

УТВЕРЖДЕН
ЮФКВ.50001- 02УД

ЯЧЕЙКА МВ11.04. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Библиотека программ интерфейса ячейки МВ11.04 и прикладных программ

Описание программы

ЮФКВ.50001-0213 01

(ЮФКВ.50001-0213 01-001ФЛ)

Листов 24

Литера

Инв.№ подл.	Подп. и дата	Взам.инв.№

АННОТАЦИЯ

Настоящий документ является описанием расширения библиотеки программ интерфейса ячейки MB11.01 для ячейки MB11.04. Содержит сведения, необходимые для разработки прикладных программ, использующих возможности ячейки MB11.04 (для исполнения в среде OS MS WINDOWS 95/98). Ячейка MB11.04 в составе ПЭВМ типа IBM PC является реализацией устройства интерфейса в терминах ГОСТ 26765.52-87 "ИНТЕРФЕЙС МАГИСТРАЛЬНЫЙ ПОСЛЕДОВАТЕЛЬНЫЙ СИСТЕМЫ ЭЛЕКТРОННЫХ МОДУЛЕЙ" и обеспечивает выполнение функций контроллера, или оконечного устройства, монитора или тестера функциональной логики интерфейса. Возможен совмещенный режим выполнения функций оконечного устройства и монитора интерфейса. Ячейка MB11.04 обеспечивает работу в соответствии со спецификациями стандарта MIL-STD-1553A, а также MIL-STD-1553B Notes 2 и полностью функционально идентична устройствам построенным на интегральной схеме (ИС) BU-61580 фирмы DDC.

СОДЕРЖАНИЕ

1. Общие сведения и функциональное назначение	4
2. Общая структура библиотеки.....	5
3. Индикация ошибок и особенных ситуаций	6
3.1. Коды ошибок	6
4. Доступ к регистрам, ОЗУ и конфигурирование ячейки MB11.04	7
4.1. Модуль access	7
4.1.1. Распознавание типа устройства	7
Дополнительные регистры MB11.04 и Тестера.....	7
4.2. Модуль Config	8
4.2.1. Дополнительные регистры тестера:.....	8
4.2.2. Дополнительные флаги для режима ГСП	8
4.2.3. Функции конфигурирования устройства	8
4.2.4. Функция mzp_SelectMode	8
4.2.5. Функция gsw_Config	8
4.3. Модуль gsw	9
5. Функции для режима Генерации словных последовательностей (Модуль GSW)	10
5.1. Основные положения и термины	10
5.2. Объекты.....	11
5.3. Структуры данных	11
5.3.1. Структура данных, описывающая внесение ошибок в сообщения.....	11
5.3.2. Дополнение структуры данных mzp_Message.....	12
5.4. Работа с кадрами	12
5.4.1. Функция mzp_WGCreateFrame	12
5.4.2. Функция mzp_WGLoadFrame	13
5.4.3. Функция mzp_WGGetMessageCount	13
5.4.4. Функция mzp_WGDestroyFrame	13
5.5. Работа с сообщениями	13
5.5.1. Функция mzp_WGFormatMessage	13
5.5.2. Функция mzp_WG_ExtendMessage	14
5.6. Целостное создание сообщений.....	15
5.6.1. Функция mzp_WG_CreateMessage	15
5.6.2. Функции mzp_WG_SetTransmit и mzp_WG_SetCheck	15
5.6.3. Функция mzp_WG_DestroyMessage	16
5.7. Пословное создание и модификация сообщений	16
5.7.1. Функция mzp_WG_CreateEmptyMessage	16
5.7.2. Функции mzp_WG_InsertWord и mzp_WG_RemoveWord	16
5.8. Внесение ошибок в сообщение	16
5.8.1. Функция mzp_WG_DivertWord	16
5.8.2. Функция mzp_WG_DivertWordIndirect	17
5.8.3. Функция mzp_WG_DivertMessageIndirect	17
5.9. Создание специальных сообщений.....	18
5.9.1. Функция mzp_WG_CreateTerminalMessage	18
5.10. Добавление сообщения в кадр	18
5.10.1. Функция mzp_WG_AddMessage	18
5.10.2. Функция mzp_WG_AddMessageIndirect	18
5.10.3. Функция mzp_WG_AddMessagesIndirect	19
5.10.4. Функция mzp_WG_GetFrameMessage	19
5.11. Проверка ответов тестируемого устройства	19
5.11.1. Функция mzp_WG_CheckMessage	20
5.11.2. Функция mzp_WG_CheckMessages	20
5.11.3. Функция mzp_WG_GetGap	20
Приложение А Пример формирования сообщения с помощью rtl2gsw.....	21
Перечень сокращений	22
Перечень ссылочных документов.....	23

1. ОБЩИЕ СВЕДЕНИЯ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

Обозначение программы – “ЮФКВ.50001-01 13 01”. Наименование программы – “Библиотека программ интерфейса и прикладных программ ячейки MB11.04 ”. Сокращенное наименование – RTL2GSW (от Run Time Library Edition 2 и Generator of Sequences of Word).

RTL2GSW предназначена для обеспечения доступа прикладных программ к ОЗУ и регистрам ячейки MB11.01 и MB11.04. В состав программного обеспечения ячеек MB11.01 и MB11.04 входит «Драйвер устройства для OS MS WINDOWS 9X» (ЮФКВ.10016-01), который ДОЛЖЕН БЫТЬ ПРЕДВАРИТЕЛЬНО УСТАНОВЛЕН (см. [4]), он необходим для работы прикладных программ.

Библиотека поддерживает работу устройства интерфейса [2] в режимах контроллера (КШ), оконечного устройства (ОУ), монитора (МТ) (словный монитор и монитор сообщений), причем возможен совмещенный режим оконечного устройства и монитора сообщений и в режиме тестера (ТСТ) протокола логики участника обмена по каналу интерфейса ГОСТ 26765.52-87. В режиме ТСТ в работу включается так называемый Генератор Словных Последовательностей (далее ГСП), который и производит формирование тестовых сообщений.

