

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Хозрасчетная кафедра молекулярной физики

УТВЕРЖДАЮ
Директор ООО ХКМФ
_____ Породнов Б.Т.
__ . __ . 2001

ОТЧЕТ ПО НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ
***РАЗРАБОТКА МЕТОДИКИ ОПТИМИЗАЦИИ МИКРОАНАЛИЗА И
СОЗДАНИЕ ПРОГРАММНО-ИНФОРМАЦИОННОГО ОБЕСПЕЧЕНИЯ
ДЛЯ РЕЖИМА СЧЕТА ИОНОВ***

(заключительный)

Х.Д. № 42/16/1778

Руководитель НИР к. ф.-м. н., зав. лабораторией	_____ __ . __ . 2001	Атанов В.Е.
Ответственный исполнитель к. ф.-м. н., доцент	_____ __ . __ . 2001	Калинин Б.А.
Исполнитель к. ф.-м. н., доцент	_____ __ . __ . 2001	Александров О.Е.

Екатеринбург 2001

РЕФЕРАТ

Отчет: 53 с., рис. 18, табл. 10, 6 источников, 0 прил.

МАСС-СПЕКТРОМЕТРИЯ, ГЕКСАФТОРИД УРАНА, РАЗРЕШАЮЩАЯ СПОСОБНОСТЬ, МИКРОАНАЛИЗ, ФОНОВЫЙ ИОННЫЙ ИОК, ВТОРИЧНО-ЭЛЕКТРОННЫЙ УМНОЖИТЕЛЬ, АВТОМАТИЗАЦИЯ ИЗМЕРЕНИЙ, ПРОГРАММНЫЙ МОДУЛЬ УПРАВЛЕНИЯ

Описаны принципы функционирования драйвера режима ядра ОС Windows NT 4.0, предназначенного для обеспечения доступа к портам ввода-вывода МИ 1201-АГМ через файловую систему ОС. Описан драйвер виртуальной машины DOS, предназначенный для обеспечения функционирования программы управления масс-спектрометром МИ 1201-АГМ, разработанной для DOS.

Приведена математическая модель для обработки масс-спектра (пиков) с целью повышения разрешающей способности и чувствительности измерений при уровнях сигнала, сравнимых с шумом.

Приведен пример синтеза модельных данных для масс-спектра и рассмотрены на их примере проблемы, возникающие при анализе масс-спектров.

Проведен анализ одного из важных механизмов образования фонового ионного тока (ФИТ) — распад метастабильных ионов в поле электромагнита. Экспериментально показано существование метастабильных ионов и их существенный вклад в ФИТ. Проведен эксперимент по уменьшению ФИТ методом задерживающего потенциала и продемонстрирована возможность снижения ФИТ при уменьшении напряжения ионизации.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1. СОЗДАНИЕ И ТЕСТИРОВАНИЕ ДРАЙВЕРОВ УПРАВЛЕНИЯ МАСС-СПЕКТРОМЕТРОМ МИ-1201 АГМ В СРЕДЕ WINDOWS NT	6
1.1. Общие принципы работы Windows NT с внешними устройствами	6
1.2. Соображения о месте и функциях драйвера в программном обеспечении	7
1.3. Общая структура драйвера ядра	8
1.4. Процесс инициализации	8
1.5. Обработка запросов	9
1.6. Процесс выгрузки драйвера “PortIO.sys”	10
1.7. Работа с драйвером “PortIO.sys” через вызовы функций ОС	10
1.8. Взаимодействие модуля управления МИ-1201 АГМ с драйвером	13
1.9. Установка драйвера “PortIO.sys”	13
1.10. Принципы работы виртуального драйвера DOS	14
1.11. Общая структура виртуального драйвера DOS	15
1.12. Загрузка “MI1201.DLL”	15
1.13. Обработка запросов к портам ввода-вывода	16
1.14. Процесс выгрузки драйвера	16
1.15. Установка драйвера	16
2. СОЗДАНИЕ МАТЕМАТИЧЕСКОЙ МОДЕЛИ МАСС-СПЕКТРА ПРИ ИЗМЕРЕНИЯХ В РЕЖИМЕ СЧЕТА ИОНОВ С УЧЕТОМ ШУМОВ И НАЛОЖЕНИЯ ПИКОВ	17
2.1. Измерение интенсивности сигнала	17
2.2. Исходные данные и математическая формулировка задачи	17
2.3. Упрощенный алгоритм поиска центра для симметричного пика	18
2.4. Поиск центра пика на основе аппроксимации экспериментальных точек функцией	18
2.5. Эквивалентность усреднения в пространстве и времени	19
2.6. Измерение при наложении пиков	20
2.7. Аппаратная функция	21
2.8. Аппроксимирующая функция для пика	21
2.9. Восстановление функции по данным измерения	23
2.10. О программе для численного решения уравнения	26
2.11. Основная процедура модуля <i>uGenFit</i> и ее аргументы	27
2.12. Вспомогательные процедуры модуля <i>uGenFit</i>	28
2.13. Процедуры подготовки данных	29
2.14. Задание аппроксимирующих функций	30

	4
2.15. Порядок работы с процедурой <i>GeneralFit</i>	30
3. МОДЕЛИРОВАНИЕ И МАТЕМАТИЧЕСКАЯ ОБРАБОТКА МАСС-СПЕКТРОВ	31
3.1. Оптимизация сбора и обработки данных масс-спектрометрических измерений.....	31
3.2. Графическое представление и эмуляция экспериментальных данных.....	34
3.3. Определение параметров пиков масс-спектра	38
3.4. Калибровка шкалы масс	40
4. АНАЛИЗ ПРИЧИН ВОЗНИКНОВЕНИЯ ФОНОВОГО ИОННОГО ТОКА	42
4.1. Измерения с задерживающим потенциалом	42
4.2. Метастабильные ионы.....	45
4.3. Пути уменьшения фонового ионного тока	48
ВЫВОДЫ	52
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	52

ВВЕДЕНИЕ

Повышение точности и чувствительности масс-спектрометрического анализа требует разработки соответствующих программных средств и методик анализа.

Для удобной и эффективной работы по созданию новых программных средств предпочтительнее использование операционной системы Windows NT. В этой ОС имеются развитые средства разработки программ, но для управления внешним устройством (например, масс-спектрометром МИ 1201-АГМ) требуется драйвер. Разработка такого драйвера для МИ 1201-АГМ позволит: 1) использовать готовые решения при создании ПО обработки масс-спектров, 2) запускать существующие программы для DOS, т.е. сделать переход на новую ОС менее болезненным, 3) перейти (в перспективе) к более защищенной и устойчивой среде, чем DOS и 4) получить опыт в разработке драйверов, который будет полезен при создании нового масс-спектрометра.

Для повышения чувствительности масс-спектрометрического анализа в режиме счета ионов необходима прежде всего разработка математической модели этого режима измерения с учетом шумов, формы пиков и их наложения. Создание такой модели масс-спектра даст основу для более осмысленного анализа и расшифровки спектра, а в дальнейшем приведет к созданию алгоритма автоматизации обработки данных измерения в режиме счета ионов.

Важным моментом для повышения чувствительности анализа в режиме счета является уменьшение фонового ионного тока (ФИТ). Для этого необходимо выяснить происхождение ФИТ и на этой основе выбрать методику измерений, устраняющую (снижающую) возникновение фона.

Одним из возможных подходов может служить анализ ионов по энергиям и отсеечение низкоэнергетических импульсов при счете. Разработка таких методик требует модификации тракта счета ионов МИ 1201-АГМ — дополнения его возможностью анализа ионов по энергиям, измерения амплитуды и формы импульсов, возбуждаемых ионами на вторичном электронном умножителе (ВЭУ).

Использование указанных технологий позволит повысить точность анализа микропримесей в гексафториде урана и разработать методику определения микропримесей при прямом (без концентрирования) анализе.

1. СОЗДАНИЕ И ТЕСТИРОВАНИЕ ДРАЙВЕРОВ УПРАВЛЕНИЯ МАСС-СПЕКТРОМЕТРОМ МИ-1201 АГМ В СРЕДЕ WINDOWS NT

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 1. ОБЩИЕ ПРИНЦИПЫ РАБОТЫ WINDOWS NT С ВНЕШНИМИ УСТРОЙСТВАМИ

Для управления внешними устройствами в архитектуре Intel-совместимых процессоров¹⁾ предусмотрены специальные программные регистры — порты. Программы могут записывать данные в порты и считывать данные из портов, используя инструкции процессора “in регистр_процессора, номер_порта” и “out номер_порта, регистр_процессора”²⁾. Внешнее устройство, посредством специальной интерфейсной платы (обычно подключаемой к шине ISA или PCI), может получать ввод портов процессора, производить какие-то операции в соответствии с полученными данными и возвращать свои данные через порты процессору. Управление через порты используется и для большинства собственных “внешних” (по отношению к процессору) устройств компьютера.

Поскольку неправильное взаимодействие программы с портами может привести компьютер в неработоспособное состояние, то операционные системы, стремящиеся к повышенной отказоустойчивости, блокируют доступ к портам со стороны прикладных программ. Именно таким образом поступает ОС Windows NT.

Доступ к портам в таком случае имеют только специальные программы — драйверы. Драйверы, по замыслу разработчиков ОС, должны обеспечивать управление внешними устройствами и препятствовать неверному использованию портов. Дополнительно (в некоторых ОС) драйверы могут использоваться для авторизации доступа к оборудованию и исключения множественного доступа в многозадачных ОС.

Наряду с указанными достоинствами, необходимость написания драйверов требует дополнительных усилий со стороны разработчиков ПО. Так драйвер должен иметь повышенную надежность, т.е. разработка и тестирование этого типа ПО требует относительно больших усилий.

В самом общем и упрощенном виде схема взаимодействия прикладных программ Windows NT с портами ввода-вывода приведена на рис. 1.1.

¹⁾ Подобную организацию управления внешними устройствами через порты применяют и в других архитектурах микропроцессоров.

²⁾ Например: *in AL,0123h* или *out 0123h, BH*. Есть и еще несколько инструкций работы с портами.

Для управления масс-спектрометром МИ1201-АГМ создан драйвер ядра “PortIO.sys”.

