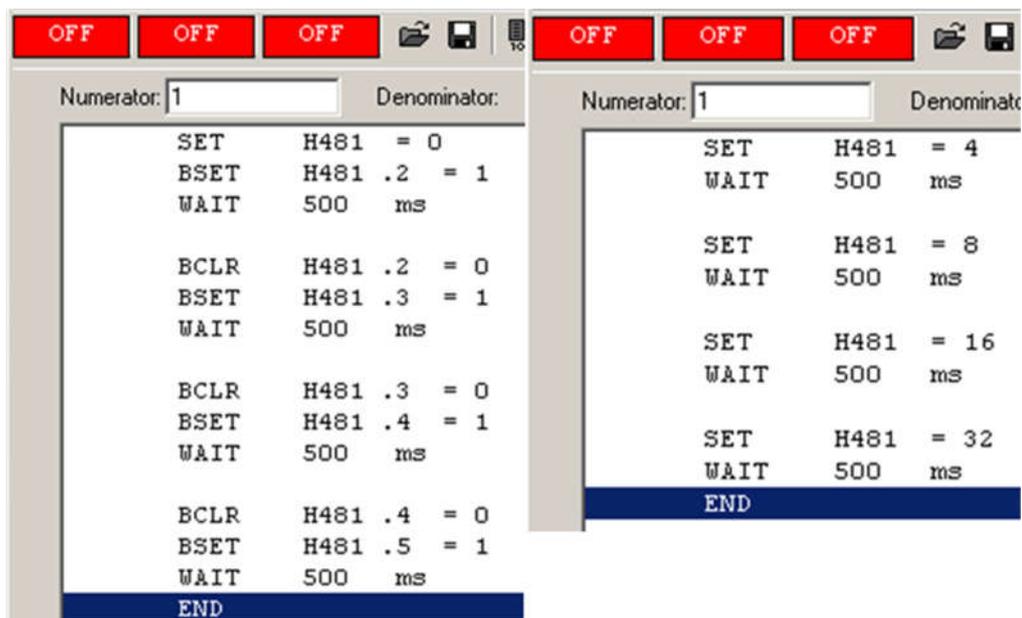


Рис. 19 пример программ управления дискретными выходами «бегущие огни»

В системе команд IPOSplus имеется группа Arithmetic commands, куда включены арифметические и логические побитовые операции над целочисленными переменными и константами, а также операции сдвига. Команды, необходимые для выполнения операций сравнения двух переменных или переменной и константы, собраны в группе Comparison commands (см. [1]).

1.4. Управление задачами.

Команды управления задачами входят в группу Program Commands. В одной прикладной программе можно организовать выполнение до 3-х задач параллельно.

Задача 1 – это основная программа. Запускается она на исполнение с помощью кнопки Start program или через меню Run\Start (Рис. 4). Задачи 2 и 3 инициализируются и запускаются программными командами из задачи 1. Если задача 1 останавливается щелчком на кнопке Stop в панели инструментов, то останавливаются все задачи IPOS-программы.

Программа IPOSplus® обрабатывает каждую задачу циклически: т.е. после завершения выполнения последней команды задачи – происходит переход снова к первой команде.

Скорость выполнения 1 и 2 задач можно изменять с помощью параметров P938 и P939 (Рис. 20).

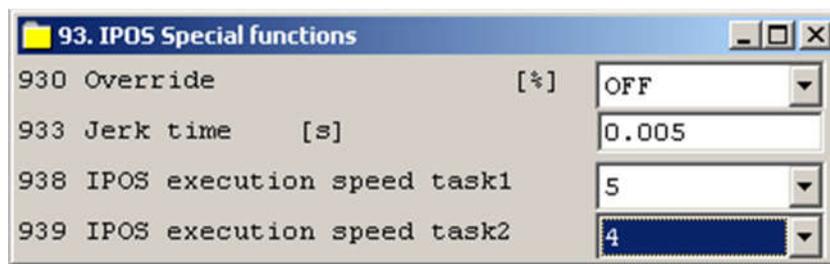


Рис. 20

Заводская настройка (по умолчанию) скорости выполнения задач:

- задача 1: 1 команда/мс (при P938 = 0);
- задача 2: 2 команды / мс (P939 = 0).

Но ее можно изменять в соответствии с Таблица 7. Суммарно для 1 и 2 задач можно увеличить скорость выполнения не более чем на 9 команд/мс.

Таблица 7

Задача 1		Задача 2	
P938	команд / мс	P939	команд / мс
0	1	9	11
1	2	8	10
2	3	7	9
3	4	6	8
4	5	5	7
5	6	4	6
6	7	3	5
7	8	2	4
8	9	1	3
9	10	0	2

Весь оставшийся после обработки команд 1 и 2 задач ресурс контроллера отдается 3-й задаче. Поэтому типичная скорость для нее – 20...40 команд/мс. Блоки прикладной программы, для которых гарантированное время выполнения отдельных программных строк не существенно, в задаче 3 обрабатываются быстрее. Но скорость 3 задачи зависит от аппаратной конфигурации преобразователя и настройки параметров P938, P939 (в самых сложных случаях она гарантированно не менее 1 команды/мс).

Для запуска и остановки, а также для определения начальных адресов (меток) задач 2 и 3 служит команда **TASK**. Эта команда имеет два аргумента.

Первый (Condition/type) может принимать следующие значения: TASK2START – для запуска задачи 2; TASK2STOP – для остановки задачи 2; TASK3START – для запуска задачи 3; TASK3STOP – для остановки задачи 3. Второй аргумент (Jump destination) представляет собой ссылку на метку, с которой начинаются команды запускаемой/останавливаемой задачи (Рис. 21).

ПРИМЕР: TASK TASK3 START M3

После выполнения этой команды параллельно задаче 1 запускается задача 3, программный код задачи 2 начинается с метки M3.

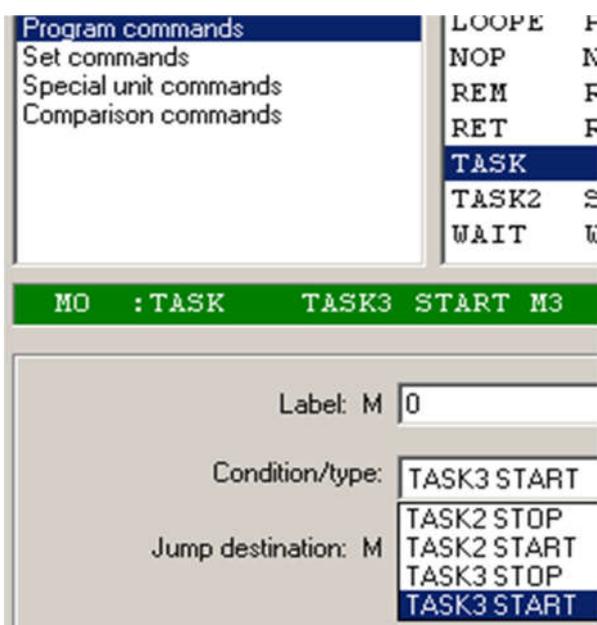


Рис. 21

В группе Program Commands имеется команда **TASK2**, позволяющая запускать и останавливать только задачу 2. Она использовалась в предыдущей версии MOVIDRIVE. Для MOVIDRIVE® В эта команда также может применяться, но представляется целесообразным заменять ее командой **TASK**.

Необходимо отметить, что набор команд каждой из задач 2 и 3 следует завершать командой **RET**, или командой безусловного перехода на первую команду задачи. Пример использования двух задач приведен на Рис. 22.

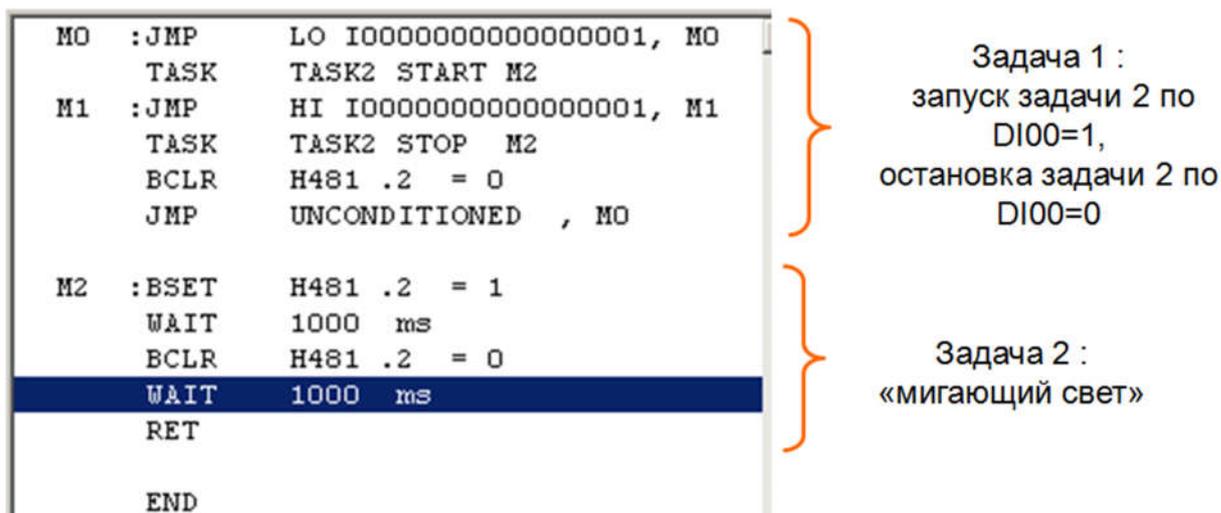


Рис. 22 пример использования двух задач в программе

1.5. Работа с прерываниями.

Рассмотрим возможности работы с прерываниями в IPOSplus. Команды инициализации прерываний входят в группу Set commands. Существует две разновидности этих команд:

- **SETINT** – позволяющая инициализировать прерывания задачи 1;
- **VARINT** – команда инициализации прерываний задач 2 или 3.

Прерывание, вызванное каким-либо событием, приостанавливает выполнение соответствующей задачи. Обработчик прерывания выполняется один раз от начала до конца, и может быть прерван только другим прерыванием той же задачи с большим приоритетом.

Прерывание, обработчик которого инициализируется командой **SETINT**, останавливает выполнение задачи 1, может вызываться следующими событиями (указывается в аргументе команды):

1. системной /аппаратной ошибкой (ERROR)
2. сигналом на вход обучения DI02 (TOUCH PROBE)
3. переполнением таймера 0 (TIMER 0)

Соответственно, прерывание по ошибке имеет наивысший приоритет, а прерывание по таймеру – низший.

Команда **SETINT** устанавливает начальный адрес обработчика прерывания. Адрес указывается в команде меткой. Сама подпрограмма обработки прерывания должна заканчиваться командой **RET**.

Переход в подпрограмму обработки прерывания выполняется сразу и независимо от того, какая строка основной программы обрабатывается в данный момент. Когда эта подпрограмма заканчивается (командой **RET**), выполнение основной программы продолжается с места прерывания.

Пример: Переход к обработке прерывания в случае аппаратной ошибки приведен на Рис. 23. В данном примере после возникновения аппаратной ошибки выполняется немедленный вызов подпрограммы обработки прерывания. Подпрограмма обработки прерывания «заставляет» двоичный выход DO02 «мигать» с периодом в 2 с.

```

      SETINT  ERROR  M100
M1  :NOP
      JMP    UNCONDITIONED  , M1

M100:BSET  H481 .2 = 1
      WAIT  1000 ms
      BCLR  H481 .2 = 0
      WAIT  1000 ms
      JMP  LO I0000000000000000100, M100
      RET

      END

```

Рис. 23 Пример прерывания задачи 1.

Как только на клемме DI02 появляется сигнал высокого уровня, происходит возврат (RET) в основную программу. Для возможности сброса сообщения об ошибке вход DI02 необходимо запрограммировать на "FAULT RESET" (см. Рис. 24).

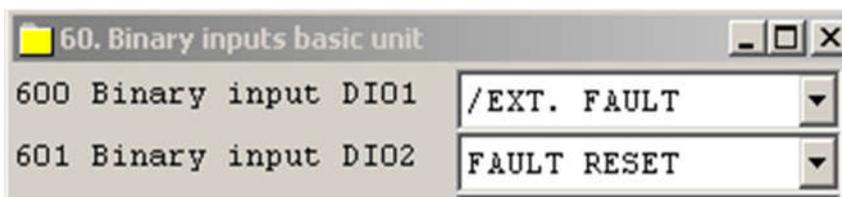


Рис. 24

Командой **VARINT**, можно дополнительно инициализировать до 4 обработчиков прерываний по контрольной переменной, которые прерывают выполнение задачи 2 или 3. С помощью этой команды можно, например, организовать прерывания:

- при каком-либо количественном значении контрольных переменных;
- при определенном значении любого таймера (0, 1 и 2);
- при выходе привода в определенное положение или в точку синхронизации с другим приводом;
- при изменении состояния какого-либо I/O-сигнала (Входа или Выхода);
- при определенном статусе преобразователя (H473);
- на прием или передачу новых данных по шине S-Bus.

Команда **VarInt** Hxx, Mxx инициализирует прерывание по контрольной переменной, используя структуру данных, начинающуюся с переменной Hxx. Если выполняется условие прерывания и запущена задача 2 или 3, в которой это прерывание обрабатывается, то начинается выполнение команд с метки Mxx.

Таблица 8

Переменная	Элементы структуры VARINT	Описание
H+0	Control	0: Все VarInterrupt = ВЫКЛ / Сброс 1: Прерывается задача 2, прерывание обрабатывается со скоростью выполнения этой задачи. 2: Прерывается задача 3, прерывание обрабатывается со скоростью выполнения этой задачи.
H+1	IntNum	0...3: Задаёт порядковый номер VarInterrupt. Какое-либо уже инициализированное прерывание под номером x во время выполнения программы можно полностью переинициализировать через вызов команды VarInt Hxx, Mxx с другой структурой данных, если в этой новой структуре данных на месте H+1 указан тот же самый номер переменной. Для прерывания задачи 1 это свойство не используется.
H+2	SrcVar	Номер контрольной переменной, значение которой сравнивается с эталонным значением. SrcVar – это значение контрольной переменной, на которую указывает SrcVar.
H+3	CompVar	Эталонное значение или маска, с которой сравнивается значение контрольной переменной H+2.
H+4	Mode	0: Нет события прерывания. Блокируется только данное прерывание без отключения всех остальных прерываний. 1: Один из битов контрольной переменной, выделяемых маской CompVar, изменил свое значение: $([*SrcVar(t) \wedge *SrcVar(t-T)] \& CompVar) \neq 0$ 2: Если значение контрольной переменной равно эталонному значению ($*SrcVar == CompVar$) 3: Если значение контрольной переменной не равно эталонному значению ($*SrcVar \neq CompVar$) 4: Если значение контрольной переменной больше либо равно эталонному значению ($*SrcVar \geq CompVar$) 5: Если значение контрольной переменной меньше либо равно эталонному значению ($*SrcVar \leq CompVar$) 6: Логическое произведение значения контрольной переменной и эталонного значения не равно 0 ($(*SrcVar \& CompVar) \neq 0$) 7: Логическое произведение значения контрольной переменной и эталонного значения равно 0 ($(*SrcVar \& CompVar) == 0$) 8: Установка бита, выделенного маской CompVar. 9: Сброс бита, выделенного маской CompVar. 10: См. пункт 2, однако прерывание обрабатывается только один раз при каждом выполнении условия (вызов по изменению). 11: См. пункт 3, однако прерывание обрабатывается только один раз при каждом выполнении условия (вызов по изменению). 12: См. пункт 4, однако прерывание обрабатывается только один раз при каждом выполнении условия (вызов по изменению). 13: См. пункт 5, однако прерывание обрабатывается только один раз при каждом выполнении условия (вызов по изменению).
H+5	Priority	Приоритет прерывания (1...10), задача 2 и задача 3 имеют приоритет = 0.
H+6	IntEvent	Отображение контрольной переменной *SrcVar в момент прерывания.