Расширение библиотеки RTL2 введено, в основном, для поддержки функционирования ячейки MB11.04 в режиме Тестера. В отличие от ячейки MB11.01 у MB11.04 расширен диапазон как внутренней (128КСлов), так и внешней (128КСлов) памяти, введены дополнительные регистры см. [5]. Для управления ячейкой и формирования тестовых сообщений и служат дополнения, описанные в настоящем документе. В остальных режимах, управление ячейкой MB11.04 осуществляется аналогично MB11.01 см. [1].

2. ОБЩАЯ СТРУКТУРА БИБЛИОТЕКИ

Библиотека состоит из нескольких модулей, назначение которых следующее:

- **access** - обеспечивает доступ к регистрам и ОЗУ устройства, получение управления по прерыванию; этот модуль образует "нижний" уровень библиотеки, услугами которого пользуются остальные модули;
- **errors** - обеспечивает индикацию ошибок и особых ситуаций;
- **types** - объявления используемых примитивных типов данных;
- **config** - в этом модуле объединены функции конфигурирования устройства, а также функции, ориентированные на протокол MIL-STD-1553, общие для всех режимов работы устройства;
- **bc** - функции для работы в режиме контроллера шины (КШ);
- **rt** - функции для работы в режиме оконечного устройства (ОУ);
- **mt** - функции для работы в режиме монитора шины (МТ);
- **int** - диспетчер обработки прерываний;
- **gsw** - функции для работы в режиме тестера шины (ТСТ).

Каждый модуль представлен файлом заголовка <имя модуля>.h и объектным модулем в библиотечном файле rtl2gsw.lib .

Дальнейшее описание функций библиотеки производится, в основном, по модулю **gsw** , описание остальных модулей производится в документе «библиотека программ интерфейса ячейки MB11.01 и прикладных программ. Описание программы» [1], за исключением изменений, которые описываются в настоящем документе.

3. ИНДИКАЦИЯ ОШИБОК И ОСОБЕННЫХ СИТУАЦИЙ

Каждая функция RTL2GSW при обнаружении ошибки устанавливает код ошибки. В дополнение к существующим кодам ошибки для RTL2 [1] вводятся дополнительные:

3.1. Коды ошибок

Таблица 1

Идентификатор	Комментарий
msp_ERROR_TOO_MANY_FRAMES	Создано недопустимо большое кол-во кадров
msp_ERROR_INVALID_MESSAGE_HANDLE	Недействительный хэндл сообщения
msp_ERROR_INVALID_FRAME_HANDLE	Недействительный хэндл кадра
msp_ERROR_ANSWER_NOT_MATCH	Ошибка при сравнении (результат не равен ожидаемому)
msp_ERROR_ANSWER_MISSING_WORD	Ошибка при сравнении (отсутствие ожидаемого слова)
msp_ERROR_ANSWER_EXTRA_WORDS	Ошибка при сравнении (присутствуют лишние слова)
msp_ERROR_INVALID_MESSAGE	Параметры сообщения – недопустимы.
msp_ERROR_INTERNAL_ALGORITHM	Отладочная ошибка (при появлении – обратиться к разработчику.)

4. ДОСТУП К РЕГИСТРАМ, ОЗУ И КОНФИГУРИРОВАНИЕ ЯЧЕЙКИ MB11.04

4.1. Модуль access

Функции этого модуля обеспечивают доступ к управляющим регистрам МСП, ОЗУ ячейки MB11.04, установку подпрограммы обработки прерываний и получение управления по прерыванию.

MB11.04 взаимодействует с ПЭВМ по интерфейсу PCI в режиме подчинённого устройства. Дополнительные функции модуля **access** обеспечивают доступ к управляющим регистрам контроллера интерфейса шины PCI ПЭВМ. Эти функции не являются частью спецификации RTL2GSW и могут не поддерживаться в последующих версиях и портах.

Функции данного модуля также обеспечивают установление и прекращение связи с MB11.04, а также получение информации об установленных ячейках MB11.0X и выбор ячейки по набору признаков.

4.1.1. Распознавание типа устройства

Распознавание типа устройства осуществляется на основе регистра ячейки - "идентификатор типа функциональной логики" (mspRR_EXTERNAL_FLOG_ID, 0x23) см. [5]. От типа функциональной логики также зависят признаки VendorID/DeviceID в конфигурационном пространстве PCI, которые могут использоваться как дополнительные признаки.

Тип устройства (название модуля)	Функциональные особенности	Идентификатор типа функциональной логики	VendorID / DeviceID
Адаптер на осн. BU-61580 (MB11.01)	КШ, ОУ, МТ	0xFF и 0x00	0x6403/ 0x0430
Адаптер на осн. КМК101/001 (MB11.04)	КШ, ОУ, МТ	0x01	0x6403 / 0x0431
Адаптер на осн. КМК101/001 (MB11.04)	Тестер	0x02	0x6403 / 0x0431
Адаптер на осн. КМК101/001 (MB11.04)	КШ, ОУ, МТ, Тестер	0x03	0x6403 / 0x0431

Значения в таблице приведены на настоящий момент с возможным последующим обновлением.

Для обозначения устройств внутри RTL2GSW введены константы (файл access.h):

mspLT_BU (0xFF) – для MB11.01,
mspLT_PLM1 (0x00) – для MB11.04,
mspLT_TESTER1 (0x01) – для режима работы - Тестер.

Далее под термином MB11.04 будут подразумеваться устройства MB11.04 с наличием тестера и без него, если иное не оговорено отдельно. А под термином Тестер – режим тестера ячейки MB11.04.

Для распознавания типа устройства приложением в структуру msp_DeviceInfo вводится дополнительное поле msp_DWORD LType, содержащее одно из значений mspLT_XXXX.

Дополнительные регистры MB11.04 и Тестера.

Регистр - делитель тактовой частоты.
mspRR_EXTERNAL_CDIVIDER (0x22) см.[5].

Для значений этого регистра вводятся константы:

msp_CLOCK_12MHz	1
msp_CLOCK_16MHz	0
msp_CLOCK_24MHz	2
msp_CLOCK_48MHz	3

При открытии (msp_OpenAccessXX()) устройства MB11.04 в этот регистр заносится значение msp_CLOCK_16MHz, а при открытии тестера - msp_CLOCK_24MHz.

