



УСТРОЙСТВО
ЧИСЛОВОГО ПРОГРАММНОГО УПРАВЛЕНИЯ
НС - XXX

Руководство по эксплуатации.
«АРМД»

АННОТАЦИЯ

Данное руководство распространяется на устройства числового программного управления (УЧПУ) **NC-110, NC-200, NC-201, NC-201M, NC-202, NC-210, NC-220, NC-230, NC-301, NC-302** и **NC-310**.

Программа автоматической регистрации машинных данных (далее **«АРМД»**), встроенная в ПрО УЧПУ, выполняет сбор и сохранение в двоичных файлах (далее файлы мониторинга) информацию об эксплуатации оборудования. Анализ файлов мониторинга и формирование отчетов об эксплуатации оборудования должна выполнять внешняя программа мониторинга (далее ВПМ), разработанная самим пользователем УЧПУ или приобретенная в организации, специализирующейся на разработке ВПМ.

В данном документе рассмотрены особенности конфигурации **«АРМД»** и, данные необходимые для разработки ВПМ.

СОДЕРЖАНИЕ

1. КОНФИГУРАЦИЯ «АРМД»	5
1.1. СТРУКТУРА «АРМД»	5
1.2. НОМЕР ВЕРСИИ «АРМД»	6
1.3. ПОДГОТОВКА «АРМД»	6
1.4. СИНХРОНИЗАЦИЯ ВРЕМЕНИ УЧПУ И СЕРВЕРА	6
1.5. СРЕДА «АРМД»	8
1.6. ВВОД ПАРАМЕТРОВ И ПЕРЕМЕННЫХ «АРМД»	9
1.7. СОЗДАНИЕ ФАЙЛА СО СПИСКОМ ПРИЧИН ПРОСТОЯ	9
1.8. ЗАПУСК «АРМД»	10
1.9. ОСОБЕННОСТИ ПЕРЕМЕННЫХ «АРМД»	10
1.10. РЕЗУЛЬТАТ РАБОТЫ «АРМД»	10
1.11. ОСОБЕННОСТИ КОНФИГУРАЦИИ ИНСТРУКЦИЙ «АРМД»	10
1.11.1. Рекомендации для выбора объема дисков	11
1.11.2. Рекомендации для выбора дисков	11
1.11.3. Программная поддержка дисков	11
1.12. ФАЙЛ КОНФИГУРАЦИИ «АРМД» (MONCFG.INI)	12
1.12.1. Инструкции «АРМД»	13
1.12.1.1. Инструкция Path	13
1.12.1.2. Инструкция IP	14
1.12.1.3. Инструкция FileSize	14
1.12.1.4. Инструкция LocalFileSize	15
1.12.1.5. Инструкция MaxLocalSize	15
1.12.1.6. Инструкция DowntimeList	15
1.12.1.7. Инструкция MachineName	16
1.12.1.8. Инструкция RecordTime	16
1.12.1.9. Инструкция Port	17
1.12.1.10. Инструкция NetTimeSynchroInterval	17
1.12.1.11. Инструкция LocalTimeSynchroInterval	18
1.12.1.12. Инструкция Pro	18
1.12.2. Переменные «АРМД»	19
1.12.2.1. Принцип обработки APPROXIMATE	20
1.12.2.2. Переменная WorkMode	25
1.12.2.3. Переменная SystemState	25
1.12.2.1. Переменная MUSP	26
1.12.2.2. Переменная CONP	26
1.12.2.3. Переменная RISP	26
1.12.2.4. Переменная ASPEPN	27
1.12.2.5. Переменная SPEPNREQ	27
1.12.2.6. Переменная REAZ	27
1.12.2.7. Переменная COMU	28
1.12.2.8. Переменная CEFA	28
1.12.2.9. Переменная GFunctions	28
1.12.2.10. Переменная ProgramName	29
1.12.2.11. Переменная NumberProgramBlock	29
1.12.2.12. Переменная ToolNumber	30
1.12.2.13. Переменная CorrectorToolNumber	30
1.12.2.14. Переменная SwitchJOG	30
1.12.2.15. Переменная SwitchSpindle	30
1.12.2.16. Переменная SwitchFeed	30
1.12.2.17. Переменная Feed	31
1.12.2.18. Переменная SpindleSpeed	31
1.12.2.19. Переменная MachineDowntime	31
1.12.2.20. Переменная ProcessCommandLine	31
1.12.2.21. Переменная ProcessBlockLine	32
1.12.2.22. Переменная CommandLine	32
1.12.2.23. Переменная UAS	32
1.12.2.24. Переменная UVR	32
1.12.2.25. Переменная URL	32

1.12.2.26.	Переменная MesErrProg	33
1.12.2.27.	Переменная MesErrPLC.....	33
1.12.2.28.	Переменная AlarmErrPLC.....	34
1.12.2.29.	Переменная PartFinished	34
1.12.2.30.	Текстовые переменные.....	35
1.12.2.1.	Переменная SpindlePower.....	37
1.12.2.2.	Пример файла MONCFG.INI.....	38
1.12.1.	Обработка ошибок в инструкциях и переменных.....	39
1.13.	ФАЙЛЫ МОНИТОРИНГА	39
1.13.1.	Порядок создания и заполнения файлов мониторинга в «АРМД».....	39
1.13.2.	Принцип создания имени для файлов мониторинга	41
1.14.	ПОСЛЕДОВАТЕЛЬНОСТЬ ДЕЙСТВИЙ ДЛЯ ЗАПУСКА «АРМД»	42
2.	ПРИЛОЖЕНИЕ А (ОБЯЗАТЕЛЬНОЕ)	43
2.1.	СТРУКТУРА ЗАПИСИ ДАННЫХ В ФАЙЛ МОНИТОРИНГА	43
3.	ПРИЛОЖЕНИЕ В (ОБЯЗАТЕЛЬНОЕ)	49
3.1.	ВСПОМОГАТЕЛЬНАЯ ПРОГРАММА «MONITOR1»	49
4.	ПРИЛОЖЕНИЕ С (ОБЯЗАТЕЛЬНОЕ)	50
4.1.	ВСПОМОГАТЕЛЬНАЯ ПРОГРАММА «BSSERVER»	50
5.	ПРИЛОЖЕНИЕ D	51
5.1.	ОСОБЕННОСТИ ПРИМЕНЕНИЯ ПЕРЕМЕННЫХ.....	51
6.	ПРИЛОЖЕНИЕ E	52
6.1.	СООТВЕТСТВИЕ ПЕРЕМЕННЫХ «АРМД» СИГНАЛАМ ИНТЕРФЕЙСА PLC.	52
7.	ПРИЛОЖЕНИЕ F	53
7.1.	ПРОГРАММА ARMD CONVERTER.....	53
	ПЕРЕЧЕНЬ СОКРАЩЕНИЙ.....	56
	ПЕРЕЧЕНЬ ТЕРМИНОВ	56

1. Конфигурация «АРМД»

1.1. Структура «АРМД»

Структура выполнения автоматической регистрации машинных данных в файлах мониторинга представлена на рисунке 1.

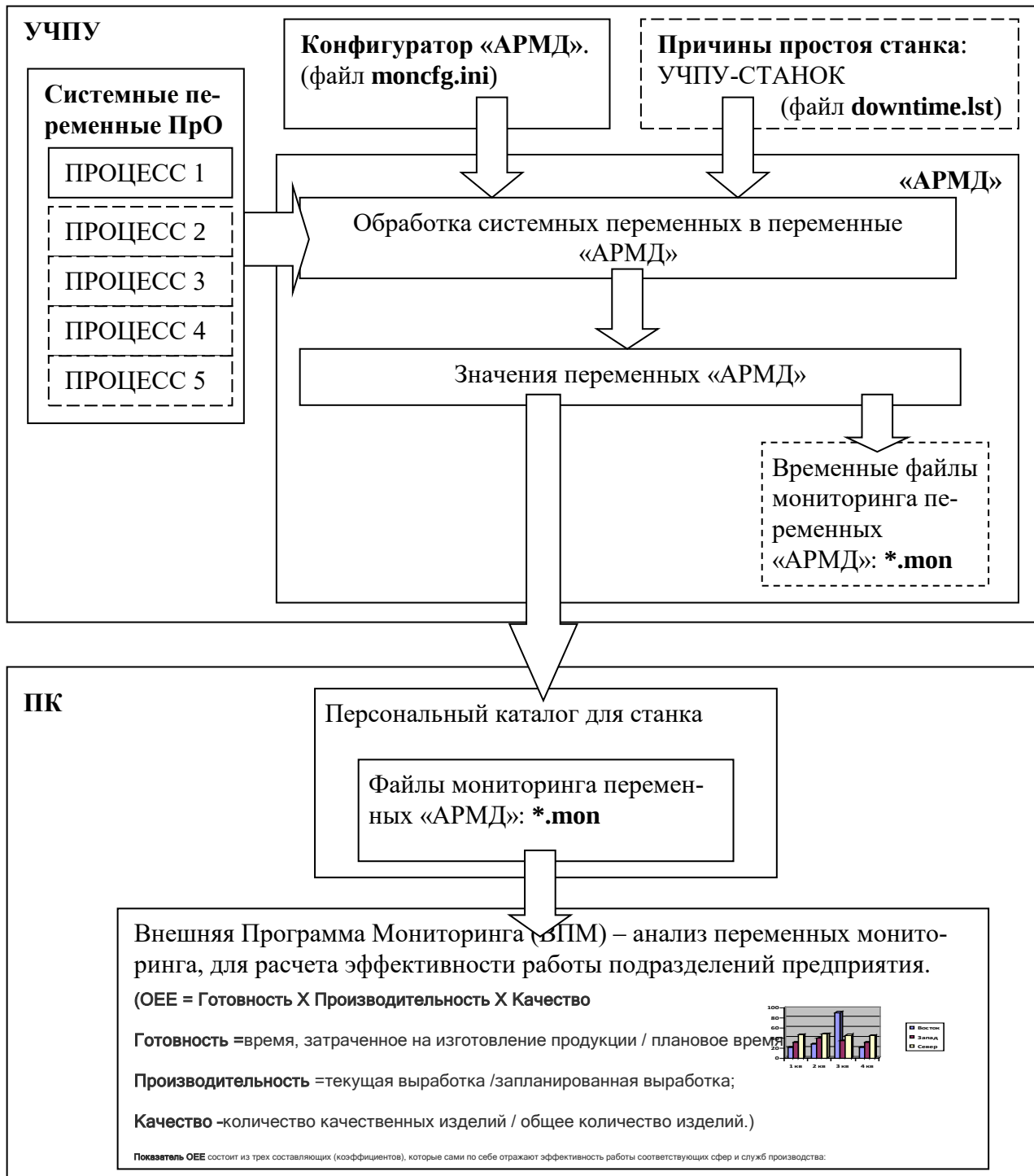


Рисунок 1.

1.2. Номер версии «АРМД»

Задача «АРМД» в ПрО не имеет собственного номера. Все инструкции и переменные «АРМД» в этом документе соответствуют номеру версии ПрО X.86.12P.

1.3. Подготовка «АРМД»

Конфигурация инструкций и переменных в «АРМД» определяется на основании задач, поставленных для ВПМ предприятием, использующим УЧПУ.

Параметры работы «АРМД» определяют 2 файла:

1. Файл конфигурации «АРМД» (обязательный): **moncfg.ini**
2. файл причин простоя (не обязательный): **downtime.lst**

Все значения инструкций для конфигурации «АРМД» ее переменные и их параметры должны быть установлены разработчиком ВПМ в файле **moncfg.ini**.

В УЧПУ файл **moncfg.ini** должен быть расположен в том каталоге, откуда запускается программа **vidmem.exe**. Путь к файлу **vidmem.exe** определен в файле **autoexec.bat** (обычно C:\CNC32\MP0).

Если «АРМД» должна обеспечить выбор и сохранение причин простоя оборудования, то в каталоге УЧПУ C:\CNC32\MP0 должен быть файл, содержащий список причин его простоя. Рекомендуются назвать этот файл именем: **downtime.lst**. Способы создания файла **downtime.lst** в УЧПУ рассмотрены в параграфе 1.7.

1.4. Синхронизация времени УЧПУ и сервера

Время УЧПУ базируется на таймере 8254, расположенном на периферийной плате УЧПУ. Это время может постоянно изменяться по отношению ко времени, установленному на сервере, на котором работает ВПМ. Поскольку время события, произошедшее в УЧПУ, фиксируется в файлах мониторинга со временем УЧПУ, а анализ этого события на сервере выполняется по времени сервера, то рекомендуется настроить параметры синхронизации времени УЧПУ по времени сервера. В противном случае результаты обработки данных файла мониторинга в ВПМ могут быть не корректными.

Синхронизация времени в УЧПУ может выполняться следующими способами:

- 1) Сетевая синхронизация времени - синхронизация времени УЧПУ по времени на сервере, на котором выполняется ВПМ.

Для установки интервала времени между последовательными запросами сетевой синхронизации в файле **moncfg.ini** должно быть установлено значение в инструкции **NetTimeSynchroInterval** больше 0;

- 2) Локальная синхронизация времени - синхронизация времени УЧПУ по времени в **BIOS** на материнской плате УЧПУ.

Для установки интервала времени между последовательными запросами локальной синхронизации в файле **moncfg.ini** должно быть установлено значение в инструкции **LocalTimeSynchroInterval** больше 0.

Сетевая синхронизация времени выполняется на этапе запуска задачи «АРМД» и далее через равные интервалы времени **NetTimeSynchroInterval** до выключения УЧПУ или разрыва связи с сервером. В случае восстановления связи между УЧПУ и сервером сетевая синхронизация времени УЧПУ также восстанавливается.

Выполнение локальной синхронизации зависит от значения интервала локальной синхронизации времени **LocalTimeSynchroInterval** и значения сетевой синхронизации времени **NetTimeSynchroInterval**. Существуют следующие 4 случая этой зависимости:

- a) **NetTimeSynchroInterval=0** и **LocalTimeSynchroInterval=0**
- в этом случае синхронизация времени в УЧПУ не выполняется.
- b) **NetTimeSynchroInterval=0** и **LocalTimeSynchroInterval>0**
- в этом случае выполняется однократная попытка сетевой синхронизации времени УЧПУ по времени сервера сразу после запуска задачи «АРМД», и далее выполняется только локальная синхронизация времени в УЧПУ.
- c) **NetTimeSynchroInterval>0** и **LocalTimeSynchroInterval<=NetTimeSynchroInterval**
в этом случае оба способа синхронизации выполняются независимо друг от друга с заданными для них интервалами времени.
- d) **NetTimeSynchroInterval>0** и **LocalTimeSynchroInterval>NetTimeSynchroInterval**
в этом случае локальная синхронизация времени будет выполняться только в случае разрыва связи между УЧПУ и сервером. Точка начала отсчета времени интервала локальной синхронизации будет соответствовать точке последней успешной сетевой синхронизации. Локальная синхронизация будет действовать до восстановления связи между УЧПУ и сервером.