Структура данных для прерывания по контрольной переменной показана в Таблица 8. Событием для прерывания является результат сравнения (N+4 в Таблица 8) контрольной переменной (указывается в N+2) с эталонным значением (содержащимся в N+3). В этой же структуре задаются:

- какая задача прерывается для обработки прерывания (N+0);
- порядковый № прерывания (N+1);
- приоритет (N+5);

Информация о том, запрашивалось ли какое-либо прерывание по контрольной переменной, доступна через биты 0...3 системной переменной N530 (см. Таблица 9). Эти биты "запроса" в программе IPOsplus® можно не только считывать, но и изменять (т.е. устанавливать запрос). Через эту переменную можно для проверки установить бит запроса и вызвать прерывание по контрольной переменной независимо от фактического условия прерывания (если предварительно инициализировать соответствующий обработчик прерывания).

Таблица 9

№	Имя в компиляторе / ассемблере	Описание
530	VarIntReq / VARINTREQ	Если в этой переменной устанавливается бит запроса какого-либо прерывания, то независимо от фактического условия прерывания вызывается прерывание по контрольной переменной. Но перед этим нужно инициализировать соответствующий обработчик прерывания. Бит 0: Запрос на прерывание 0 по контрольной переменной Бит 1: Запрос на прерывание 1 по контрольной переменной Бит 2: Запрос на прерывание 2 по контрольной переменной Бит 3: Запрос на прерывание 3 по контрольной переменной

2. Управление замкнутым электроприводом через программу IPOSplus.

С помощью преобразователей частоты **MOVIDRIVE B** можно создавать электроприводы, замкнутые по моменту, скорости, положению или технологическому параметру. В рамках данного пособия рассмотрим только особенности реализации и управления через IPOSplus приводами замкнутыми по скорости и положению.

2.1. Замкнутый электропривод в режиме регулирования скорости.

Замкнутые по скорости вращения приводы можно создать только при наличии инкрементного (sin/cos-го, HIPERFACE) датчика на валу двигателя. Такой датчик называется motor encoder. При этом могут использоваться векторные законы частотного управления (Рис. 25):

- VFC-n-control, CFC – для асинхронного двигателя;
- SERVO – для вентильного серводвигателя.

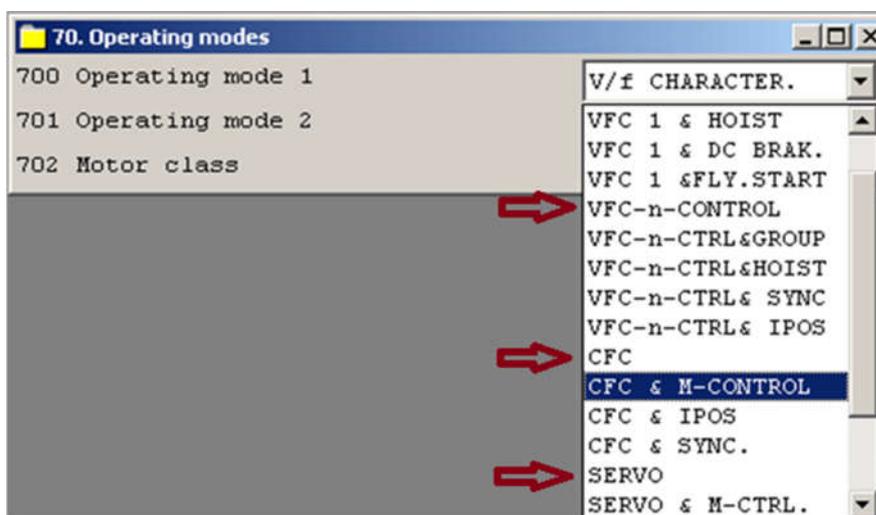


Рис. 25

Структура контура скорости такого привода показана на Рис. 26. Она может включать в себя подчиненный контур тока (как показано на Рис. 26), если используются законы частотного управления CFC или SERVO.

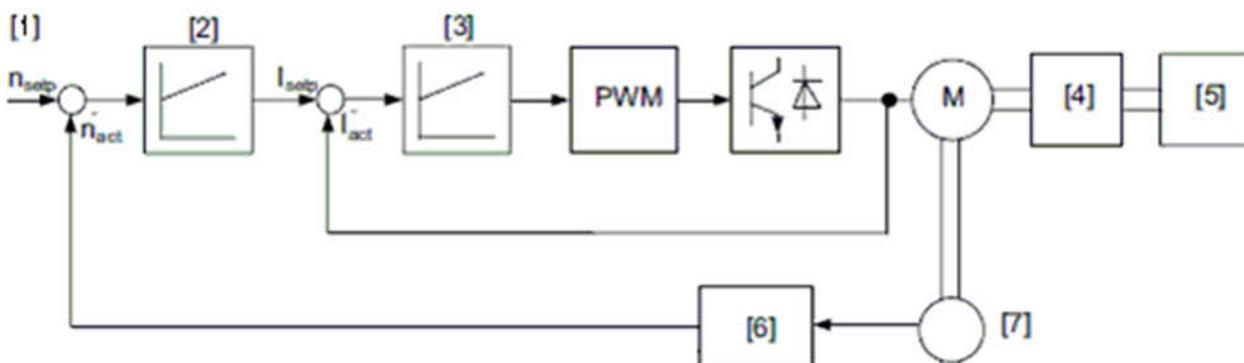


Рис. 26 Контур скорости: 1 – задание скорости (уставка, требуемое значение); 2 – регулятор скорости; 3 – регулятор тока (момента); 4 – редуктор (передача); 5 – исполнительный механизм; 6 – преобразователь сигнала инкрементного датчика в величину значения скорости; 7 – инкрементный датчик

Регулятор скорости представляет собой ПИ-регулятор с упреждающей дифференцирующей связью по заданию (по заданию ускорения), структурная схема которого показана на Рис. 27.



Рис. 27 Структура регулятор скорости.

Все параметры регулятора скорости при вводе в эксплуатацию преобразователя настраиваются автоматически, но, в дальнейшем, доступны для изменения их значений через программу SHELL (см. Рис. 28).

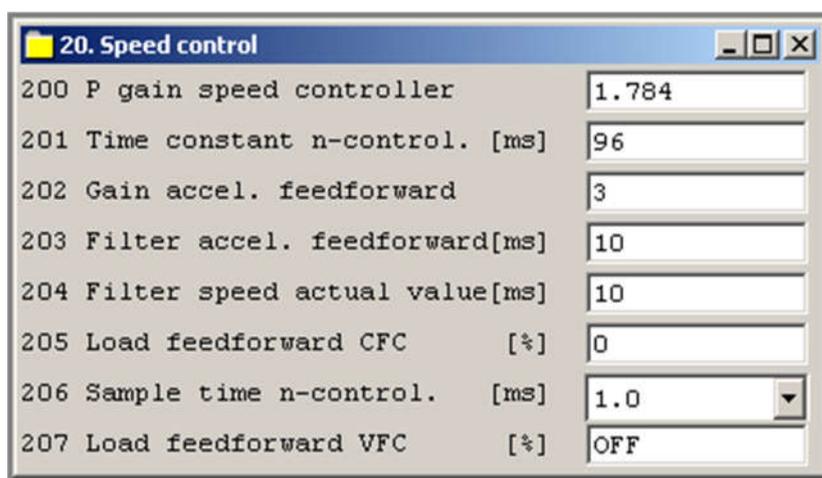


Рис. 28 Параметры регулятора скорости

Управление приводом в режиме регулирования скорости из программы IPOS реализуется двумя способами:

1. *Через системную переменную Control word H484* (Таблица 2). Для этого способ задания уставки скорости (P100) должен быть задан как: фиксированные уставки (однополярные или биполярные) или мотор потенциометр. Дискретные управляющие сигналы разрешения (ENABLE), выбора направления вращения (CW, CCW), уставки (n11, n12, n13...) и т.п. задаются не от дискретных входов, а с помощью битов H484.
2. *С использованием системной переменной H524* (см. Таблица 10) для задания уставки скорости. В этом случае требуется в параметре P100 установить значение IPOS SETPOINT (Рис. 29).

Таблица 10

№	Имя в компиляторе / ассемблере	Описание
524	IPOS_Setp / IPOS_SETP	IPOS-уставка, управляющее воздействие ПИД-регулятора при H540 = 1. Если H540 = 0 или 2, возможна запись значения прямо из прикладной программы. H524 можно использовать как уставку момента или частоты вращения, если задано P100 "Источник уставки" = IPOS, а P700 "Режим работы 1" = xxx&M-control или CFC или SERVO. В этом случае 1 инкременту в H524 соответствуют 0,2 об/мин уставки частоты вращения или 0,01 % I _{ном} уставки момента.

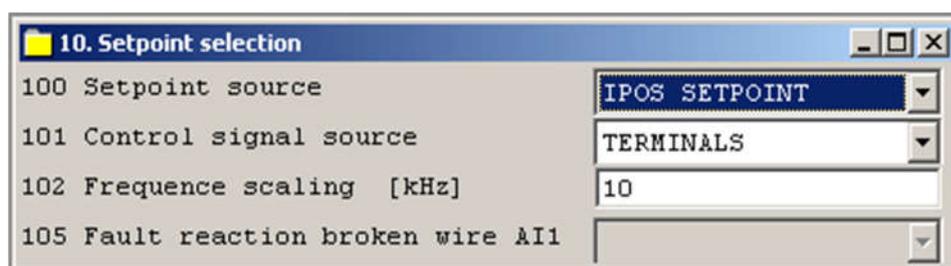


Рис. 29

Единственным обязательно используемым аппаратным сигналом для обоих случаев является DI00. Остальные дискретные входы могут быть запрограммированы на значение NO FUNCTION.

2.2. Электропривод в режиме позиционирования.

Привод в режиме позиционирования строится как система подчиненного управления и предполагает использование подчиненного контура скорости. Как отмечалось ранее, контур скорости может работать только с датчиком на валу двигателя (motor encoder), который подключается к разъему X15 платы датчика (например, платы DEH11b).

Контур регулирования положения можно замкнуть с использованием инкрементного или абсолютного датчика. Причем, эти датчики могут устанавливаться как на валу двигателя, так и на рабочем органе механизма. Для инкрементного датчика в первом случае используются сигналы уже подключенного к разъему X15 датчика, во втором – внешний датчик (external encoder) подключается к разъему X14 платы датчика [2]. Позиционирование по внешнему датчику позволяет компенсировать проскальзывание или люфт в соединении между приводом и перемещаемым механизмом (пробуксовка колес, зазоры в зубчатом зацеплении и т. п.) или механический люфт в редукторе.

Если в качестве датчика положения предполагается использовать SSI-датчик абсолютного отсчета (absolute encoder), то его подключают к разъему X62 специальной платы (например, DIP11b) [2,3].

Выбранный для замыкания контура положения вид датчика должен быть обязательно установлен в параметре «Источник действительного положения» P941 (см. Рис. 30).

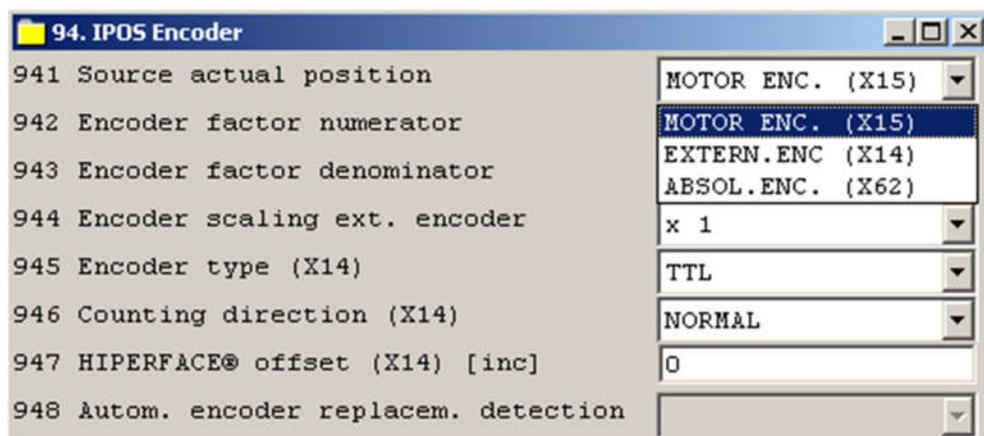


Рис. 30

Чтобы привод работал в режиме позиционирования, при вводе в эксплуатацию преобразователя частоты следует выбрать Operating mode: positioning, что установит в параметре P700 один из режимов "... & IPOS" (см. Рис. 31).

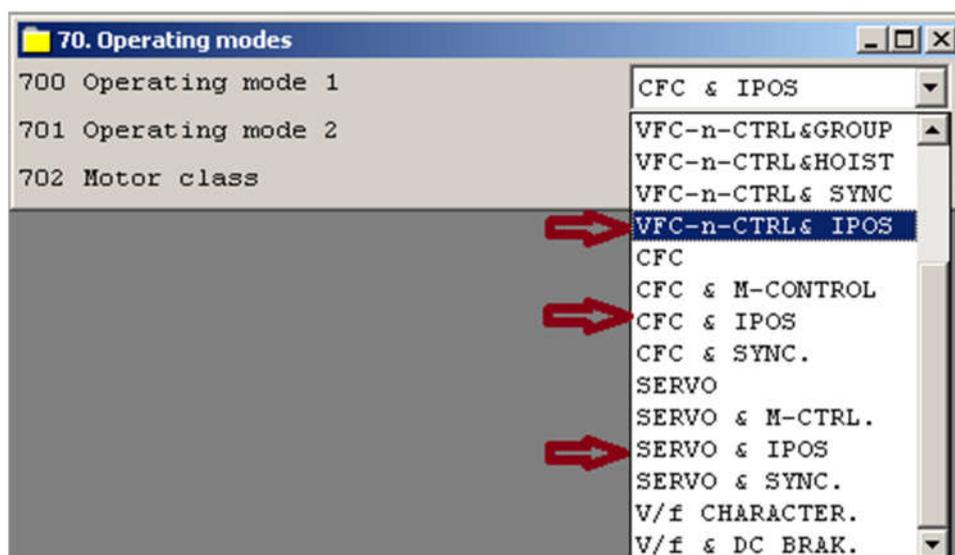


Рис. 31

Структурная схема электропривода, у которого контур положения замкнут по датчику на валу двигателя, показана на Рис. 32. В этом контуре использован П-регулятор. Система включает в себя интерполятор, который рассчитывает траекторию (задание на контур положения $x_{\text{интерп}}$ - setpoint