4.2. Модуль **Config**

4.2.1. Дополнительные регистры тестера:

Указатель стека данных ГСП.
mspRR_WG_DATA_STACK_POINTER (0x03),

Начальный указатель стека данных ГСП
mspRR_WG_INITIAL_DATA_STACK_POINTER (0x04),

Подробнее эти регистры описаны в [5].

4.2.2. Дополнительные флаги для режима ГСП

Генератор активен - mspF_GENERATOR_ACTIVE .

4.2.3. Функции конфигурирования устройства

Вводится новый режим msp_MODE_WG (0x0020) (config.h) и 2 новых опции для msp_SelectMode(): msp_WG_FIRST (0x0000) и msp_WM_FIRST (0x4000) в дополнение к [1].

4.2.4. Функция **msp_SelectMode**

Выбор режима (описана в [1]). Для MB11.01/04 допустимы режимы в соответствии со спецификацией RTL2 вер. 1.01.

В режиме тестера нет выделенных областей памяти; никакого резервирования не требуется !

Заголовок:

msp_ERROR msp_SelectMode(msp_DEVHANDLE device, msp_MODE mode);

Параметры: device - хэндл устройства
mode - режим работы. Для тестера - допустим режим msp_MODE_WG с опциями msp_WM_FIRST, msp_WG_FIRST (по умолчанию, msp_WG_FIRST).

Возврат: Код ошибки или 0.

Ошибки: msp_ERROR_INVALID_DEVICE_HANDLE
msp_ERROR_UNSUPPORTED_MODE
msp_ERROR_NO_HOST_MEMORY
msp_ERROR_FRAM_NOT_RESERVED

4.2.5. Функция **gsw_Config**

Конфигурирование устройства в режиме ГСП

Заголовок:

msp_ERROR gsw_Config(msp_DEVHANDLE device, msp_WORD select_mode);

Параметры: device - хэндл устройства
select_mode - одна из опций : msp_WG_FIRST или msp_WM_FIRST (запуск в подрежиме мониторинга или генерации).

Возврат: Код ошибки или 0.

Ошибки: msp_ERROR_INVALID_DEVICE_HANDLE
msp_ERROR_UNSUPPORTED_MODE
msp_ERROR_NO_HOST_MEMORY
msp_ERROR_FRAM_NOT_RESERVED

4.3. Модуль gsw

Функции этого модуля обеспечивают доступ к ОЗУ ячейки MB11.04, при работе ячейки в режиме тестера (TST). Для доступа к ОЗУ стека данных ГСП, которое представляет собой отдельный (дополнительный) блок из 64Кслов, вводятся п\п доступа к ОЗУ по аналогии с "одноименными" п\п для доступа к основному ОЗУ.

Для доступа к ОЗУ стека данных ГСП, которое представляет собой отдельный (дополнительный) блок из 64Кслов, вводятся п\п доступа к ОЗУ по аналогии с "одноименными" п\п [1] для доступа к основному ОЗУ. (файл gsw.c). Отличие в написании заключается только в префиксе WG имени функции, что указывает на обращение к памяти ГСП, и в параметре Address, который поменял тип на msp_DWORD для обращения к памяти свыше 32КБ:

```
msp_WORD  msp_WGReadRamW      (msp_DEVHANDLE device, msp_DWORD Address);  
  
msp_ERROR msp_WGWriteRamW     (msp_DEVHANDLE device, msp_DWORD Address, msp_WORD Data);  
  
msp_DWORD msp_WGReadRamDW     (msp_DEVHANDLE device, msp_DWORD Address);  
  
msp_ERROR msp_WGWriteRamDW    (msp_DEVHANDLE device, msp_DWORD Address, msp_DWORD Data);  
  
msp_ERROR msp_WGReadRamBLK    (msp_DEVHANDLE device, msp_WORD Address, void *Data,  
msp_WORD Length);  
  
msp_ERROR msp_WGWriteRamBLK   (msp_DEVHANDLE device, msp_WORD Address, void *Data,  
msp_WORD Length);  
  
msp_ERROR msp_WGFillRamBLK    (msp_DEVHANDLE device, msp_WORD Address, msp_WORD Data,  
msp_WORD Length);
```

5. ФУНКЦИИ ДЛЯ РЕЖИМА ГЕНЕРАЦИИ СЛОВНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ (МОДУЛЬ GSW)

5.1. Основные положения и термины.

Основная идея заключается в том, чтобы построить в ОЗУ УВ полный образ сообщения (транзакции) MIL-STD-1553, часть слов которого используется для генерации сообщений генератором словных последовательностей, а оставшаяся часть - для сравнения ответов тестируемых устройств с ожидаемыми.

Вводятся 2 новых объекта - стек данных (кадр) ГСП (msp_WG_FRMHANDLE) и сообщение ГСП (msp_WG_MSGHANDLE).

Кадр состоит из сообщений.

В отличие от стека данных КШ, кадр формируется в ОЗУ УВ. При операции "загрузки" необходимые части сообщений переписываются в ОЗУ ГСП.

В настоящий момент не допускается одновременное создание нескольких кадров. Для загруженного кадра резервируются все 64Кслова ОЗУ ГСП независимо от фактического размера кадра.

Сообщение - это образ сообщения MIL-STD-1553 в ОЗУ УВ плюс необходимая сопроводительная и управляющая информация. Работа с сообщениями ГСП по форме аналогична работе с сообщениями КШ (см.), но по сути имеются существенные различия.

- сообщение ГСП содержит (может содержать) все командные слова, слова данных и состояния, предполагаемые в данном сообщении, независимо от того, какое из устройств должно передавать эти слова;
- сообщение ГСП метится признаком "роли" или нескольких ролей, которые исполняет ГСП. В зависимости от этих меток, ГСП генерирует те или иные части сообщения, а оставшиеся части ("эталонные") используются для проверки ответа других устройств;
- Имеется механизм для внесения ошибок в генерируемые слова сообщений;
- Имеется механизм, регулирующий степень "придирчивости" операции сравнения;
- Сообщение может быть создано целиком, аналогично сообщениям КШ, или же оно может быть создано по-словными операциями. Имеются также п/п для инспекции и модификации сообщений, в том числе для добавления и удаления слов.

При операции загрузки кадра генерируемые части сообщений переписываются в ОЗУ ГСП.