Для сетевой синхронизации времени в инструкции **Port** в файле **moncfg.ini** должен быть задан номер порта устанавливаемого соединения (по умолчанию **53847**), а также на компьютере, IP адрес которого указан в файле **moncfg.ini** в инструкции **IP**, должно работать приложение «BSServer» к задаче «АРМД» или аналогичное приложение, встроенное в ВПМ.

Исходные тексты приложения «BSServer», поставляются в электронном виде по запросу пользователя.

Если синхронизация выполнена успешно, то в файле мониторинга появляется соответствующее сообщение: «**TIME_SYNC**».

1.5. Среда «АРМД»

В режиме работы УЧПУ «КОМАНДА» для «АРМД» создана специальная среда, представленная на рисунке 2. Вход в среду «АРМД» выполняется функциональной клавишей «F5» из главного меню режима «КОМАНДА». Если при входе в среду «АРМД» выводится сообщение «АРМД не сконфигурирована», то это означает, что в каталоге C:\CNC32\MP0 отсутствует файл **moncfg.ini**. В среде «АРМД» можно выполнить следующие действия:

- 1) Выбор причины простоя оборудования. Этот выбор может быть сделан функциональной клавишей «F1» (Причины простоя).
- 2) Просмотр списка переменных «АРМД» сконфигурированных в файле **moncfg.ini**. Этот выбор может быть сделан функциональной клавишей «F2» (АРМД парам.). Если при просмотре списка переменных «АРМД» переменная отсутствует, то это означает, что она не сконфигурирована в файле **moncfg.ini**, либо записана в файле **moncfg.ini** не правильно.
- 3) Просмотр инструкций и переменных «АРМД», в определении которых были обнаружены ошибки в момент ее запуска. Этот выбор может быть сделан функциональной клавишей «F3» (Анализ конфигу).
- 4) Просмотр сетевого ресурса, предназначенного для создания в нем файлов мониторинга, и их состояние может быть выбран функциональной клавишей «F4» (Сост-ние ресурса)
- 5) Просмотр предупреждения от «АРМД», выведенного на экран при загрузке УЧПУ, может быть выбран функциональной клавишей «F5» (Внутрен.сообщ.)

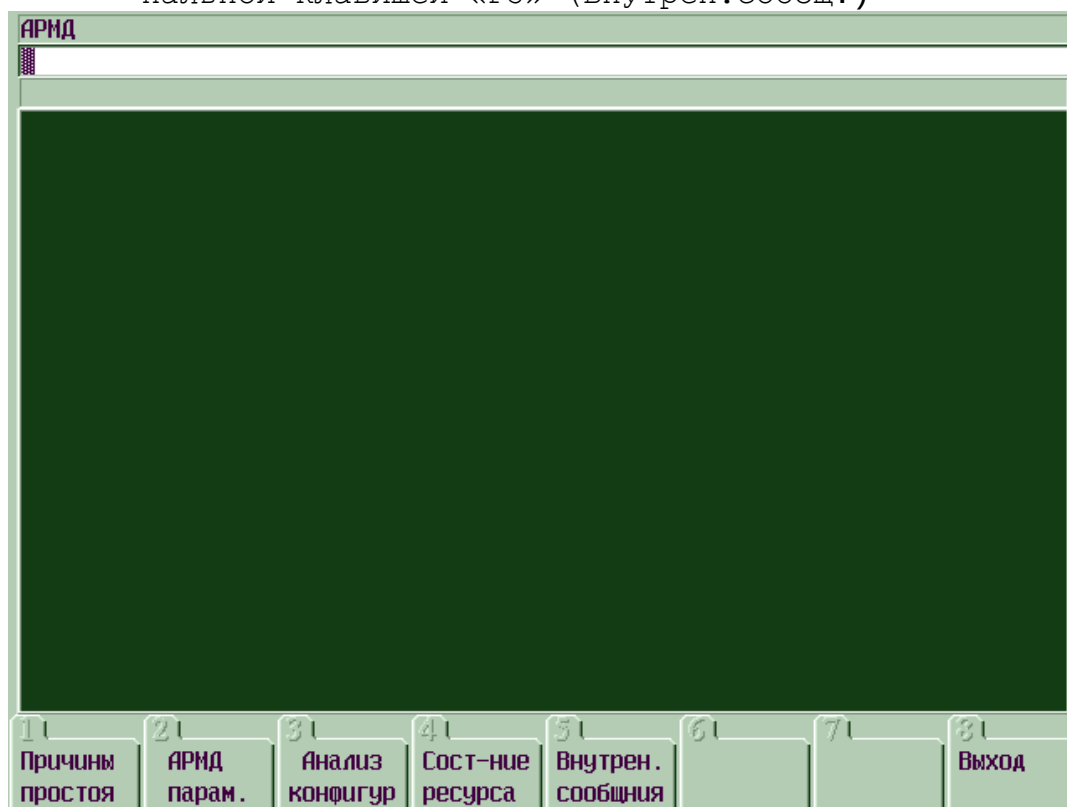


Рисунок 2

1.6. Ввод параметров и переменных «АРМД»

Ввод инструкций и переменных «АРМД» выполняется в файле **C:\CNC32\MP0\moncfg.ini**.

Создание и редактирование этого файла может быть выполнено только из внешней файловой оболочки, например «**Norton Commander**» или на внешнем ПК с последующим его копированием в каталог УЧПУ **C:\CNC32\MP0**.

Список параметров переменных «АРМД» представлен в таблицах 1 и 2. Выбор инструкций и переменных выполняется разработчиком ВПМ.

Любые изменения, выполненные в файле **moncfg.ini** с удаленного ПК по сети, будут активированы только после перезапуска УЧПУ.

1.7. Создание файла со списком причин простоя

Файл, содержащий список причин простоя УЧПУ, не является обязательным файлом задачи «АРМД». Поскольку этот файл загружается только при запуске задачи «АРМД», то любые изменения в этом файле будут активированы только после перезапуска УЧПУ.

Создание файла причин простоя УЧПУ в рабочем каталоге УЧПУ, обычно **C:\CNC32\MP0**, может быть выполнено одним из 2-х способов:

- 1) С поддержкой создания этого файла в ВПМ, если в ней встроен программный модуль аналогичный программе «**BSserver**».

В этом случае в рабочем каталоге УЧПУ автоматически создается дубликат внешнего файла причин простоя, расположенного в сетевом каталоге, путь к которому записан в инструкции **Path**, сразу после запуска «АРМД». Имя файла простоя в сетевом каталоге ПК должно быть идентично имени, записанному в файле **moncfg.ini** в инструкции **DownTimeList**.

Примечание. Применение программы «**BSserver**» представлено в параграфе 4.

- 2) Без поддержки создания этого файла в ВПМ. В этом случае файл причин простоя создается в рабочем каталоге УЧПУ в текстовом редакторе внешней файловой оболочки «**Norton Commander**», или на внешнем ПК с последующим его копированием в УЧПУ в каталог **C:\CNC32\MP0**.

Файл причин простоя должен быть создан разработчиком ВПМ совместно с ее заказчиком, и может иметь любое имя (в этом документе этот файл назван именем **downtime.lst**). Имя файла причин простоя должно быть определено в файле **moncfg.ini** в инструкции **DownTimeList**.

Стандартный список причин простоя приведен в Руководстве оператора «АРМД».

1.8. Запуск «АРМД»

Если файл **moncfg.ini** сконфигурирован, то запуск **«АРМД»** после включения УЧПУ выполняется автоматически сразу после загрузки файлов характеристики станка. Если файл **moncfg.ini** отсутствует, то **«АРМД»** не запускается, и ее запуск в текущем сеансе работы УЧПУ не возможен.

1.9. Особенности переменных «АРМД»

Значения переменных **«АРМД»** формируются на основании им соответствующих значений системных переменных ПрО, принадлежащих сконфигурированным в УЧПУ процессам. Исключением является формирование переменной причины простоя, значение которой для каждого процесса устанавливает оператор станка из среды **«АРМД»**.

1.10. Результат работы «АРМД»

Значения переменных **«АРМД»** сохраняются в двоичных файлах мониторинга по указанному пути в инструкции **Path** в файле **moncfg.ini**.

1.11. Особенности конфигурации инструкций «АРМД»

Конфигурация инструкций **«АРМД»** в файле **moncfg.ini** позволяет настроить обработку данных в ВПМ, как в реальном времени, так и по требованию.

1) Обработка файлов мониторинга в ВПМ в реальном времени.

Если обработку файлов мониторинга ВПМ должна выполнять в реальном времени, то в инструкциях **«АРМД»** для создания файлов мониторинга необходимо указать путь к внешнему сетевому ресурсу локальной сети (рекомендуется), или к сетевому ресурсу, принадлежащим УЧПУ.

ВПМ должна иметь доступ к сетевым ресурсам, на которых создаются файлы мониторинга.

2) Обработка файлов мониторинга в ВПМ по требованию.

Если обработку файлов мониторинга ВПМ должна выполнять по требованию, то в инструкциях **«АРМД»** для создания файлов мониторинга необходимо указать путь к любому сетевому ресурсу локальной сети (рекомендуется), или к сетевому ресурсу, принадлежащем УЧПУ, или к любому диску, подключенному к УЧПУ.

Для обработки файлов мониторинга они должны быть скопированы в каталог, доступный для ВПМ.

Конфигурация инструкций и переменных в **«АРМД»** позволяет настроить обработку всех необходимых данных для любого подразделения, использующего ВПМ. Количество используемых пере-

менных **«АРМД»** и периодичность их считывания позволяет определить скорость накопления регистрируемых данных в файлах мониторинга. На основании скорости накопления регистрируемых данных, способе обработки данных в ВПМ и необходимости резервного копирования файлов мониторинга можно определить:

- 1) объем каждого файла мониторинга;
- 2) объем диска для хранения файлов мониторинга.

1.11.1. Рекомендации для выбора объема дисков

Основным требованием к автоматической регистрации машинных данных обычно является исключение возможности потери информации о контролируемом объекте. В случае потери доступа к сетевому ресурсу, выделенного для файлов мониторинга, **«АРМД»** создаст локальный файл мониторинга в каталоге `C:\CNC32\MP0` с размером `LocalFileSize`. Так как минимальное количество локальных файлов мониторинга равно 2, то минимальный размер свободного места (V) на диске `C:` можно рассчитать по следующей формуле:

$$V = \text{LocalFileSize} * 2 + 1 \text{ [Мб]}$$

Объем памяти на диске `C:` для создания локальных файлов мониторинга резервируется созданием файла `C:\CNC32\MP0\allocsp.spc` с объемом V сразу при загрузке **«АРМД»**.

Если размер свободного места на диске `C:` при включении УЧПУ будет меньше, чем V , то **«АРМД»** прекращает работу.

Примечания.

1. **«АРМД»** не контролирует размер свободного места на сетевом ресурсе, в который записываются файлы мониторинга. Эту функцию должна обеспечить ВПМ или персонал, имеющий доступ к этому диску.
2. Несвоевременная обработка файлов мониторинга может повлечь потерю части регистрируемых машинных данных при переполнении объема диска.

1.11.2. Рекомендации для выбора дисков

Файлы мониторинга могут создаваться на сетевом ресурсе (рекомендуется), или любом диске доступном для УЧПУ, в том числе, USB-диске или USB-flash.

Для создания файлов мониторинга не рекомендуется использовать внутренний IDE(SATA)-flash-диск УЧПУ, если он является системным или содержит файлы характеристики и программу логики станка, т.к. это приведет к его быстрому износу и возможной потере этих файлов. Обычно внутренний диск представлен разделом `C`, но также это может быть любой другой раздел.

Тип и местоположение диска, на котором создаются файлы мониторинга, а также организация доступа к этим файлам определяется разработчиком ВПМ совместно с заказчиком ВПМ.

1.11.3. Программная поддержка дисков

Все диски и сетевые ресурсы, предназначенные для создания на них файлов мониторинга, должны быть определены и доступны

для базового программного обеспечения УЧПУ. Для этого должны быть выполнены настройки:

- драйверов для подключенных дисков мониторинга; инструкции **USB** или **IDE** в секции 4 файла **FCRSYS/MP0** (см. руководство по характеристике УЧПУ);
- локальной сети с использованием утилиты **SNT** (см. руководство по характеристике УЧПУ).

Если файлы мониторинга создаются на сетевом ресурсе, то необходимо выполнить следующие условия:

- на удаленном ПК должна быть установлена **OS** поддерживающая сетевой протокол с **RTOS32**, например: **WINDOWS NT, WINDOWS XP, WINDOWS 7**;
- УЧПУ и ПК объединены в единую рабочую группу;
- сетевой ресурс доступен для УЧПУ.

Примечание. Доступность дисков и сетевых ресурсов, подключенных к УЧПУ, определяется утилитой **SMP**.

1.12. Файл конфигурации «АРМД» (**moncfg.ini**).

Файл **moncfg.ini** содержит инструкции и переменные «АРМД».

Инструкции определяют параметры файлов мониторинга и их местоположение.

Переменные «АРМД» определяют список регистрируемых машинных данных (контроль состояния УЧПУ, станка и технологического процесса обработки детали).

Особенности записи инструкций и переменных «АРМД» в файле **moncfg.ini**:

- Каждая инструкция или переменная должна быть записана на отдельной строке.
- Порядок записи инструкций и переменных не имеет значения.
- Если в файле одна и та же инструкция или переменная записана дважды на разных строках, то действительно значение, записанное для нее ниже.
- Строки комментариев должны начинаться с символа «;».
- Строка считается комментарием, если имя инструкции или имя переменной в этой строке записаны не правильно.
- В записи имен инструкций и переменных прописные и заглавные буквы не разбираются и могут быть использованы пользователем для формального разбиения сложных имен переменных и инструкций.

Имена инструкций и переменных, записанных в файле **moncfg.ini**, проверяются при загрузке УЧПУ. Список инструкций и переменных, записанных с ошибками, может быть выведен на экране УЧПУ в среде «АРМД» элемент меню «**Анализ конфигурации**».