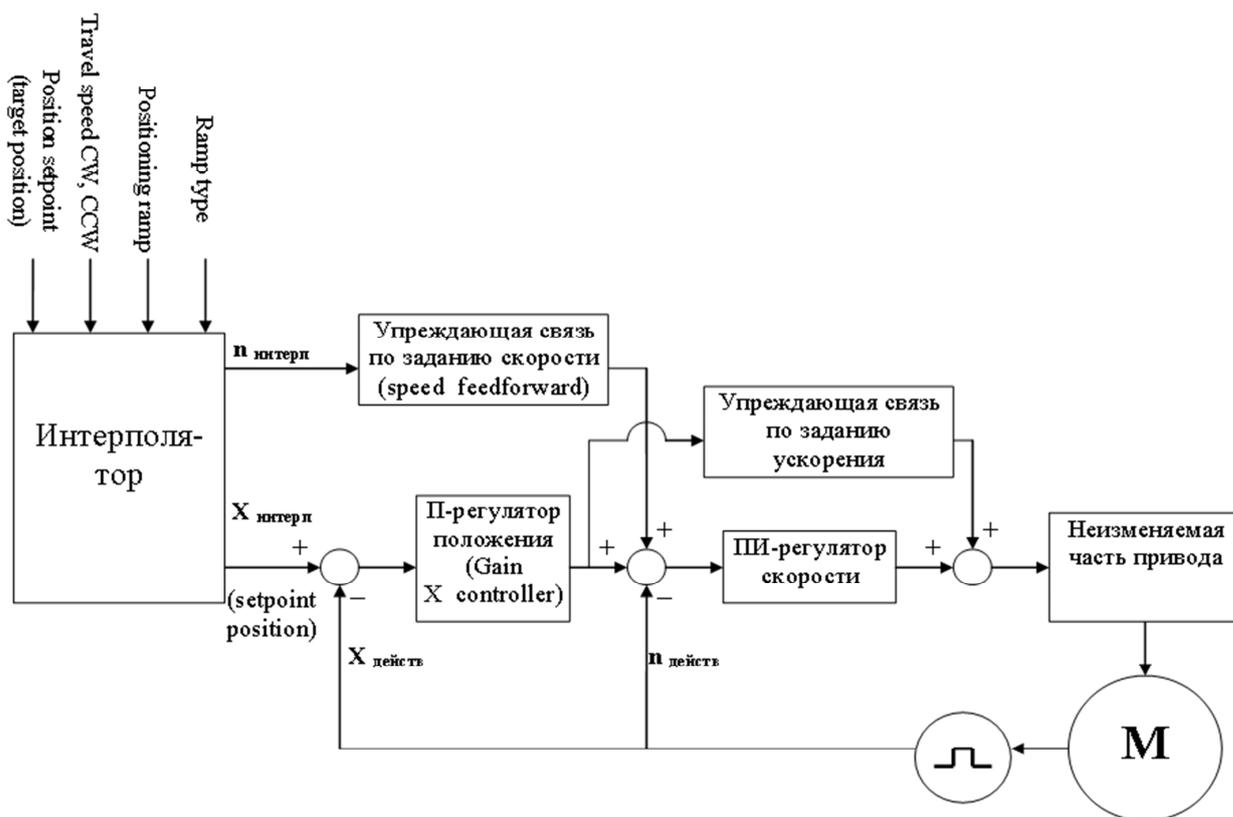


Рис. 32 Структурная схема электропривода в режиме позиционирования.

position) и формирует сигнал задания на контур скорости $n_{\text{интерп}}$ в процессе движения на основании следующих входных параметров:

1. Задания конечной точки позиционирования (target position);
2. Скоростей перемещения в конечную точку, которые задаются для движения по часовой и против часовой стрелки (travel speed CW, travel speed CCW);
3. Темпов разгона и торможения (positioning ramp);
4. Формы кривой разгона/замедления (ramp type).

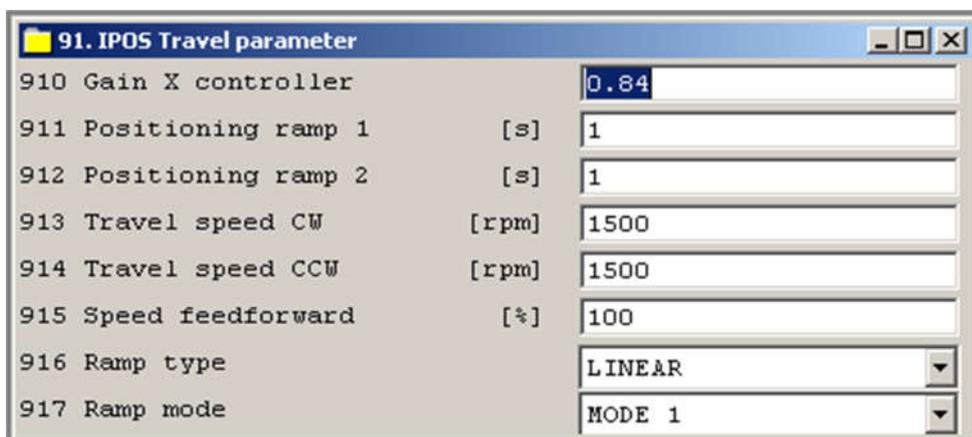


Рис. 33

Доступ к большинству из этих параметров, а также к коэффициентам П-регулятора и упреждающей связи по ускорению возможен через параметры в программе SHELL (см. Рис. 33). К другим из вышеназванных величин можно обращаться через системные переменные (Таблица 11).

Таблица 11

№	Имя в компиляторе / ассемблере	Описание
491	SetpointPos / SETP.POS.	 СЧИТЫВАНИЕ текущего значения уставки положения. ВНИМАНИЕ: Системная регулируемая величина! Изменять значение запрещается! Независимо от числа импульсов датчика за один оборот уставка положения имеет фиксированную единицу измерения: 4096 инкр. / оборот вала двигателя (дискретность датчика ≥ 512). Текущее значение уставки положения представляет собой действительное в данный момент для позиционного регулятора абсолютное положение привода в рамках данного задания перемещения. Изменение уставки положения во времени происходит вследствие расчета профиля перемещения с учетом темпа и скорости позиционирования, формы генератора темпа и т. п. После выполнения задания перемещения и остановки привода значение H491 равно значению H492. Диапазон значений: $-2^{31} \dots 0 \dots 2^{31} - 1$ инкр.
492	TargetPos / TARGET POS	СЧИТЫВАНИЕ и УСТАНОВКА текущего значения конечного положения. Независимо от числа импульсов датчика за один оборот конечное положение имеет фиксированную единицу измерения: 4096 инкр. / оборот вала двигателя (дискретность датчика ≥ 512). Эта переменная отображает текущее значение конечного положения в рамках данного задания перемещения. Значение положения в H492 выражено в абсолютной форме. Пример: 1. Текущее положение привода: 50000 инкр. 2. GOR NOWAIT #–8000 инкр. 3. Текущее конечное положение: 42000 инкр. Диапазон значений: $-2^{31} \dots 0 \dots 2^{31} - 1$ инкр. Если значение в H492 записывается напрямую (не через GO-команду), то в переменной H473 бит 19 "Конечное положение достигнуто" остается активным еще до 1 мс.
509	ActPos_Abs / ACTPOS ABS	 СЧИТЫВАНИЕ текущего значения действительного положения по DIP-датчику абсолютного отсчета (SSI). ВНИМАНИЕ: Системная регулируемая величина! Изменять значение запрещается! Это действительное положение определяется по сигналам, поступающим на разъем X62 (опция DIP11A). Единица измерения: инкременты в зависимости от дискретности датчика.
510	ActPos_Ext / ACTPOS EXT	 СЧИТЫВАНИЕ текущего значения действительного положения по внешнему датчику. ВНИМАНИЕ: Системная регулируемая величина! Изменять значение запрещается! Это действительное положение определяется по инкрементным сигналам, поступающим на разъем X14. Такое определение положения выполняется только в том случае, если разъем X14 используется как вход датчика. Единица измерения: инкременты в зависимости от дискретности датчика.
511	ActPos_Mot / ACTPOS MOT	 СЧИТЫВАНИЕ текущего значения действительного положения по датчику двигателя. ВНИМАНИЕ: Системная регулируемая величина! Изменять значение запрещается! Независимо от числа импульсов датчика за один оборот это действительное положение имеет фиксированную единицу измерения: 4096 инкр. / оборот вала двигателя (дискретность датчика от 512 до 2048)

Необходимо сделать некоторые пояснения, к параметру P916 Ramp type, который задает характер кривых разгона и торможения в процессе движения. В простейшем случае можно задать линейный закон изменения скорости (LINEAR), как указано в Таблица 12. При этом траектория и задание скорости, рассчитываемые интерполятором, будут иметь вид, показанный на Рис. 34. Кривая ускорения $a(t)$ имеет прямоугольную форму. Основным недостатком такого вида траекторий являются слишком большие рывки (производные от ускорения) на фронтах кривой $a(t)$, приводящие к быстрому износу редукторов и механики приводов.

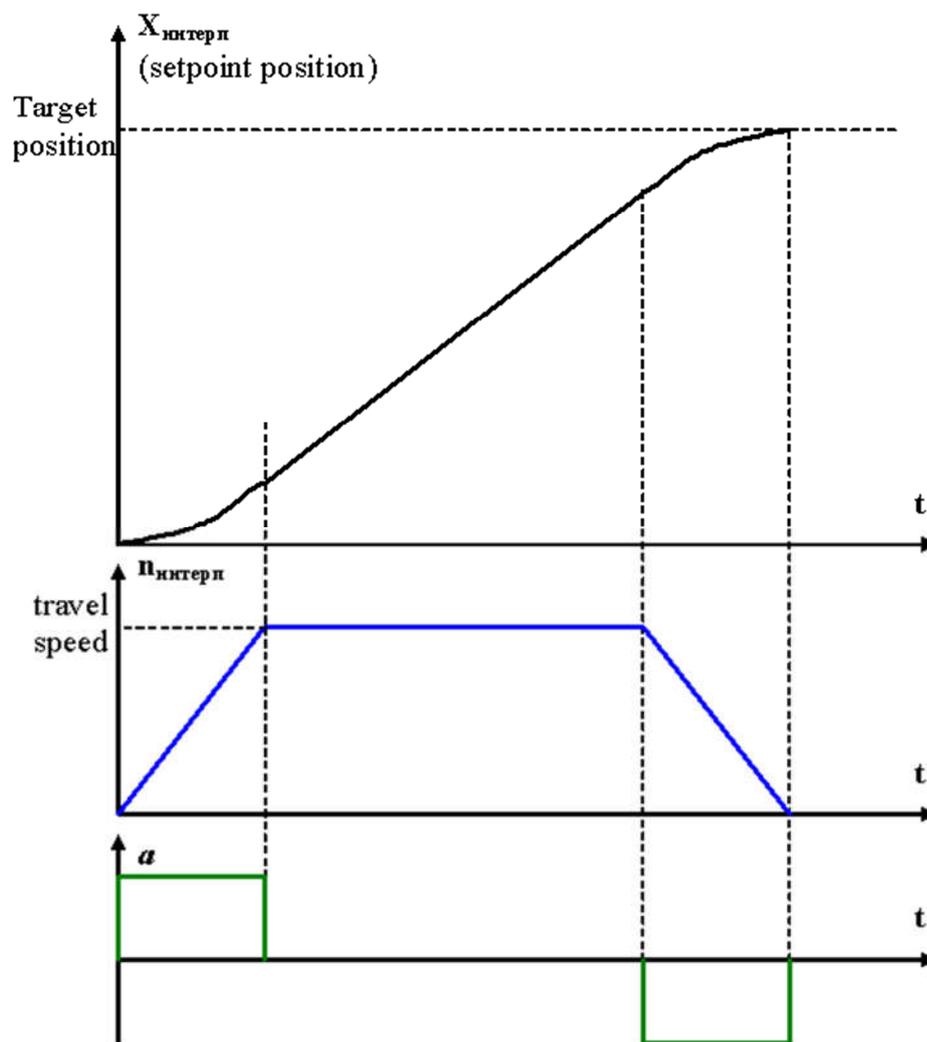
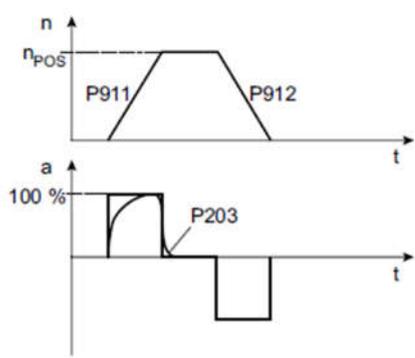
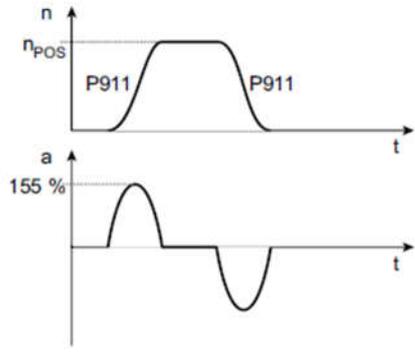
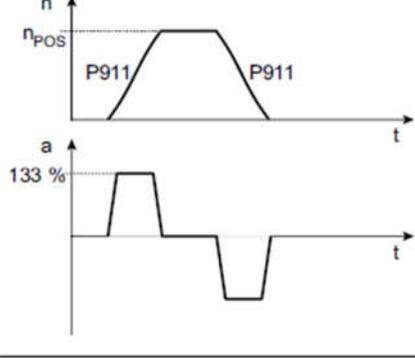
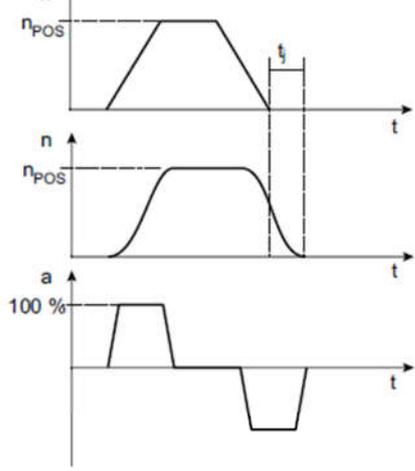


Рис. 34 Работа интерполятора при линейном законе разгона и торможения.

Ramp type	Характер позиционирования	Применение
<p>LINEAR</p> 	<p>Кривая частоты вращения Кривая частоты вращения $n(t)$ – линейная и зависит от темпа и частоты вращения при позиционировании (трапеция).</p> <p>Кривая ускорения Кривая ускорения $a(t)$ (т. е. и кривая вращающего момента) – ступенчатая (рывковая нагрузка). Параметр P203 "Фильтр упреждения по ускорению" ослабляет рывок, но при настройке на слишком большое значение вызывает задержку выхода в конечное положение. Характер темпа позиционирования можно сгладить с помощью параметра P915 "Упреждение по скорости".</p>	<ul style="list-style-type: none"> Мало подверженные вибрации и низкоэластичные приводные системы. Быстрые операции позиционирования с быстрым разгоном/ торможением. Из-за рывка (скачкообразное возрастание вращающего момента) перемещаемый материал подвергается сильной нагрузке. (Не подходит для перемещения жидких материалов и пластичных масс (например, на установках бутылочного розлива)).
<p>SINE</p> 	<p>Кривая частоты вращения Кривая частоты вращения – \sin^2-подобная и зависит от темпа и частоты вращения при позиционировании.</p> <p>Кривая ускорения Кривая ускорения – синусоидальная. В начале и в конце разгона ускорение ниже, чем при линейном темпе. При том же значении темпа (как у линейного) на среднем участке разгона ускорение выше примерно в 1,55 раза. Функция ограничения рывка при пуске не активна.</p>	<ul style="list-style-type: none"> Быстрые операции позиционирования на подверженных вибрации или эластичных приводных системах. <p>Ускорение (т. е. и необходимый вращающий момент) примерно на 55 % выше, чем при линейном темпе. При проектировании учитывайте, что двигатель и преобразователь должны быть способны развивать необходимый вращающий момент (иначе может возникнуть погрешность запаздывания).</p>
<p>SQUARED</p> 	<p>Кривая частоты вращения В начале и в конце разгона/замедления кривая частоты вращения – квадратичная и зависит от темпа и частоты вращения при позиционировании.</p> <p>Кривая ускорения Кривая ускорения – трапецевидная. То есть, рывок слабее, чем при линейном темпе. Данная форма генератора темпа – компромисс между необходимым вращающим моментом и сглаживанием темпа. Функция ограничения рывка при пуске не активна.</p>	<ul style="list-style-type: none"> Быстрые операции позиционирования на подверженных вибрации или эластичных приводных системах. <p>Ускорение (т. е. и необходимый вращающий момент) примерно на 33 % выше, чем при линейном темпе. При проектировании учитывайте, что двигатель и преобразователь должны быть способны развивать необходимый вращающий момент (иначе может возникнуть погрешность запаздывания).</p>
<p>JERK LIMITED</p> 	<p>В основе ограничения рывка (P916) используется линейный темп. При активной функции ограничения рывка вращающий момент, а вместе с ним и ускорение изменяются по трапецевидной кривой. При разгоне вращающий момент линейно возрастает во времени, пока не достигнет максимального значения. Таким же образом он уменьшается во времени до нуля при торможении. За счет этого система почти не испытывает колебательного процесса. Диапазон настройки составляет от 5 до 2000 мс (P933). Время позиционирования в сравнении с использованием линейного темпа увеличивается на указанное время рывка t_j. Величина ускорения и вращающего момента не больше, чем при линейном темпе.</p>	<p>Все варианты применения, где использовались формы генератора темпа "LINEAR", "SINE" или "SQUARED".</p>

Чтобы избежать этого недостатка, можно применять другие характеристики движения с ускорением, показанные в Таблица 12.