При операции проверки сообщения, кадра эталонные части сообщений используются для сравнения с данными в ОЗУ словного монитора.

Сообщение состоит из слов. Помимо собственно слов сообщения в ОЗУ УВ формируется сопроводительная информация для интерпретации этих слов.

В зависимости от типа сообщения и заданной роли ГСП в этом сообщении слова делятся на генерируемые и эталонные. Признак передаваемое/эталонное сопровождает каждое слово.

Каждое генерируемое и эталонное слово сопровождается классификационным признаком, отличающим командные слова, слова данных и статуса.

Генерируемое слово сопровождается структурой msp_Diversion (см. ниже), на основе которой при загрузке кадра формируется "слово сопутствующих параметров" и - при необходимости - "слово описания ошибки" (см. "Тестер соответствия требованиям стандарта MIL-STD-1553B").

Для эталонных слов хранятся признаки, определяющие один из трех режимов сравнения этих слов с фактически переданными в сообщении:

- сравнение и проверка наличия слова в сообщении не производится;
- производится проверка наличия данного слова в сообщении;
- производится сравнение значения данного слова.

При начальном формировании сообщений формируется "нейтральная" сопроводительная информация:

- устанавливаются нормальные задержки между словами в соответствии с MIL-STD-1553 (различные для первого и последующих слов в сообщении);
- ошибки не вносятся;
- производится проверка наличия всех эталонных слов, предусмотренных данным типом сообщения;

- производится сравнение значений всех эталонных слов, предусмотренных данным типом сообщения, значения которых заданы при формировании сообщения.

В дальнейшем характеристики как генерируемых, так и эталонных слов могут быть изменены при помощи предназначенных для этих целей функций.

Формирование последовательности выдачи слов сообщения в канал, в связи с особенностями форматов, производится в следующем порядке:

CmdWord1	(Командное слово 1)
CmdWord2	(Командное слово 2)
StatusWord1	(Ответное слово 1)
Data	(Данные (до 32 слов))
StatusWord2	(Ответное слово 2).

5.2. Объекты

msp_WG_FRMHANDLE - хэндл кадра (программа) ГСП.

msp_WG_MSGHANDLE - хэндл сообщения ГСП.

(файл gsw.h)

5.3. Структуры данных.

5.3.1. Структура данных, описывающая внесение ошибок в сообщения.

```
typedef struct {  
    msp_BYTE wordref;  
    msp_BYTE gap;  
    msp_word flags;  
    msp_word error;  
} msp_Diversion;
```

wordref - указатель на слово сообщения. Могут использоваться

- константы msp_MW_XXXX (bc.h) и целые числа, как в msp_ReadReadMessageWord() (bc.h),
- комбинация из константы msp_MW_ABS (0x40) + <целое> как указатель абсолютной позиции слова в сообщении,

причем порядок расположения по абсолютным значениям следующий:

0	CmdWord1	(Командное слово 1)
1	CmdWord2	(Командное слово 2)
2	StatusWord1	(Ответное слово 1)
3	StatusWord2	(Ответное слово 2)
4	Data 1	(Слово данных 1)
5	Data 2	(Слово данных 2)
.....
35	Data 32	(Слово данных 32).

- специальное значение msp_MW_NO_MORE (0xFF) как признак конца массива структур msp_Diversion.

gap - величина паузы перед данным словом, единица младшего разряда соответствует 0,25мкс.

flags - флажковая часть управляющее слово ГСП (без gap). Состоит из

```
msp_WGCW_LOOP (1<<5; dsa_roll)  
msp_WGCW_ERROR (1<<4; error)  
msp_WGCW_COMMAND (1<<3; com_dat)  
msp_WGCW_CHANNEL_B(1<<2; ch_b_a)  
msp_WGCW_CONTIGUOUS (1<<1; cd)
```

В функциях, использующих msp_Diversion, используются только флажки (msp_CHANNEL_B, msp_COMMAND), остальные игнорируются. (Зато используются в других местах).

К этому полю может быть добавлено одно из специальных значений

- `msp_WGCW_SET (0x01)` - флаги остаются без изменений (позволяет сохранить установленные биты слова сопутствующих параметров (CCP))
- `msp_WGCW_INVERT (0x80)` – поле `flags` инвертируется, т.е. при записи `msp_WGCW_SET | msp_WGCW_INVERT | msp_WGCW_COMMAND` в поле `flags` будет инвертирован бит **com_dat** слова сопутствующих параметров.

error - слово описания ошибки. Специальное значение `msp_INJ_NONE (0xC000)` интерпретируется как отмена внесения ошибки, в остальных случаях слово описания ошибки соответствует [5] и формируется при помощи макроса

```
FormatErrorWord(error_type, error_pos, error_dur)
```

`error_type` - тип ошибки - одно из значений

- `msp_INJ_ERROR_INVERT_LEFT (0x00)`
- `msp_INJ_ERROR_INVERT_RIGHT (0x01)`
- `msp_INJ_ERROR_SKIP (0x02)`
- `msp_INJ_ERROR_WORDLEN (0x03)`

Остальные параметры интерпретируются в зависимости от типа ошибки

Тип ошибки	<code>error_pos</code>	<code>error_dur</code>
<code>msp_INJ_ERROR_INVERT_LEFT</code>	Позиция ошибки от начала слова, единица младшего разряда соответствует в 0,25мкс	Продолжительность ошибки, единица младшего разряда соответствует 41,6666нс
<code>msp_INJ_ERROR_INVERT_RIGHT</code>		
<code>msp_INJ_ERROR_SKIP</code>		
<code>msp_INJ_ERROR_WORDLEN</code>	число бит, на которые изменяется длина слова	-

5.3.2. Дополнение структуры данных `msp_Message`.

В дополнение к [1] введены дополнительные поля для сообщений ГСП (файл `config.h`).

```
msp_BYTE present, transmit, check, role;
```

Каждое из полей, кроме *role*, является набором следующих битовых флажков, соответствующих словам сообщений MIL-STD-1553:

- `msp_MSG_CMD1 (0x01)`,
- `msp_MSG_CMD2 (0x02)`,
- `msp_MSG_DATA (0x08)`,
- `msp_MSG_STATUS1 (0x04)`,
- `msp_MSG_STATUS2 (0x10)`.

present - отмечены заполненные поля `msp_Message`

transmit - отмечены поля, которые должен передавать ГСП

check - отмечены эталонные поля для проверки сообщений тестируемого устройства

Если поле отмечено в `transmit`, оно будет заполнено и в `present`.