1.12.1. Инструкции «АРМД»

Инструкции «АРМД» в файле **moncfg.ini** определяют путь к файлам мониторинга, параметры файлов мониторинга и список номеров процессов сконфигурированных в системе, переменные которых необходимо контролировать.

Формат записи инструкций:

Имя инструкции = <значение>

В файле **moncfg.ini** можно записывать инструкции «АРМД», приведенные в таблице 1.

Таблица 1

Список инструкций «АРМД» в файле **moncfg.ini**

Имена инструкций	Назначение
Path	Путь к каталогу мониторинга.
IP	IP-адрес сетевого компьютера, если файлы мониторинга создаются на его сетевом ресурсе.
FileSize	Размер файла мониторинга, Мб
LocalFileSize	Размер каждого из 2-х временных файлов мониторинга [Мб] в каталоге: C:\CNC32\MP0
MachineName	Номер УЧПУ или номер станка.
DownTimeList	Имя и расширение файла, содержащее список причин простоя.
RecordTime	Период времени между записями в файл мониторинга.
Port	порт синхронизации (по умолчанию порт: 53847)
NetTimeSynchroInterval	Временной интервал синхронизации с сервером; [минут].
LocalTimeSynchroInterval	Временной интервал синхронизации с BIOS УЧПУ; [минут].
Pro	Список номеров процессов, подлежащих мониторингу.
MaxLocalSize	объем части диска C:, который может использовать «АРМД» для создания локальных файлов мониторинга

1.12.1.1. Инструкция Path

Инструкция **Path** определяет путь к каталогу на сетевом ресурсе или к каталогу диска, в которой должны быть созданы файлы мониторинга.

Семантика:

Path=сетевой ресурс или диск\каталог

Формат записи:

- для сетевого ресурса:

\\сетевое имя компьютера\имя ресурса\каталог

- для диска:

Path=Имя диска:[\Имя каталога]

Для каждого элемента «УЧПУ - станок» предприятия необходимо создать персональный каталог на сетевом ресурсе или персональный каталог на диске, в котором будут создаваться файлы мониторинга. Имя каталога должно отражать модель станка и его номер на предприятии. Если для нескольких УЧПУ будет записан путь к одному и тому же каталогу на сетевом ресурсе, то это приведет к разрушению файлов мониторинга для этих УЧПУ.

Если сетевой ресурс, установленный в инструкции **Path**, не доступен, то информацию об этом можно найти в среде «АРМД» элемент меню «**Анализ конфигурации**» и элемент меню «**Состояние ресурса**».

Пример определения каталога STANOK1 на сетевом ресурсе D для создания файлов мониторинга:

Path=\\STAND\D\STANOK1

Пример определения каталога для создания файлов мониторинга на диске УЧПУ:

Path=D:\MONITOR\STANOK1

1.12.1.2. Инструкция IP

Инструкция **IP** определяет IP-адрес ПК в рабочей группе, на котором запущена программа синхронизации времени УЧПУ и времени этого ПК.

Семантика:

IP=<IP-адрес>

Формат записи:

IP=байт.байт.байт.байт

Пример:

IP=192.168.1.129

1.12.1.3. Инструкция FileSize.

Инструкция **FileSize** определяет объем файла мониторинга на ресурсе, указанном в инструкции **Path**.

Семантика:

FileSize=<значение объема в Мб>

Формат записи:

FileSize=<целое число>

Пример определения объема файла мониторинга 5 Мб:

FileSize=5

1.12.1.4. Инструкция LocalFileSize

Инструкция **LocalFileSize** определяет максимальный объем одного локального мониторинга, создаваемого в каталоге C:\CNC32\MP0.

Семантика:

LocalFileSize=<значение объема в Мб>

Формат записи:

LocalFileSize=<целое число>

В случае потери доступа к сетевому ресурсу, записанному в инструкции **Path**, «АРМД» автоматически создает локальные файлы мониторинга в каталоге C:\CNC32\MP0. После восстановления доступа к сетевому ресурсу накопленные локальные файлы мониторинга будут автоматически скопированы на этот сетевой ресурс.

Пример определения объема файла мониторинга 5 Мб:

LocalFileSize=5

1.12.1.5. Инструкция MaxLocalSize

Инструкция **MaxLocalSize** определяет объем части диска C:, который может использовать «АРМД» для создания локальных файлов мониторинга.

Семантика:

MaxLocalSize=<значение объема в Мб>

Формат записи:

MaxLocalSize=<целое число>

MaxLocalSize

– значение параметра должно быть больше, чем значение:
2*LocalFileSize.

Если инструкция **MaxLocalSize** не задана, то «АРМД» отводит для размещения локальных файлов мониторинга 50% свободного объема диска C:.

Пример резервирования на диске C: объема 20 Мб для размещения локальных файлов мониторинга:

MaxLocalSize=20

1.12.1.6. Инструкция DowntimeList

Инструкция **DowntimeList** определяет имя файла причин простоя.

Семантика:

DowntimeList=<имя файла>.<расширение файла>

Формат записи:

DowntimeList=<до 8 ASCII-символов>.<3 ASCII-символов>

Автоматическое обновление файла причин простоя выполняется по следующему алгоритму.

1. После перезапуска УЧПУ при 1-ом запросе синхронизации задача «АРМД» формирует путь и имя файла причин простоя на основании значений инструкций **path** и **DownTimeList** и передает эту информацию в программу **bsserver** или ей подобной, исполняемой в ПК.

Пример

Path=\\STAND\D\STANOK1

DownTimeList=downtime.lst

Результрующий путь: \\STAND\D\STANOK1\downtime.lst

2. Если файл причин простоя существует по указанному пути в инструкции **path**, то программа **bsserver.exe** формирует запрос копирования этого файла в УЧПУ в каталог C:\CNC32\MP0.
3. Если файл причин простоя не существует по указанному пути в инструкции **path**, то в УЧПУ остается действующим файл причин простоя, имя которого записано в инструкции **DownTimeList** из каталога C:\CNC32\MP0.
4. Если инструкция **DownTimeList** отсутствует в файле **moncfg.ini** или в каталоге УЧПУ C:\CNC32\MP0 файл причин простоя не существует, то выбор причины простоя в УЧПУ будет не возможен.

Примечание.

- 1) Установленные в процессе работы УЧПУ причины простоя сохраняются в файле **downtime.sav**.
- 2) Инструкция **DownTimeList** ранее имела имя **IdleTimeList**. Имя инструкции **IdleTimeList** остается действительным в «АРМД».

1.12.1.7. Инструкция **MachineName**

Инструкция **MachineName** определяет имя УЧПУ или номер станка.

Семантика:

MachineName=<номер УЧПУ или номер станка>

Формат записи:

MachineName=<25 ASCII-символов>

Значение **MachineName** является справочной информацией, которая может быть использована для идентификации оборудования.

Пример:

MachineName= N102346#16A20

1.12.1.8. Инструкция **RecordTime**

Инструкция **RecordTime** определяет период времени между записями в файл мониторинга.

Семантика:

RecordTime =<Значение периода времени>

Формат записи:

RecordTime =<целое число>

Значение периода времени - определяет максимальный период времени между очередной записью регистрируемых переменных в файл мониторинга; выражено в секундах. Минимальное значение периода времени между записями в файл может быть установлено равным 3 с.

Если значение параметра задано равным 1 с или 2 с, то автоматически **Значение периода времени** принимает значение равным 3 с.

Если значение параметра задано равным 0 с, или инструкция **RecordTime** отсутствует в файле **moncfg.ini**, то автоматически **Значение периода времени** принимает значение равным 60 с.

Пример определения временного периода между записями в файл мониторинга равного 5 с:

RecordTime=5

1.12.1.9. Инструкция Port

Инструкция **Port** определяет номер порта синхронизации времени УЧПУ и сервера.

Семантика:

Port =<Номер порта синхронизации>

Формат записи:

Port =<WORD>

Номер порта синхронизации - определяет порт синхронизации.

Если значение параметра задано равным 0 с, или инструкция **Port** отсутствует в файле **moncfg.ini**, то автоматически **Port** принимает значение **53847**.

Пример определения номера порта:

Port=53847

1.12.1.10. Инструкция NetTimeSynchroInterval

Инструкция **NetTimeSynchroInterval** определяет интервал времени между запросами синхронизации УЧПУ с сервером.

Семантика:

NetTimeSynchroInterval=<Значение интервала времени>

Формат записи:

NetTimeSynchroInterval=<DWORD>

Значение интервала времени - определяет максимальный интервал времени между запросами сетевой синхронизации.

зации времени сервера и УЧПУ; значение выражено в минутах.

Если значение параметра задано равным 0, то выполняется однократный сетевой запрос синхронизации сразу после запуска «АРМД».

Минимальное время интервала равно 1 минута.

Пример определения интервала времени сетевой синхронизации равного 1-ой минуте:

NetTimeSynchroInterval=1

1.12.1.11. Инструкция LocalTimeSynchroInterval

Инструкция **LocalTimeSynchroInterval** определяет интервал времени между запросами синхронизации УЧПУ со временем **BIOS** на материнской плате УЧПУ.

Семантика:

LocalTimeSynchroInterval=<Значение интервала времени>

Формат записи:

LocalTimeSynchroInterval=<DWORD>

Значение интервала времени - определяет максимальный интервал времени между запросами локальной синхронизации времени сервера и УЧПУ; значение выражено в минутах.

Если значение параметра задано равным 0, то локальная синхронизация времени выключена.

Минимальное время интервала равно 1 минута.

Пример определения интервала локальной синхронизации времени равного 1-ой минуте:

LocalTimeSynchroInterval=1

1.12.1.12. Инструкция Pro

Инструкция **Pro** определяет номера процессов от 1 до 5, сконфигурированных в характеристиках УЧПУ, переменные которых необходимо записывать в файлы мониторинга.

Семантика:

Pro=<номер процесса, номер процесса, номер процесса, номер процесса, номер процесса>

Формат записи:

Pro=<число, число, число, число, число>

Примечание.

Все процессы, номера которых перечислены в инструкции **Pro**, должны быть сконфигурированы в системе. Количество процессов, сконфигуриро-

ванных в системе, определено в 1-ом параметре инструкции **NBP** в файле характеристики **AXCFIL**.

ВНИМАНИЕ !

Не записывайте в инструкцию **Pro** номера процессов, отсутствующих в характеристике системы, в противном случае в файлах мониторинга для переменных этих процессов будут записываться нулевые значения.

Пример записи инструкции **Pro** для системы, в которой сконфигурированы 5 процессов и для мониторинга из них выбраны только процессы 1, 2 и 5:

Pro=1,2,5

Пример записи инструкции **Pro** для системы, в которой сконфигурирован только один процесс:

Pro=1

1.12.2. Переменные «АРМД»

Переменные «АРМД» являются автоматически регистрируемыми машинными данными в файлах мониторинга.

Переменные «АРМД» предназначены для связи между системными переменными, сконфигурированных процессов в УЧПУ и переменными ВПМ.

«АРМД» записывает в файлы мониторинга только те переменные, имена и параметры которых определены в файле **moncfg.ini**.

При определении каждой переменной «АРМД» в файле **moncfg.ini** должны быть сформированы значения параметров, определяющих способ обработки связанной с ней системной переменной.

Формат записи переменных:

Имя переменной, Тип учета [= Параметр учета]

Имя переменной

- определяет имя переменной «АРМД» в файле **moncfg.ini**. Список допустимых имен переменных «АРМД» приведен в таблице 2. Каждая переменная «АРМД» соответствует своей системной переменной в ПрО.

Тип учета

- определяет условие обработки системной переменной до присвоения ее значения переменной «АРМД».

Допускаются следующие значения «Тип учета»:

- 1) **Timer** - определяет присвоение переменной «АРМД» значения соответствующей ей системной переменной через заданный интервал времени в «Параметр учета».
- 2) **SPUTICK** - определяет присвоение переменной «АРМД» значения соответствующей ей системной переменной через интервал, равный **ТИК** системы.

Примечание. Если для переменной «АРМД» в параметре «Тип учета» установлено значение **SPUTICK**, то значение «Параметр учета», за-

данное для этой переменной в файле **moncfg.ini**, игнорируется.

- 3) **MEANVALUE** - определяет обработку системной переменной, как среднее арифметическое значение за интервал времени, заданный для переменной «АРМД» в «**Параметр учета**».
- 4) **APPROXIMATE** определяет аппроксимированную обработку системной переменной. Пример аппроксимации системной переменной представлен на рисунке 3.

Параметр учета

- определяет значение условия присвоения в переменную «АРМД» значения системной переменной:

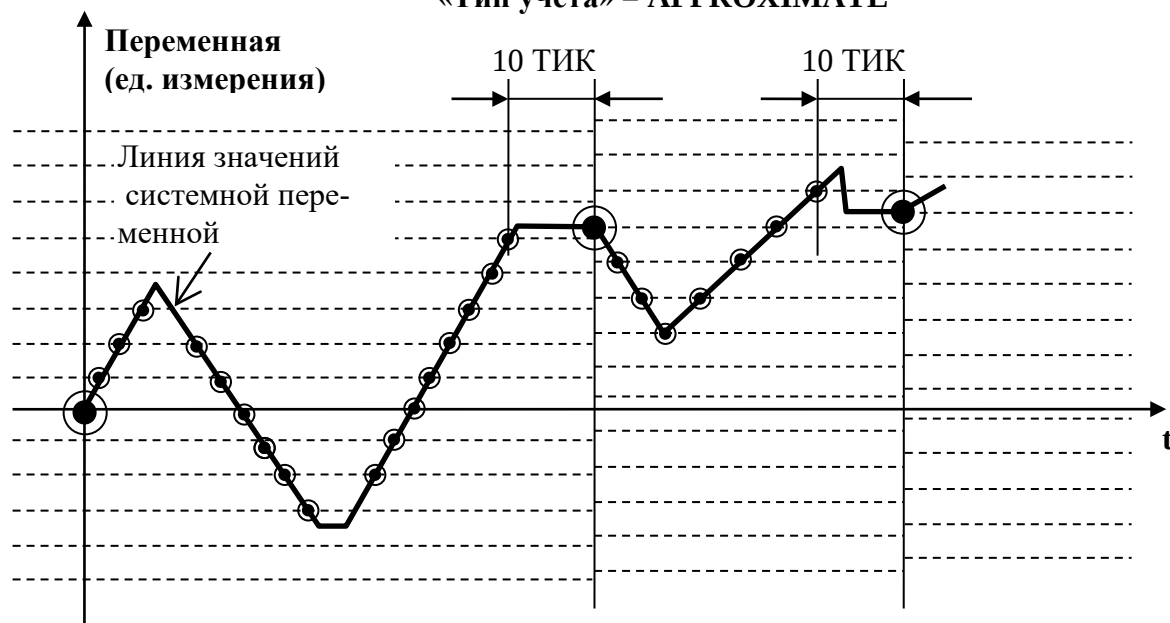
- 1) Если значение «**Тип учета**» равно **Timer**, то значение «**Параметр учета**» определяет интервал времени, выраженный в миллисекундах, между очередным присвоением переменной «АРМД» значения ее системной переменной. Значение «**Параметр учета**» можно не записывать, если значение «**Тип учета**» равно **SPUTICK**.
- 2) Если значение «**Тип учета**» равно **MeanValue**, то значение «**Параметр учета**» определяет интервал времени, выраженный в миллисекундах, за который должно быть рассчитано среднее значение ее системной переменной.
- 3) Если значение «**Тип учета**» равно **APPROXIMATE**, то значение «**Параметр учета**» определяет уровень изменения системной переменной относительно ее текущего значения; выражается в текущих единицах измерения системной переменной.