Отметим, что чтение текущих значений скорости и положения (по датчику указанному в H941), а также величин target position и setpoint position возможны с использованием команды **GETSYS** (Рис. 35). В свою очередь, с помощью команды **SETSYS** можно изменять скорости (travel speed CW, CCW - P913, P914), ускорения (positioning ramp1, 2 - P911, P912) и характер движения с ускорением (ramp type - P916) задаваемые интерполятору (см. Рис. 33 и Рис. 36).

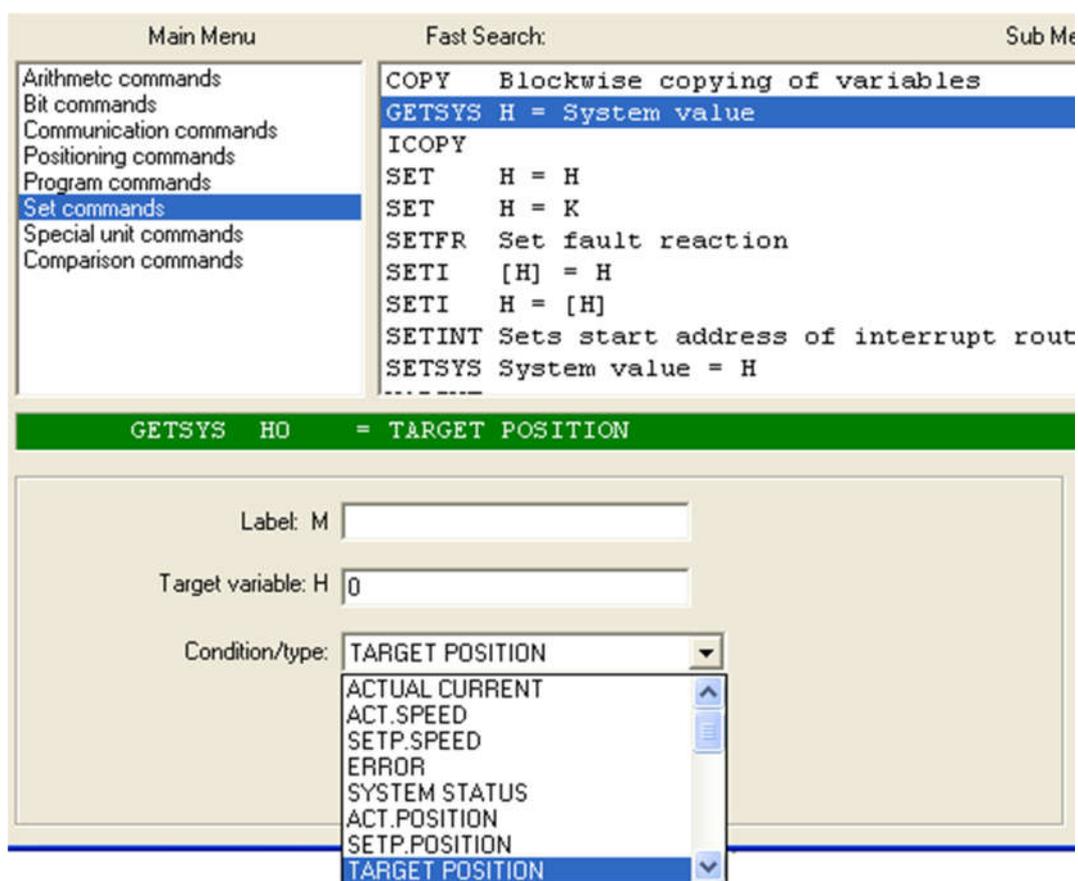


Рис. 35

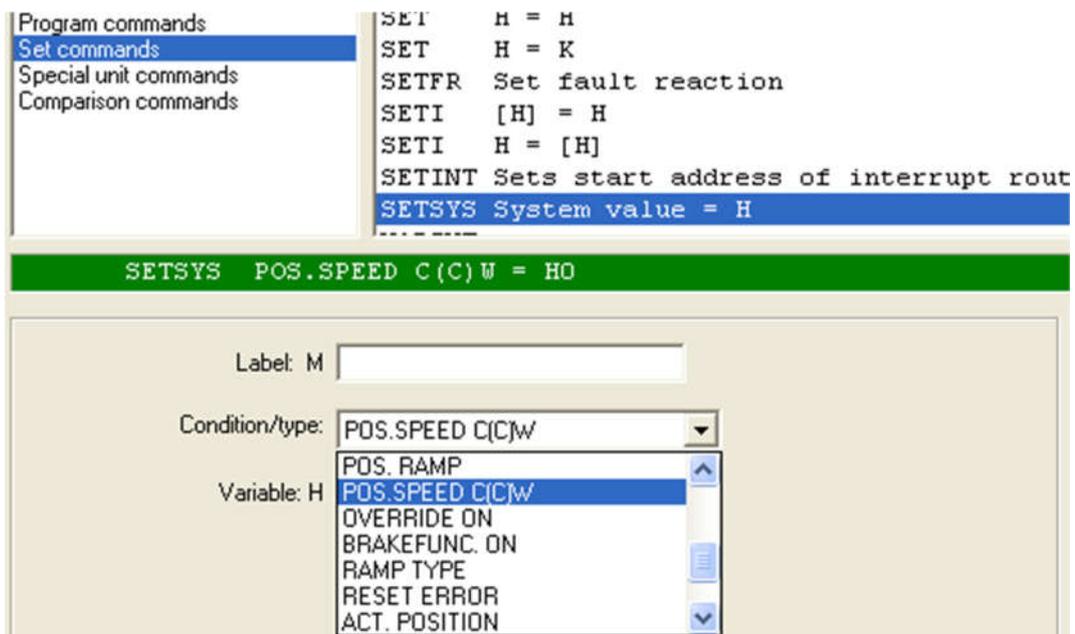


Рис. 36

Показанный на Рис. 32 вариант замыкания контура положения - не единственный. Другие возможные варианты приведены на Рис. 37 - Рис. 41.