ВЗАИМОДЕЙСТВИЕ ПРИКЛАДНЫХ ПРОГРАММ WINDOWS NT С ПОРТАМИ ВВОДА-ВЫВОДА

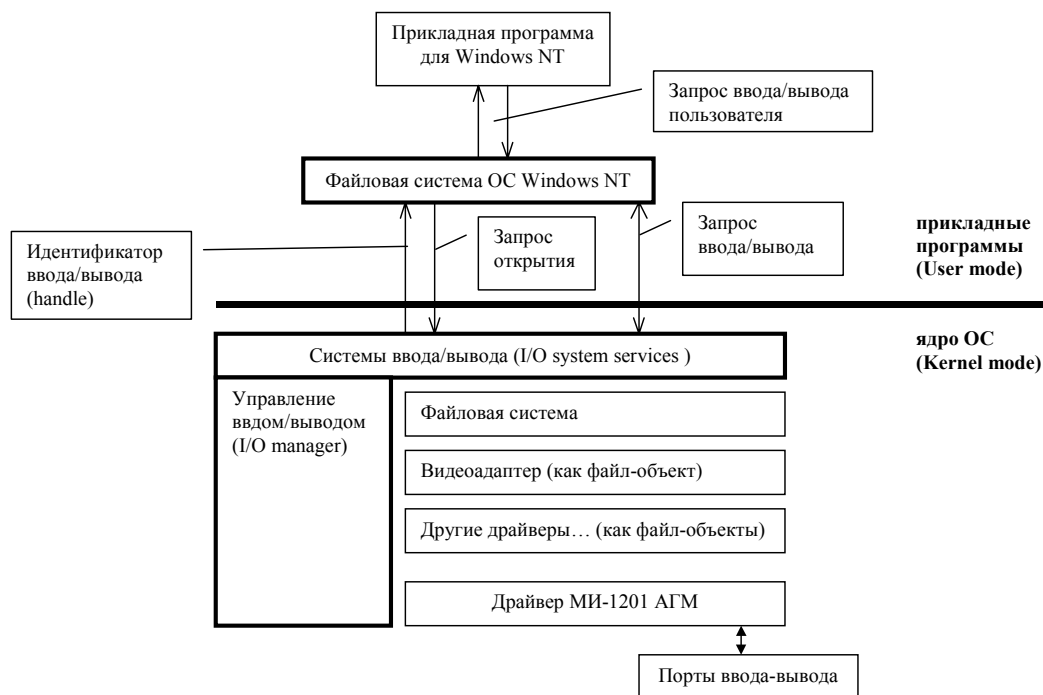


Рис. 1

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 2. СООБРАЖЕНИЯ О МЕСТЕ И ФУНКЦИЯХ ДРАЙВЕРА В ПРОГРАММНОМ ОБЕСПЕЧЕНИИ

Объем функций управления, которые можно разместить в драйвере неограничен, но для нашего случая этот объем разумно минимизировать. Аргументы следующие:

- 1) Разработка и тестирование драйвера очень трудоемкий процесс.
- 2) Драйвер критичен к ошибкам, ошибки в драйвере могут привести к неустойчивой работе ОС.
- 3) Программное обеспечение должно выполняться под управлением как Windows NT, так и Windows 9x/Windows ME. В операционных системах Windows 9x/ Windows ME нет жесткой необходимости в драйвере (можно непосредственно работать с портами).

Следовательно, функции драйвера должны быть ограничены реализацией аналогов инструкций In/Out процессора. В этом случае, все более сложные функции управления реализуются в программе.

При работе под различными ОС программа должна опознать тип ОС и, соответственно, выбрать путь ввода/вывода в порты. Все остальные функции программы будут работать без изменений.

Единственное соображение в пользу усложнения функций драйвера: каждое обращение к портам при работе в Window NT требует больших накладных расходов — запрос ввода/вывода обрабатывается операционной системой.

Для сравнения, при прямом доступе в порты (Windows 9x) на операцию ввода вывода затрачивается несколько тактов процессора, даже с учетом некоторых дополнительных операций программы, она может осуществлять более 10000 запросов ввода/вывода в миллисекунду (на процессоре с частотой 400 МГц). Для Window NT аналогичный показатель около 100 запросов ввода/вывода в миллисекунду.

Хотя для задачи управления МИ-1201 АГМ этого вполне достаточно — необходимое характерное время реакции программы порядка 1 мс, но интенсивная работа системы ввода/вывода ОС создает лишнюю нагрузку на процессор и может мешать выполнению других программ. Из этих соображений возможно было бы разумным некоторое усложнение функций драйвера, например в реализации “пакетного” ввода/вывода в несколько портов за один вызов.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 3. ОБЩАЯ СТРУКТУРА ДРАЙВЕРА ЯДРА

Драйвер ядра (далее KMD) “PortIO.sys” — программа на языке Си. Исходный текст программы находится см. в папке “NT-драйверы для МИ 1201-АГМ\Исходный код драйверов\Драйвер ядра NT”.

Драйвер является переработанной версией стандартного драйвера ввода-вывода в порты, поставляемого Microsoft Corp. в составе “Microsoft® Windows NT® Version 4.0 Device Driver Kit”.

В результате модификации драйвер поддерживает несколько диапазонов портов (до 10).

Перечень функций “PortIO.sys” приведен в табл. 1.1.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 4. ПРОЦЕСС ИНИЦИАЛИЗАЦИИ

- 1) ОС вызывает функцию драйвера DriverEntry() (эта функция присутствует в любом KMD).
- 2) DriverEntry() считывает из реестра ключ
`NKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\PortIO\Parameters`

Таблица 1

ФУНКЦИИ “PORTIO.SYS”

Функция	Назначение
1. DriverEntry()	Инициализация, чтение параметров из реестра, создание устройства вызовом GpdCreateDevice()
2. GpdCreateDevice()	Создание устройства
3. GpdDispatch()	Обработка запросов (чтение, запись, выгрузка) вызовом соответствующих функций (GpdioctlReadPort(), GpdioctlWritePort(), GpdUnload())
4. GpdioctlReadPort()	Чтение данных из порта
5. GpdioctlWritePort()	Запись данных в порт
6. GpdUnload()	Выгрузка

где хранятся параметры устройства. В данном случае, это диапазоны используемых портов ввода-вывода. Задание диапазонов портов осуществляется с помощью значений вида “IoPortAddressN” и “IoPortCountN”, где N может принимать значения от 0 до 9.

- 3) DriverEntry() сообщает ОС о предполагаемом использовании портов с целью исключить возможные конфликты между существующими драйверами.
- 4) В случае отсутствия конфликтов DriverEntry() создает устройство и передает ОС адрес функции-диспетчера запросов (GpdDispatch()).

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 5. ОБРАБОТКА ЗАПРОСОВ

KMD обрабатывает три типа запросов. Запросы перечислены в табл. 1.2. Запрос поступает в функцию GpdDispatch() и обрабатывается ею. Идентификатор запроса является просто числовой константой, ее значение можно найти в файлах-заголовках поставляемых в составе “Microsoft® Windows NT® Version 4.0 Device Driver Kit”.

Таблица 2

ЗАПРОСЫ НА ВВОД-ВЫВОД ДЛЯ “PORTIO.SYS”

Запрос	Условие вызова	Реакция
1. IRP MJ_CREATE	Вызывается ОС во	Сообщение ОС об успешном

Запрос	Условие вызова	Реакция
	время инициализации драйвера	запуске драйвера
2. IRP_MJ_CLOSE	Вызывается ОС во время выгрузки драйвера	Сообщение ОС об успешной выгрузке драйвера
3. IRP_MJ_DEVICE_CONTROL	Вызывается при использовании драйвера для чтения/записи	Вызов GpdiocctlReadPort()/GpdiocctlWritePort()

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 6. ПРОЦЕСС ВЫГРУЗКИ ДРАЙВЕРА “PORTIO.SYS”

Для выгрузки ОС вызывает функцию GpdUnload(), которая выполняет следующие действия.

- 1) Сообщает ОС о том, что драйвер более не использует порты.
- 2) Разрушает устройство.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 7. РАБОТА С ДРАЙВЕРОМ “PORTIO.SYS” ЧЕРЕЗ ВЫЗОВЫ ФУНКЦИЙ ОС

Взаимодействие прикладной программы с “PortIO.sys” осуществляется через вызовы системных функций создания, уничтожения и передачи данных, приведенные в табл. 1.3.

Таблица 3

СИСТЕМНЫЕ ФУНКЦИИ ДЛЯ РАБОТЫ С KMD

Функция	Назначение
1. CreateFile()	Создает файловое устройство, представляющее диапазон портов.
2. CloseHandle()	Уничтожает дескриптор, созданный предыдущей функцией.
3. DeviceIOControl()	Работа с файловым устройством - чтение/запись.

Пропущен номер предыдущего уровня: 1.1. Пункт! Вставьте его и нажмите клавиши: [Ctrl]+A, [F9]. 1. CreateFile()

Для работы с портами необходимо создать файловое устройство посредством вызова CreateFile(). В качестве параметров передается имя файлового устройства, созданного "PortIO.sys", режим доступа (запись/чтение), режим совместного доступа, параметры защищенности, разрешение на создание файла, атрибуты и шаблон. Большинство этих параметров для работы несущественны, типичные значения параметров приведены в табл. 1.4. Пример подключения к драйверу:

```
HDevice := CreateFile(
    '\\.\PortIODev',    // имя драйвера
    GENERIC_READ or GENERIC_WRITE, // тип доступа
    FILE_SHARE_READ,  // разрешение совместного доступа
    nil,               // адрес описания прав доступа
    OPEN_EXISTING,    // как создавать файл
    0,                 // атрибуты файла
    0                  // идентификатор файла для копирования атрибутов файла
);
```

В переменной *HDevice* будет возвращен идентификатор файла (handle) для доступа к нему.

В прикладной программе рекомендуется один раз создать файловое устройство и затем использовать его по мере необходимости.

Пропущен номер предыдущего уровня: 1.1. Пункт! Вставьте его и нажмите клавиши: [Ctrl]+A, [F9]. 2. CloseHandle()

Для завершения работы с драйвером необходимо уничтожить файловое устройство посредством вызова CloseHandle(). В качестве параметра передается дескриптор файлового устройства, созданного ранее вызовом CreateFile(). Например:

Таблица 4

ПАРАМЕТРЫ, ПЕРЕДАВАЕМЫЕ ФУНКЦИИ CREATEFILE()³

Тип	Значение
1. строка	\\.\PortIODev
2. двойное слово	GENERIC_READ or GENERIC_WRITE
3. двойное слово	FILE_SHARE_READ
4. указатель	nil
5. двойное слово	OPEN_EXISTING
6. двойное слово	0
7. дескриптор	0

CloseHandle(*HDevice*);

данный вызов освобождает системные ресурсы и прекращает доступ к файлу.

В прикладной программе рекомендуется освобождать файловое устройство один раз при выходе из программы.

Пропущен номер предыдущего уровня: 1.1. Пункт! Вставьте его и нажмите клавиши: [Ctrl]+A, [F9]. 3. DeviceIOControl()

Работа с портами происходит посредством вызовов функции работы с файловыми устройствами DeviceIOControl(). В качестве параметров передается дескриптор устройства, управляющий код, буферы для входной и выходной информации и их размеры, а также длина возвращенной информации и указатель на структуру для асинхронной работы с устройством (асинхронный ввод/вывод не поддерживается драйвером). Управляющий код формируется из битовых полей, говорящих системе о функциональности драйвера и вызываемой функции. Например, запись в порт:

```
DeviceIOControl(
    HDevice, // идентификатор файла
    IOCTL_GPD_WRITE_PORT_UCHAR, // код операции
    @buf, // указатель на буфер с данными вывода
    SizeOf(buf), // размер буфера с данными вывода
    nil, // указатель на буфер для получения данных ввода
    0, // размер буфера для получения данных
    ret, // переменная для размера возвращаемых данных
    nil // указатель на структуру асинхронного ввода-вывода
);
```

³) Все параметры здесь и далее приведены в синтаксисе Delphi.

Формат буфера *buf* определяется так:

```
_GENPORT_WRITE_INPUT=record
  PortNumber: ULONG; // номер порта
  case byte of // данные
    0:(LongData:ULONG);
    1:(ShortData:WORD);
    2:(CharData:BYTE);
  end;
```

Вызов с кодом IOCTL_GPD_WRITE_PORT_UCHAR ничего не возвращает.