Поля `transmit` и `check` задействуются только при установленном поле `present`.

Поля `bccw`, `loopback` в структуре `msp_Message` в режиме ГСП не используются.

5.4. Работа с кадрами

5.4.1. Функция `msp_WGCreateFrame`

Функция создает кадр ГСП. В каждый момент времени может существовать не более одного кадра (должно проверяться). Кадр занимает все ОЗУ ГСП.

Заголовок:

```
msp_WG_FRMHANDLE msp_WGCreateFrame(msp_DEVHANDLE device);
```

Параметры: device - хэндл устройства

Возврат: хэндл кадра или 0 в случае ошибки.

Ошибки: msp_ERROR_INVALID_DEVICE_HANDLE
msp_ERROR_TOO_MANY_FRAMES
msp_ERROR_NO_HOST_MEMORY

5.4.2. Функция **msp_WGLoadFrame**

Загрузка кадра. Передаваемые части сообщений переписываются в ОЗУ ГСП.
Допустимо вызывать функцию повторно неограниченное число раз.

Заголовок:

```
msp_WG_FRMHANDLE msp_WGLoadFrame(msp_WG_FRMHANDLE frame);
```

Параметры: frame - хэндл кадра

Возврат: хэндл кадра .

Ошибки: msp_ERROR_INVALID_DEVICE_HANDLE
msp_ERROR_INVALID_FRAME_HANDLE

5.4.3. Функция **msp_WGGetMessageCount**

Возвращает число сообщений в кадре.

Заголовок:

```
msp_WORD msp_WGGetMessageCount(msp_WG_FRMHANDLE frame);
```

Параметры: frame - хэндл кадра

Возврат: Кол-во сообщений.

Ошибки: msp_ERROR_INVALID_FRAME_HANDLE

5.4.4. Функция **msp_WGDestroyFrame**

Уничтожение кадра.

Заголовок:

```
msp_ERROR msp_WGDestroyFrame(msp_WG_FRMHANDLE frame);
```

Параметры: frame - хэндл кадра

Возврат: Код ошибки или 0.

Ошибки: msp_ERROR_INVALID_FRAME_HANDLE

5.5. Работа с сообщениями

5.5.1. Функция **msp_WGFormatMessage**

Предварительное форматирование сообщения.

Функция аналогична функции msp_FormatMessage() (bc.h) см.[1], со следующими отличиями:

- слова данных могут вноситься для любого типа сообщений, дополнительно введен новый тип **mspM_EXTENDED** (файл config.h), для формирования нестандартных сообщений.
- можно задать слова состояния ST1, ST2;
- функция отмечает все заданные поля в поле present
- параметр wgmcsw - комбинация следующих флажков:
msp_WGM_CHANNEL_B (0x80) всем словам сообщения приписывается канал А, если этот флаг не задан, и канал В в противном случае,
msp_ROLE_BC,
msp_ROLE_RT,
msp_ROLE_RT2 - признаки ролей (BC – контроллера шины, RT – оконечного устройства), действия которых выполняет ГСП;
- msp_ROLE_RT2 - роль принимающего ОУ при пересылке ОУ-ОУ.
- функция формирует поля transmit и check в соответствии с типом сообщения и заданным набором исполняемых ролей; к полю check добавляется (OR) значение параметра checkalso.

Заголовок:

```
msp_Message* msp_WG_FormatMessage(  
    msp_Message* buffer,  
    msp_BYTE type,  
    msp_BYTE RT,  
    msp_BYTE SA,  
    msp_BYTE RTR_MC,  
    msp_WORD SAR_MCD,  
    msp_BYTE dataWordCount,  
    msp_WORD *data,  
    msp_WORD ST1,  
    msp_WORD ST2,  
    msp_BYTE wgmcsw,  
    msp_BYTE checkalso);
```

Параметры: buffer - указатель на структуру msp_Message.
type - тип сообщения см. функцию msp_FormatMessage() (bc.h) [1]. дополнительно введен новый тип **mspM_EXTENDED** (файл config.h), для формирования нестандартных сообщений.
RT - см. функцию msp_FormatMessage() (bc.h) [1].
SA - см. функцию msp_FormatMessage() (bc.h) [1].
RTR_MC - см. функцию msp_FormatMessage() (bc.h) [1].
SAR_MCD - см. функцию msp_FormatMessage() (bc.h) [1].
dataWordCount - см. функцию msp_FormatMessage() (bc.h) [1].
data - см. функцию msp_FormatMessage() (bc.h) [1].
ST1 - значение ответного слова
ST2 - значение ответного слова 2 (только для сообщений RT→RT)
wgmcsw - комбинация следующих флажков:
msp_WGM_CHANNEL_B,
msp_ROLE_BC,
msp_ROLE_RT,
msp_ROLE_RT2.
checkalso - флаги для проверки дополнительных слов

Возврат: указатель на структуру msp_Message.
Ошибки: msp_ERROR_INVALID_FRAME_HANDLE

5.5.2. Функция **msp_WG_ExtendMessage**

Функция предназначена для заполнения дополнительных полей после того, как образ сообщения сформирован при помощи функции msp_FormatMessage() (bc.h).

Кроме этого, функция должна установить поля present, transmit и check в соответствии с типом сообщения, фактически заданными полями и указанной ролью (ролями) ГСП в данном сообщении.

Параметры аналогичны параметрам функции msp_WG_FormatMessage().