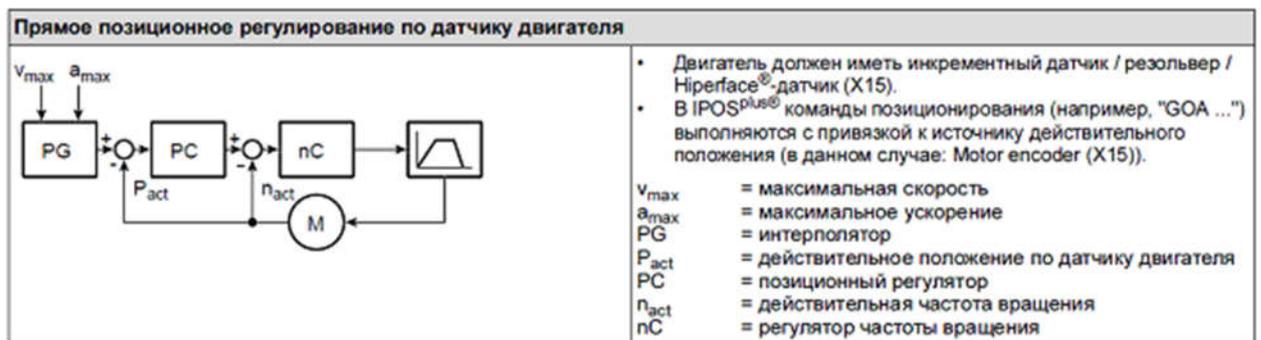


Рис. 37

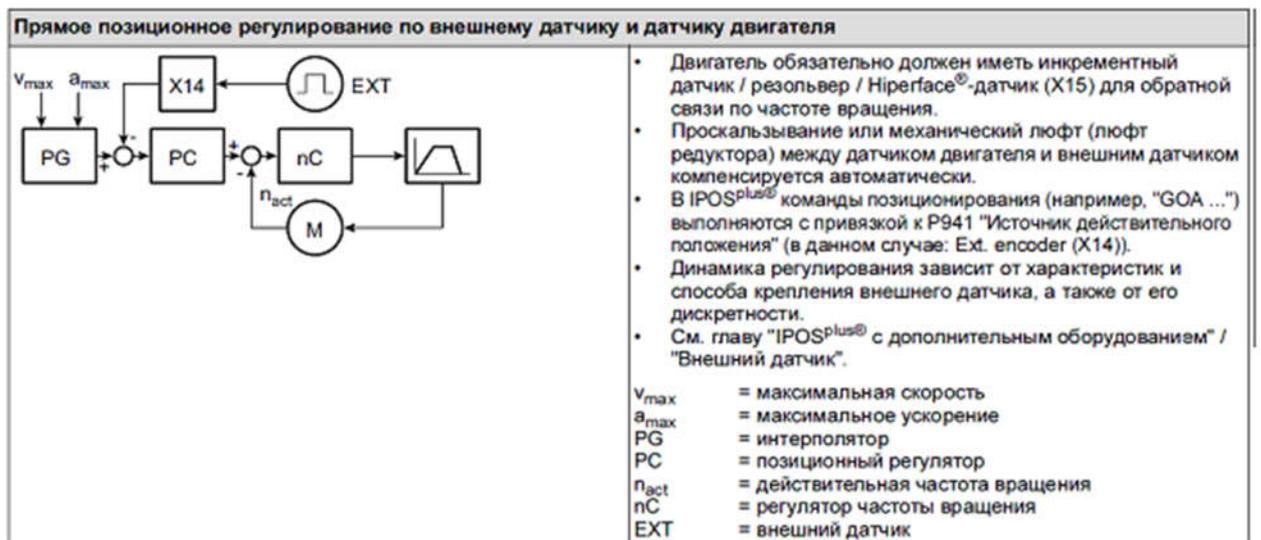


Рис. 38

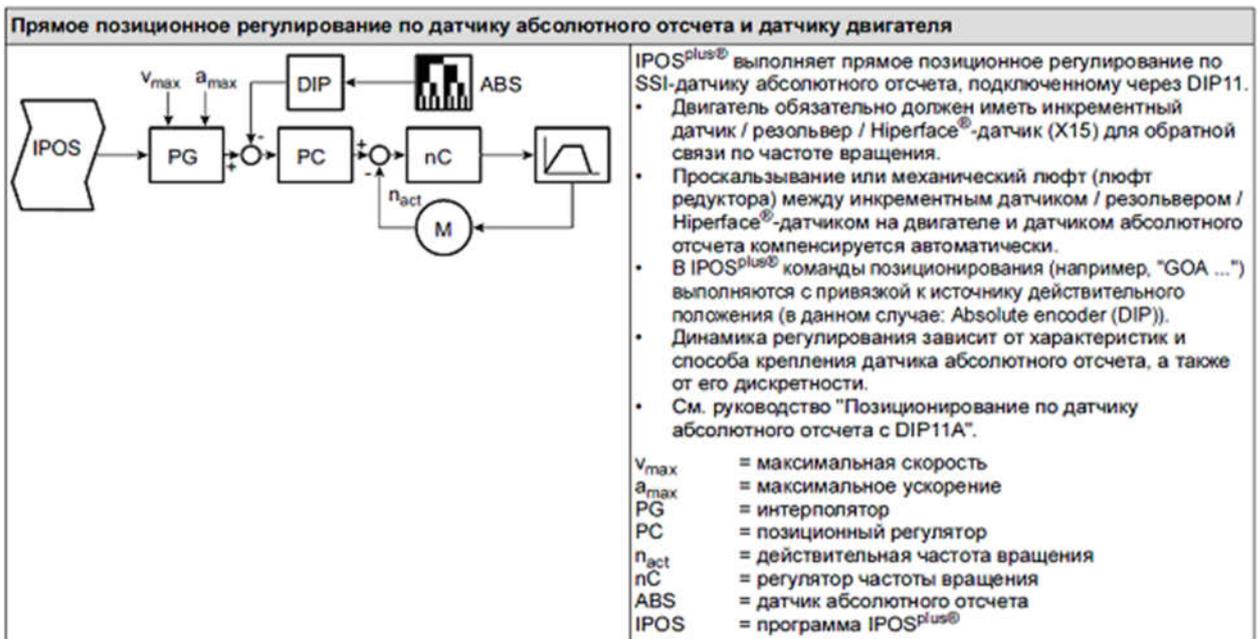


Рис. 39

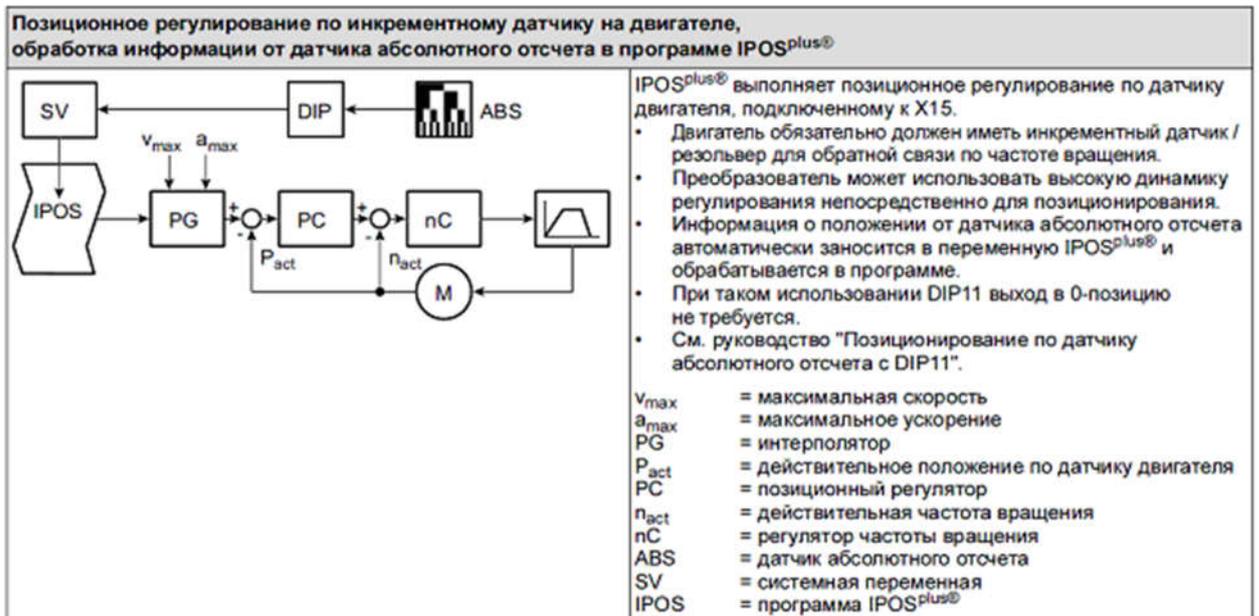


Рис. 40

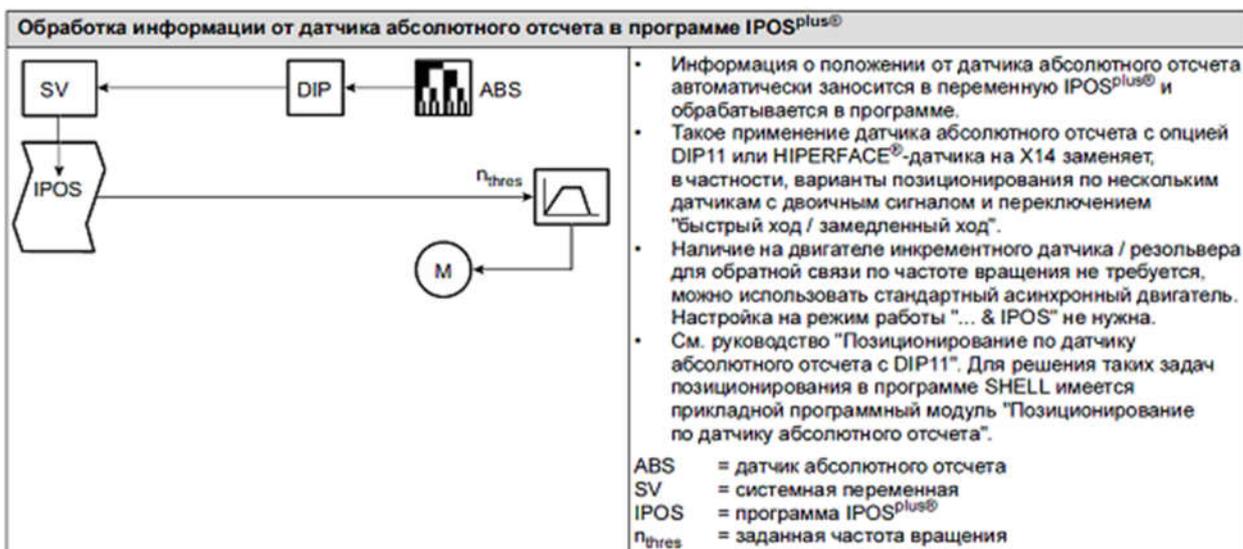


Рис. 41

2.3. Выход в 0-позицию.

Для работы с командами абсолютного позиционирования необходимо определить точку отсчета (машинный ноль). Процедура поиска машинного ноля называется выходом 0-позицию (reference travel). В зависимости от типа используемого датчика эта операция выполняется:

- один раз при первом вводе в эксплуатацию или при замене датчика (с датчиком абсолютного отсчета SSI или HIPERFACE);
- при каждом включении электропривода (со всеми другими датчиками).

В большинстве случаев машинный ноль определяется с использованием сигналов конечных выключателей и датчика 0-позиции, подключаемых к двоичным входам. При этом (см. Рис. 42):

- вход, к которому подключен датчик 0-позиции программируется на REFERENCE CAM;
- входы для подключения аппаратных конечных выключателей - /LIM.SWITCH CW и /LIM.SWITCH CCW.

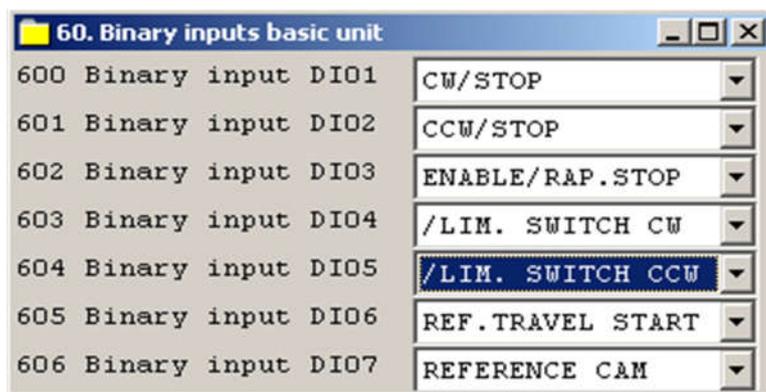


Рис. 42

Запуск режима выхода в 0-позицию возможен 3-мя способами:

1. С помощью двоичного входа, запрограммированного на функцию REF.TRAVEL START (см. Рис. 42);
2. используя бит переменной N484.18 Control word (Таблица 2);
3. командой IPOSplus (см. Рис. 43).

Для команды **GOO** необходимо выбрать ее атрибуты, смысл которых разъяснен в Таблица 13.

Таблица 13

C	(conditional)	Выход в 0-позицию только в том случае, если он еще не выполнен (т. е. N473, бит 20 = 0).
U	(unconditional)	Выход в 0-позицию в любом случае.
W	(wait)	Ожидание завершения выхода привода в 0-позицию. До этого ни одна другая команда этой задачи не выполняется.
NW	(non-wait)	Выполнение следующей команды, пока устанавливается 0-позиция (рекомендация).
ZP	(zero pulse)	Выход в 0-позицию на нулевой импульс сигнала датчика (игнорируется при 903 = 0 или P903 = 5).
CAM	(reference cam)	Выход в 0-позицию на датчик 0-позиции (игнорируется при 903 = 0 или P903 = 5).
RESET		Начатый выход в 0-позицию прерывается (торможение с темпом позиционирования), и вызов отменяется. Если привод уже выведен в 0-позицию, то сигнал "0-позиция определена" сбрасывается, и появляется сообщение "Конечное положение достигнуто".

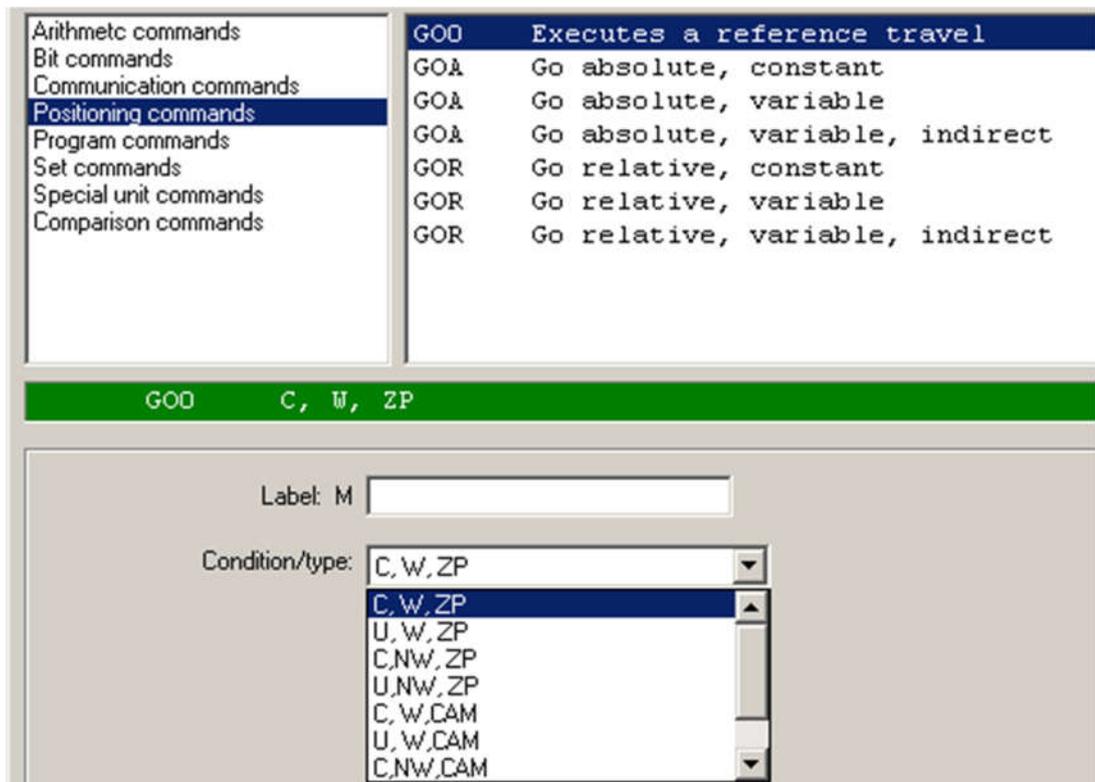


Рис. 43

Имеется специальная группа параметров используемых в процессе выполнения выхода в 0-позицию (см. Рис. 44). Рассмотрим их подробнее:

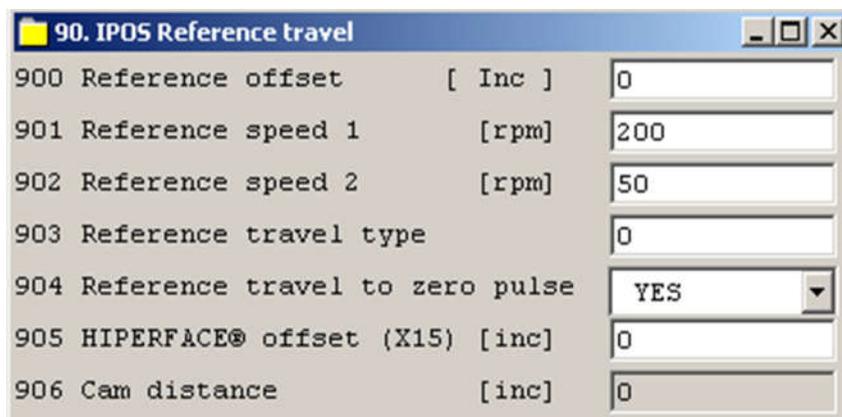


Рис. 44

P903 - определяет способ (стратегию) выхода в 0-позицию. Всего имеется 9 способов.

P901 и P902 - задают две скорости движения (соответственно быструю и медленную), используемые в процессе выхода в 0-позицию.

P904 - определяет точку «привязки» (точку окончания выхода в 0-позицию):

при “NO” - точкой «привязки» назначается спадающий фронт датчика, определяющего 0-позицию (точка «схода» с этого датчика);

при “YES” - точкой «привязки» назначается ближайшая к точке «схода» оборотная метка дорожки С энкодера (нулевой импульс).

P900 - смещение 0-позиции инкрементного или sin/cos-датчика. Используется в тех случаях, когда удобнее считать, что найденное положение соответствует не нулю, а некоторому другому значению. Это требуемое значение и вносится в P900. Параметру P900 полностью соответствует системная переменная H498 (см. Таблица 14).

Таблица 14

№	Имя в компиляторе / ассемблере	Описание
498	RefOffset / REF.OFFSET	<p>СЧИТЫВАНИЕ и УСТАНОВКА значения для смещения 0-позиции. Значения H498 и P900 идентичны. Эта функция смещения 0-позиции обеспечивает сдвиг машинного нуля без физического перемещения привода в новую 0-позицию. Действительно следующее уравнение:</p> <p>Машинный нуль = 0-позиция + смещение 0-позиции.</p> <p>Привод перемещается в 0-позицию и останавливается в этом положении. После выхода в 0-позицию машинный нуль только рассчитывается по значениям 0-позиции и ее смещения.</p> <p>Значение смещения 0-позиции задается пользователем в своих единицах измерения.</p> <p>Диапазон настройки: $-2^{31} \dots 0 \dots +2^{31} - 1$</p>

P905 - смещение для датчика Hiperface (если он установлен на валу двигателя и подключен к разъему X15).

Рассмотрим подробнее стратегии выхода в 0-позицию (P903). Они представлены в виде диаграмм на Рис. 45 - Рис. 52.

На диаграммах использованы следующие обозначения:

$nRef1$ = Скорость 1 выхода в 0-позицию (P901);

$nRef2$ = Скорость 2 выхода в 0-позицию (P902);

Возможные исходные положения привода перед выходом в 0-позицию:

- [1] между датчиком 0-позиции и правым аппаратными конечными выключателями (АКВ);
- [2] на зоне срабатывания датчика 0-позиции;
- [3] между датчиком 0-позиции и левым АКВ;

LHWLS = левый (limit switch CCW) АКВ (нормально замкнутый);

RHWLS = правый (limit switch CW) АКВ (нормально замкнутый);

CAM = дискретный датчик 0-позиции (нормально разомкнутый);

RefCAM = 0-позиция на датчике 0-позиции: привод перемещается в это положение, если в аргументе команды выхода в 0-позицию **GOO** указано CAM.

RefZP = 0-позиция на нулевом импульсе: привод перемещается в это положение, если в аргументе команды выхода в 0-позицию **GOO** указано ZP (Zero Pulse);

RefOffCAM = смещение 0-позиции при выходе на датчик CAM;

RefOffZP = смещение 0-позиции при выходе на нулевой импульс ZP;

MZP = машинный нуль.

Выполняя 0 способ выхода в 0-позицию (Рис. 45), привод из исходного положения движется против часовой стрелки **до первого нулевого импульса** энкодера. Для выхода в 0-позицию используется только медленная скорость $nRef2$ (P902). Если выход в 0-позицию запускается сигналом на входе "REF. TRAVEL START", то в параметре P904 должно быть установлено "YES". Если выход в 0-позицию запускается IPOSplus - командой **Go0**, то в ней должен быть указан аргумент "ZP", при этом P904 не учитывается. Если P904 = "NO", или в команде **Go0** указан аргумент "CAM", то привод реагирует как в способе 5 и за 0-позицию принимает текущее положение.

В 5 и 8 способах (Рис. 46) 0-позицией является **текущее положение привода**. Аргументы "ZP" или "CAM" команды **Go0** и параметр P904 – не активны. Этот способ поиска 0-позиции подходит для работы с датчиками абсолютного отсчета и для приводов, требующих привязки к 0-позиции в режиме останова. В отличие от 5-го способа выполнение способа 8 возможно не только когда привод введен в эксплуатацию в режиме позиционирования, но в любом другом режиме.