Аналогично чтение порта:

```
DeviceIOControl(
  HDevice, // идентификатор файла
  IOCTL_GPD_READ_PORT_UCHAR, // код операции
  @buf, // указатель на буфер с данными вывода
  SizeOf(buf), // размер буфера с данными вывода
  @Result, // указатель на буфер для получения данных ввода
  1, // размер буфера для получения данных
  ret, // переменная для размера возвращаемых данных
  nil // указатель на структуру асинхронного ввода-вывода
);
```

В этом случае формат буфера *buf* определяется так:

```
_GENPORT_WRITE_OUTPUT=record
  PortNumber: ULONG; // номер порта
end;
```

Вызов с кодом IOCTL_GPD_READ_PORT_UCHAR возвращает один байт в буфере *Result*.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9].

8. ВЗАИМОДЕЙСТВИЕ МОДУЛЯ УПРАВЛЕНИЯ МИ-1201 АГМ С ДРАЙВЕРОМ

Все взаимодействие модуля управления МИ-1201 АГМ с драйвером сосредоточено в файле "PortsIO.pas". Здесь определены три типа процедур ввода/вывода в порты:

- 1) Прямой ввод/вывод для Windows 9x.
- 2) Ввод/вывод для Windows NT, через драйвер.
- 3) Заглушка для Windows NT для случая отсутствия драйвера.

При запуске программы код в “PortIO.pas” определяет тип ОС и подключает соответствующие процедуры ввода/вывода.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 9. УСТАНОВКА ДРАЙВЕРА “PORTIO.SYS”

- 1) Драйвер (файл “PortIO.sys”) должен быть помещен в папку
System32\Drivers
главной папки системы (обычно это C:\WinNT, т.е. полный путь будет выглядеть как C:\WinNT\System32\Drivers).
- 2) В системном реестре в ветви
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\
должен быть создан ключ с именем, совпадающим с именем файла драйвера без расширения:
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\PortIO
Значения, которые необходимо создать в данном ключе, перечислены в табл. 1.5. Данный ключ дает информацию о драйвере системе.
- 3) В указанном ключе создается ключ Parameters, который служит для указания параметров драйверу. В данном случае здесь перечисляются диапазоны портов ввода-вывода. Значения типа двойное слово, записанные в этот ключ должны быть названы IoPortAddress*N* и IoPortCount*N*, где “*N*” пробегает значения от 0 до 9. Например, если необходимы два диапазона портов, то записываются значения IoPortAddress0, IoPortCount0, IoPortAddress1, IoPortCount1. IoPortAddress отвечает базовому порту (порту с наименьшим адресом), а IoPortCount - количеству последовательных портов.

Необходимые установки записаны в файле “PortIO.reg”, их можно переписать в системный реестр дважды щелкнув мышкой на указанном файле.

Установки файла “PortIO.reg” предписывают для драйвера старт “вручную” (“Manual”), см. Control Panel→ Devices. Это сделано с целью обеспечить безопасность системы, если при запуске драйвер приведет к остановке ОС, то можно просто загрузить ОС заново. После успешного запуска можно изменить значение на “Automatic” — драйвер будет запускаться при каждом старте ОС.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 10. ПРИНЦИПЫ РАБОТЫ ВИРТУАЛЬНОГО ДРАЙВЕРА DOS

Таблица 5

ПАРАМЕТРЫ РЕЕСТРА ДЛЯ “PORTIO.SYS”

Название значения	Тип значения	Значение	Комментарии
1. Type	Двойное слово	1	
2. Start	Двойное слово	3	Старт производится вручную (“Manual”), после успешного запуска можно изменить значение на “Automatic” (см. Control Panel→ Devices).
3. Group	Строка	Extended Base	
4. ErrorControl	Двойное слово	1	
5. DisplayName	Строка	МИ-1201 ports I/O	Определяет название драйвера, которое будет отображаться в Devices(Устройства) панели управления

Для работы уже существующей программы для ОС MS-DOS под Windows NT 4.0 необходим драйвер виртуальной машины DOS (далее VDD), который получает управление при перехвате ОС обращения к портам ввода-вывода и заменяет их на вызовы функций “PortIO.sys”.

Схема взаимодействия прикладных программ DOS с портами ввода-вывода в DOS приведена на рис. 2 и в ОС Windows NT приведена на рис. 1.3.

ВЗАИМОДЕЙСТВИЕ ПРИКЛАДНЫХ ПРОГРАММ DOS С ПОРТАМИ ВВОДА-ВЫВОДА В РЕАЛЬНОМ РЕЖИМЕ

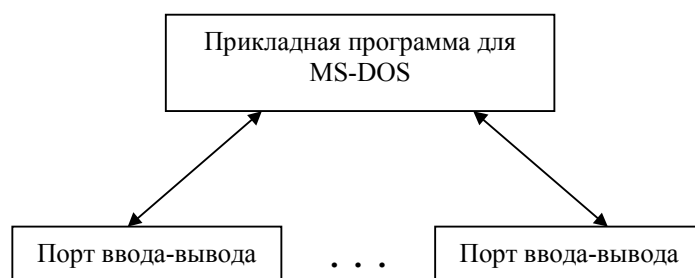


Рис. 2

ВЗАИМОДЕЙСТВИЕ ПРИКЛАДНЫХ ПРОГРАММ DOS С ПОРТАМИ ВВОДА-ВЫВОДА В СРЕДЕ WINDOWS NT

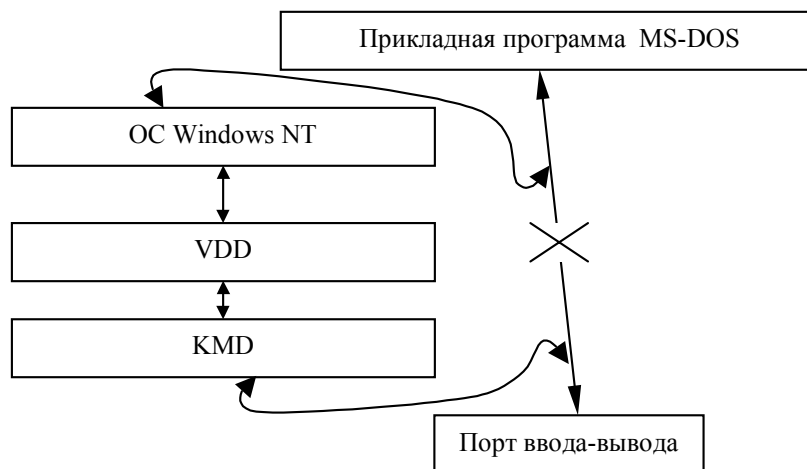


Рис. 3

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 11. ОБЩАЯ СТРУКТУРА ВИРТУАЛЬНОГО ДРАЙВЕРА DOS

Виртуальный драйвер DOS “MI1201.DLL” является подключаемой библиотекой, написанной на языке Си. Исходный текст программы см. в папке “NT-драйверы для МИ 1201-АГМ\Исходный код драйверов\Виртуальный драйвер DOS (VDD)\”.

Перечень функций “MI1201.DLL” приведен в табл. 1.6.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 12. ЗАГРУЗКА “MI1201.DLL”

При инициализации “MI1201.DLL” сообщает системе, что при попытке прямого использования портов из указанных VDD диапазонов программой DOS управление необходимо передать функции VDD, которая является обработчиком таких запросов.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 13. ОБРАБОТКА ЗАПРОСОВ К ПОРТАМ ВВОДА-ВЫВОДА

Все попытки программы DOS обратиться к портам перехватываются ОС и она производит поиск запущенного VDD, диапазоны портов которого включают в себя данный порт. Если система находит такой VDD, то она передает управление функции, которая при инициализации VDD была объявлена как обработчик подобных запросов. В описываемом “MI1201.DLL” такими функциями являются MyInB() и MyOutB(), отвечающие соответственно за ввод и вывод из/в порты. Данные функции обращаются к драйверу ядра “PortIO.sys”. С точки зрения программы DOS подобные цепочки обращений абсолютно прозрачны, т.е. программа не может отличить реальное обращение к портам от их перехвата и передачи управления цепочке драйверов.

Единственным заметным отличием может оказаться время, затрачиваемое на обращение к порту (см. пункт **Текст вывода не может распространяться больше чем на одну строку!**).

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 14. ПРОЦЕСС ВЫГРУЗКИ ДРАЙВЕРА

При выгрузке “MI1201.DLL” просто сообщает ОС то, что ОС не должна более передавать “MI1201.DLL” запросы об обращении к портам и завершает работу.

Таблица 6

Функции “MI1201.DLL”

Функция	Назначение
1. VDDInitialize()	Инициализация/освобождение драйвера.
2. MyInB()	Чтение байта из порта. На самом деле происходит обращение к KMD.
3. MyOutB()	Запись байта в порт. На самом деле происходит обращение к KMD.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 15. УСТАНОВКА ДРАЙВЕРА

В ключе реестра

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\VirtualDeviceDrivers находится значение с именем “VDD” мультистрокового типа. В это значение необходимо добавить строку, содержащую полный путь доступа к файлу драйвера, например, c:\winnt\system32\mi1201.dll). После этого ОС будет запускать “MI1201.DLL” при каждом старте виртуальной машины DOS.

Для правильной работы “MI1201.DLL” должен быть установлен драйвер ядра “PortIO.sys”.

2. СОЗДАНИЕ МАТЕМАТИЧЕСКОЙ МОДЕЛИ МАСС-СПЕКТРА ПРИ ИЗМЕРЕНИЯХ В РЕЖИМЕ СЧЕТА ИОНОВ С УЧЕТОМ ШУМОВ И НАЛОЖЕНИЯ ПИКОВ

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 16. ИЗМЕРЕНИЕ ИНТЕНСИВНОСТИ СИГНАЛА

Измерение сигнала в точке, соответствующей центру пика на масс-спектре необходимо для корректного сравнения интенсивности сигналов и определения доли вещества в пробе.

Точное измерение интенсивности особенно важно при анализе микропримесей.

Измерение интенсивности сигнала в пике может быть разделено на два этапа⁴⁾:

- 1) Определение положения центра пика⁵⁾. Центром пика считается точка сигнала максимальной интенсивности при отсутствии шумов⁶⁾.

⁴⁾ Ниже будет предложена и иная схема измерения.

⁵⁾ Положение центра пика может быть задано различно, например, значением тока электромагнита, в единицах счетчика импульсов развертки или в единицах отношения массы иона к заряду. Для дальнейшего это несущественно, но мы будем оперировать числом импульсов контроллера развертки, поскольку это, фактически, минимальный шаг, на который мы можем изменять эл.-магн. поле.

⁶⁾ На этот счет существуют и другие мнения.

- 2) Измерение интенсивности сигнала в центре пика достаточно длительное время⁷⁾. Длительное время измерения сигнала позволяет увеличить точность измерения, за счет усреднения шумовой составляющей сигнала.