Заголовок:

```
msp_Message* msp_WG_ExtendMessage(  
    msp_Message* buffer,  
    msp_WORD *data,  
    msp_WORD ST1,  
    msp_WORD ST2,  
    msp_BYTE wgmcs,  
    msp_BYTE checkalso);
```

Параметры: buffer - указатель на структуру msp_Message.
ST1 - значение ответного слова
ST2 - значение ответного слова 2 (только для сообщений RT→RT)
Wgmcs - комбинация следующих флажков:
msp_WGM_CHANNEL_B,
msp_ROLE_BC,
msp_ROLE_RT,
msp_ROLE_RT2.
checkalso - флаги для проверки дополнительных слов

Возврат: указатель на структуру msp_Message.
Ошибки: msp_ERROR_INVALID_MESSAGE_FORMAT

5.6. Целостное создание сообщений

5.6.1. Функция msp_WG_CreateMessage

Создает внутреннее представление сообщения по заготовке сообщения msp_Message

Заголовок:

```
msp_WG_MSGHANDLE msp_WG_CreateMessage(msp_DEVHANDLE device, msp_Message *  
buffer)
```

Параметры: device - хэндл на устройство.
buffer - указатель на структуру msp_Message.

Возврат: указатель на сообщение.
Ошибки: msp_ERROR_INVALID_DEVICE_HANDLE
msp_ERROR_NO_HOST_MEMORY

5.6.2. Функции msp_WG_SetTransmit и msp_WG_SetCheck

Явные изменения наборов передаваемых (msp_WG_SetTransmit) и сравниваемых слов (msp_WG_SetCheck).

Заголовки:

```
msp_ERROR msp_WG_SetTransmit(msp_WG_MSGHANDLE message,  
                             msp_BYTE set,  
                             msp_BYTE reset);  
msp_ERROR msp_WG_SetCheck(msp_WG_MSGHANDLE message,  
                           msp_BYTE set,  
                           msp_BYTE reset);
```

Параметры: message - сообщение.
set, reset - перечень полей сообщения, которые включаются (set) или исключаются (reset) из набора передаваемых (msp_WG_SetTransmit) и сравниваемых (msp_WG_SetCheck) слов. Могут содержать комбинации флажков msp_MSG_CMD1, msp_MSG_CMD2, msp_MSG_DATA, msp_MSG_STATUS1, msp_MSG_STATUS2.

Возврат: Код ошибки или 0.
Ошибки: msp_ERROR_INVALID_MESSAGE_HANDLE

5.6.3. Функция **msp_WG_DestroyMessage**

Функция уничтожает сообщение.

Заголовок:

```
msp_ERROR msp_WG_DestroyMessage(msp_WG_MSGHANDLE message);
```

Параметры: message - сообщение.

Возврат: Код ошибки или 0.

Ошибки: msp_ERROR_INVALID_MESSAGE_HANDLE

5.7. Пословное создание и модификация сообщений

5.7.1. Функция **msp_WG_CreateEmptyMessage**

Функция создает исходно пустое сообщение (число слов = 0).

Заголовок:

```
msp_WG_MSGHANDLE msp_WG_CreateEmptyMessage(msp_DEVHANDLE device);
```

Параметры: device - устройство.

Возврат: сообщение или 0 при ошибке.

Ошибки: msp_ERROR_INVALID_DEVICE_HANDLE

msp_ERROR_NO_HOST_MEMORY

5.7.2. Функции **msp_WG_InsertWord** и **msp_WG_RemoveWord**

Вставка (в позицию wordref), добавление (в конец сообщения) или исключени слова.

При добавлении передаваемого слова устанавливается нулевая задержка от предыдущего слова и не вносятся ошибки. Изменить эти значения можно при помощи msp_WG_DivertWord() (см. ниже).

Заголовки:

```
msp_ERROR msp_WG_InsertWord(msp_WG_MSGHANDLE message, msp_BYTE wordref,  
                             msp_BYTE status, msp_WORD data);
```

```
msp_ERROR msp_WG_RemoveWord(msp_WG_MSGHANDLE message, msp_BYTE wordref);
```

Параметры: message - редактируемое сообщение

wordref - ссылка на позицию слова, в которую производится вставка или из которой производится удаление; аналогично полю wordref структуры msp_Diversion.

status - признак, является ли слово генерируемым или эталонным, и как осуществлять проверку эталона; одна из констант msp_WG_TRANSMIT (0x01), msp_WG_CHECK_NONE (0x00) или msp_WG_ABSENT (0x00), msp_WG_CHECK_PRESENT (0x02), msp_CHECK_VALUE (0x03).

data - значение самого слова

Возврат: сообщение или 0 при ошибке.

Ошибки: msp_ERROR_INVALID_MESSAGE_HANDLE

msp_ERROR_INVALID_PARAMETER

5.8. Внесение ошибок в сообщение

5.8.1. Функция **msp_WG_DivertWord**

Внесение ошибки и/или временного промежутка в слово.

Все параметры интерпретируются так, как одноименные поля структуры msp_Diversion.

Следует учесть, что бит `msp_WGCW_COMMAND` в `flags` замещает признак команда/данные, установленный `msp_WG_InsertWord()` или `msp_WG_CreateMessage()`, это справедливо и для остальных функций из данного пункта.

Заголовок:

```
msp_WG_MSGHANDLE msp_WG_DivertWord(  
    msp_WG_MSGHANDLE message,  
    msp_BYTE wordref,  
    msp_WORD gap,  
    msp_BYTE flags,  
    msp_WORD error)
```

Параметры: `message` - редактируемое сообщение
`wordref` - указатель изменяемого слова
`gap` - значение задержки перед передачей слова
`flags` - флаги BCCW
`error` - слово описания ошибки

Возврат: параметр `message`
Ошибки: `msp_ERROR_INVALID_MESSAGE_HANDLE`
`msp_ERROR_INVALID_PARAMETER`

5.8.2. Функция `msp_WG_DivertWordIndirect`

Внесение ошибки и/или временного промежутка в слово.
Все параметры интерпретируются так, как одноименные поля структуры `msp_Diversion`.

Заголовок:

```
msp_WG_MSGHANDLE msp_WG_DivertWordIndirect(  
    msp_WG_MSGHANDLE message,  
    msp_Diversion *diversion)
```

Параметры: `message` - редактируемое сообщение
`diversion` - структура `msp_Diversion`, задающая вносимую ошибку.

Возврат: параметр `message`
Ошибки: `msp_ERROR_INVALID_MESSAGE_HANDLE`
`msp_ERROR_INVALID_PARAMETER`

5.8.3. Функция `msp_WG_DivertMessageIndirect`

Внесение ошибок и/или временные промежутки в сообщение.
Все параметры интерпретируются так, как одноименные поля структуры `msp_Diversion`.