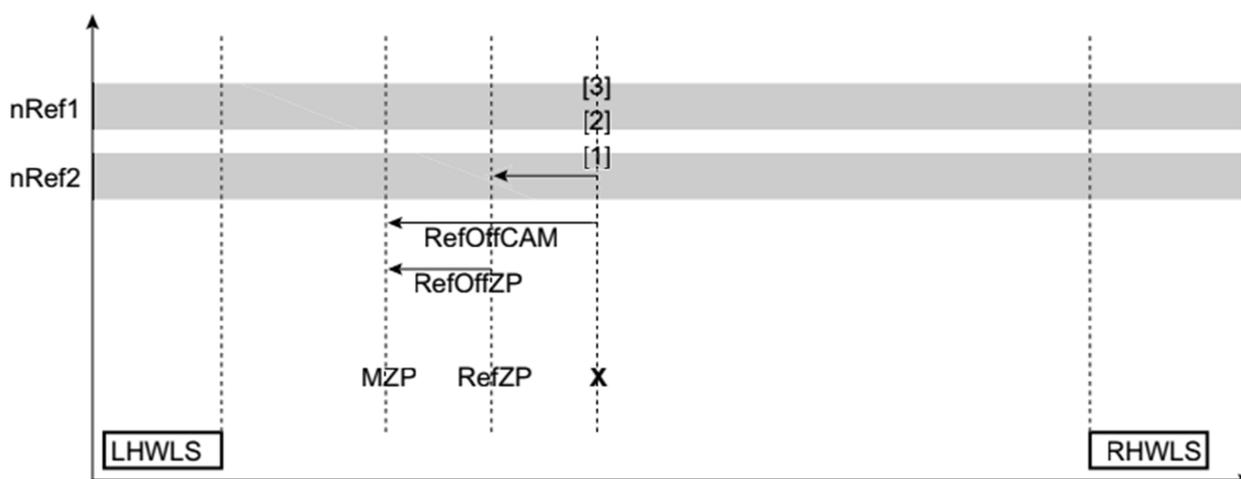


Рис. 45 Способ 0: Выход в 0-позицию на нулевой импульс

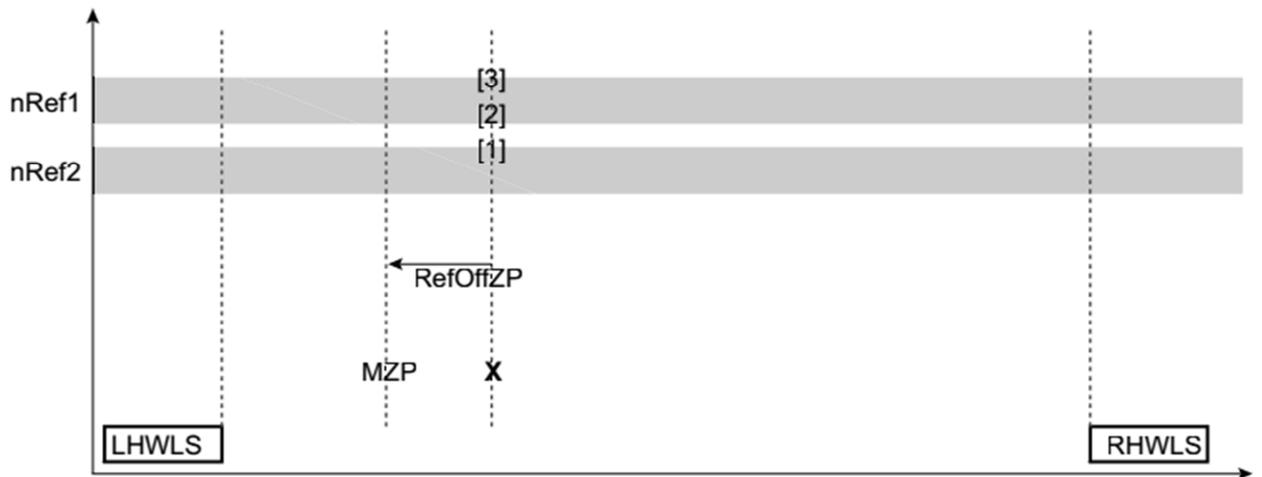


Рис. 46 Способы 5 и 8: Выход в 0-позицию не выполняется

В 1-м и 2-м способах выхода в нулевую позицию привязка машинного нуля происходит соответственно к левому и правому краям датчика нулевой позиции или первому нулевому импульсу за этим краем (Рис. 47, Рис. 48). Датчик нулевой позиции подключается к одному из дискретных входов запрограммированному на функцию "REFERENCE CAM". Поиск 0-позиции начинается на высокой скорости $nRef1$ (P901), соответственно для 1-го способа вращением против часовой стрелки, а для 2-го – по часовой. После срабатывания датчика нулевой позиции происходит переход на пониженную скорость $nRef2$. В момент «схода» с датчика (на спадающем фронте) происходит привязка датчика, если в параметре P904 установлено "NO" или в команде **Go0** установлен аргумент "CAM". Если в параметре P904 установлено "YES" или в команде **Go0** установлен аргумент "ZP", то привязка осуществляется к ближайшей нулевой метке энкодера после спадающего фронта.

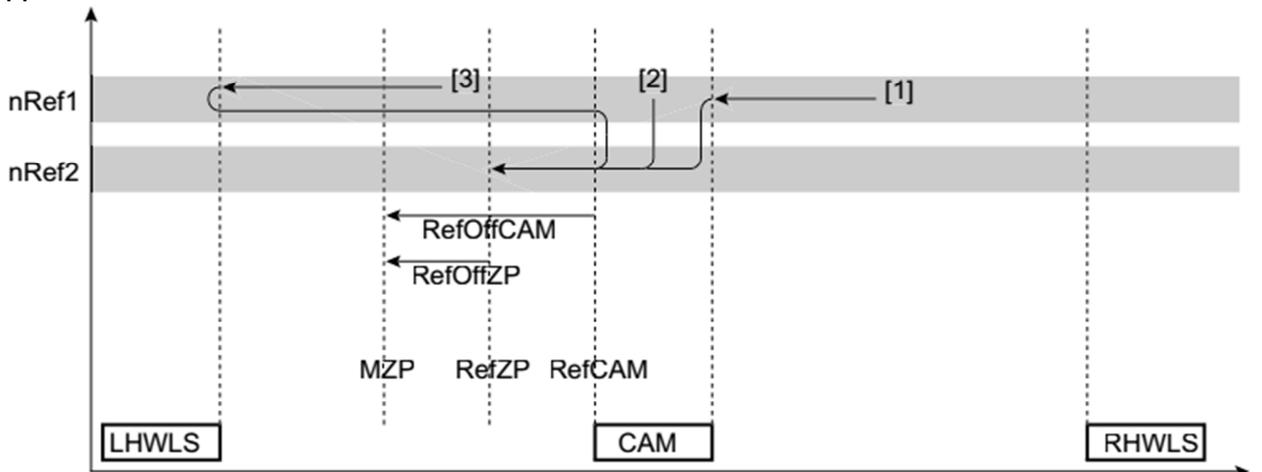


Рис. 47 Способ 1: Привязка к левому краю датчика 0-позиции

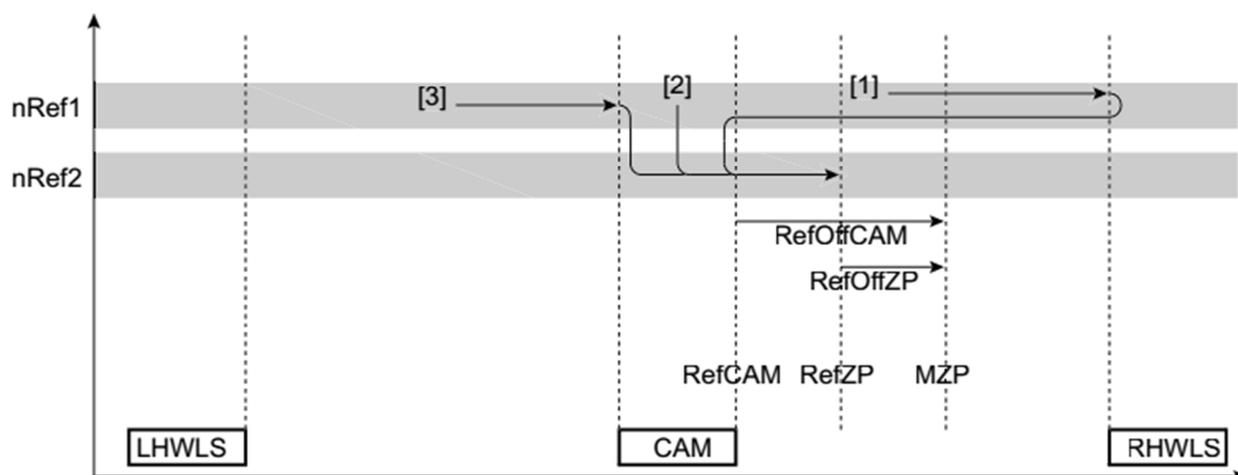


Рис. 48 Способ 2: Привязка к правому краю датчика 0-позиции

В 3-м и 4-м способах привязка происходит соответственно к правому и левому аппаратным конечным выключателям (АКВ) и датчик нулевой позиции не требуется (Рис. 49, Рис. 50). В обоих случаях привод осуществляет перемещение на высокой скорости $nRef1$ в направлении требуемого АКВ. Достигнув его, понижает скорость и изменяет направление движения. Привязка делается в момент «схода» с АКВ или к следующей за ним нулевой метке энкодера, в зависимости от настроек P904 или аргументов команды **Go0**.

Датчики АКВ подключаются к двоичным входам, которые программируются на следующие функции: /lim.switch CW – правый АКВ, /lim.switch CCW - левый АКВ.

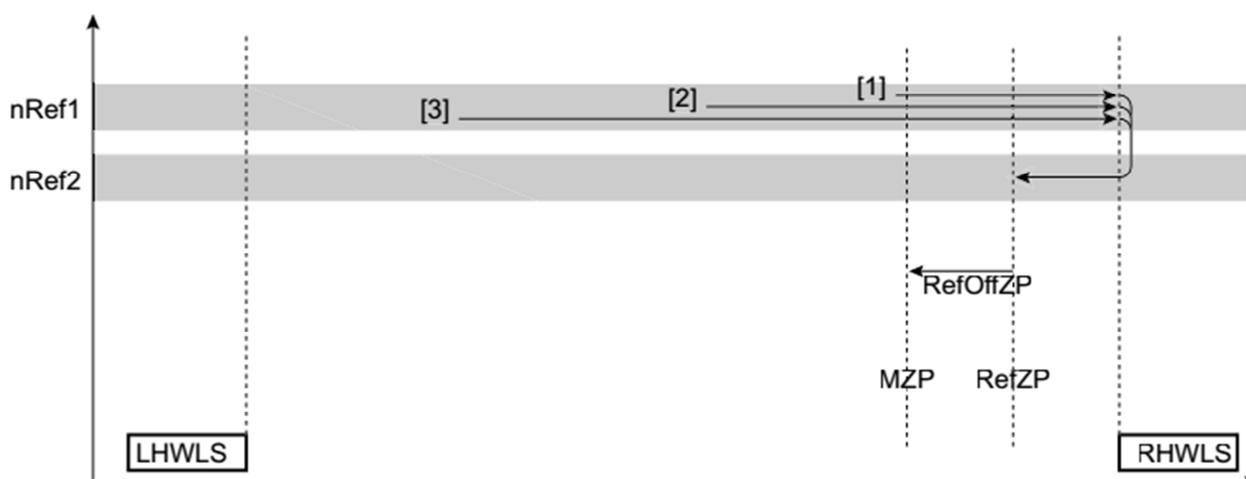


Рис. 49 Способ 3: Правый конечный выключатель

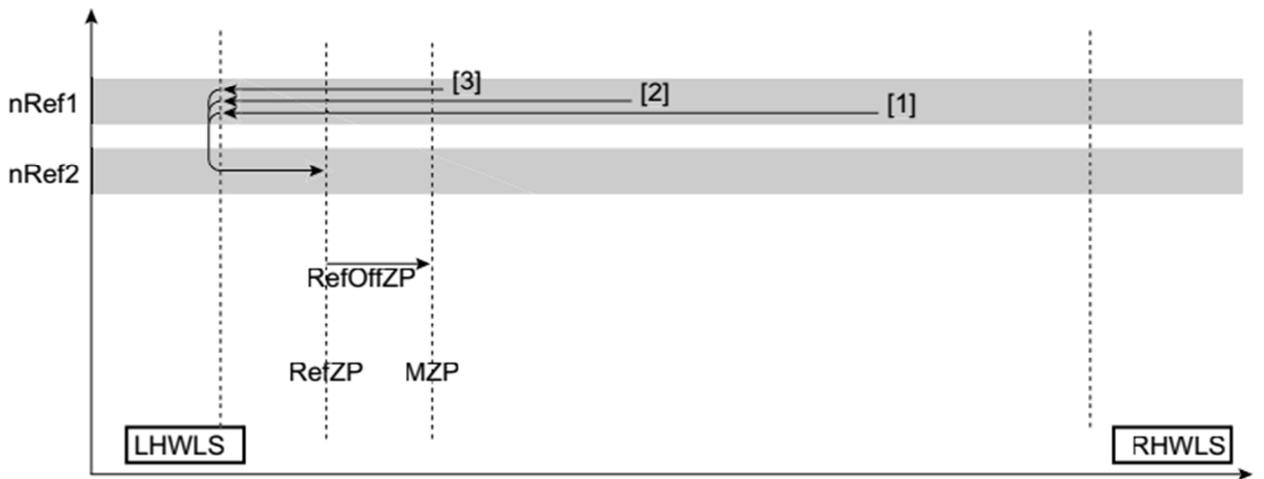


Рис. 50 Способ 4: Левый конечный выключатель

В случаях, когда датчик нулевой позиции установлен вплотную к АКВ используют 6 и 7 способы выхода в нулевую позицию (Рис. 51, Рис. 52).

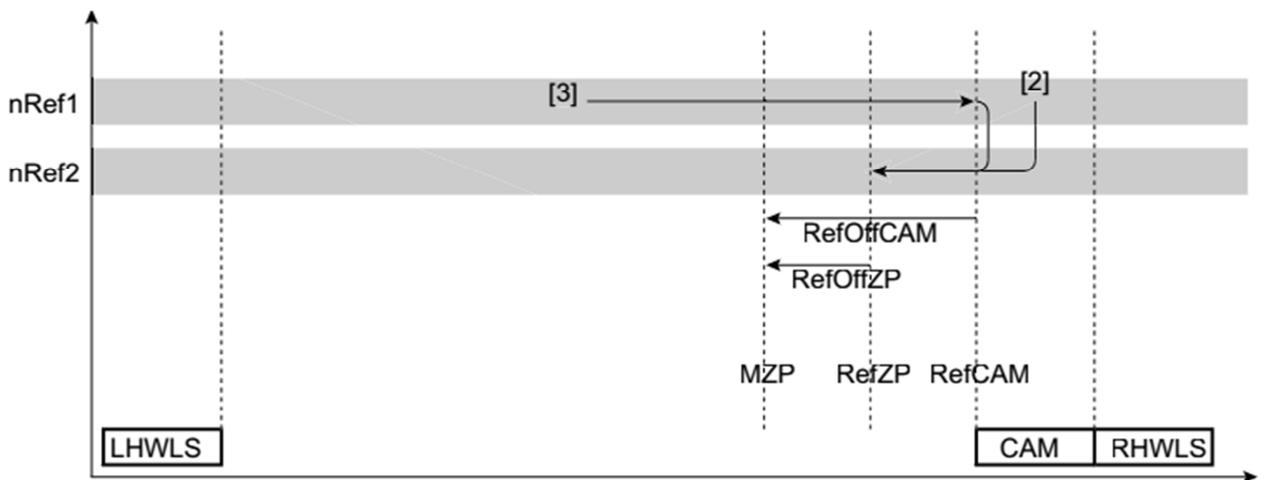


Рис. 51 Способ 6: Датчик 0-позиции в зоне срабатывания правого конечного выключателя

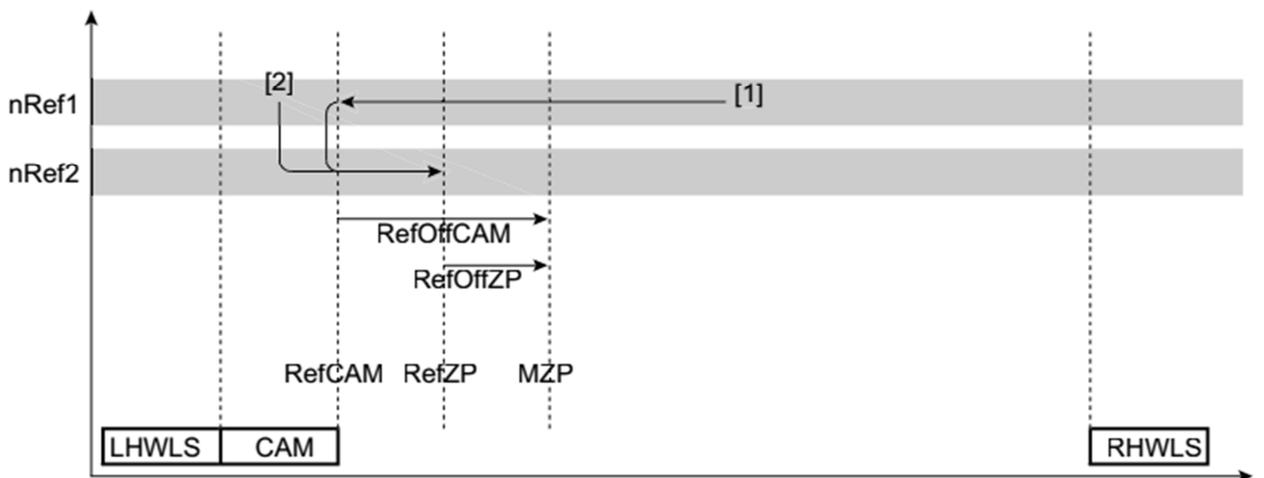


Рис. 52 Способ 7: Датчик 0-позиции в зоне срабатывания левого конечного выключателя

2.4. Команды управления позиционированием.