1.12.2.1. Принцип обработки APPROXIMATE.

В первом ТИКе системы после запуска «АРМД» в переменную «АРМД» присваивается точное значение ее системной переменной, и от этого значения симметрично в положительном и отрицательном направлении рассчитывается шкала изменения системной переменной. Каждое деление шкалы равно значению «**Параметр учета**» и определяет линию контролируемого уровня.

Если текущее значение системной переменной выходит за линию ближайшего уровня, то в переменную «АРМД» присваивается значение, равное ближайшему уровню шкалы. Если после этого значение системной переменной за 10 ТИКов не выходит за линию ближайшего уровня, то переменной «АРМД» присваивается точное значение системной переменной, считанное в 10-ом ТИКе, и выполняется привязка шкалы уровней к ее точному значению.

Рисунок, отображающий обработку системной переменной, если
«Тип учета» = APPROXIMATE



----- - линии контрольных уровней изменения системной переменной.

- - точки присвоения точных значений системной переменной в переменную «АРМД» и привязка шкалы к точному значению (значения переменной в этих точках сохраняются в файле мониторинга).
- - точки присвоения аппроксимированных значений системной переменной в переменную «АРМД» (значения переменной в этих точках сохраняются в файле мониторинга).

Рисунок 3

Список переменных «АРМД», конфигурируемых в файле **moncfg.ini**

№	Имя переменной	Тип учета	Параметр учета		Назначение Переменной
			Мин .	Макс .	
1	WorkMode	CPUTICK	ТИК	ТИК	Режим работы: MDI, AUTO, STEP, MANU, MANJ, PROF, HOME, RESET
		TIMER	1мс	10000000мс	
2	SystemState	CPUTICK	ТИК	ТИК	Состояния системы: IDLE, RUN, HOLD, RUNH, RESE, RTCE
		TIMER	1мс	10000000мс	
3	MUSP	CPUTICK	ТИК	ТИК	Состояние станка: MUSP=1-выключен; MUSP=0-включен
		TIMER	1мс	100000мс	
4	CONP	CPUTICK	ТИК	ТИК	Состояние готовно- сти процесса для обслуживания осей: CONP=1-готов; CONP=0-не готов
		TIMER	1мс	100000мс	
5	RISP	CPUTICK	ТИК	ТИК	Запрос выключения станка: RISP=1-включен; RISP=0-выключен
		TIMER	1мс	100000мс	
6	ASPEPN	CPUTICK	ТИК	ТИК	Состояние станка: ASPEPN=0-АВАРИЯ; ASPEPN=1-Нет АВАРИИ
7	SPEPNREQ	CPUTICK	ТИК	ТИК	Готовность УЧПУ для включения станка: SPEPNREQ=1-готово; SPEPNREQ=0-не гото- во
		TIMER	1мс	100000мс	
8	REAZ	CPUTICK	ТИК	ТИК	Состояние запроса «Общий сброс» из ПЛ
		TIMER	1мс	100000мс	
9	COMU	CPUTICK	ТИК	ТИК	Состояние разреше- ния начала движения COMU=1 - разрешено COMU=0 - запрещено
		TIMER	1мс	100000мс	
10	CEFA	CPUTICK	ТИК	ТИК	Состояние разреше- ния выполнения M, S, T-функций нача- ла/конца движения: CEFA=1 - разрешено CEFA=0 - запрещено
		TIMER	1мс	100000мс	
11	GFunctions	CPUTICK	ТИК	ТИК	G-функцию для каж- дой модальной груп- пы
		TIMER	1мс	1000000мс	

№	Имя переменной	Тип учета	Параметр учета		Назначение Переменной
			Мин.	Макс.	
12	SwitchJOG	CPUTICK	ТИК	ТИК	% коррекции скорости ручных перемещений (поле «JOG%» на видеостранице «ПРОЦЕСС #»)
		TIMER	1мс	10000000мс	
13	SwitchSpindle	CPUTICK	ТИК	ТИК	% скорости вращения шпинделя (поле «S%» на видеостранице «ПРОЦЕСС #»)
		TIMER	1мс	10000000мс	
14	SwitchFeed	CPUTICK	ТИК	ТИК	% коррекции подачи (поле «F%» на видеостранице «ПРОЦЕСС #»)
		TIMER	1мс	10000000мс	
15	Feed	CPUTICK	ТИК	ТИК	Подача по контуру с учетом корректора F% (поле «F» на видеостранице «ПРОЦЕСС #»).
		TIMER	1мс	10000000мс	
		MEANVALUE	1	1000000	
		APPROX	1.0	10000000.0	
16	SpindleSpeed	CPUTICK	ТИК	ТИК	Скорость вращения шпинделя с учетом корректора S% (поле «S» на видеостранице «ПРОЦЕСС #»)
		TIMER	1мс	10000000мс	
		MEANVALUE	1	1000000	
		APPROX	1.0	10000000.0	
17	ProgramName	CPUTICK	ТИК	ТИК	Имя управляющей программы, загруженной на выполнение (поле «Имя УП» на видеостранице «ПРОЦЕСС #»).
		TIMER	1мс	100000мс	
18	NumberProgramBlock	CPUTICK	ТИК	ТИК	Номер выполняемого кадра N (поле «N» на видеостранице «ПРОЦЕСС #»)
		TIMER	1мс	100000мс	
19	ToolNumber	CPUTICK	ТИК	ТИК	Номер активного инструмента в «ПРОЦЕСС #».
		TIMER	1мс	100000мс	
20	CorrectorTool-Number	CPUTICK	ТИК	ТИК	Номер активного корректора для инструмента в «ПРОЦЕСС #».
		TIMER	1мс	100000мс	
21	MachineDownTime	CPUTICK	ТИК	ТИК	Причина простоя.

№	Имя переменной	Тип учета	Параметр учета		Назначение Переменной
			Мин .	Макс .	
22	ProcessCommandLine	CPUTICK	ТИК	ТИК	Текст команды, выполняемой в «ПРОЦЕСС #» из строки «Ввода и редактирования» с клавишей «ENTER».
		TIMER	1мс	100000мс	
23	ProcessBlockLine	CPUTICK	ТИК	ТИК	Текст кадра, выполняемого в «ПРОЦЕСС #» из строки «Ввода и редактирования» с клавишей «ПУСК».
		TIMER	1мс	100000мс	
24	CommandLine	CPUTICK	ТИК	ТИК	Текст команды, выполняемой в режиме «КОМАНДА» из строки «Ввода и редактирования» с клавишей «ENTER».
		TIMER	1мс	100000мс	
25	UVR	CPUTICK	ТИК	ТИК	Состояние UVR: UVR=1 – выполнение УП на быстром ходу; UVR=0 – выполнение УП на запрограммированной подаче.
		TIMER	1мс	100000мс	
26	URL	CPUTICK	ТИК	ТИК	Коррекция скорости быстрого хода, заданной в кадре УП: UVR=1 – включена; UVR=0 – выключена.
		TIMER	1мс	100000мс	
27	UAS	CPUTICK	ТИК	ТИК	Состояние осей: UAS=1 – отключены; UAS=0 – подключены.
		TIMER	1мс	100000мс	
28	MesErrProg	CPUTICK	ТИК	ТИК	Номер строки файла RUMES4, текст которой был выведен на экран в поле «Сообщения 4 __хх».
		TIMER	1мс	10000000мс	
29	MesErrPLC	CPUTICK	ТИК	ТИК	Номер строки файла RUMES5, текст которой был выведен на экран в поле: «Сообщения 5 (1) __ххХ»
		TIMER	1мс	10000000мс	
30	AlarmErrPLC	CPUTICK	ТИК	ТИК	Номер строки файла RUMES5, текст которой был выведен на экран в поле: «Сообщения 5 (2) __ххХ»
		TIMER	1мс	10000000мс	
31	PartFinished	CPUTICK	ТИК	ТИК	Конец выполнения детали.
		TIMER	1мс	10000000мс	

№	Имя переменной	Тип учета	Параметр учета		Назначение Переменной
			Мин.	Макс.	
32	WNCMT	CPUTICK	ТИК	ТИК	Текстовая переменная для ввода служебной информации
33	WNPRT	CPUTICK	ТИК	ТИК	Текстовая переменная для ввода служебной информации
34	WPROG	CPUTICK	ТИК	ТИК	Текстовая переменная для ввода служебной информации
35	WIZKD	CPUTICK	ТИК	ТИК	Текстовая переменная для ввода служебной информации
36	SpindlePower	CPUTICK	ТИК	ТИК	Мощность шпинделя
		TIMER	1мс	10000000мс	

Примечание.

- 1) В таблице 2 переменные записаны в том порядке, в котором их рекомендуется записывать в файл **moncfg.ini**. Этот порядок учитывает причинно-следственный механизм возникновения событий.
- 2) Для ввода новых переменных «АРМД» (таблица 2) необходимо обратиться к разработчику ВПМ.

1.12.2.2. Переменная WorkMode

Переменная **WorkMode** предназначена для контроля режимов работы в процессах, номера которых записаны в инструкции **Pro**.

Пример:

WorkMode, Timer = 5

или

WorkMode, CPUTICK

1.12.2.3. Переменная SystemState

Переменная **SystemState** предназначена для контроля состояния процессов, номера которых записаны в инструкции **Pro**.

Пример:

SystemState, Timer = 5

или

SystemState, CPUTICK

1.12.2.1. Переменная MUSP

Переменная **MUSP** предназначена для контроля состояния станка (включен; выключен) в процессах, номера которых записаны в инструкции **Pro**.

Пример:

MUSP, Timer = 1

или

MUSP, CPUTICK

Примечание. Для правильного анализа состояния станка в процессе необходимо, чтобы ПЛ устанавливала сигнал интерфейса PLC MUSPE в соответствующем процессе в состояние:

- «1», если станок ВЫКЛЮЧЕН;
- «0», если станок ВКЛЮЧЕН.

1.12.2.2. Переменная CONP

Переменная **CONP** предназначена для контроля готовности обслуживания осей в процессах, номера которых записаны в инструкции **Pro**.

Пример:

CONP, Timer = 1

или

CONP, CPUTICK

Переменная **CONP** отображает состояние интерфейсных сигналов PLC, представленных в параграфе 6.

Примечание. Анализ значения переменной **CONP** может быть использован совместно с анализом значений переменных **MUSP**, **RISP**, **ASPEPN** и **SPEPNREQ** для точного определения состояния станка.

1.12.2.3. Переменная RISP

Переменная **RISP** предназначена для контроля состояния станка в процессах, номера которых записаны в инструкции **Pro**.

Пример:

RISP, Timer = 1

или

RISP, CPUTICK

Переменная **RISP** отображает состояние интерфейсных сигналов PLC, представленных в параграфе 6.

Примечание. Анализ значения переменной **RISP** может быть использован совместно с анализом значений переменных **MUSP**, **CONP**, **ASPEPN** и **SPEPNREQ** для точного определения состояния станка.

1.12.2.4. Переменная **ASPEPN**

Переменная **ASPEPN** предназначена для контроля состояния станка в процессах, номера которых записаны в инструкции **Pro**.

Пример:

ASPEPN, Timer = 1

или

ASPEPN, CPUTICK

Переменная **ASPEPN** отображает состояние интерфейсных сигналов **PLC**, представленных в параграфе 6.

Примечание. Анализ значения переменной **ASPEPN** может быть использован совместно с анализом значений переменных **MUSP**, **CONP**, **RISP** и **SPEPNREQ** для точного определения состояния станка.

1.12.2.5. Переменная **SPEPNREQ**

Переменная **SPEPNREQ** предназначена для контроля состояния станка в процессах, номера которых записаны в инструкции **Pro**.

Пример:

SPEPNREQ, Timer = 1

или

SPEPNREQ, CPUTICK

Переменная **SPEPNREQ** отображает состояние интерфейсных сигналов **PLC**, представленных в параграфе 6.

Примечание. Анализ значения переменной **SPEPNREQ** может быть использован совместно с анализом значений переменных **MUSP**, **CONP**, **RISP** и **ASPEPN** для точного определения состояния станка.

1.12.2.6. Переменная **REAZ**

Переменная **REAZ** предназначена для контроля выполнения запроса «ОБЩИЙ СБРОС» из ПЛ в процессах, номера которых записаны в инструкции **Pro**.

Пример:

REAZ, Timer = 1

или

REAZ, CPUTICK

Переменная **REAZ** отображает состояние интерфейсных сигналов **PLC**, представленных в параграфе 6.

Примечания.

- 1) Запрос выполнения «ОБЩИЙ СБРОС» может быть выполнен оператором станка и базовым Про, но переменная **REAZ** устанавливается только в том случае, если запрос «ОБЩИЙ СБРОС» был выполнен в ПЛ.
- 2) Другие причины выполнения «ОБЩИЙ СБРОС» в процессе можно анализировать на основании следующих данных:
 - Запрос выполнен оператором станка в режиме работы **RESET** с клавишей **ПУСК** или сигналом интерфейса **PLC: CYST**;
 - Запрос выполнен базовым Про по изменению состояния интерфейсного сигнала **PLC: MUSPE**;

- Запрос выполнен базовым Про по входу на концевики ограничения перемещения или в объемную защищенную зону **CUB**.
- Запрос выполнен базовым Про по аварии. Список причин аварии приведен в таблице 3.

1.12.2.7. Переменная COMU

Переменная **COMU** предназначена для контроля разрешения начала движения осей в процессах, номера которых записаны в инструкции **Pro**.