Заголовок:

```
msp_WG_MSGHANDLE msp_WG_DivertMessageIndirect(  
    msp_WG_MSGHANDLE message,  
    msp_Diversion *diversions)
```

Параметры: `message` - редактируемое сообщение
`diversion` - массив структур `msp_Diversion`, задающих вносимые ошибки. Последняя структура в массиве должна иметь значение `msp_MW_NO_MORE` в поле `wordref`
Возврат: параметр `message`
Ошибки: `msp_ERROR_INVALID_MESSAGE_HANDLE`
`msp_ERROR_INVALID_PARAMETER`

5.9. Создание специальных сообщений.

5.9.1. Функция `msp_WG_CreateTerminalMessage`

Создание специального сообщения. Тип сообщения определяется параметром `code`.

Заголовок:

```
msp_WG_MSGHANDLE msp_WG_CreateTerminalMessage(msp_DEVHANDLE device, int code, msp_WORD time)
```

Параметры: `device` - устройство
`code` - код специального сообщения; одна из констант:
`msp_WG_TERM_WM` (0x0001) - переход (с возвратом) в режим WM (словного монитора);
`msp_WG_TERM_LOOP` (0x0020) - циклический переход к началу кадра;
`msp_WG_TERM_STOP` (0xFFC0) - остановка ГСП;
`msp_WG_TERM_PAUSE` (0x0003) - пауза в работе ГСП, на время заданное в `time`;
`msp_WG_TERM_EXIT_WM` (0x0007) - переход из режима WM в режим ГСП, по истечении времени задаваемого в `time`;
`time` - время, мзр = 0,25 мкс

Возврат: параметр `message`
Ошибки: `msp_ERROR_INVALID_DEVICE_HANDLE`
`msp_ERROR_INVALID_PARAMETER`

5.10. Добавление сообщения в кадр

5.10.1. Функция `msp_WG_AddMessage`

Добавление одиночного сообщения в кадр. В отличие от режима КИШ, каждое сообщение может быть лишь один раз добавлено в кадр.

Заголовок:

```
msp_ERROR msp_WG_AddMessage(msp_WG_FRMHANDLE frame, msp_WG_MSGHANDLE message);
```

Параметры: `frame` - кадр
`message` - сообщение

Возврат: Код ошибки или 0.
Ошибки: `msp_ERROR_INVALID_FRAME_HANDLE`
`msp_ERROR_INVALID_MESSAGE_HANDLE`
`msp_ERROR_FRAME_EXCEEDS_SIZE_LIMIT`

5.10.2. Функция `msp_WG_AddMessageIndirect`

Добавление одиночного сообщения в кадр.

Заголовок:

```
msp_ERROR msp_WG_AddMessageIndirect(msp_WG_FRMHANDLE frame, msp_Message *message, msp_Diversion *diversions);
```

Параметры: `frame` - кадр
`message` - сообщение
`diversions` - ошибки, вносимые в сообщение (массив структур `msp_Diversion`);

Возврат: Код ошибки или 0.
Ошибки: msp_ERROR_INVALID_FRAME_HANDLE
msp_ERROR_INVALID_MESSAGE_HANDLE
msp_ERROR_FRAME_EXCEEDS_SIZE_LIMIT

5.10.3. Функция **msp_WG_AddMessagesIndirect**

Добавление множества сообщений в кадр.

Заголовок:

```
msp_ERROR msp_WG_AddMessagesIndirect(msp_WG_FRMHANDLE frame, int count,  
msp_Message *messages, msp_Diversion *diversions);
```

Параметры: frame – кадр
count - количество добавляемых сообщений
message – сообщения
d iversions - ошибки, вносимые в сообщения (массив структур msp_Diversion);

Возврат: Код ошибки или 0.
Ошибки: msp_ERROR_INVALID_FRAME_HANDLE
msp_ERROR_INVALID_MESSAGE_HANDLE
msp_ERROR_FRAME_EXCEEDS_SIZE_LIMIT

5.10.4. Функция **msp_WG_GetFrameMessage**

Возвращает хендл сообщения по его номер в кадре

Заголовок:

```
msp_WG_MSGHANDLE msp_WG_GetFrameMessage(msp_WG_FRMHANDLE frame, int entry);
```

Параметры: frame – кадр
entry - номер позиции в кадре

Возврат: Код ошибки или 0.
Ошибки: msp_ERROR_INVALID_FRAME_HANDLE
msp_ERROR_INVALID_PARAMETER

5.11. Проверка ответов тестируемого устройства

Функции сравнения возвращают следующие значения:

msp_NOERROR - при успешном сравнении
msp_ERROR_ANSWER_NOT_MATCH - при неуспешном сравнении, а также при ошибке передачи ответа,
msp_ERROR_ANSWER_EXTRA_WORDS - при наличии в стеке монитора лишних слов,
msp_ERROR_ANSWER_MISSING_WORDS – при отсутствии слов в стеке монитора, предусмотренных эталоном,
msp_ERROR_INVALID_MESSAGE – при ошибке формирования сообщения,

Функции сравнивают с образцом (сгенерированным msp_WG_FormatMessage() и msp_WG_CreateMessage()) - ответ тестируемого устройства на сообщение. При этом учитываются признаки типа сравнения.

В структуру details записываются результаты сравнения, которые могут принимать следующие значения:

msp_WG_CHK_NONE – сравнение не производится

msp_WG_CHK_OK (0x00) – сравнение успешно

msp_WG_CHK_ERROR - ошибка при передаче/приеме слова
msp_WG_CHK_MISSING - данное слово отсутствует в ответе
msp_WG_CHK_ERROR_GAP – истечение времени ожидания отклика по MIL-STD-1553B
msp_WG_CHK_NOT_MATCH - данное слово присутствует в ответе, но не совпадает с ожидаемым значением.

Поля details, соответствующие полям сообщения, не подлежащим сравнению, заполняются значением msp_WG_CHK_NONE.

5.11.1. Функция **msp_WG_CheckMessage**

Проверка ответа на сообщение (путем сопоставление с образцом).

Заголовок:

```
msp_ERROR msp_WG_CheckMessage(msp_WG_FRMHANDLE frame, int entry, msp_Message *details);
```

Параметры: frame – кадр
entry - номер позиции в кадре
details - детальное описание ошибок сравнения (см. выше)

Возврат: Код ошибки или 0.