Простейшим вариантом поиска могло бы быть простое измерение сигнала с необходимой длительностью при каждом допустимом значении магнитного поля (т.е. с минимальным шагом развертки). Недостаток такого подхода заключается в чрезмерных затратах времени на измерение.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 17. ИСХОДНЫЕ ДАННЫЕ И МАТЕМАТИЧЕСКАЯ ФОРМУЛИРОВКА ЗАДАЧИ

В качестве исходных данных выступает некоторая последовательность точек (x_i, y_i) , $i = -N, -(N+1), \dots, -1, 0, 1, \dots, M$ и в общем случае $N \neq M$. При этом все y_i удовлетворяют условию $y_i \geq y_{min}$, где y_{min} некоторый априорно заданный уровень сигнала.

Величина x есть интенсивность магнитного поля и обычно изменяется по закону: $x_i = x_0 + \Delta x \cdot i$, где Δx - некоторый постоянный шаг развертки (ШР) с которым изменяется напряженность поля электромагнита (ЭМ). Величина ШР удовлетворяет условию $\Delta x \ll \sigma$, где 2σ - ширина пика на половине его высоты. Значение x_0 соответствует максимальному значению y в последовательности ($y_0 > y_i$ для любого $i \in [-N..M]$)⁸⁾. Значение x_0 не обязательно совпадает с истинным центром пика (экстремумом).

Величина y_i - характеристика интенсивности выходного сигнала масс-спектрометра (ионный ток), измеренная в точке x_i .

Предполагается, что мы можем получить такую последовательность в произвольном интервале шкалы масс.

Необходимо найти значение x_{max} , при котором y_{max} соответствует истинному экстремуму пика.

⁷⁾ Предполагается, что мы имеем возможность установить любое требуемое значение положения с необходимой точностью (фактически эта точность определяется контроллером развертки и кое-какими привходящими обстоятельствами).

⁸⁾ Если существует несколько $y_j = y_{max}$, $j = j_1 \dots j_2$ и $j \in [-N..M]$ то в качестве искомого y_i принимается $i = [(j_1 + j_2)/2]$.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 18. УПРОЩЕННЫЙ АЛГОРИТМ ПОИСКА ЦЕНТРА ДЛЯ СИММЕТРИЧНОГО ПИКА

Если мы предположим, что пик симметричен относительно x_{max} , то можно отыскать x_{max} следующим образом:

$$x_{max} = \frac{\sum_{i=-N}^{-1} (x_i + x'_i) + \sum_{i=1}^M (x_i + x'_i)}{2(N + M)}, \quad (1)$$

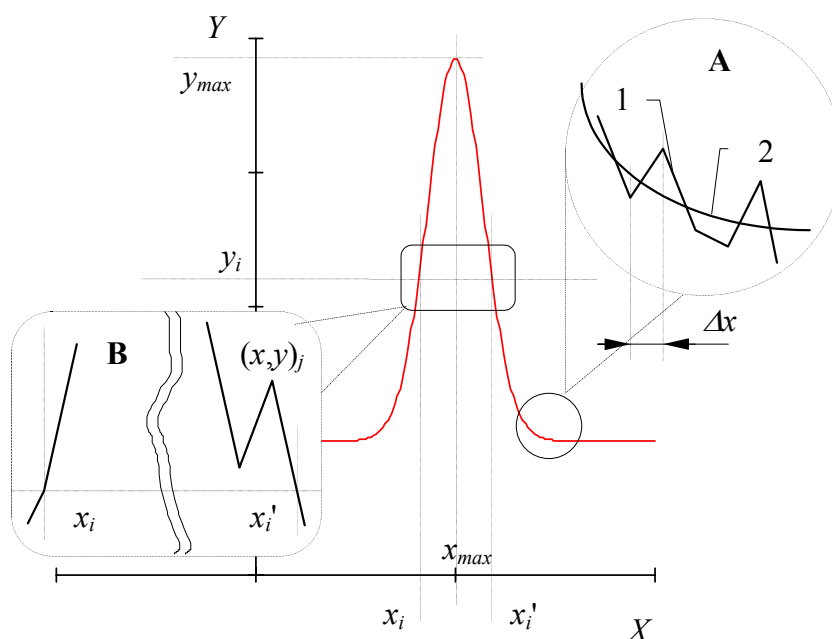
где x_i - точка исходной последовательности; x'_i - соответствует точке, в которой интенсивность сигнала $y = y_i$, но расположенной на противоположном склоне пика, как это изображено на рис. 2.1. Поскольку такой точки в исходной последовательности скорее всего нет, то x'_i следует вычислить как пересечение прямой $y = y_i$ с соответствующим сегментом ломаной линии, определяемой точками $(x, y)_j$ и $(x, y)_{j+1}$ такими, что $y_j > y_i$ и $y_{j+1} < y_i$ (см. рис. 2.1 выноска В).

Недостаток этого алгоритма — явная неработоспособность при частичном наложении пиков (несимметричный пик).

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 19. ПОИСК ЦЕНТРА ПИКА НА ОСНОВЕ АППРОКСИМАЦИИ ЭКСПЕРИМЕНТАЛЬНЫХ ТОЧЕК ФУНКЦИЕЙ

Основными преимуществами восстановления функции по экспериментальным точкам являются возможности: а) восстановления одновременно центра пика и интенсивности сигнала в центре; в) восстановление параметров для перекрывающихся пиков.

ОТЫСКАНИЕ ПОЛОЖЕНИЯ ЦЕНТРА СИММЕТРИЧНОГО ПИКА



А - вид экспериментальной 1 и реальной 2 кривой. Экспериментальная кривая это ломаная линия, вершины которой не совпадают с реальной кривой из-за случайных помех.
В - схема вычисления x_i' .

Рис. 4

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9].

20. ЭКВИВАЛЕНТНОСТЬ УСРЕДНЕНИЯ В ПРОСТРАНСТВЕ И ВРЕМЕНИ

Альтернативой алгоритму пункта

Текст вывода не может распространяться больше чем на одну строку!
является отыскание центра пика методом восстановления истинной⁹⁾ кривой.

Для этого, как минимум, необходим выбор функционального описания пика: $y = f(x, \mathbf{P})$, где $\mathbf{P} = (p_1, p_2, \dots, p_K)$ - вектор параметров функции, который надо определить по экспериментальным данным из условия наилучшего совпадения.

Прямолинейный подход состоял бы в использовании метода наименьших квадратов (МНК). Основной недостаток МНК — сложность применения для

⁹⁾ Имеется ввиду, что есть возможность установить вид функциональной зависимости для пика теоретически.

функций, которые невозможно линеаризовать. Вспомним, что МНК, лишь способ устранить случайные возмущения, возникающие при экспериментальном измерении. Он использует множество точек $(x, y)_i$ для восстановления параметров функции. В некотором приближении это можно интерпретировать как усреднение в пространстве x_i , для точек с ограниченным временем измерения (чем меньше время измерения в точке, тем больше относительная случайная погрешность).

Но с другой стороны, мы можем повысить достоверность измерения одной точки, за счет уменьшения их числа. В частности мы можем потратить все время измерения для отыскания K точек на пике, где K - число параметров выбранной функции. В этом случае, указанные K точек будут определены с большей достоверностью.

Если принять гипотезу, что погрешность измерения интенсивности сигнала в точке обратно пропорциональна времени измерения, то затратив равное время на измерение меньшего числа точек, мы не ухудшим совокупную погрешность нового набора исходных данных.

Здесь есть тонкость, связанная с тем, что если мы потратим все время на измерение M точек для $M < K$, то мы явно ухудшаем результат — восстановить функцию станет невозможно. Аналогичный результат может быть достигнут при выборе неудачных точек для измерения, например, на одной стороне пика или очень близко друг к другу. Поэтому тезис об эквивалентности усреднения в пространстве и времени может быть в некоторых случаях неверен.

Восстановление искомой функции $y = f(x, \mathbf{P})$ по K точкам сводится к решению системы K уравнений:

$$y_j = f(x_j, \mathbf{P}), j = 1, 2, K \quad (2)$$

относительно неизвестных параметров \mathbf{P} . Такое решение может оказаться проще (для небольшого числа параметров), чем применение МНК.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 21. ИЗМЕРЕНИЕ ПРИ НАЛОЖЕНИИ ПИКОВ

Для случая изолированного пика алгоритм восстановления функции не дает существенного выигрыша. Единственный плюс — не надо проводить дополнительное точное измерение интенсивности в центре пика, поскольку автоматически восстанавливается и амплитуда сигнала.

Существенно большее преимущество дает этот метод при измерении в условиях наложения S пиков¹⁰⁾. В этом случае искомой функцией будет суперпозиция функций $f(x, \mathbf{P}_s)$:

$$y = \sum_{s=1}^S f(x, \mathbf{P}_s), \quad (3)$$

содержащая $S \cdot K$ параметров.

Реально число параметров должно возрастать медленнее, поскольку можно предположить, что все пики геометрически подобны и отличаются только амплитудой. Тогда, число параметров при наложении S пиков составит $K + (S-1)$, т.е. потребует для разрешения S пиков дополнительно измерить S точек.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 22. АППАРАТНАЯ ФУНКЦИЯ

Восстановление параметров аппроксимирующей функции для “плохих” пиков (слабых или перекрывающихся) затруднено, в этом случае разумнее восстановить основные параметры пика, например, такие параметры формы, как σ или подобные ему на “хорошем” пике (пиках) в ближайшей окрестности. Поскольку форма пика в масс-спектрометре не должна сильно изменяться, то эти параметры могут быть использованы при разрешении “плохих” пиков. При этом для “плохих” пиков рассчитываются только положение центра и амплитуда.

Такой подход называется “определение аппаратной функции”, он позволяет снизить вычислительные затраты на разрешение пиков и, более того, снижает вероятность несходимости процедур численных расчетов.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 23. АППРОКСИМИРУЮЩАЯ ФУНКЦИЯ ДЛЯ ПИКА

Распределение плотности потока ионов в пучке¹¹⁾, падающем на приемную щель детектора ионов (рис. 2.2) может быть аппроксимировано нормальным распределением (график 1 на рис. 2.2)

¹⁰⁾ Наложение пиков в реальном масс-спектре существует всегда, но им обычно пренебрегают если расстояние между центрами пика много больше или просто больше их суммарной ширины.

¹¹⁾ Поскольку пучок моноэнергетичен, то можно вместо “плотности потока ионов” употреблять термин “плотность ионов”.

$$y_0(x) = \frac{A_0}{\sigma\sqrt{2\pi}} e^{-\frac{(x-x_0)^2}{2\sigma^2}} + B_0, \quad (4)$$

здесь A_0 - амплитуда пика, B_0 - фон пика, σ - полуширина пика, x_0 - положение центра пика.