Пример:

COMU, Timer = 1

или

COMU, CPUTICK

Переменная **COMU** отображает состояние интерфейсных сигналов PLC, представленных в параграфе 6.

1.12.2.8. Переменная CEFA

Переменная **CEFA** предназначена для контроля разрешения выполнения **M, S, T** – функций в ПЛ в процессах, номера которых записаны в инструкции **Pro**.

Пример:

CEFA, Timer = 1

или

CEFA, CPUTICK

Переменная **CEFA** отображает состояние интерфейсных сигналов PLC, представленных в параграфе 6.

1.12.2.9. Переменная GFunctions

Переменная **GFunctions** предназначена для контроля изменения активных **G**-функций в каждой их группе модальности, в процессах, номера которых записаны в инструкции **Pro**.

Пример:

GFunctions, Timer = 1

или

GFunctions, CPUTICK

1.12.2.10. Переменная ProgramName

Переменная **ProgramName** предназначена для контроля имен УП, загруженных для выполнения командой **SPG** или вызванных из выполняемой УП 3-х буквенным кодом **CLS**, в процессах, номера которых записаны в инструкции **Pro**.

Пример:

ProgramName, Timer = 5

или

ProgramName, CPUTICK

Примечание.

В процессе одновременно может быть открыто до 3-х УП, находящихся на 3-х разных уровнях. В файле мониторинга для каждой загруженной для выполнения УП указывается ее уровень.

Пример:

- кадры **N1, N2, N3, N4** и **N5** принадлежат программе **P0/MP1** (нулевой уровень);
- кадры **N10, N20, N30, N40** и **N50** принадлежат подпрограмме **P1/MP1** (1-ый уровень);
- кадры **N100, N200, N300** принадлежат программе **P2/MP1** (2-ой уровень).

SPG, P0/MP1

Номер
уровня

Кадры программ

0	N1 X...Y...F...	
0	N2 X...Y...F...	
0	N3 (CLS, P1/MP1)	-
1		N10 X...Y...F...
1		N20 X...Y...F...
1		N30 (CLS, P2/MP1)
2		N100 X...Y...F...
2		N200 X...Y...F...
2		N300 X...Y...F...
1		N40 X...Y...F...
1		N50 X...Y...F...
0	N4 X...Y...F...	
0	N5 X...Y...F...	

1.12.2.11. Переменная NumberProgramBlock

Переменная **NumberProgramBlock** предназначена для контроля выполненных или выполняемых номеров кадров, записанных в кадре с адресом **N**, в тех в процессах, номера которых записаны в инструкции **Pro**.

Пример:

NumberProgramBlock, Timer = 2

или

NumberProgramBlock, CPUTICK

1.12.2.12. Переменная ToolNumber

Переменная **ToolNumber** предназначена для контроля номера активного инструмента в процессах, номера которых записаны в инструкции **Pro**.

Пример:

ToolNumber, Timer = 1

или

ToolNumber, CPUTICK

1.12.2.13. Переменная CorrectorToolNumber

Переменная **CorrectorToolNumber** предназначена для контроля номера активного корректора в процессах, номера которых записаны в инструкции **Pro**.

Пример:

CorrectorToolNumber, Timer = 1

или

CorrectorToolNumber, CPUTICK

1.12.2.14. Переменная SwitchJOG

Переменная **SwitchJOG** предназначена для контроля значения коррекции скорости ручных перемещений и скорости быстрого хода, если переменная URL=1 в процессах, номера которых записаны в инструкции **Pro**.

Пример:

SwitchJOG, Timer = 5

или

SwitchJOG, CPUTICK

1.12.2.15. Переменная switchSpindle

Переменная **SwitchSpindle** предназначена для контроля значения коррекции скорости вращения шпинделя в процессах, номера которых записаны в инструкции **Pro**.

Пример:

SwitchSpindle, Timer = 5

или

SwitchSpindle, CPUTICK

1.12.2.16. Переменная switchFeed

Переменная **SwitchFeed** предназначена для контроля значения коррекции подачи на профиле в процессах, номера которых записаны в инструкции **Pro**.

Пример:

SwitchFeed, Timer = 5

или

SwitchFeed, CPUTICK

1.12.2.17. Переменная Feed

Переменная **Feed** предназначена для контроля значения подачи (поле «**F**» в видеостранице #1[, #2, #3, #4, #5]) с учетом значения коррекции (поле «**F%**») в процессах, номера которых записаны в инструкции **Pro**. Значение F

Пример:

Feed, Timer = 5

или

Feed, CPUTICK

или

Feed, approximate =10

или

Feed, meanvalue =1000

1.12.2.18. Переменная SpindleSpeed

Переменная **SpindleSpeed** предназначена для контроля значения скорости вращения шпинделя (поле «**S**» в видеостранице #1[, #2, #3, #4, #5]) с учетом значения коррекции (поле «**S%**») в процессах, номера которых записаны в инструкции **Pro**.

Пример:

SpindleSpeed, Timer = 5

или

SpindleSpeed, CPUTICK

или

SpindleSpeed, approximate =10

или

SpindleSpeed, meanvalue =1000

1.12.2.19. Переменная MachineDownTime

Переменная **MachineDownTime** предназначена для контроля причины простоя оборудования в процессах, номера которых записаны в инструкции **Pro**.

Пример:

MachineDownTime, CPUTICK

1.12.2.20. Переменная ProcessCommandLine

Переменная **ProcessCommandLine** предназначена для контроля выполнения команд, введенных из строки «**Ввода и редактирования**» с клавишей **ENTER**. Эта переменная регистрируется для процессов, номера которых записаны в инструкции **Pro**.

Пример:

ProcessCommandLine, Timer = 1

или

ProcessCommandLine, CPUTICK

1.12.2.21. Переменная **ProcessBlockLine**

Переменная **ProcessBlockLine** предназначена для контроля выполнения кадров, введенных из строки «Ввода и редактирования» с клавишей **ПУСК**. Эта переменная регистрируется для процессов, номера которых записаны в инструкции **Pro**.

Пример:

ProcessBlockLine, Timer = 1

или

ProcessBlockLine, CPUTICK

1.12.2.22. Переменная **CommandLine**

Переменная **CommandLine** предназначена для контроля выполнения команд в режиме «КОМАНДА», введенных из строки «Ввода и редактирования» с клавишей **ENTER**.

Пример:

CommandLine, Timer = 1

или

CommandLine, CPUTICK

1.12.2.23. Переменная **UAS**

Переменная **UAS** предназначена для контроля отсоединения осей в случае выполнения УП без станка в процессах, номера которых записаны в инструкции **Pro**.

Пример:

UAS, Timer = 1

или

UAS, CPUTICK

1.12.2.24. Переменная **UVR**

Переменная **UVR** предназначена для контроля выполнения УП на скорости быстрого хода в процессах, номера которых записаны в инструкции **Pro**.

Пример:

UVR, Timer = 1

или

UVR, CPUTICK

1.12.2.25. Переменная **URL**

Переменная **URL** предназначена для контроля о включении установки коррекции скорости быстрого хода **G00** в процессах, номера которых записаны в инструкции **Pro**.

Пример:

URL, Timer = 1

или

URL, CPUTICK

1.12.2.26. Переменная MesErrProg

Переменная **MesErrProg** предназначена для регистрации всех сообщений, которые выводятся в поле «Сообщения 4__xx» в видеостраницах #1, #2, #3, #4, #5, #6 и #7 режима «УПРАВЛЕНИЕ СТАНКОМ» в процессах, номера которых записаны в инструкции **Pro**.

Пример:

MesErrProg, Timer = 5

или

MesErrProg, CPUTICK

Полный список кодов сообщений, которые выводятся в поле «Сообщения 4__xx» в видеостраницах **#1, #2, #3, #4, #5, #6 и #7** режима «УПРАВЛЕНИЕ СТАНКОМ» приводятся в руководстве оператора в п. «Файл RUMES4». Для справочной информации в таблице 3 приводится список кодов аварийных сообщений.

Таблица 3

Код сообщения	Назначение
Сообщение 4 61	Обрыв датчика
Сообщение 4 62	Обрыв датчика
Сообщение 4 63	Исключение расчета
Сообщение 4 64	Ось не в допуске
Сообщение 4 65	Ошибка привода оси
Сообщение 4 66	Ошибка скорости оси
Сообщение 4 68	Неисправность датчиков

1.12.2.27. Переменная MesErrPLC

Переменная **MesErrPLC** предназначена для регистрации сообщений, которые выводятся в поле «Сообщения 5__xxx (1) (FILMS5) Сообщения ПЛ» в видеостраницах #1, #2, #3, #4, #5 и #7 режима «УПРАВЛЕНИЕ СТАНКОМ» в процессах, номера которых записаны в инструкции **Pro**.

Примечание. Для регистрации аварийных сообщений от ПЛ, которые выводятся в поле «Сообщения 5__xxx (2) (FILMS5) Аварийные сообщения ПЛ», смотри описание переменной **AlarmErrPLC**.

Пример:

MesErrPLC, Timer = 1

или

MesErrPLC, CPUTICK

Текст сообщений, которые выводятся в поле «Сообщения 5__xxx(1)» в видеостраницах **#1, #2, #3, #4, #5 и #7** режима «УПРАВЛЕНИЕ СТАНКОМ» приводятся в руководстве по эксплуатации для конкретного станка.

1.12.2.28. Переменная AlarmErrPLC

Переменная **AlarmErrPLC** предназначена для регистрации сообщений, которые выводятся в поля «Сообщения 5__xxx (2) (FILMS5) Аварийные сообщения ПЛ» в видеостраницах #1, #2, #3, #4, #5 и #7 режима «УПРАВЛЕНИЕ СТАНКОМ» в процессах, номера которых записаны в инструкции **Pro**.

Примечание. Для регистрации сообщений от ПЛ, которые выводятся в поле «Сообщения 5__xxx (1) (FILMS5) Сообщения ПЛ», смотри описание переменной **MesErrPLC**.

Пример:

AlarmErrPLC, Timer = 1

или

AlarmErrPLC, CPUTICK

Текст сообщений, которые выводятся в поле «Сообщения 5__xx(2)» в видеостраницах #1, #2, #3, #4, #5 и #7 режима «УПРАВЛЕНИЕ СТАНКОМ» приводятся в руководстве по эксплуатации для конкретного станка.

1.12.2.29. Переменная PartFinished

Переменная **PartFinished** предназначена для контроля окончания выполнения детали в процессах, номера которых записаны в инструкции **Pro**.

Пример:

PartFinished, Timer = 1

или

PartFinished, CPUTICK

Если переменная **PartFinished** задана в файле **moncfg.ini** для текущего процесса, то ее значение в этом процессе будет устанавливать М-функция, содержащая во втором байте инструкции **Mxx** (секция 2 файла **IOCFIL**) код **80h** «деталь выполнена». Сброс переменной **PartFinished** в текущем процессе выполняет «АРМД» после записи этого события в файл мониторинга.

Пример характеристики М-функции «деталь выполнена» в файле **IOCFIL**:

```
*2
...
M87=02,80,FF
...
```

Примеры использования М-функции «деталь выполнена» в УП.

1) Программа для выполнения одной детали из 2-х операций.

```
N10 T1.1M6
    <Кадры УП. Операция 1>
N20 T2.2M6
    <Кадры УП. Операция 2>
N30 M87
N31 M30
```

- 2) Программа для выполнения 10-ти деталей из 2-х операций.

(RPT, 10)

N10 T1.1M6

<Кадры УП. Операция 1>

N20 T2.2M6

<Кадры УП. Операция 2>

N30 M87

(ERP)

N31 M30

- 3) Программа PART0 для выполнения 2-х разных деталей, каждая из которых состоит из 2-х операций.

N1 (CLS, PART1/MP1)

;Кадры подпрограммы PART1:

N10 T1.1M6

<Кадры УП. Операция 1>

N20 T2.2M6

<Кадры УП. Операция 2>

N30 M87

N2 (CLS, PART2/MP1)

;Кадры подпрограммы PART2:

N100 T3.3M6

<Кадры УП. Операция 1>

N200 T4.4M6

<Кадры УП. Операция 2>

N300 M87

N3 M30

1.12.2.30. Текстовые переменные

Текстовые переменные предназначены для передачи в файлы мониторинга служебной информации, введенной пользователем из строки «ввода и редактирования команды» с клавишами «ENTER» или «ПУСК» или из кадра УП. В задаче АРМД можно использовать 4 текстовые переменные со следующими именами:

WNCMT

WPROG

WNPRT

WIZKD

Примеры записи переменных WNCMT, WPROG, WNPRT, WIZKD в файле moncfg.ini:

WNCMT, CPUTICK, 1

WPROG, CPUTICK, 1

WNPRT, CPUTICK, 1

WIZKD, CPUTICK, 1

Рекомендуемое назначение переменных:

WNCMT – содержит текст, сообщающий:

- «модель УЧПУ»
- «модель станка»;

- WPROG** - содержит текст, сообщающий:
- «Номер программы», выполняемой в процессе,
 - «Табельный номер программы», выполняемой в процессе;
- WNPRT** - содержит текст, сообщающий «номер детали», выполняемой в процессе;
- WIZKD** - содержит текст, сообщающий номер извещения на конструкторскую документацию.

Примечания.

- 1) Если текст переменной должен быть разбит на несколько частей, отличающихся своим содержанием, то между такими частями в тексте рекомендуется записывать символ разделителя, например символ «+» или лидирующее ключевое слово, определяющее назначение следующего за ним текста. Все допустимые символы или ключевые слова для разделения частей текста, содержащегося в одной и той же переменной, определяет разработчик ВПМ.
- 2) Незначимые символы «пробелов» вначале или в конце текстовой строки могут быть удалены при анализе строки в ВПМ.

Определение текстовых переменных в системе.

Переменные **WNCMT**, **WPROG**, **WNPRT**, **WIZKD** являются локальными переменными пользователя для каждого процесса, сконфигурированного в системе. Максимальная длина каждой переменной может быть равна 65536 (0÷65535) ASCII-символа (байта).

