

МОДУЛИ «ОСПЧ»
БИБЛИОТЕКА ДОСТУПА РАЗДЕЛЯЕМОЙ ПАМЯТИ

Руководство программиста

Листов 6

2013-2020

СОДЕРЖАНИЕ

1. Описание функций	3
1.1. Подключение канала.....	3
1.2. Отключение канала	3
1.3. Запись данных в канал	4
1.4. Чтение канала данных.....	4
2. Описание констант	5
2.1. Режим подключения (SHMEM_MODE)	5
2.2. Возвращаемые значения (SHMEM_ERROR).....	5
3. Рекомендуемая последовательность работы.....	6

Библиотека `libshmem` предназначена для обмена данными между процессами операционной системы. Для реализации функционала библиотеки использован механизм разделяемой памяти. Память библиотеки организована как 256 независимых каналов, каждый из которых имеет 8 буферов. Запись и чтение осуществляется по принципу FIFO. Если запись в какой-либо канал была произведена более 8 раз, при этом данные ни разу не вывалились, то при следующей записи данные, которые были записаны первыми, будут перезаписаны. Размер каждого буфера канала задается при подключении к памяти. Чтение и запись канала в дальнейшем ведется заданным размером.

Все функции библиотеки при успешном выполнении возвращают ноль, в противном случае – код ошибки. Коды ошибок определены в перечислении `SHMEM_ERROR`.

1. ОПИСАНИЕ ФУНКЦИЙ

1.1. Подключение канала

Подключение к каналу общей памяти и его инициализация выполняется функцией `Connect2SharedMem`. При инициализации производится настройка буферов памяти согласно параметру `DataSize`. Если канал был инициализирован другим процессом, размер буфера (параметр `DataSize`) должен быть таким же.

<pre>int Connect2SharedMem (SHMEM_MODE Action, unsigned char Channel, unsigned int DataSize, void** phMem);</pre>		
Описание	Производит подключение к каналу разделяемой памяти	
Входные параметры	Action	режим подключения к каналу памяти
	Channel	номер канала разделяемой памяти (0..255)
	DataSize	размер буфера (2^n , $n = 10..18$)
Выходные параметры	phMem	идентификатор канала

1.2. Отключение канала

По окончании работы с каналом разделяемой памяти необходимо выполнить отключение от него. Для этого предназначена функция `DisconnectSharedMem`.

<pre>int DisconnectSharedMem (SHMEM_MODE Action, unsigned char Channel);</pre>		
Описание	Отключение от канала разделяемой памяти	
Входные параметры	Action	режим подключения к каналу памяти
	Channel	номер канала разделяемой памяти (0..255)

1.3. Запись данных в канал

Для записи в канал разделяемой памяти предназначена функция `WriteToMem`. В функцию передается идентификатор канала `hMem`, полученный при подключении, и указатель на область памяти, содержащую данные для записи в канал. Объем данных должен быть не меньше, чем размер буфера, указанный при подключении.

```
int WriteToMem (  
    void* hMem,  
    void* Ptr  
);
```

Описание	Запись в канал разделяемой памяти	
Входные параметры	<code>hMem</code>	идентификатор канала памяти
	<code>Ptr</code>	указатель на область памяти данных для записи

1.4. Чтение канала данных

Для получения данных из канала разделяемой памяти предназначена функция `ReadFromMem`. В функцию передается идентификатор канала `hMem`, полученный при подключении, и указатель на область памяти, выделенную пользователем, в которую будет произведено считывание данных. Размер массива должен быть не меньше, чем размер буфера, указанный при подключении. В случае отсутствия данных, функция зависает в ожидании, поэтому вызов данной функции желательно производить в отдельном потоке. При отключении от канала памяти, даже при отсутствии данных, будет выполнен возврат из функции.

```
int ReadFromMem (  
    void* hMem,  
    void* Ptr  
);
```

Описание	Чтение из канала разделяемой памяти	
Входные параметры	<code>hMem</code>	идентификатор канала памяти
	<code>Ptr</code>	указатель на область памяти для чтения данных

2. ОПИСАНИЕ КОНСТАНТ

2.1. Режим подключения (SHMEM_MODE)

SHMEM_MODE		
MODE_WRITE	0	Запись в разделяемую память
MODE_READ	1	Чтение из разделяемой памяти

2.2. Возвращаемые значения (SHMEM_ERROR)

SHMEM_ERROR		
ERR_NO_ERROR	0	Выполнено без ошибок
ERR_INVALID_ARGUMENT	1002	Переданы некорректные параметры
ERR_CHANNEL_BUSY	1003	Канал данных занят
ERR_NO_DATA	1005	Канал не содержит данных, считывание не выполнено
ERR_CHANNEL_NOT_INIT	1006	Канал не инициализирован
ERR_BAD_BUFFER_SIZE	1007	Некорректный размер буфера. Уже выполнено подключение к данному каналу с указанием другого размера буфера

3. РЕКОМЕНДУЕМАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ РАБОТЫ

- 1) Подключиться к каналу общей памяти: *Connect2SharedMem*;
- 2) Если процесс является источником данных, записывать данные в память: *WriteToMem*;
- 3) Если процесс является получателем данных, читать данные из памяти: *ReadFromMem*.
- 4) По окончании работы отключиться от канала общей памяти: *DisconnectSharedMem*;

Примечание 1: Функцию *ReadFromMem* необходимо вызывать в отдельном потоке, так как она содержит вызов системной функции с продолжительным временем ожидания. Вызов функции *ReadFromMem* в основном потоке и отсутствие данных со стороны источника может привести к «замораживанию» приложения-приемника данных.

Примечание 2: Отключение от канала общей памяти источника или получателя данных может производиться только процессом, выполнявшим подключение к данному каналу в качестве источника или получателя соответственно.

Примечание 3: В случае возникновения системных ошибок для переинициализации общей памяти необходимо завершить все процессы, выполнявшие операции с функциями библиотеки *libshmem*.