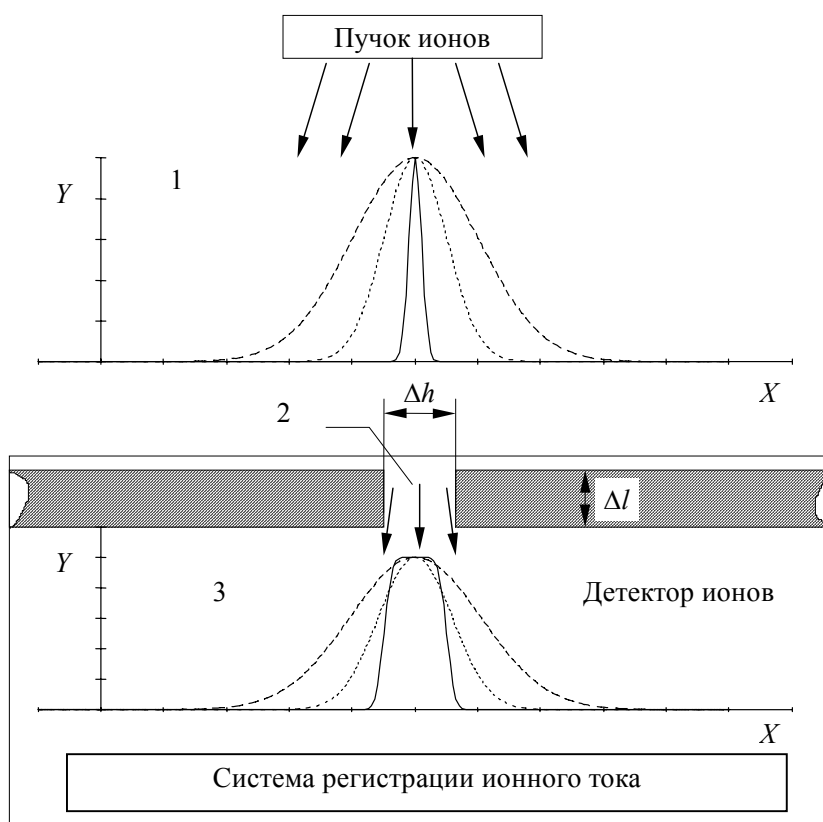
Распределение в форме **Ошибка! Источник ссылки не найден.** означает принятие гипотезы, что на распределение плотности ионов в пучке¹¹⁾ влияет значительное число независимых случайных сил, с однородным распределением плотности вероятности.

Детектор ионов представляет собой приемную щель 2 (см. рис. 2.2) шириной Δh в пластине толщиной Δl , которая вырезает из пучка ионов некоторую часть и пропускает ее к системе регистрации ионов. Высота щели полагается бесконечно большой.

Для случая бесконечно тонкой пластины ($\Delta l = 0$) интенсивность, регистрируемая детектором будет иметь вид распределения 3 (см. рис. 2.2)

$$y(x) = \frac{A_0}{\sigma\sqrt{2\pi}} \int_{x-\frac{\Delta h}{2}}^{x+\frac{\Delta h}{2}} e^{-\frac{(x-x_0)^2}{2\sigma^2}} dx + B_0\Delta h. \quad (5)$$

СВЯЗЬ РАСПРЕДЕЛЕНИЯ ПЛОТНОСТИ ПОТОКА ИОНОВ В ПУЧКЕ И
ИНТЕНСИВНОСТИ РЕГИСТРИРУЕМОГО СИГНАЛА



1 - распределение относительной¹²⁾ интенсивности сигнала до приемной щели детектора;
2 - приемная щель детектора;
3 - интегральная¹³⁾ относительная интенсивность сигнала после приемной щели детектора.
Показано три различных по ширине распределения плотности ионов в пучке и соответствующие им интенсивности регистрируемого сигнала

Рис. 5

Очевидно, что при стремлении ширины щели Δh к нулю или для $\Delta h \ll \sigma$ распределение интенсивности после щели стремится к (4). При сравнимой ширине щели Δh и ширине пика до щели σ , распределение интенсивности регистрируемого сигнала приобретает плоскую вершину в окрестности максимума и стремится к прямоугольной форме (см. рис. 5).

Функция (4) может быть представлена в форме

$$y(x) = A \cdot \left\{ \operatorname{erf} \left(\frac{x - x_0 + \Delta h/2}{\sigma\sqrt{2}} \right) - \operatorname{erf} \left(\frac{x - x_0 - \Delta h/2}{\sigma\sqrt{2}} \right) \right\} + B \quad (6)$$

где $\operatorname{erf}(x)$ - функция ошибок $\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-x^2} dx$, это известная специальная функция и для нее есть хорошие аппроксимации и методы численного вычисления.

¹²⁾ Интенсивность отнесена к ее максимальному значению.

¹³⁾ Предполагается, что все ионы, прошедшие щель, регистрируются.

Вычисление функции необходимо производить быстро. Для сложных функций, не реализованных аппаратно в процессоре, применяют либо интерполяцию по табличным значениям, либо метод суммирования ряда (хотя ряды обычно плохо сходятся).

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9].

24. ВОССТАНОВЛЕНИЕ ФУНКЦИИ ПО ДАННЫМ ИЗМЕРЕНИЯ

В этом разделе рассмотрен вопрос применения МНК для восстановления параметров функции произвольного вида.

Пропущен номер предыдущего уровня (1.1. Пункт)! Вставьте его и нажмите клавиши: [Ctrl]+A, [F9]. 4. Общая постановка задачи

Пусть дана последовательность:

$$(x_i, y_i), i = 1, 2, \dots N, \quad (7)$$

и функция

$$y = F(x, \mathbf{P}), \quad (8)$$

где \mathbf{P} - вектор параметров, $\mathbf{P} = (p_1, p_2, \dots p_k)$.

Необходимо найти такой \mathbf{P}_0 , чтобы функционал

$$\Phi(\mathbf{P} = \mathbf{P}_0) = \sum_{i=1}^N (F(x_i, \mathbf{P}_0) - y_i)^2 \quad (9)$$

принимал минимальное значение.

Необходимое условие экстремума в точке \mathbf{P}_0 :

$$\frac{\partial \Phi(\mathbf{P}_0)}{\partial P_k} = 0, k = 1, \dots N \quad (10)$$

или

$$2 \sum_{i=1}^N \frac{\partial F(x_i, \mathbf{P}_0)}{\partial P_k} \cdot (F(x_i, \mathbf{P}_0) - y_i) = 0, k = 1, \dots N$$

Таким образом, есть альтернатива, либо решать уравнение

$$\Phi(\mathbf{P}_0) = \min, \quad (11)$$

либо систему N уравнений

$$\sum_{i=1}^N \frac{\partial F(x_i, \mathbf{P}_0)}{\partial P_k} \cdot [F(x_i, \mathbf{P}_0) - y_i] = 0, k = 1, \dots N \quad (12)$$

Поскольку приближение для пика, описанное в пункте **Текст вывода не может распространяться больше чем на одну строку!**, не является абсолютно точным, более того, экспериментальные измерения дают несимметричные пики (хвост в сторону увеличения массы иона), то является

разумным ориентироваться на восстановление произвольной функции. В этом случае, невозможно использовать особенности конкретного вида функции и линеаризовать или другим образом упростить уравнения (11) или (12). Поэтому приходится ориентироваться на численный расчет.

Пропущен номер предыдущего уровня (1.1. Пункт)! Вставьте его и нажмите клавиши: [Ctrl]+A, [F9]. 5. Общие принципы численного решения уравнений

Обычная схема численного решения уравнения может быть представлена в виде:

- 1) Выбор начального приближения для решения \mathbf{P}_0 в области допустимых значений \mathbf{P} . Пусть начальное приближение выбрано и равно \mathbf{P}_{00} .
- 2) Отыскание следующего приближения \mathbf{P}_{01} такого, чтобы расстояние от \mathbf{P}_{01} до \mathbf{P}_0 , было меньше чем расстояние от \mathbf{P}_{00} до \mathbf{P}_0 .
- 3) Повтор операции 2) для вычисления последовательности \mathbf{P}_{0j} , $j = 1, 2, \dots$, сходящейся к \mathbf{P}_0 .
- 4) Вычисление \mathbf{P}_{0j} до $j = M$ такого, что расстояние от \mathbf{P}_{0M} до \mathbf{P}_0 будет меньше заданной величины ε .
- 5) Величина \mathbf{P}_{0M} принимается как искомая величина \mathbf{P}_0 .

Наиболее важным этапом является выбор алгоритма вычисления \mathbf{P}_{0j} . Этот алгоритм должен: а) обеспечивать сходимость к \mathbf{P}_0 ; б) обеспечивать максимальную скорость сходимости. т.е. минимальное число M .

Пропущен номер предыдущего уровня (1.1. Пункт)! Вставьте его и нажмите клавиши: [Ctrl]+A, [F9]. 6. Метод наискорейшего спуска

Применяется для поиска экстремумов, т.е. в нашем случае пригоден для решения уравнения (11).

В этом случае выбирают:

$$\mathbf{P}_{0j+1} = \mathbf{P}_{0j} - S_j \cdot \frac{d\Phi(\mathbf{P}_{0j})}{d\mathbf{P}}, \quad (13)$$

где S - некоторая произвольная положительная величина. В общем случае S может изменяться от шага к шагу для достижения оптимальной сходимости.

Вычисление по (13) выполняет смещение точки в направлении против градиента функции, т.е. в сторону уменьшения значения функции или в направлении наискорейшего спуска. Величину S следует определить из условий: а) обеспечения сходимости, б) максимальной скорости сходимости.

В нашем случае, когда известна точная нижняя граница функции $\Phi(\mathbf{P})$, равная нулю, выбор S можно сделать так: новая точка \mathbf{P}_{0j+1} определяется как пе-

ресеение прямой сонаправленной с вектором $\text{grad}_{\mathbf{P}}(\Phi(\mathbf{P}_{0j}))$ и проведенной через точку $\Phi(\mathbf{P}_{0j})$ с плоскостью $\mathbf{P} = 0$. Что дает выражение для S

$$S_j = \frac{\Phi(\mathbf{P}_{0j})}{\left| \frac{\partial \Phi(\mathbf{P}_{0j})}{\partial \mathbf{P}} \right|} \quad (14)$$

Скорее всего этот способ вычисления неприемлем. Простейший аргумент — при приближении к точке минимума мы получим увеличение шага и деление на ноль в точке экстремума

Пропущен номер предыдущего уровня (1.1. Пункт)! Вставьте его и нажмите клавиши: [Ctrl]+A, [F9]. 7. Модификация метода наискорейшего спуска для гарантии сходимости

Основной проблемой всех способов выбора следующего значения приближения является обеспечение сходимости в любых условиях. Для улучшения этого показателя любую методику выбора следующего значения приближения можно дополнить следующими шагами:

- 1) Вычисляется значение функции в новой точке $\Phi(\mathbf{P}_{0j+1})$.
- 2) Сравняется со значением в предыдущей точке $\Phi(\mathbf{P}_{0j})$.
- 3) Если значение $\Phi(\mathbf{P}_{0j+1}) \geq \Phi(\mathbf{P}_{0j})$, то \mathbf{P}_{0j+1} корректируется, новое значение вычисляется по формуле $\mathbf{P}_{0j+1}' = (\mathbf{P}_{0j+1} + \mathbf{P}_{0j})/2$.
- 4) Шаги 1)÷3) повторяются пока не выполнится неравенство $\Phi(\mathbf{P}_{0j+1}) < \Phi(\mathbf{P}_{0j})$. Это усовершенствование почти гарантирует сходимость, при этом скорость сходимости ухудшается незначительно.

Пропущен номер предыдущего уровня (1.1. Пункт)! Вставьте его и нажмите клавиши: [Ctrl]+A, [F9]. 8. Параболическое уточнение шага

Вычисление следующего значения приближения по формуле (14) является весьма грубым и может существенно сказываться на скорости сходимости, ухудшая ее.