Ошибки: msp_ERROR_INVALID_FRAME_HANDLE
msp_ERROR_INVALID_PARAMETER

5.11.2. Функция **msp_WG_CheckMessages**

Проверка всех сообщений кадра

Заголовок:

```
msp_ERROR msp_WG_CheckMessages(msp_WG_FRMHANDLE frame, msp_Message *details);
```

Параметры: frame – кадр
details - детальное описание ошибок сравнения, массив структур msp_Message, размер которого должен быть не меньше числа сообщений в кадре. Элементы массива details заполняются также, как параметр details функции msp_WG_CheckMessage().

Возврат: msp_ERROR_ANSWER_NOT_MATCH - при ошибке сравнения хотя бы одного сообщения кадра; в остальных случаях - как msp_WG_CheckMessage().

Ошибки: msp_ERROR_INVALID_FRAME_HANDLE
msp_ERROR_INVALID_PARAMETER

5.11.3. Функция **msp_WG_GetGap**

Считывание значения межсловного временного промежутка перед передачей указанного слова.

Заголовок:

```
msp_WORD msp_WG_GetGap(msp_WG_FRMHANDLE frame, int entry, msp_BYTE wordref);
```

Параметры: frame – кадр; должен быть загружен
entry - номер позиции сообщения в кадре
wordref - номер слова в сообщении, интерпретируется как поле wordref в структуре msp_Diversion.

Возврат: Возвращает значение межсловного временного промежутка перед передачей данного слова. Значение приведено к разрядности промежутка словного монитора мзр– 0.5 мкс.

Ошибки: msp_ERROR_INVALID_FRAME_HANDLE
msp_ERROR_INVALID_PARAMETER

ПРИЛОЖЕНИЕ А

(обязательное)

ПРИМЕР ФОРМИРОВАНИЯ СООБЩЕНИЯ С ПОМОЩЬЮ RTL2GSW

Создание сообщения реакции тестера на команду передачи данных между ОУ и ОУ, и внесение ошибки в ответное слово по каналу В(П).

```
//.....  
// Получаем хэндл у-ва  
msp_DEVHANDLE hdev = msp_Open64(i); // где i – номер ячейки  
// Конфигурируем устройство  
gsw_Config(hdev, msp_WG_FIRST);  
//.....  
  
msp_WG_FRMHANDLE frame;  
msp_WG_MSGHANDLE msg;  
msp_Message* buf = new msp_Message;  
  
// Создаем кадр и сообщение  
frame = msp_WGCreateFrame(hdev);  
if(!frame) /*Реакция на ошибку*/;  
buf = msp_WG_FormatMessage(buf, mspM_RTtoRT, adr,1,adr+1,1,32,data, adr<<11, (adr+1)<<11, \  
    msp_WGM_CHANNEL_B | msp_ROLE_RT | msp_ROLE_RT2, 0);  
if(msp_GetLastError()) /*Реакция на ошибку*/;  
msg = msp_WG_CreateMessage(hdev, buf);  
if(msp_GetLastError()) /*Реакция на ошибку*/;  
  
// Устанавливаем временной интервал перед выдачей ответного слова принимающего ОУ  
// Макрос MKS_4 должен быть определен  
msg = msp_WG_DivertWord(msg, msp_MW_STATUS2, MKS_4, msp_WGCW_SET, 0);  
if(msp_GetLastError()) /*Реакция на ошибку*/;  
  
// Вносим ошибку  
msp_WORD error = FormatErrorWord(msp_INJ_ERROR_INVERT_RIGHT, 38, 24);  
msg = msp_WG_DivertWord(msg, msp_MW_STATUS1, 0, msp_WGCW_SET |  
    msp_WGCW_ERROR, error);  
if(msp_GetLastError()) /*Реакция на ошибку*/;  
  
// Добавляем сообщение в кадр  
er = msp_WG_AddMessage(frame, msg);  
if(er) /*Реакция на ошибку*/;  
  
// Создаем сообщение останова тестера  
msg = msp_WG_CreateTerminalMessage(hdev, msp_WG_TERM_STOP,0);  
if(msp_GetLastError()) /*Реакция на ошибку*/;  
if(msp_WG_AddMessage(frame, msg)) /*Реакция на ошибку*/;  
  
// Загружаем кадр в устройство  
msp_WGLoadFrame(frame);  
if(msp_GetLastError()) /*Реакция на ошибку*/;  
  
// Запускаем тестер в работу  
gsw_Start(hdev); // Недокументированная ф-ция. Можно использовать ее эквивалент, описанный в RTL2.  
  
// Удаляем кадр и сообщение  
msp_WGDestroyFrame(frame);  
msp_WG_DestroyMessage(message);  
  
// .....
```

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

ГСП –	генератор словных последовательностей.
КШ –	режим контроллера шины.
МСП–	машина связная последовательная (т.е. по последовательному каналу).
МТ –	режим монитора шины.
ОЗУ –	оперативное запоминающее устройство (оперативная память).
ОУ –	режим оконечного устройства шины.
П/П –	подпрограмма.
ПЭВМ–	персональная электронно-вычислительная машина.
ТСТ –	режим тестера шины.
УВ –	управляющий вычислитель.
РСІ –	шина РСІ.

ПЕРЕЧЕНЬ ССЫЛОЧНЫХ ДОКУМЕНТОВ

1. ЮФКВ.50001-01 13 01. Ячейка МВ11.01. Библиотека программ интерфейса ячейки МВ11.01 и прикладных программ. Описание программы
2. ГОСТ 26765.52-87 ИНТЕРФЕЙС МАГИСТРАЛЬНЫЙ ПОСЛЕДОВАТЕЛЬНЫЙ СИСТЕМЫ ЭЛЕКТРОННЫХ МОДУЛЕЙ”
3. ЮФКВ.469555.075РЭ. Ячейка МВ11.01. Руководство по эксплуатации
4. ЮФКВ.00016-01 31 01. Ячейка МВ11.01. Программное обеспечение. Описание применения
5. ЮФКВ.469555.075РЭ. Ячейка МВ11.04. Руководство по эксплуатации