Для использования переменной в текущем процессе она должна быть определена в секции 2 файла характеристики **PGCFIL** в инструкции **SIM** для этого процесса:

```
PRO=1
SIM=NEW, WNCMT, 32, 7, 64, 1
SIM=NEW, WPROG, 32, 7, 64, 1
SIM=NEW, WNPRT, 32, 7, 64, 1
SIM=NEW, WIZKD, 32, 7, 64, 1
...
PRO=2
SIM=NEW, WNCMT, 32, 7, 64, 1
SIM=NEW, WPROG, 32, 7, 64, 1
SIM=NEW, WNPRT, 32, 7, 64, 1
SIM=NEW, WIZKD, 32, 7, 64, 1
...
```

Присвоение переменным строки ASCII-символов может быть выполнено следующими способами:

- 1) выполнением команды из строки «ввода и редактирования команды» с клавишей «ENTER»;
- 2) выполнением кадра из строки «ввода и редактирования команды» в режиме «MDI» с клавишей «ПУСК»;
- 3) выполнением кадра УП в режимах «AUTO» или «STEP» с клавишей «ПУСК».

В одной команде или кадре УП можно выполнить ввод максимально до 32-х символов.

Пример частного случая, в котором вся длина переменной **WNCMT** равная 65535 символов разбита на 2048 текстовых частей, каждая из которых равна 32 символа:

№переменной	0	1	2	2047
№символа:	0-31	32-65	66-98...	65504-65535.
	WNCMT0.32CH	WNCMT32.32CH	WNCMT66.32CH	WNCMT65504.32CH

Пример записи в переменную **WNCMT** строки «**NC-210+Г#2171+3010+255331+47601.9606.268.000+DT476.96.002765-V1**» длиной 63 символа 6-тью кадрами:

```
Программа P3010:
<НАЧАЛО ПРОГРАММЫ P3010>
;WNCMT0.6CH - модель УЧПУ:
WNCMT0.6CH="NC-210"
;WNCMT6.7CH - модель станка:
WNCMT6.7CH="+Г#2171"
;WNCMT13.5CH - номер программы:
WNCMT13.5CH="+3010"
;WNCMT18.7CH - табельный номер программы:
WNCMT18.7CH="+255331"
;WNCMT25.19CH - номер детали:
WNCMT25.19CH="+47601.9606.268.000"
;WNCMT44.19CH - номер извещения на конструкторскую документацию:
WNCMT44.19CH="+DT476.96.002765-V1"
<кадры программы P3010>
.....
<КОНЕЦ ПРОГРАММЫ P3010>
```

Примеры рекомендуемого использования переменных:

Пример 1. Присвоение текстовой переменной **WNCMT** строки из 13 символов, содержащей служебную информацию о модели УЧПУ: «**NC-210**» (6 символов) и модели станка: «**Г#2171**» (6 символов), разделенных символом: «**+**» (1 символ):

WNCMT0.13CH="NC-210+Г#2171"

Пример 2. Присвоение текстовой переменной **WPROG** строки из 11-ти символов, содержащей служебную информацию о номере программы: «**3010**» (4 символа) и табельном номере программы: «**255331**» (6 символов), разделенных символом: «**+**» (1 символ):

WPROG0.11CH="3010+255331"

Пример 3. Присвоение текстовой переменной **WNPRT** строки из 18-ти символов, содержащей служебную информацию о номер детали «**47601.9606.268.000**»:

WNPRT0.18CH="47601.9606.268.000"

Пример 4. Присвоение текстовой переменной **WIZKD** строки из 18-ти символов, содержащей служебную информацию о номере извещения на конструкторскую документацию «**DT476.96.002765-V1**»:

WIZKD0.18CH="DT476.96.002765-V1"

1.12.2.1. Переменная SpindlePower

Переменная **SpindlePower** предназначена для контроля мощности шпинделя в процессах, номера которых записаны в инструкции **Pro**.

Пример:

SpindlePower, Timer = 1000

или

SpindlePower, CPUTICK

Если переменная **SpindlePower** задана в файле **moncfg.ini** для текущего процесса, то ее значение в этом процессе будет устанавливаться на основании данных, записанных в инструкции **ADP** в файле характеристики **AXCFIL**. Значение **SpindlePower** выражено в процентах.

1.12.2.2. Пример файла MONCFG.INI

Пример файла конфигурации «АРМД» для автоматической регистрации машинных данных для УЧПУ, управляющего 5-ю процессами.

```
Path=\\BS-VASIA\D\0
FileSize=1
LocalFileSize=5
DownTimeList=downtime.lst
RecordTime=5
IP=192.168.1.23
MachineName=stanok1
MaxLocalSize=11
PRO=1,2,3,4,5
; режим работы:
WorkMode, timer =100
; состояние системы:
SystemState, timer =1
; состояние станка:
MUSP, timer =1
; состояние готовности обслуживания осей:
CONP, timer =1
; состояние станка:
ASPEPN, timer =1
; состояние готовности для включения станка:
SPEPNREQ, timer =1
; запрос "Общий сброс" из ПЛ:
REAZ, timer=1
; разрешение движения:
COMU, timer=1
; разрешение MST и индексных осей:
CEFA, timer=1
; % корректора JOG:
SwitchJOG, timer =10
; % корректора S:
SwitchSpindle, timer =10
; % корректора F:
SwitchFeed, timer =10
; скорость шпинделя:
;SpindleSpeed, approximate =10
;SpindleSpeed, meanvalue =1000
SpindleSpeed, timer =1000
; подача:
Feed, cputick,1
; имя УП и подУП:
ProgramName, timer =10
; номер кадра:
NumberProgramBlock, timer =2
; номер инструмента:
ToolNumber,timer=1
; номер инструмента:
CorrectorToolNumber,timer=1
; причины простоя:
MachineDownTime,cputick,1
; команда в ПРОЦЕССе № из строки "Ввода и редактирования":
ProcessCommandLine = cputick,1
; кадр в ПРОЦЕССе № из строки "Ввода и редактирования":
ProcessBlockLine = cputick,1
; команда в режиме "КОМАНДА" из строки "Ввода и редактирования":
CommandLine = cputick,1
; отработка УП на скорости "Б.Х.":
UVR,timer=1
; коррекция скорости "Б.Х.":
```

```

URL,timer=1
;                                     отключение осей станка:
UAS,timer=1
;                                     сообщения и ошибки "Сообщение_4 xxx ":
MesErrProg,      timer      =1
;                                     сообщения ПЛ "Сообщение_5 xxx (1)":
MesErrPLC,       timer      =1
;                                     аварийные сообщения ПЛ "Сообщение_5 xxx (2)":
AlarmErrPLC,     timer      =1
;                                     конец управляющей программы:
PartFinished, Timer = 1
;                                     текстовая переменная для служебной информации
WNCMT, cputick,1
;                                     текстовая переменная для служебной информации
WNPRT, cputick,1
;                                     текстовая переменная для служебной информации
WPROG, cputick,1
;                                     текстовая переменная для служебной информации
WIZKD, cputick,1
;                                     мощность шпинделя [%]
SpindlePower, Timer = 1000

```

1.12.1. Обработка ошибок в инструкциях и переменных

При запуске «АРМД» все инструкции, переменные и их значения, записанные в файле **moncfg.ini**, проверяются. В результате проверки создается список инструкций и переменных, записанных с ошибками, который может быть выведен на экран УЧПУ в среде «АРМД»->«Анализ конфигурации».

1.13. Файлы мониторинга

Файлы мониторинга предназначены для сохранения истории работы оборудования. Создание файлов мониторинга и их управление выполняет «АРМД» по указанному пути в файле **moncfg.ini**.

1.13.1. Порядок создания и заполнения файлов мониторинга в «АРМД»

При загрузке базового Про выполняется анализ инструкций и переменных «АРМД» в файле **C:\CNC32\MP0\moncfg.ini**. Если «АРМД» запустилась, то и выполняется попытка синхронизации времени УЧПУ и ПК, а также отправка пути доступа к внешнему файлу причин простоя для его копирования в УЧПУ. Если ресурс, записанный в инструкции **Path**, доступен, то на этом ресурсе выполняется поиск файла мониторинга, имя которого записано в файле **current.inf**.

Если файл мониторинга существует, и его текущий объем меньше максимального, то запись переменных «АРМД» продолжается в этот же файл. Если текущий файл мониторинга отсутствует или переполнен, то автоматически создается новый файл мониторинга и выполняется запись в его шапку начальных значений переменных, записанных в файле **moncfg.ini**. Далее выполняется накопление значений переменных «АРМД» в соответствии с типом их учета (типы учета приведены в таблице 2) в буфере, расположенном в ОЗУ. Накопленные данные в буфере ОЗУ записываются в файл мониторинга с перио-

дом времени, заданным в переменной **RecordTime** или по превышению размера буфера ОЗУ, в зависимости, что наступит раньше.

Если в УЧПУ не происходит никаких событий, то с периодом времени **RecordTime** в файл мониторинга записывается событие "NO EVENT" (НЕТ СОБЫТИЯ). Событие "NO EVENT" используется, как сторожевой таймер на внезапное отключение УЧПУ. Событие "NO EVENT" позволяет в ВПМ определить, что ПРОЦЕСС в УЧПУ работает, например, выполняет длительное перемещение с постоянной скоростью, при котором нет событий, или находится в состоянии IDLE ожидании команды, при котором нет событий, или давно уже выключено ключом, после чего также нет событий.

Принудительная запись данных буфера в файл выполняется при выключении станка установкой сигнала **MUSPE** в состояние «1» или при перезагрузке УЧПУ клавишами «**Ctrl**» + «**Alt**» + «**Del**». Текущий файл мониторинга заполняется данными до тех пор, пока текущее значение его объема меньше максимального объема, заданного в инструкции **FileSize**. Когда объем текущего файла мониторинга достигнет объема, заданного в инструкции **FileSize**, то он будет закрыт и создан следующий файл мониторинга по тому же принципу. Имя текущего файла мониторинга всегда записывается на указанном ресурсе в файле **info.mon**.

Если ресурс для создания файлов мониторинга становится не доступным, то автоматически «АРМД» будет выполнять запись регистрируемых машинных данных в 2-х временных файлах мониторинга в каталоге **C:\CNC32\MP0**. Имена временных файлов мониторинга генерируются базовым Про аналогично файлам мониторинга, создаваемых на ресурсе. Структура временных файлов мониторинга аналогична структуре файлов мониторинга на ресурсе. Максимальный объем каждого временного файла мониторинга может быть равен значению, определенному в инструкции **Local FileSize**. Имя текущего временного файла мониторинга записывается в файле **C:\CNC32\MP0\localmon.inf**. Если 2-ой временный файл переполнится, то 1-ый временный файл мониторинга будет удален, и на его месте будет создан очередной временный файл мониторинга.

Внимание! Длительное выполнение мониторинга во временные файлы может повлечь потерю части регистрируемых машинных данных.

Как только ресурс для создания файлов мониторинга становится доступным, то автоматически временные файлы мониторинга копируются на ресурс и далее «АРМД» выполняет запись регистрируемых машинных данных в файлы мониторинга на ресурсе, пока он доступен.

1.13.2. Принцип создания имени для файлов мониторинга

Имя файла мониторинга (макс. 16 символов) генерируется Про УЧПУ на основании текущей даты и времени в момент его создания.

Семантика:

YYYYMMDDUUNNSS.mon

Где:

YYYY – год создания файла (4 знака);
MM – месяц создания файла (2 знака);
DD – дата создания файла (2 знака);
UU – час создания файла (2 знака);
NN – минута создания файла (2 знака);
SS – секунда создания файла (2 знака);
mon – расширение файла (3 знака).

Пример имен файлов мониторинга, создаваемых на внешнем ПК:

20111216151415.mon
 20111216152232.mon
 20111216152309.mon

Файлы мониторинга.

Формат данных файлов мониторинга: двоичный.

Каждый файл мониторинга содержит заголовок с его идентификацией и данные мониторинга.

В файле **info.mon** в 1-ой строке система записывает имя файла мониторинга, в который в данный момент времени выполняется запись данных. Файл **info.mon** имеет только одну строку длиной 25 символов, в которых 1-ые 14 символов идентифицируют имя текущего файла мониторинга в формате **YYYYMMDDUUNNSS.mon**, а оставшиеся 11 символов с кодом 0 являются резервными.

Пример файла **info.mon**:

20120116152346.mon

1.14. Последовательность действий для запуска «АРМД»

Для запуска «АРМД» необходимо выполнить следующие действия:

- 1) установить в УЧПУ Про - «АРМД»;
Для получения Про - «АРМД» для УЧПУ можно обратиться к ООО «Балт-Систем» или к разработчику ВПМ.
- 2) установить на ПК ВПМ.
Для получения ВПМ для ПК необходимо обратиться к ее разработчику.
- 3) выполнить настройку локальной сети, создать каталог на сетевом ресурсе для сохранения в нем файлов мониторинга и сделать к нему доступ для УЧПУ, если файлы мониторинга должны создаваться на сетевом ресурсе.
Запрет доступа к сетевому ресурсу могут создавать различные антивирусные программы защиты и безопасности ПК от внешних сетевых атак. В этом случае такие программы необходимо отключить или создать для них особые правила, разрешающие создание файлов мониторинга на сетевом ресурсе ПК.
- 4) выполнить подключение внешнего диска (USB-диск, USB-FLASH) к УЧПУ и создать каталог для сохранения в нем файлов мониторинга, если файлы мониторинга должны создаваться на внешнем диске;
- 5) создать файл `c:\cnc32\mp0\moncfg.ini` и сконфигурировать в нем инструкции и переменные «АРМД»;
- 6) создать файл причин простоя `c:\cnc32\mp0\downtime.lst` и записать в него список причин простоя;
- 7) включить УЧПУ. Если файлы конфигурации «АРМД» изменялись, то рекомендуется сделать анализ конфигурации, проверку состояния ресурса и проверку внутренних сообщений. Для этого в главном меню режима «КОМАНДА» должна быть выбрана опция «АРМД».

Примечания.

1. если файлы мониторинга создаются на сетевом ресурсе, то изменение значений переменных «АРМД» на УЧПУ (см. таблицу 2) можно контролировать на экране ПК посредством запуска программы «**monitor1**» или ВПМ.
2. если файлы мониторинга создаются на диске, то порядок обработки данных определяет пользователь.

2. ПРИЛОЖЕНИЕ А (обязательное)

2.1. Структура записи данных в файл мониторинга

При запуске системы УЧПУ по указанному пути в файле **moncfg.ini** выполняется попытка открыть файл **info.mon** (местоположение см. инструкцию **Path** в **moncfg.ini**). В данном файле находится имя файла мониторинга, работа с которым происходила в последнем сеансе. Далее происходит попытка открыть этот файл. Если он существует, то из него считывается заголовок файла. Если заголовок совпадает с текущими параметрами мониторинга и датой создания файла, и его размер не превышает размеров указанных в **moncfg.ini**, тогда происходит дальнейшая запись в данный файл мониторинга, во всех других случаях создается новый (следующий) файл мониторинга. При этом записывается событие о старте системы, и перечисляются все начальные значения переменных в типе «сообщений», которые подвергаются мониторингу, далее система отслеживает события в обычном порядке. Если некоторое время не происходило ни одного события, то записывается событие «нет события», затем, если все же происходит какое-либо событие, то оно записывается поверх события «нет события». Сделано это для того, чтобы при длительном простое УЧПУ не было неучтенного времени.