Для ускорения сходимости можно использовать аппроксимацию многочленом второй степени для более точного вычисления следующего значения \mathbf{P}_{0j+1} .

В этом случае предполагается, что функцию вблизи экстремума можно представить приближенно в виде:

$$\Phi(\mathbf{P}) = a \cdot \mathbf{P}^2 + \mathbf{b} \cdot \mathbf{P} + c. \quad (15)$$

Такое приближение является весьма точным для ближайшей окрестности минимума (это следует из разложения в ряд Тейлора).

Если мы знаем коэффициенты в (15), то можно вычислить приближение для точки экстремума, решив уравнение

$$2a \cdot \mathbf{P} + \mathbf{b} = 0. \quad (16)$$

Сами коэффициенты a и \mathbf{b} могут быть вычислены, например, по значению функции $\Phi(\mathbf{P}_{0j})$, ее производной $\frac{d\Phi(\mathbf{P}_{0j})}{d\mathbf{P}}$ в точке \mathbf{P}_{0j} и значению функции в предыдущей точке $\Phi(\mathbf{P}_{0j-1})$.

Однако, поскольку у начального приближения \mathbf{P}_{00} нет предыдущей точки, то параболическая аппроксимация далее используется по следующему алгоритму:

- 1) Вычисляется \mathbf{P}_{0j+1} по методу описанному в пунктах
Текст вывода не может распространяться больше чем на одну строку!
и

Текст вывода не может распространяться больше чем на одну строку!

При этом оказываются известными $\Phi(\mathbf{P}_{0j})$, $\frac{d\Phi(\mathbf{P}_{0j})}{d\mathbf{P}}$ и $\Phi(\mathbf{P}_{0j+1})$.

- 2) На основе $\Phi(\mathbf{P}_{0j})$, $\frac{d\Phi(\mathbf{P}_{0j})}{d\mathbf{P}}$ и $\Phi(\mathbf{P}_{0j+1})$ восстанавливаются коэффициенты параболической аппроксимации (15) и вычисляется уточненное значение \mathbf{P}'_{0j+1} по формуле (16).

Вычисление \mathbf{P}'_{0j+1} на основе параболического уточнения может быть выполнено по следующему алгоритму:

- 1) Модифицируем (15) к виду: $\Phi(\mathbf{P}) = a \cdot (\mathbf{P} - \mathbf{P}_{0j})^2 + \mathbf{b} \cdot (\mathbf{P} - \mathbf{P}_{0j+1}) + c$. Это не ухудшает общности, но упрощает восстановление коэффициентов.
- 2) Для вычисления значений коэффициентов следует решить систему уравнений:

$$\begin{aligned} \Phi_{0j} &= \mathbf{b} \cdot (\mathbf{P}_{0j} - \mathbf{P}_{0j+1}) + c \\ \frac{d\Phi(\mathbf{P}_{0j})}{d\mathbf{P}} &= \mathbf{b} \\ \Phi_{0j+1} &= a \cdot (\mathbf{P}_{0j+1} - \mathbf{P}_{0j})^2 + c \end{aligned} \quad (17)$$

- 3) Решая уравнение (16), с учетом модификации, получим: $\mathbf{P}'_{0j+1} = -\frac{\mathbf{b}}{2a} + \mathbf{P}_{0j}$.

Откуда для уточненного S' получим:

$$S'_j = \frac{1}{2} \frac{(\mathbf{P}_{0j+1} - \mathbf{P}_{0j})^2}{\Phi_{0j+1} - \Phi_{0j} - \mathbf{grad}(\Phi(\mathbf{P}_{0j})) \cdot (\mathbf{P}_{0j+1} - \mathbf{P}_{0j})}. \quad (18)$$

Вычисление \mathbf{P}'_{0j+1} можно провести по формуле (13).

С учетом предыстории вычисления \mathbf{P}_{0j+1} по формуле (13), можно записать вместо (18)

$$S'_j = \frac{1}{2} \frac{(S_j \cdot \mathbf{grad}(\Phi(\mathbf{P}_{0j})))^2}{\Phi_{0j+1} - \Phi_{0j} - S_j \cdot (\mathbf{grad}(\Phi(\mathbf{P}_{0j})))^2}. \quad (19)$$

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 25. О ПРОГРАММЕ ДЛЯ ЧИСЛЕННОГО РЕШЕНИЯ УРАВНЕНИЯ

Здесь приведены рассуждения о возможной реализации метода наискорейшего спуска. Для других методов суть подхода не меняется.

Реализация программы должна обеспечивать применение алгоритма для произвольной аппроксимирующей функции. В качестве примера для подражания можно взять реализацию функции “genfit” в пакете MathCAD.

Программа должна позволять передать процедуре численного поиска минимума: 1) начальные значения приближения для параметров — \mathbf{P}_0 , 2) процедуру вычисления значения функции в произвольной точке \mathbf{P} — $\Phi(\mathbf{P})$, 3) процедуру вычисления вектора из частных производных функции $\Phi(\mathbf{P})$ в точке \mathbf{P} — $\mathbf{grad}_{\mathbf{P}}(\Phi(\mathbf{P}))$, 4) желаемую точность вычисления ε .

Программа должна вернуть значение \mathbf{P}_0 с указанной точностью или информировать о невозможности отыскания такового. Приемлемым вариантом может быть возврат значения \mathbf{P}_0 с указанием достигнутой точности.

Передачу процедур вычисления функции и параметрического градиента функции следует сделать через функциональные параметры типа $F: \text{function}(\dots)$; или $F: \text{procedure}(\dots)$.

После реализации такой процедуры поиск приближений для новых аппроксимирующих функций будет заключаться только в программировании процедур вычисления значения функции и градиента функции.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 26. ОСНОВНАЯ ПРОЦЕДУРА МОДУЛЯ *UGENFIT* И ЕЕ АРГУМЕНТЫ

Алгоритм обобщенного поиска наилучшего приближения набора двумерных данных N -параметрической функцией реализован в модуле *uGenFit* (см. файл *uGenFit.pas*).

Основные процедуры модуля: *GeneralFit* и *GeneralFitX*. Они различаются тем, что *GeneralFitX* дополнительно возвращает значение достигнутой точности вычислений.

Процедуры выполняют поиск значений параметров произвольной функции, при которых наилучшим образом (минимум суммы квадратов отклонений) аппроксимируется набор заданных точек x_i, y_i .

Их заголовки определены следующим образом:

```
procedure GeneralFit(aF:tFunc; aGradF:tGradF; var P:tParameters; const
X,Y:tData; Eps:Double; var ErrCode:tErrorCode);
```

Описание аргументов см. *GeneralFitX*.

procedure GeneralFitX(aF:tFunc; aGradF:tGradF; var P:tParameters; const X,Y:tData; Eps:Double; var ReachedEps:Double; var ErrCode:tErrorCode);

Аргументы процедуры имеют следующее значение: *aF* - вычисляет значение аппроксимирующей функции в точке *x* при параметрах **P**, тип определен как *tFunc = function (const P:tParameters; X:Double):Double*; *aGradF* - вычисляет значения всех частных производных аппроксимирующей функции по всем *P_i* в точке *x* при параметрах **P**, тип определен как *tGradF = procedure(const P:tParameters; X:Double; var GradF: tParameters)*; *P* - ВХОД: значения начального приближения для параметров, ВЫХОД: значения найденного наилучшего приближения для параметров, типы определены так

```
tParameterNumber=1..cMaxParameterSize;
tParametersArray=array[tParameterNumber] of Double;
tParameters=record
    Size:tParameterNumber;
    Par:tParametersArray;
end;
```

X, *Y* - массивы данных *x_i* и *y_i*, типы определены так

```
tDataNumber=1..cMaxDataSize;
tdataArray=array[tDataNumber] of Double;
tData=record
    Size:tDataNumber;
    Data:tdataArray;
end;
```

Eps - желаемая погрешность вычислений; *ReachedEps* - возвращает реально достигнутую погрешность, в любом случае; *ErrCode* - возвращает код ошибки или 0 — если нет ошибки.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9].

27. ВСПОМОГАТЕЛЬНЫЕ ПРОЦЕДУРЫ МОДУЛЯ UGENFIT

procedure AllocateP(var pP:tPtrParameters; Size:tParameterNumber);

procedure AllocateD(var pD:tPtrData; Size:tDataNumber);

Размещают в памяти структуру *tParameters* или *tData* размером *Size* и устанавливают значение *Size* в поле *Size* структуры (см. определение типа *tParameters* и *tData*). Размер — это число элементов массива РЕАЛЬНО РАЗМЕЩЕННЫХ В ПАМЯТИ. Обычно *Size* < *cMaxParameterSize* или *cMaxDataSize*. НЕ РЕКОМЕНДУЕТСЯ изменять значение поля *Size* структуры.

procedure *FreeP*(var *pP*:*tPtrParameters*);

procedure *FreeD*(var *pD*:*tPtrData*);

Освобождают память, занимаемую структурой *tParameters* или *tData*, размер освобождаемой памяти берется из поля *Size* структуры (см. определение типа *tParameters* и *tData*).

procedure *MoveP*(const *p1*:*tParameters*; var *p2*:*tParameters*);

procedure *MoveD*(const *d1*:*tData*; var *d2*:*tData*);

Присваивает *p2* (*d2*) значение *p1* (*d1*) с учетом их размера. Возбуждает ошибку *RunError*(201), если размеры структур не совпадают. Размер берется из поля *Size* структуры (см. определение типа *tParameters* и *tData*).

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 28. ПРОЦЕДУРЫ ПОДГОТОВКИ ДАННЫХ

Для увеличения скорости и точности вычислений процедуры *GeneralFit* рекомендуется нормализовать исходные данные с помощью процедур модуля *uDataPreparation*, см. файл *uDataPre.pas*.

Модуль содержит процедуру нормализации данных, которая приводит все точки исходных массивов к диапазону [0..1] и возвращает заполненную структуру данных для обратного преобразования (денормализации). Алгоритм нормализации вектора данных:

- 1) В массиве (векторе) ищется наименьшее значение, запоминается в структуре данных обратного преобразования.
- 2) Из всех элементов массива вычитается наименьшее значение, найденное на шаге 1).
- 3) Ищется наибольшее значение в массиве, запоминается в структуре данных обратного преобразования.
- 4) Все элементы массива делятся на наибольшее значение, найденное на шаге 3).

procedure *NormalizeVector*(var *NP*:*tNormalizeParameter*; var *V*:*tData*);

Нормализует вектор данных *V* и возвращает параметры преобразования в структуре *NP*. Тип *tNormalizeParameter* определен так:

```
tNormalizeParameter=record
    Origin, Scale:Double;
end;
```


procedure NormalizeData(var ND:tNormalizeParameters; var X,Y:tData);

Нормализует данные X , Y и возвращает параметры преобразования в структуре ND . Тип $tNormalizeParameters$ определен так:

```
tNormalizeParameters=record
    x,y:tNormalizeParameter;
end;
```

function GetUnNormalized(const NP:tNormalizeParameter; const X:Double):Double;

Возвращает значение X преобразованное к ненормализованному значению в соответствии с данными NP .