По умолчанию все вычисления и обработку команд позиционирования программа IPOSplus выполняет из соотношения 4096 инкремента = 1 оборот вала двигателя. Даже если контур положения замкнут по датчику с другим числом меток на оборот, то контроллер осуществляет приведение этого числа к 4096. Поэтому задание перемещения интерполятору также должно задаваться исходя из этого соотношения.

Все команды позиционирования (за исключением команды **Go0**) делятся на две группы (Рис. 53):

- 1) Команды абсолютного позиционирования **GOA**, задающие перемещение в конечную точку (target position) относительно машинного нуля;
- 2) Команды относительного позиционирования **GOR**, задает перемещение в конечную точку (target position) относительно текущего положения.

У этих команд есть атрибут (Рис. 53), который может принимать значения WAIT или NOWAIT. В случае если атрибут = WAIT, то следующая за командой позиционирования инструкция начнет выполняться только когда привод придет в заданную точку (войдет в окно конечного положения (P922)). При другом значении этого атрибута, сразу после запуска начала обработки команды позиционирования, программа переходит к выполнению следующей команды. Т.е. выполнение программы продолжается, в то время как привод еще совершает перемещение.

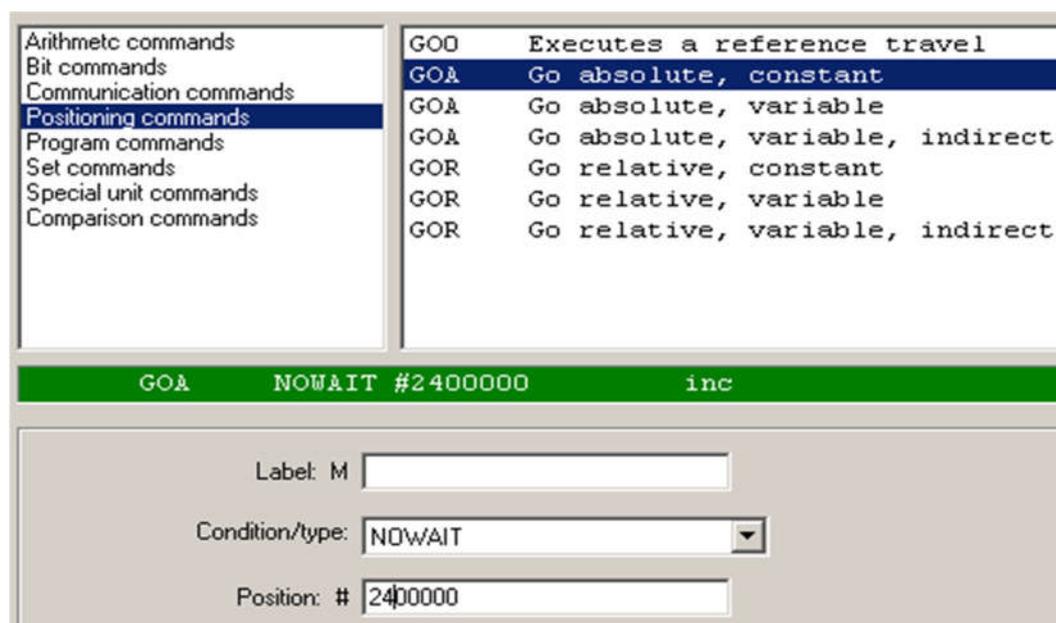


Рис. 53 команды управления позиционированием

Для команд и абсолютного, и относительного перемещений применяется три способа задания значения перемещения:

- С помощью константы **K**;
- Заданием номера переменной **H**, в которой хранится значение перемещения;
- Путем указания на переменную **[H]**, в которой находится адрес другой переменной, где хранится величина перемещения (indirect – косвенная адресация).

Поэтому имеется всего 6 разновидностей команд перемещения (Рис. 53).

В командах с указанием величины перемещения в виде константы есть возможность задавать перемещение не в инкрементах, а своих (пользовательских) единицах измерения (например: мм, об. или др.). Чтобы сделать это потребуются в окне редактора ASSEMBLER настроить масштабные коэффициенты “Numerator” и “Denominator” и указать единицы измерения “Unit” (Рис. 54). Пересчет выполняется по следующей формуле:

$$\text{Число инкрементов} = \frac{\text{NUMERATOR}}{\text{DENOMINATOR}} \cdot \text{Пользовательская единица измерения}$$

Пример: если **12376** инкрементам датчика положения соответствуют **120 мм** перемещения рабочего органа. Тогда: NUMERATOR (числитель) = 12376, DENOMINATOR (знаменатель) = 120, UNIT (единица) = мм.

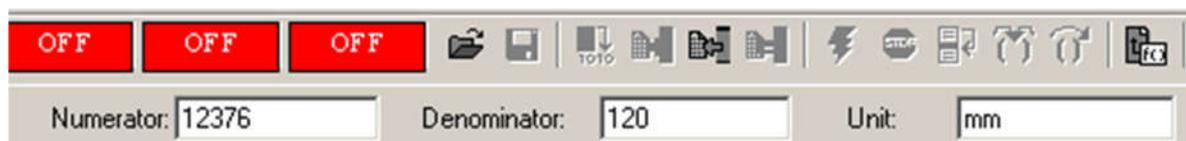


Рис. 54

В двух других способах задания перемещения (с помощью переменной и путем косвенной адресации) его величина задается только в инкрементах!

Для остановки привода находящегося в процессе отработки заданного перемещения в группе специальных команд есть команда **ASTOP** (Рис. 55).

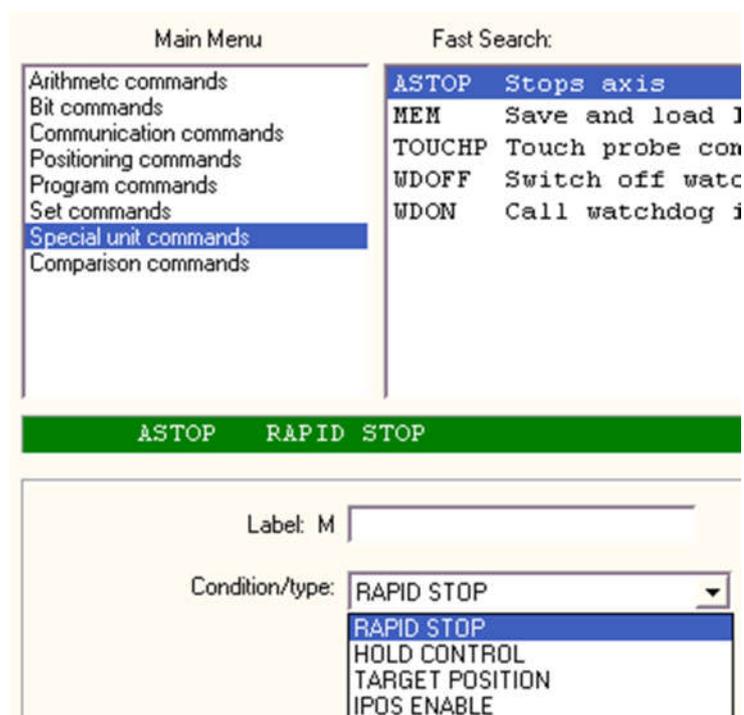


Рис. 55

Каким образом должно быть прервано начатое ограничение определяет значение аргумента команды:

- При выполнении команды с аргументом **RAPID STOP**, в управляющем слове N484 устанавливается 1-й бит (“no enable”). Это приводит к торможению привода с темпом быстрой остановки с последующим наложением механического тормоза (при его наличии и активированной функции торможения). При этом значение переменной N492 (target position) – сохраняется.
- При выполнении команды с аргументом **HOLD CONTROL**, в управляющем слове N484 устанавливается 13-й бит (“hold control”). Это приводит к торможению привода с темпом, который задан в P131/133, с последующим электрическим удержанием вала двигателя в неподвижном положении (без использования механического тормоза). Значение переменной N492 – также сохраняется.
- При выполнении команды с аргументом **TARGET POSITION**, значения битов переменной N484 не меняются. Но привод начинает торможение с темпом P911/912, для этого пересчитывается и заменяется конечное значение точки перемещения N492 (target position).

При использовании аргументов **RAPID STOP** и **HOLD CONTROL**, для продолжения движения следует выполнить команду **ASTOP** с аргументом

IPOS ENABLE. Движение будет возобновлено в точку, значение которой сохранилось в H492. После выполнения команды **ASTOP** с аргументом TARGET POSITION, перемещение можно возобновить только командами **GOA** или **GOR**. Выполнение команды **ASTOP** с аргументом IPOS ENABLE в этом случае не требуется.

2.5. Контроль работы привода в режиме позиционирования.

Кроме аппаратных конечных выключателей диапазон значений перемещений, в котором принимаются и обрабатываются команды позиционирования, можно ограничить с помощью программных конечных выключателей P920 и P921 (Рис. 56). Если значения P920 и P921 равны 0, то это значит, что программные конечные выключатели не задействованы.

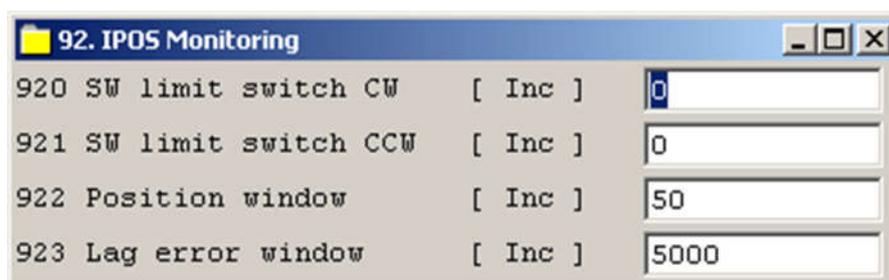


Рис. 56

P922 – определяет окно вокруг значения конечной точки позиционирования (target position). В конце отработки заданного перемещения, когда привод, приближаясь к значению target position, входит в внутрь этого окна системой управления формируется сигнал о достижении заданного конечного положения “IPOS in position” (в слове состояния H473, бит 19). Именно этот сигнал указывает на завершение перемещения и используется, в частности, для организации работы атрибута WAIT в командах позиционирования и др.

P923 – определяет допустимое значение отклонения в процессе движения действительного положения от setpoint position (H491), формируемого интерполятором для управления процессом движения. По-другому эту величину можно назвать допустимым отклонением от траектории движения. Реакция на выход за пределы допустимого отклонения задается в P834.

Значениям параметров P920-P923 полностью соответствуют величины системных переменных H493, H494, H496 и H497 (Таблица 15).

Таблица 15

№	Имя в компиляторе / ассемблере	Описание
493	PosWindow / POS.WINDOW	СЧИТЫВАНИЕ и УСТАНОВКА значения для окна положения. Значения H493 и P922 идентичны. Окно положения устанавливает диапазон удаленности от конечного положения (H492), заданного какой-либо командой перемещения или останова (GOx или ASTOP TARGET POSITION). Как только привод входит в окно положения, генерируется сигнал IPOS In Position (IPOS: заданное положение достигнуто). Этот сигнал можно использовать через какой-либо двоичный выход (которому назначается функция IPOS In Position) и через системную переменную H473, бит 19. Сигнал IPOS In Position сбрасывается сразу, если подается новая команда GOx. Контроль окна положения выполняется обязательно, если активен режим работы с IPOS (P700). Размер окна на точность позиционирования не влияет. Диапазон настройки: 0...50...2 ¹⁵ – 1 инкр.
494	LagWindow / LAG WINDOW	СЧИТЫВАНИЕ и УСТАНОВКА значения для окна допуска погрешности запаздывания. Значения H494 и P923 идентичны. Окно допуска погрешности запаздывания задает максимально допустимую разность между текущей уставкой положения, которую генератор темпа постоянно обновляет с циклом в 1 мс, и действительным положением привода. Если эта разность превышает, то подается сигнал о погрешности запаздывания (F42). Реакция на F42 задается в параметре P834 "Реакция на ПОГРЕШНОСТЬ ЗАПАЗДЫВАНИЯ". Отмена: настройка P923 "Окно допуска погрешности запаздывания" = 0 отменяет контроль погрешности запаздывания. Диапазон настройки: 0...5000...2 ³¹ – 1 инкр.
495	LagDistance / LAG DISTAN	СЧИТЫВАНИЕ значения интервала запаздывания. Абсолютное значение текущего интервала запаздывания при позиционировании (разность между уставкой положения и действительным положением привода). Диапазон значений: 0...2 ³¹ – 1 инкр.
496	SLS_right / SLS RIGHT	СЧИТЫВАНИЕ и УСТАНОВКА значения для правого ПКВ. Значения H496 и P920 идентичны. Ограничивает перемещение привода в правом направлении. Значение задается пользователем в своих (пользовательских) единицах измерения. Диапазон настройки: – 2 ³¹ ...0 своих единиц измерения...2 ³¹ – 1 инкр.
497	SLS_left / SLS LEFT	СЧИТЫВАНИЕ и УСТАНОВКА значения для левого ПКВ. Значения H497 и P921 идентичны. Ограничивает перемещение привода в левом направлении. Значение задается пользователем в своих (пользовательских) единицах измерения. Диапазон настройки: – 2 ³¹ ...0 своих единиц измерения...2 ³¹ – 1 инкр.

Среди системных переменных имеется переменная H495, которая отображает текущее значение отклонения от обрабатываемой траектории.

3. Задания для самостоятельной работы.

УПРАЖНЕНИЕ 1.

На Рис. 57 показана установка для приготовления водных растворов веществ. Она включает в себя:

- емкость для приготовления раствора со сливной магистралью и клапаном, отпираемым при подаче питания на электромагнит YA3;
- напорную водяную магистраль с клапаном, отпираемым электромагнитом YA1;
- бункер для растворяемого вещества с шибером, открываемым магнитом YA2;
- преобразователь MOVIDRIVE B, с подключенным к нему двигателем перемешивателя M1;
- дискретный датчик S1 верхнего уровня жидкости, подключенный к входу преобразователя DI05;
- тумблер «ВКЛ», через который на вход преобразователя DI4 подается сигнал запуска установки.

Электромагниты YA1-YA3 управляются от дискретных выходов преобразователя DO1-DO3.