При включении УЧПУ в текущий файл мониторинга записываются начальные состояния контролируемых событий (инициализация) и далее выполняется запись в файл мониторинга только тех событий, в которых произошли изменения.

Каждый раз после создания очередного файла мониторинга в него записывается заголовок и после заголовка записываются произошедшие в УЧПУ события. При смене текущей даты проявляется событие «новая дата».

Любые события записываются в файле мониторинга, как сообщения. Сообщение может содержать несколько событий в разных процессах. Записываются только те события, значения которых изменились. Если значения событий принадлежащих процессу не изменятся, процесс не будет записан. Например: время опроса подачи 10 тиков, если за это время подача не изменит свое значение, то событие не будет сформировано. Формат сообщений представлен ниже.

Запись выполняется в строку.

«АРМД». Руководство по эксплуатации.

Структура записи для заголовка файла мониторинга представлена на рисунке А.1.

Идентификатор файла мониторинга для УЧПУ серии NC-XXX 4 байт(char)	Версия структуры файла мониторинга 4 байта (char)	Длина файла мониторинга 4байта (unsigned long)	Имя предыдущего файла 25 байт (char) без признака конца строки	Имя следующего файла 25 байт(char) без признака конца строки	Длина строки N1(BYTE)	Версия ПРО N1*(BYTE)
---	--	---	---	---	--------------------------	-------------------------

Имя станка 25 байт(char) без признака конца строки	ТИК УЧПУ (инструкция TIM файл AXCFIL) 2 байта(short)	Год 2 байта (WORD)	Месяц 2 байта (WORD)	День 2 байта (WORD)
---	---	-----------------------	-------------------------	------------------------

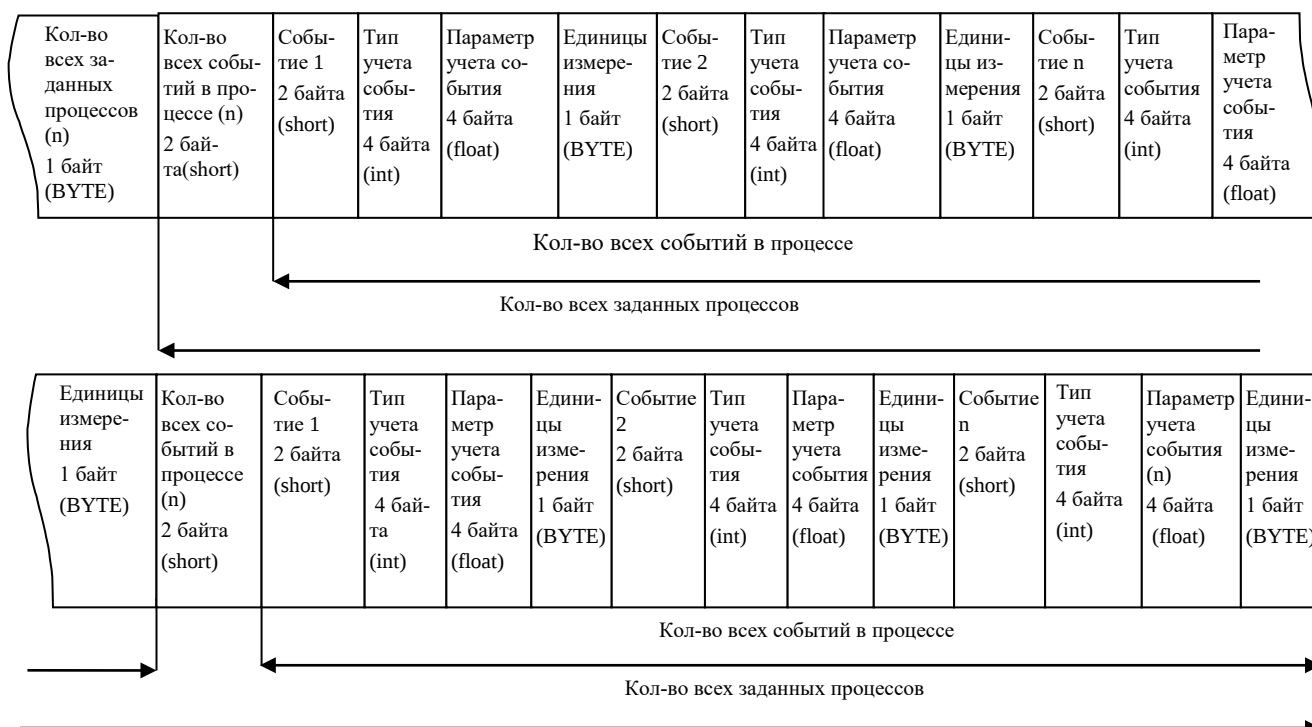


Рисунок А.1

Значения переменных при их инициализации и значения переменных в процессе работы (события) записываются в виде сообщения. Структура сообщения представлена на рисунке А.2.

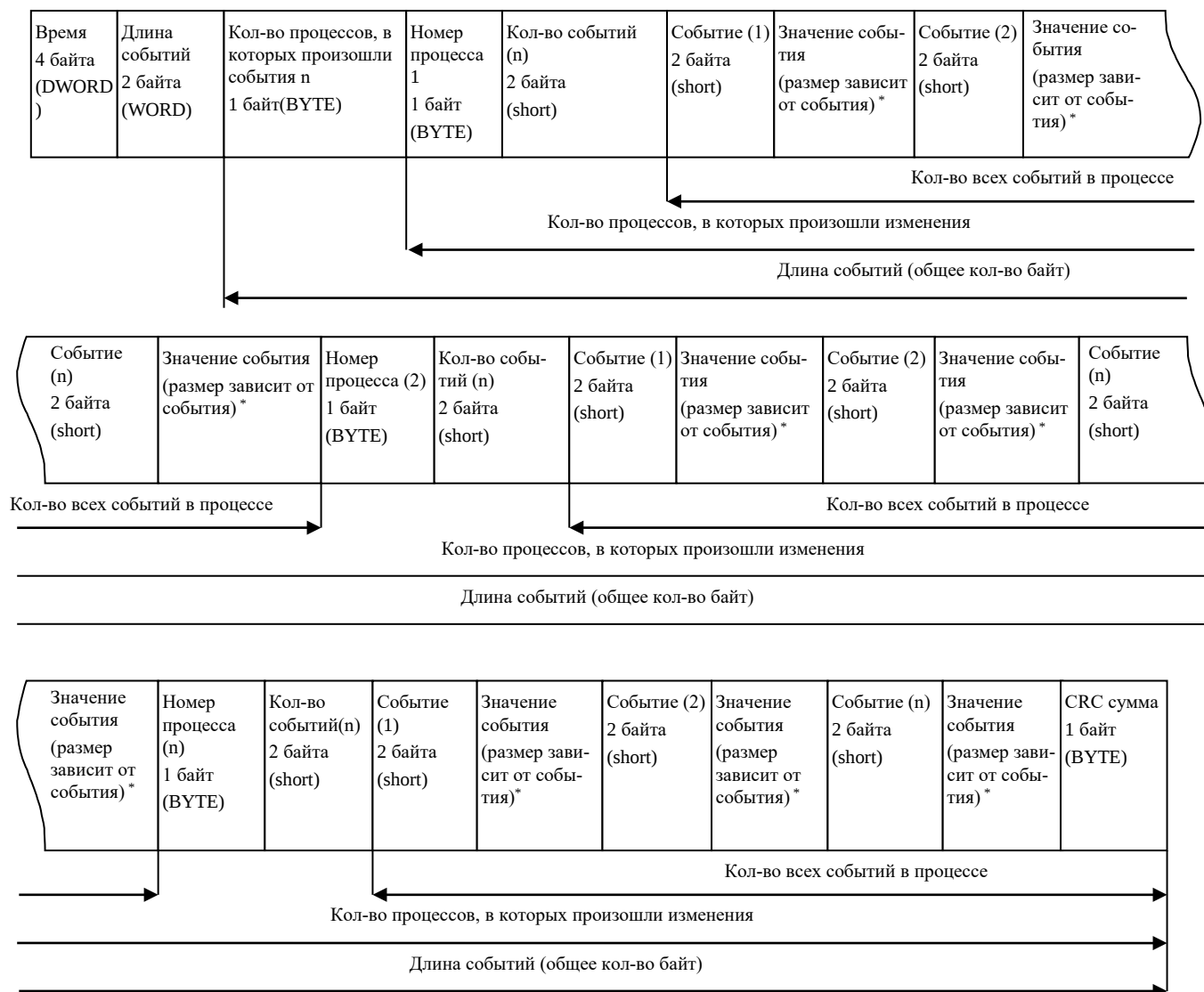


Рисунок А.2.

Примечание. Поля структуры, отмеченные символом *, определены в таблице А.1.

Структура данных событий

Таблица А.1

Имя переменной в файле moncfg.ini	Мнемоническое название номера события	Номер (идентификатор) события	Структура данных события				
Не задается	EVENT_NO_EVENT	1	отсутствует				
Не задается	EVENT_SYSTEM_START	2	Год (WORD)	Месяц (WORD)	День (WORD)	Время (DWORD)	
Не задается	EVENT_NEW_DATE	3	Год (WORD)	Месяц (WORD)	День (WORD)		
WorkMode	EVENT_WORK_MODE	4	Значение (char)				
Feed	EVENT_FEED	5	Значение (float)				
SpindleSpeed	EVENT_SPINDLE_SPEED	6	Значение (float)				
SystemState	EVENT_SYSTEM_STATE	7	Значение (float)				
MesErrProg	EVENT_EMERGENCY_ERROR	8	Номер Ошибки (char)	Длина Строки N (char)	Строка (N*char)		
ProgramName	EVENT_PROGRAM_NAME	9	Уровень (BYTE)	Кол-во символов N1 (BYTE)	Имя файла (N1*char)	Кол-во символов N2 (BYTE)	Путь (N2*char)
SwitchJOG	EVENT_CONTROL_PANEL_SWITCH_JOG	10	Значение (float)				
SwitchFeed	EVENT_CONTROL_PANEL_SWITCH_FEED	11	Значение (float)				
SwitchSpindle	EVENT_CONTROL_PANEL_SWITCH_SPINDLE	12	Значение (float)				
NumberProgramBlock	EVENT_BLOCK_NUMBER_CTRL_PROG	13	Значение (long)				
ToolNumber	EVENT_TOOL_NUMBER	14	Значение (WORD)				
Corrector-ToolNumber	EVENT_CORRECTOR_NUMBER	15	Значение (WORD)				

Продолжение таблицы А.1

Имя переменной в файле moncfg.ini	Мнемоническое название номера события	№ (идентификатор) события	Структура данных события			
UAS	EVENT_UAS	16	Значение (char)			
UVR	EVENT_UVR	17	Значение (char)			
URL	EVENT_URL	18	Значение (char)			
COMU	EVENT_COMU	19	Значение (char)			
CEFA	EVENT_CEFA	20	Значение (char)			
MUSP	EVENT_MUSP	21	Значение (char)			
REAZ	EVENT_REAZ	22	Значение (char)			
MACHINE_DOWN TIME_CAUSE	EVENT_MACHINE_DOW NTIME_CAUSE	23	Кол-во (char) N1 N1* Значение(char)Кол-во символов(char) N2 строка N2*(char)Кол-во символов(char) N3строка N3*(char)			
AlarmErrPLC	EVENT_ALARM_PLC_E RROR	24	Кол-во символов(char) N1 Строка N1*(char)			
MesErrPLC	EVENT_MESS_PLC_ER ROR	25	Кол-во символов(char) N1 Строка N1*(char)			
ProcessCom- mandLine	EVENT_PROCESS_COM MAND_LINE	26	Кол-во символов(char) N1 Строка N1*(char)			
Pro- cessBlock- Line	EVENT_PROCESS_BLO CK_LINE	27	Кол-во символов(char) N1 Строка N1*(char)			
CommandLine	EVENT_COMMAND_LIN E	28	Кол-во символов(char) N1 Строка N1*(char)			
PartFinished	EVENT_PART_FINISH ED	29	Значение (char)			

Продолжение таблицы А.1

Имя переменной в файле moncfg.ini	Мнемоническое название номера события	Номер (идентификатор) события	Структура данных события	
GFunctions	EVENT_G_FUNCTIONS	30	Кол-во символов(BYTE) N1	Значение(BYTE) N1*(BYTE)
RISP	EVENT_RISP	31	Значение (char)	
CONP	EVENT_CONP	32	Значение (char)	
SPEPNREQ	EVENT_SPEPN_REQ	33	Значение (char)	
ASPEPN	EVENT_A_SPEPN	34	Значение (char)	

3. ПРИЛОЖЕНИЕ В (обязательное)

3.1. Вспомогательная программа «monitor1»

Основным назначением программы «**monitor1**» является преобразование двоичного формата данных файла мониторинга в текстовый вид и вывод текста на экран. Для изучения структуры файлов мониторинга разработчику ВПМ рекомендуется рассмотреть исходные тексты программы «**monitor1**» и комментарии в теле программы.

Если использовать программу «**monitor1**», как самостоятельную программу для вывода текста файлов мониторинга на экран, то необходимо выполнить следующие действия:

1. Скопировать файл **monitor1.exe** в каталог с файлами мониторинга.
2. Выполнить запуск программы «**monitor1**» командой:
monitor1.exe [имя файла мониторинга.mon] с клавишей **ENTER**.

Если в команде запуска программы «**monitor1**» параметр: **[имя файла мониторинга.mon]** не указан, то будет использовано имя файла мониторинга, записанное в файле **info.mon**. В этом случае после вывода на экран данных уже сохраненных в файле мониторинга программа «**monitor1**» будет ожидать появления в файле мониторинга очередной пачки данных.

Если необходимо выполнить перенаправление вывода двоичных данных файла мониторинга в текстовый файл, то команда запуска программы «**monitor1**» должна иметь следующий вид:

monitor1.exe [имя файла мониторинга.mon] > имя файла.txt с клавишей **ENTER**.