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 29. ЗАДАНИЕ АППРОКСИМИРУЮЩИХ ФУНКЦИЙ

Данная реализация процедуры *GeneralFit* принимает в качестве функциональных параметров аппроксимирующую функцию $F(x, P)$, а не функционал $\Phi(P)$ (см. пункт

Текст вывода не может распространяться больше чем на одну строку!).

Функционал $\Phi(P)$ вычисляется внутри процедуры *GeneralFit*.

Функциональные параметры aF и $aGradF$ передают процедуре *GeneralFit* указатели на процедуры вычисления аппроксимирующей функции и ее параметрического градиента. Пользователь должен самостоятельно запрограммировать эти процедуры.

В качестве примера можно использовать модуль *uFuncDef* (см. файл *uFuncDef.pas*), где содержится код вычисления аппроксимирующей функции и ее параметрического градиента описанных в пункте **Текст вывода не может распространяться больше чем на одну строку!**

Процедуры должны иметь заголовки следующего типа:

***tFunc* = function(const P:tParameters; X:Double):Double;**

При вызове процедура типа *tFunc* должна вычислить и вернуть значение аппроксимирующей функции в точке X при значении параметров P .

***tGradF* = procedure(const P:tParameters; X:Double; var GradF:tParameters);**

При вызове процедура типа *tGradF* должна вычислить и вернуть в переменной *GradF* значение параметрического градиента аппроксимирующей функции в точке X при значении параметров P . В элементе массива данных структуры *GradF* должна возвращаться производная по соответствующему ему параметру. Порядок следования параметров внутри структу-

ры несущественен для работы процедуры *GeneralFit*, но пользовательские функции не должны менять его в процессе вычислений.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ (1. ГЛАВА)! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 30. ПОРЯДОК РАБОТЫ С ПРОЦЕДУРОЙ *GENERALFIT*

При использовании процедуры *GeneralFit* следует придерживаться следующего порядка:

- 1) Запрограммировать и протестировать процедуры вычисления аппроксимирующей функции и ее параметрического градиента. Следует позаботиться, чтобы при вычислении не возникало переполнения или деления на ноль в особых точках функции. Для этого в особых точках следует возвращать числовой результат. Например, в особой точке, где функция стремится к бесконечности, следует возвращать максимальное число для математического сопроцессора (максимальное значение для типа *Double* равно $\sim 10^{300}$).
- 2) Разместить в памяти необходимые структуры данных нужного размера: 1) массивы данных X , Y и 2) массив параметров P . Рекомендуется динамическое размещение процедурами *AllocateD* и *AllocateP*.
- 3) Заполнить массивы данных X , Y значениями. Порядок следования точек в массиве несущественен, но x_i и y_i , относящиеся к одной точке должны находиться в элементах массивов с одинаковым индексом i .
- 4) Рекомендуется нормализовать данные (см. пункт **Текст вывода не может распространяться больше чем на одну строку!**).
- 5) Заполнить массив параметров начальными приближениями. Здесь имеет смысл постараться выбрать наилучшее приближение — чем ближе исходная точка P к истинному значению P_0 — тем быстрее сосчитает *GeneralFit*.
- 6) Вызвать процедуру *GeneralFit*, передав ей подготовленные аргументы.
- 7) После возврата из *GeneralFit* проверить код ошибки и, если он равен нулю, то в переменной P возвращены искомые значения параметров.

3. МОДЕЛИРОВАНИЕ И МАТЕМАТИЧЕСКАЯ ОБРАБОТКА МАСС-СПЕКТРОВ

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 31. ОПТИМИЗАЦИЯ СБОРА И ОБРАБОТКИ ДАННЫХ МАСС-СПЕКТРОМЕТРИЧЕСКИХ ИЗМЕРЕНИЙ

Качество масс-спектрометрических измерений наряду с паспортными данными и настройкой прибора зависит от оптимизации сбора и обработки экспериментальных данных [1, стр. 497].

С этой точки зрения очевидно преимущество цифровых методов, так как:

- 1) Аналоговые данные подвержены погрешностям в определении положения и площади пиков, что связано с зависимостью этих величин от постоянной времени регистрации и скорости сканирования.
- 2) Собранные цифровые данные можно обработать различными методами (например пакет TableCurve имеет на входе цифровой фильтр), а аналоговые данные, как только они зарегистрированы, больше нельзя изменить, так как они уже зафиксированы аналоговой системой регистрации.
- 3) Одним из наиболее серьезных ограничений аналоговых данных является невозможность их накопления с целью увеличения в спектре отношения сигнала/шум.

Кроме того, как было показано ранее [2, стр. 9], цифровая регистрация (режим счета ионов) позволяет проводить планирование эксперимента (находить оптимальное соотношение времен измерения сигнала и фона). В частности при слабых сигналах, много меньших уровня фона, соответствующих измерениям микроконцентрации, времена измерения сигнала и фона должны быть одинаковыми.

Для достижения заданного отношения сигнал/шум используется накопление данных [1, стр. 500]. Выбирая способ накопления данных необходимо учитывать, что в некоторых выпускаемых приборах автоматически осуществляется предварительная обработка данных (сглаживание и вычитание фона). Это очень неудачная практика. Как будет ясно из последующего, не существует однозначного способа проведения подобных операций, и такой подход может приводить к неверным результатам. Поэтому, как правило, ко всем методам обработки данных необходимо относиться с осторожностью, а первоначальные необработанные данные всегда следует хранить в неизменном виде. В наших измерениях результаты измерений представлены в текстовом формате и, ввиду нестабильности шкалы масс, с неоткорректированной массовой шкалой. Эти данные хранятся как исходные. При необходимости (измерение положения от-

дельных пиков) может быть сформирован текстовый файл с откорректированной шкалой масс. Подобная корректировка неоднозначна, и хотя значение реперных точек на шкале остается известным, вносимые изменения могут привести к некоторой потере информации.

При проведении процедуры сглаживания (например для определения максимума и положения пика) используются метод преобразования Фурье и метод наименьших квадратов (МНК).

Выбор количества точек, используемых при сглаживании МНК, определяется следующими соображениями [1, стр. 503].

- 1) Степень сглаживания увеличивается с увеличением числа точек в интервале сглаживания. Оптимальная величина этого интервала равна 0,7 от ширины пика на половине его высоты.
- 2) Степень сглаживания уменьшается при увеличении степени полинома. В общем случае применение квадратичных и кубических полиномов более эффективно, чем применение полиномов более высокой степени.
- 3) Для увеличения степени сглаживания процедуру сглаживания можно повторить с новым сглаживающим полиномом.

Сглаживание осуществляют также различными методами, использующими преобразование Фурье. Преобразование Фурье спектра, содержащего шум, приводит к разложению по пространственным гармоникам как полезного сигнала, так и шума. При этом можно уменьшить шумовую компоненту путем умножения пространственного спектра на некоторую весовую функцию, после чего обратное преобразование приводит к сглаживанию спектра.

Сложность такого метода заключается в выборе подходящей весовой функции, выбор которой во многом зависит от желаемой степени сглаживания. Увеличение отношения сигнал/шум приводит к уменьшению разрешения. В литературе [1, стр. 507] есть ссылки на оптимальный в этой точке зрения фильтр.

Методы сглаживания содержат многочисленные субъективные факторы и поэтому использовать их следует осторожно.

Для увеличения вероятности правильной интерпретации масс-спектров необходимо использовать возможно большее число методов их анализа.

Пропущен номер предыдущего уровня: 1.1. Пункт! Вставьте его и нажмите клавиши: [Ctrl]+A, [F9]. 9. Метод обратной свертки

В наблюдаемую ширину линий в масс-спектрах вносят вклад факторы инструментального уширения. К ним относятся конечная разрешающая сила, немоноэнергетичность ионов и другие. Если вклад этих факторов известен, и его можно описать некоторой функцией (B), то наблюдаемый спектр будет сверткой “истинного” спектра (f_i) и этой аппаратной функции. Истинный спектр можно было бы получить путем применения операции обратной свертки к на-

блюдаемому спектру (f_o), конкретный вид которого зависит от характеристик масс-спектрометра. Наблюдаемый спектр связан с “истинным” спектром уравнением “свертки”:

$$f_o = f_i B \quad (20)$$

Восстановление функции f_i по функции f_o называется “операцией обратной свертки”. Статистический высокочастотный шум приводит к появлению шумовой компоненты в уравнении **Ошибка! Источник ссылки не найден.** Наличие этой шумовой компоненты означает, что операция обратной свертки не является однозначной, при этом в спектре, полученном в результате обратной свертки, возможно появление шума и ложных осцилляций. Это служит объяснением того факта, что получаемое улучшение разрешения достигается за счет уменьшения отношения сигнал/шум точно так же, как увеличение отношения сигнал/шум при сглаживании в общем случае сопровождается потерей разрешения.

Пропущен номер предыдущего уровня: 1.1. Пункт! Вставьте его и нажмите клавиши: [Ctrl]+A, [F9]. 10. Аппроксимация кривых нелинейным методом наименьших квадратов

Масс-спектр можно синтезировать цифровыми или аналоговыми методами путем суммирования функций, соответствующих отдельным пикам, и получить результирующую функцию, которая близка к экспериментальному спектру. Функция, описывающая отдельный пик, конструируется таким образом, чтобы она зависела от таких параметров пика, как его положение, интенсивность, ширина, тип функциональной зависимости, уровень фона. Такой синтез кривых дает полезную начальную информацию о спектре для дальнейшей его более строгой обработки по нелинейному методу наименьших квадратов. Последний представляет собой попытку оптимизации синтеза кривых с учетом того, что параметры, описывающие спектр, входят в алгебраические выражения нелинейным образом. В качестве аппроксимирующей функции в первом приближении может быть взята функция Гаусса. В литературе [1] приводят преимущества метода наименьших квадратов:

- 1) Обеспечивает воспроизводимость параметров пиков.
- 2) Уменьшает влияние субъективных факторов.
- 3) Учитывает вес отдельных точек спектра.
- 4) Позволяет указать точность описания экспериментальных данных и ошибки определяемых параметров.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 32. ГРАФИЧЕСКОЕ ПРЕДСТАВЛЕНИЕ И ЭМУЛЯЦИЯ ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ

Первичную обработку масс-спектров: определение положения и величины максимумов пиков удобно проводить в графическом представлении. Виртуальный инструмент (ВИ) Mass-spectr.vi, созданный в пакете LabVIEW для вывода на экран масс-спектра, записанного в текстовом формате, приведен на рис. 3.1 (Панель управления) и рис. 3.2 (Блок - диаграмма).