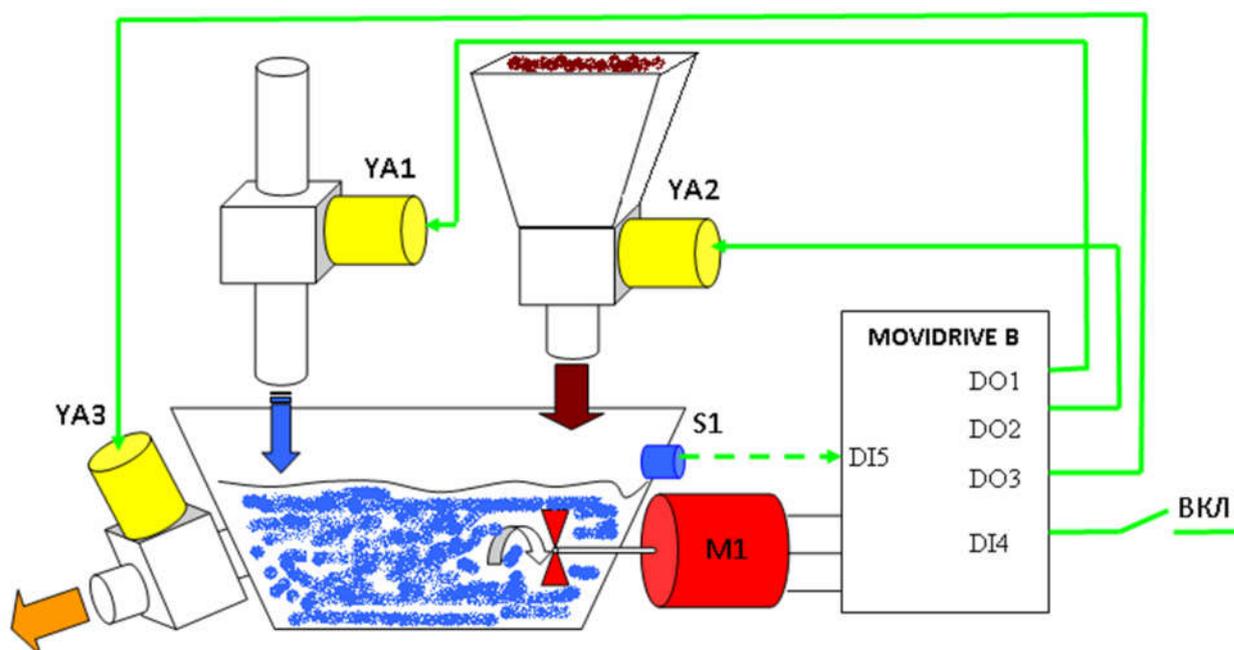


Рис. 57

Необходимо написать и отладить программу для контроллера IPOsplus, которая обеспечит работу установки по нижеописанному алгоритму.

Цикл работы установки:

1. При наличии сигнала «ВКЛ» - открыть напорную магистраль для заливки воды (YA1).
2. Через 15 секунд (или по срабатыванию датчика S1) закрыть напорную магистраль.
3. включить двигатель M1 для перемешивания с малой скоростью 200 об/мин.
4. Открыть шибер для засыпки порошка (YA2) на 10сек.
5. Для ускоренного растворения вещества включить на 15сек. перемешивание с большой скоростью 1500 об/мин.
6. Открыть сливной клапан (YA3) на 10сек. для выпуска готовой смеси.
7. Перейти к п.1.

Если в процессе работы установки сигнал снимается, то технологический процесс должен быть остановлен: двигатель – выключен, все клапаны закрыты.

Вариант решения: Привод должен быть введен в эксплуатацию в режиме регулирования скорости с использованием в качестве источника задания скорости внутренних фиксированных уставок. Для этого необходимо сделать настройки, показанные на Рис. 58 - Рис. 60.



Рис. 58

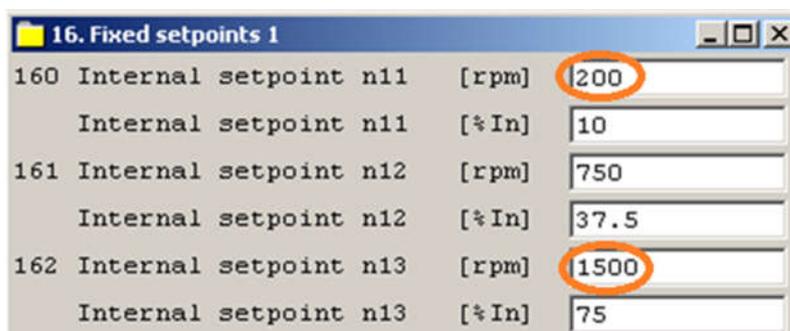


Рис. 59

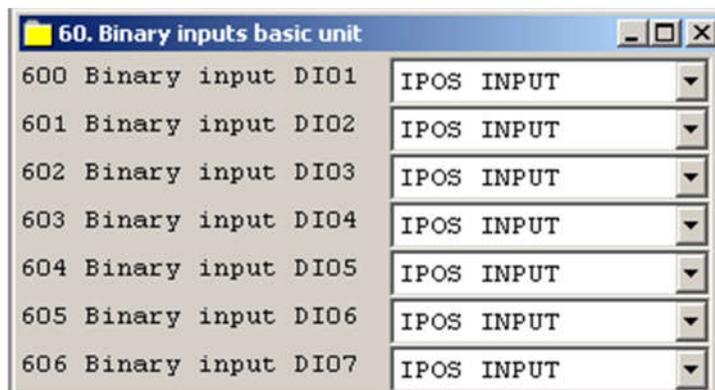


Рис. 60

Текст программы приведен ниже.

```

M0 :TASK    TASK2 START M10
M2 :JMP     LO I00000000000010000, M3
      JMP     LO I000000000000100000, M0
M3 :TASK    TASK2 STOP  M10
      SET     H484 = 2
      SET     H481 = 0
      JMP     UNCONDITIONED , M2

```

-----task2-----

```

M10 :SET    H484 = 2
M1  :SET    H481 = 0
      BSET   H481 .1 = 1
      WAIT   15000 ms

      BCLR   H481 .1 = 0
      BCLR   H484 .1 = 0
      BSET   H484 .2 = 1
      BSET   H484 .4 = 1
      BSET   H481 .2 = 1
      WAIT   10000 ms

      BCLR   H481 .2 = 0
      BSET   H484 .5 = 1
      WAIT   15000 ms

      SET    H484 = 2
      BSET   H481 .3 = 1
      WAIT   10000 ms
      JMP    UNCONDITIONED , M1
      RET

      END

```

УПРАЖНЕНИЕ 2.

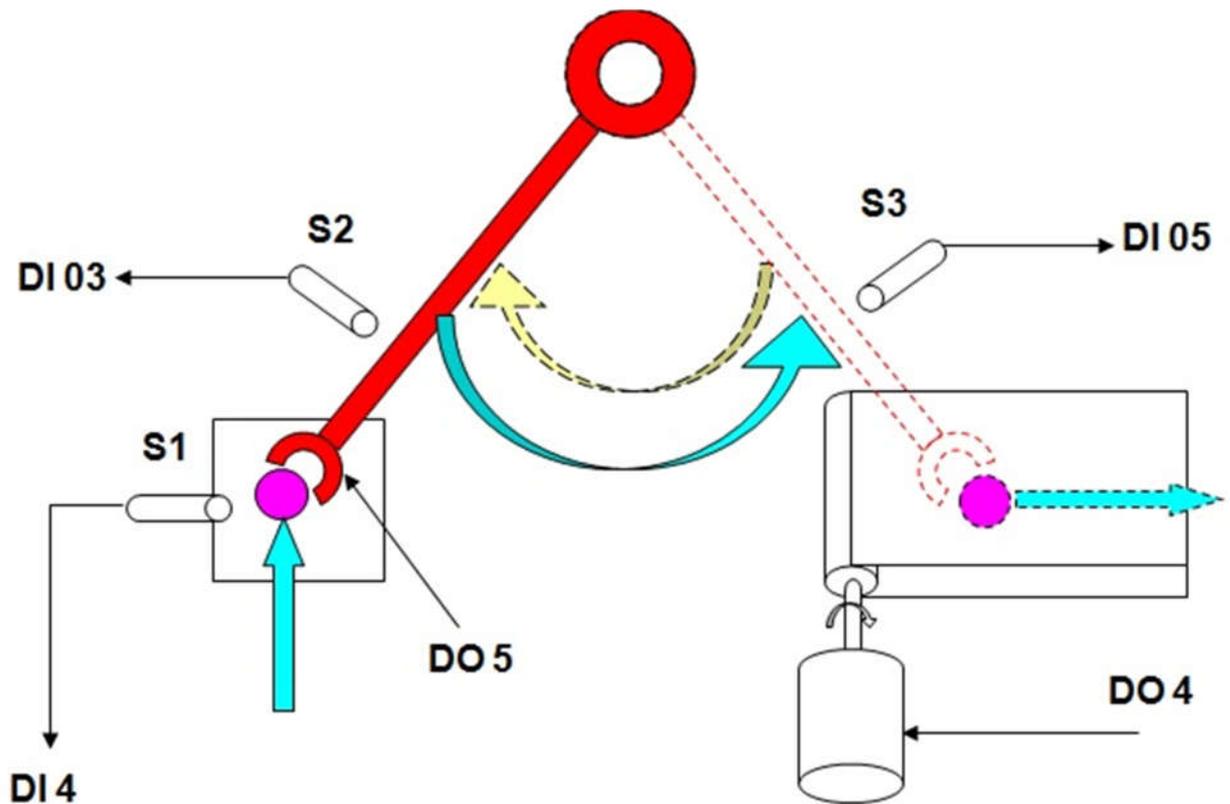


Рис. 61

Необходимо написать и отладить программу для контроллера IPOsplus, которая обеспечит работу манипулятора по нижеописанному алгоритму.

Цикл работы установки

1. По сигналу датчика наличия детали S1 повернуть схват манипулятора к технологической площадке с большой скоростью (1000об/мин). Остановка по сигналу датчика S2.
2. Включить схват (сигналом DO5). Пауза на срабатывание схвата 2 секунды.
3. Повернуть схват к транспортеру с низкой скоростью (200об/мин.). Остановка по сигналу датчика S3.
4. Выключить схват. Пауза на разжимание схвата - 2 секунды.
5. Включить сигналом DO6 транспортер на 10секунд.

УПРАЖНЕНИЕ 3.

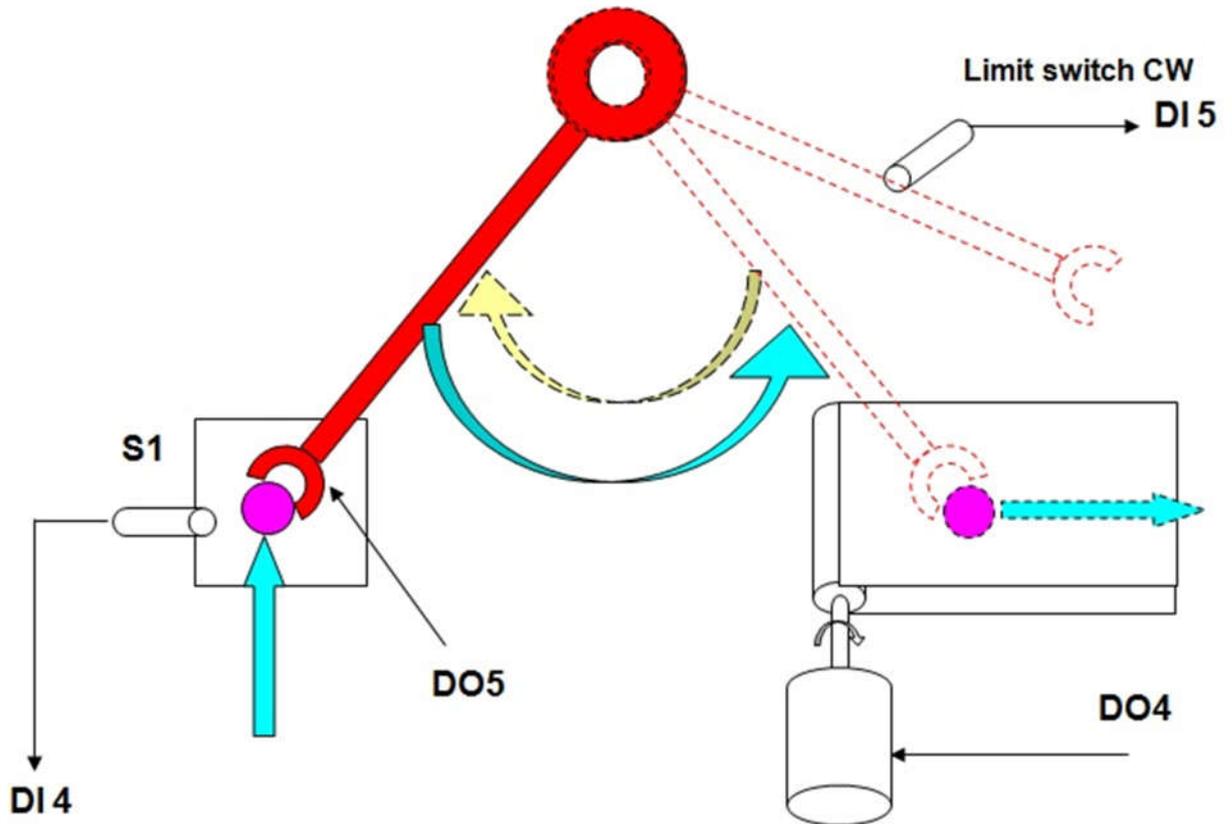


Рис. 62

Необходимо написать и отладить программу для контроллера IPOsplus, которая обеспечит работу манипулятора по нижеописанному алгоритму.

Работа установки начинается после получения сигнала «ВКЛ» на DI01. После его получения, необходимо сделать привязку начала отсчета датчика к Limit switch CW. Положение над технологической площадкой: -900000inc. Положение над транспортером: -300000inc

Цикл работы устройства

1. По сигналу датчика наличия детали S1 повернуть схват манипулятора к технологической площадке.
2. Включить схват (сигналом DO5). Пауза на срабатывание схвата 2 секунды.
3. Повернуть схват к транспортеру.
4. Выключить схват. Пауза на разжимание схвата - 2 секунды.
5. Включить сигналом DO4 транспортер на 10секунд.

Список литературы:

1. Руководство. Позиционирование и автоматическое управление циклом работы IPOsplus®. Издание 11320451/RU. 11/2004г. 354с.
2. Системное руководство MOVIDRIVE® MDX60B/61B. Издание 16838068/RU. 09/2010г. 650с.
3. Практика приводной техники – Сервоприводы. Издание 11322853/RU. 09/2006г. 141с.

Оглавление:

Введение.....	3
1. Система автоматического позиционирования и управления циклом работы IPOsplus.	4
1.1. Общее описание. Работа в редакторе ASSEMBLER.....	4
1.2. IPOs переменные.	8
1.3. Базовые команды IPOsplus.....	14
1.4. Управление задачами.	21
1.5. Работа с прерываниями.	24
2. Управление замкнутым электроприводом через программу IPOsplus. ...	28
2.1. Замкнутый электропривод в режиме регулирования скорости.....	28
2.2. Электропривод в режиме позиционирования.	31
2.3. Выход в 0-позицию.....	40
2.4. Команды управления позиционированием.	48
2.5. Контроль работы привода в режиме позиционирования.	51
3. Задания для самостоятельной работы.....	53
Список литературы:.....	59