Для завершения работы программы нажмите одновременно клавиши:
«Ctrl» + «C».

4. ПРИЛОЖЕНИЕ С (обязательное)

4.1. Вспомогательная программа «BSserver»

Назначением программы «BSserver» является синхронизация времени УЧПУ со временем сервера и отправка файла причин простоя в УЧПУ. Для изучения процесса синхронизации времени УЧПУ и сервера, а также обработка запроса передачи данных файла причин простоя в УЧПУ разработчику ВПМ рекомендуется рассмотреть исходные тексты программы «BSserver» и комментарии в теле программы.

Если использовать программу «BSserver», как самостоятельную программу для указанных выше целей, то необходимо выполнить следующие действия:

1. Выключить УЧПУ.
2. Скопировать файл **BSserver.exe** в отдельный каталог ПК.
3. Выполнить запуск программы «BSserver» командой:
BSserver.exe с клавишей **ENTER**.
4. Создать на ПК в сетевом каталоге, путь к которому записан в файле **moncfg.ini** в инструкции **Path**, файл причин простоя УЧПУ.
5. Включить УЧПУ.
6. Контролировать:
 - a. создание файла причин простоя УЧПУ в рабочем каталоге УЧПУ с именем, записанным в инструкции **DownTimeList** в файле **moncfg.ini**;
 - b. запись в файлы мониторинга сообщений о синхронизации времени УЧПУ со временем сервера «**TIME_SYNC**» с интервалом, заданным в инструкции **NetTimeSynchroInterval** в файле **moncfg.ini**.

Для завершения работы программы нажмите одновременно клавиши:
«Ctrl» + «C».

5. ПРИЛОЖЕНИЕ D

5.1. Особенности применения переменных.

Определение состояния оборудования возможно в том случае, если ВПМ обеспечит одновременный анализ нескольких переменных.

Например, для определения полезного времени работы оборудования можно использовать переменные **WorkMode**, **SystemState** и **MUSP**

Пример:

(1) + (RUN) + (AUTO+STEP) = (1)
MUSP + SystemState+ WorkMode = Отладка УП без станка

(0) + (RUN) + (AUTO+STEP)+ (1) = (1)
MUSP + SystemState+ WorkMode + UVR = Отладка УП без детали

(0) + (RUN) + (STEP) + (0) = (1)
MUSP + SystemState+ WorkMode + UVR = Отладка УП в пошаговом режиме работы

(0) + (RUN) + (AUTO) + (0) = (1)
MUSP + SystemState+ WorkMode + UVR = Выполнение УП в автоматическом режиме работы

Если установлено, что оборудование выполняет УП в автоматическом режиме, то можно выполнить разделение контроля «выполнения перемещений на скорости рабочей подачи» (**G01/G02/G03/G6/G33/G34**) и «выполнения перемещений на скорости быстрого хода» (**G00**) и обеспечить контроль выполнения технологии в процессе обработки деталей. Для этого можно контролировать следующие параметры:

- 1) коррекцию скорости выполняемых перемещений (**Feed**), заданную корректорами F% (**SwitchFeed**) и S% (**SwitchSpindle**), если перемещения выполняются на рабочей подаче
- 2) коррекцию скорости выполняемых перемещений (**Feed**), заданную корректором JOG% (**SwitchJOG** при условии **URL(=1)**), если перемещения выполняются на скорости быстрого хода;
- 3) контролировать изменение времени выполнения каждой детали из одной партии (время от запуска УП до появления признака «Конец обработки детали» (**PartFinished**));
- 4) оценивать изменение времени обработки каждой детали по отношению ко времени обработки 1-ой контрольной детали.

Для определения эффективности работы оборудования на основании полученных значений переменных мониторинга можно рассчитать параметры: «**Готовность**», «**Производительность**» и «**Качество**».

Некоторые особенности применения других переменных мониторинга.

NumberProgramBlock + MesErrProg – позволяет идентифицировать кадр УП в момент аварии на станке. Обычно в момент аварии выполняется «общий сброс» и информация о текущем выполняемом кадре теряется.

6. ПРИЛОЖЕНИЕ Е

6.1. Соответствие переменных «АРМД» сигналам интерфейса PLC.

Таблица D.1

Имя пере- менной «АРМД»	ПРОЦЕСС 1	ПРОЦЕСС 2	ПРОЦЕСС 3	ПРОЦЕСС 4	ПРОЦЕСС 5
CONP	I0K2	I26K2	I52K2	I78K2	I104K2
ASPEPN	I6K21	I32K21	I58K21	I84K21	I110K21
MUSP	U10K0	U36K0	U62K0	U88K0	U114K0
REAZ	U10K1	U36K1	U62K1	U88K1	U114K1
RISP	U10K7	U36K7	U62K7	U88K7	U114K7
SPEPNREQ	U10K20	U36K20	U62K20	U88K20	U114K20
COMU	U10K24	U36K24	U62K24	U88K24	U114K24
CEFA	U10K25	U36K25	U62K25	U88K25	U114K25
MesErrPLC	W17K3	W43K3	W69K3	W95K3	W121K3
AlarmErrPLC	21K÷22K	47K÷48K	73K÷74K	99K÷100K	125K÷126K

7. ПРИЛОЖЕНИЕ F

7.1. Программа ARMD Converter.

Программа **ARMD Converter** (далее программа) предназначена для ручной обработки событий, сохраненных в файлах мониторинга. Программа выполняет следующие операции с файлами мониторинга:

- 1) фильтрацию выбранных событий за указанный период времени.
- 2) преобразование двоичного вида отфильтрованных событий в текстовый вид.
- 3) Сохранение отфильтрованных событий в текстовом виде в файл.

Примечание. Программа высылается по e-mail по заказу.

Для работы программы необходимо выполнить следующие действия:

- Запустить файл ARMDConverter.exe. Вид основной видеостраницы программы представлен на рисунке F.1

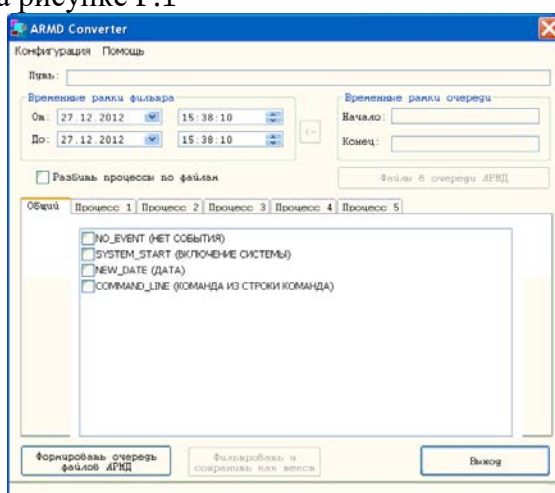


Рисунок F.1

- Сформировать очередь из файлов АРМД (мониторинга) для выполнения выборки событий в этих файлах. Для этого нажмите экранную клавишу «Формировать очередь файлов АРМД». В окне «Обзор папок» установите курсор на папку, содержащую файлы мониторинга и нажмите клавишу «ENTER». Выбранная папка с файлами мониторинга на экране должна иметь иконку в виде открытой папки и нажать экранную клавишу «ОК». Пример видеостраницы представлен на рисунке F.2.

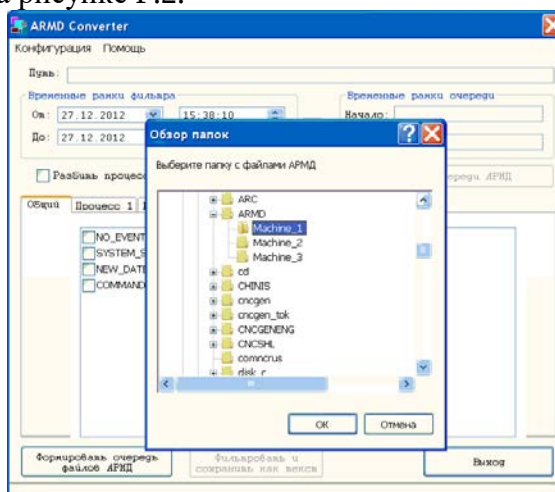


Рисунок F.2

- ввести начальные значения даты и времени и конечные значения даты и времени, интересующего периода. Если выборка событий должна быть выполнена по всем файлам мониторинга, находящимся в указанной папке, нажмите экранную клавишу «<—». Если выборка событий должна быть выполнена за конкретный период времени, введите начальные и конечные значения даты и времени в окне «Временные рамки фильтра» файлам мониторинга. Пример видеостраницы представлен на рисунке F.3.

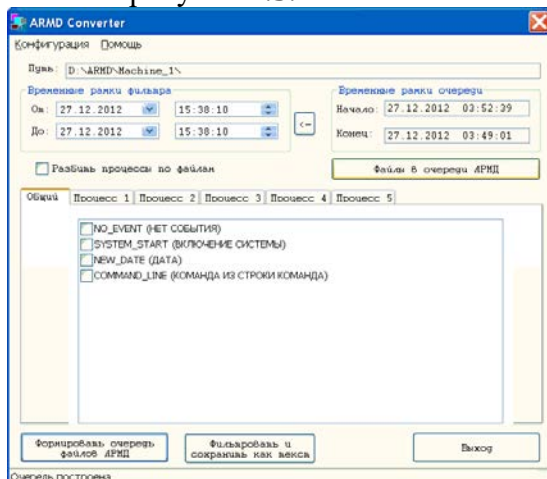


Рисунок F.3

- выбрать события. Пример видеостраницы для выбора общих переменных для всех процессов представлен на рисунке F.4. Пример видеостраницы для выбора переменных 1-ого процесса представлен на рисунке F.5. Для формирования в выходном файле-отчете временной последовательности событий рекомендуется всегда включать в список конвертируемых переменных две общие переменные: NO_EVENT и SYSTEM_START.

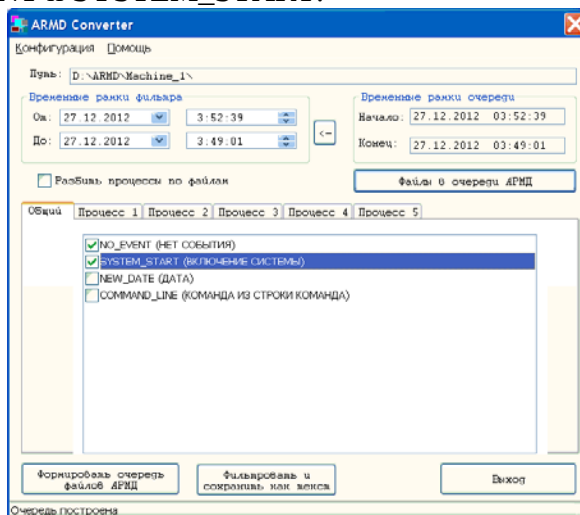


Рисунок F.4

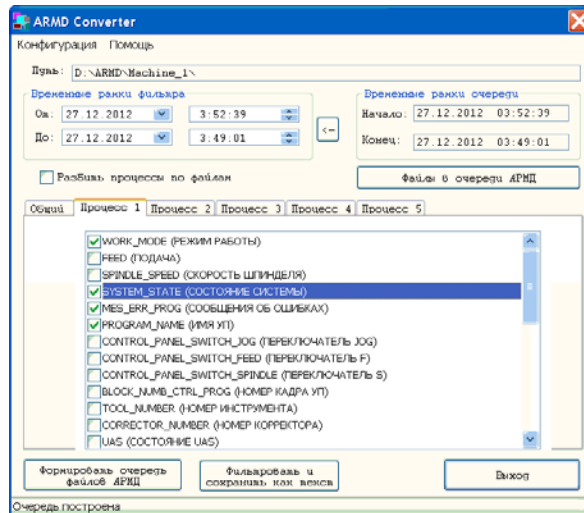


Рисунок F.5

- ввести имя текстового файла-отчета. Пример видеостраницы для ввода имени текстового файла-отчета по выбранным переменным представлен на рисунке F.6.

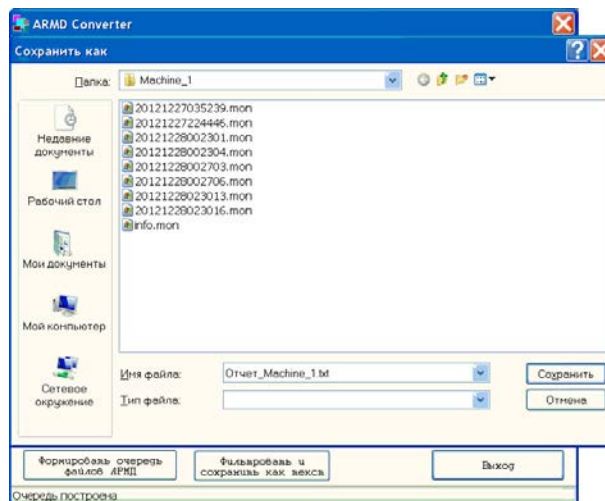


Рисунок F.6

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

ДЭС	- датчик обратной связи;
ОЗУ	- оперативное запоминающее устройство;
ПЛ	- программа логики станка;
Про	- программное обеспечение;
УП	- управляющая программа (для обработки детали);
УЧПУ	- устройство числового программного управления;
СРU	- центральный процессор;
АРМД	- автоматическая регистрация машинных данных;
ВРМ	- внешняя программа мониторинга.

ПЕРЕЧЕНЬ ТЕРМИНОВ

Инициализация	- установка начальных данных для функционирования Про УЧПУ в соответствии со значениями, объявленными при характеристике, и их диагностики.
ПЛ	- программа логики станка разрабатывается на языке PLC (см. документ «Программирование интерфейса PLC»).
УП	- управляющая программа, определяет технологию обработки детали.
Характеризация	- запись параметров и характеристик управляемого оборудования, а также аппаратных и программных модулей УЧПУ в соответствии с требованиями, предъявляемыми к составлению файлов FCSYS , AXCFIL , IOCFIL и PGCFIL , описанных в данном документе.
АРМД	- задача Про для автоматической регистрации состояния данных в УЧПУ и на станке. Запуск задачи « АРМД » выполняется в том случае, если в каталоге C:\CNC32\MP0 существует файл конфигурации moncfg.ini .
Файл мониторинга	- двоичный файл, в котором автоматически регистрируются машинные данные. Файлы мониторинга являются результатом выполнения « АРМД ». Файлам мониторинга присвоено расширение « mon ».
Мониторинг	- процесс обработки данных, созданных « АРМД » и расположенных в файлах мониторинга, для контроля и планирования работы подразделений предприятия.