ПАНЕЛЬ УПРАВЛЕНИЯ ВИРТУАЛЬНОГО ИНСТРУМЕНТА ДЛЯ ГРАФИЧЕСКОГО ПРЕДСТАВЛЕНИЯ МАСС-СПЕКТРА

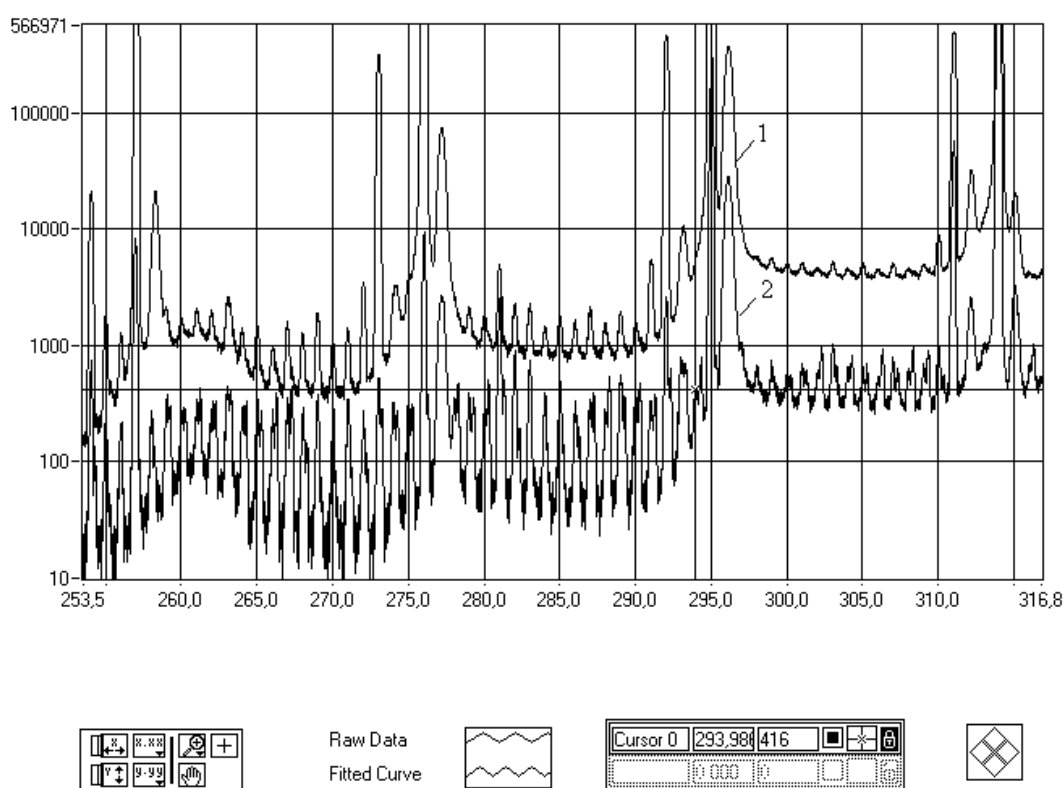


Рис. 6

БЛОК - ДИАГРАММА ВИРТУАЛЬНОГО ИНСТРУМЕНТА ДЛЯ
ГРАФИЧЕСКОГО ПРЕДСТАВЛЕНИЯ МАСС-СПЕКТРА

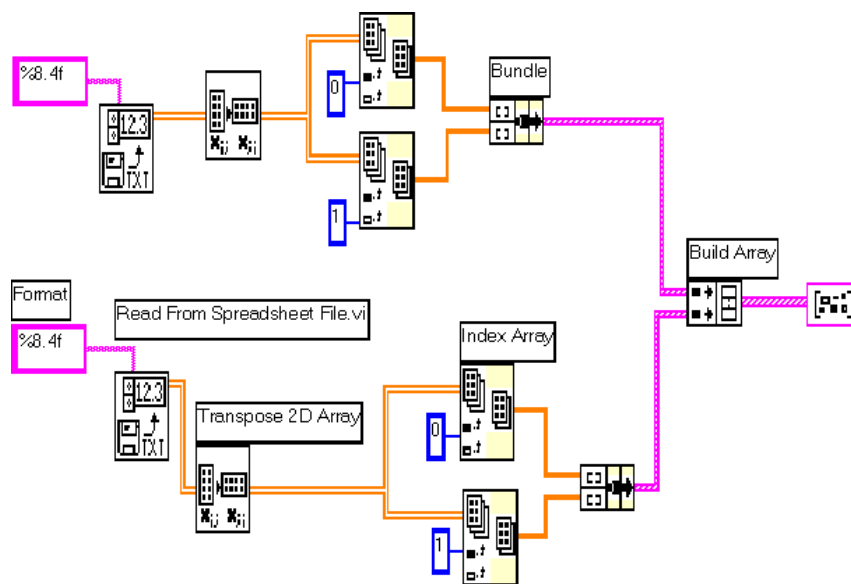


Рис. 7

Участки масс-спектра могут быть увеличены и положение отдельных точек спектра измерено курсором. Результаты измерений могут быть скопированы из редактора курсора для последующей обработки.

На панели управления (рис. 6) приведены масс-спектры UF_6 , записанные при ионизирующем напряжении 30,1 В и 50 В (верхний масс-спектр). Шкала масс масс-спектра скорректирована по методике изложенной ниже.

Эмуляция экспериментальных данных позволяет проверить эффективность процедуры сглаживания и методы анализа масс-спектров, в том числе целесообразность применения концепции аппаратной функции.

Для аппроксимации отдельных пиков выбрана функция Гаусса с наложением фонового ионного тока и статистических флуктуаций, имеющих также гауссовское распределение. Средняя квадратичная флуктуация равна корню квадратному из числа регистрируемых ионов [3, с.11]. Функция Гаусса используется как начальное приближение в предположении узкой выходной щели масс-спектрометра.

Кривая Гаусса имеет вид [4 с.186]

$$y = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-(x - x_0)^2 / 2\sigma^2\right],$$

где σ^2 - дисперсия; x_0 - положение вершины кривой.

В одном из способов определения разрешающей способности R [5 с.28] используют ширину пика ΔM на 50% его высоты (рис. 3.3) и записывают $R = M/\Delta M$. При $x - x_0 = \Delta M/2$

$$y = y_{\max} \exp\left[-(\Delta M)^2 / 8\sigma^2\right] = 0,5y_{\max}.$$

$$\text{Откуда } \sigma^2 = \frac{\Delta M^2}{8 \ln 2} = \frac{M^2}{8R^2 \ln 2}.$$

АППРОКСИМАЦИЯ ОТДЕЛЬНОГО ПИКА ФУНКЦИЕЙ
ГАУССА

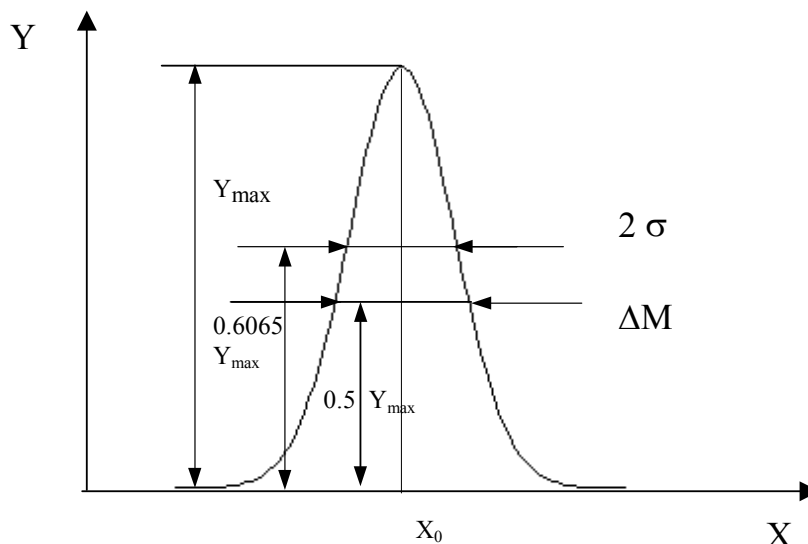


Рис. 8

Подставляя полученное значение дисперсии в формулу кривой Гаусса получаем кривую аппроксимации пика:

$$y = y_{\max} \exp\left[-\left(1.6651 \cdot (M - M_0) \frac{R}{M}\right)^2\right]. \quad (21)$$

Формула (21) позволяет проводить математическую обработку отдельных пиков, в том числе неразрешенных, с вычислением действительной разрешающей способности (ширины пиков). Кроме того, формула (21) использована для эмуляции участков масс-спектра.

В качестве примера приведены панель управления и диаграмма виртуального инструмента эмуляции триплета: рис. 3.4 и рис. 3.5. Триплет включает полностью разрешенный пик $M = 280$ и два взаимно неразрешенных пика $M = 280,3$ и $M = 280,4$. Используемые в блок-диаграмме подпрограммы (Subvi) Subpeak и Subnoise представлены на рис. 3.6 и рис. 3.7.

Поскольку работа подпрограммы Subnoise неочевидна (выбирается произвольный элемент из генерируемого массива гауссовского шума), проведена проверка эмуляции фона при заданном его значении равном 400. Результаты нескольких обработок эмуляции: $401,35 \pm 20,36$; $399,88 \pm 19,14$; $402,15 \pm 19,08$; $399,21 \pm 19,60$, говорят о правильной работе подпрограммы Subnoise.

ПАНЕЛЬ УПРАВЛЕНИЯ ВИРТУАЛЬНОГО ИНСТРУМЕНТА ЭМУЛЯЦИИ ТРИПЛЕТА

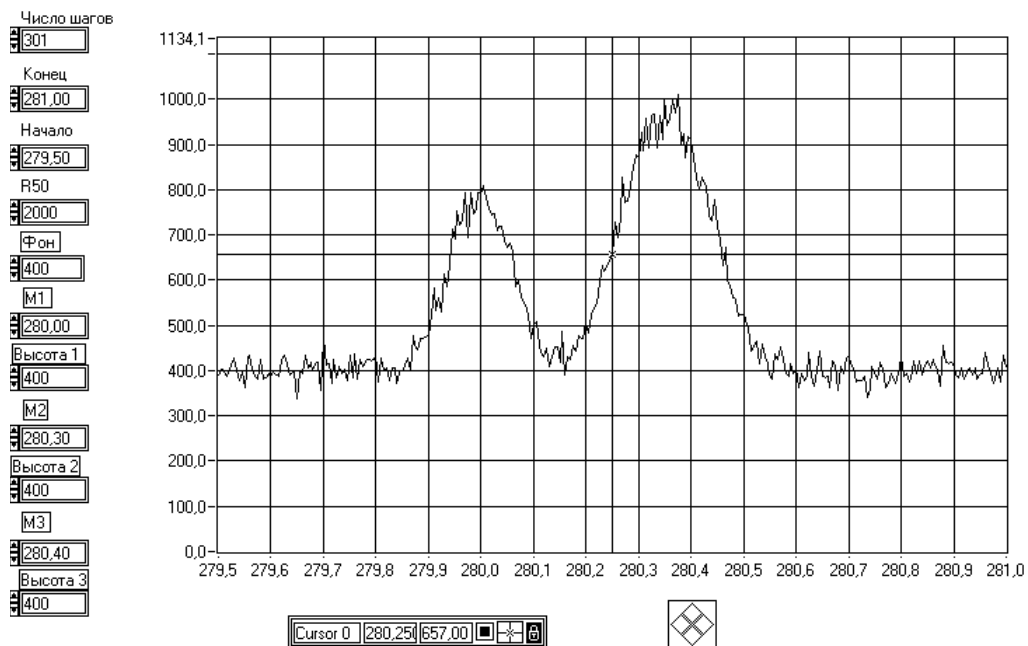


Рис. 9

БЛОК- ДИАГРАММА ВИРТУАЛЬНОГО ИНСТРУМЕНТА ЭМУЛЯЦИИ ТРИПЛЕТА

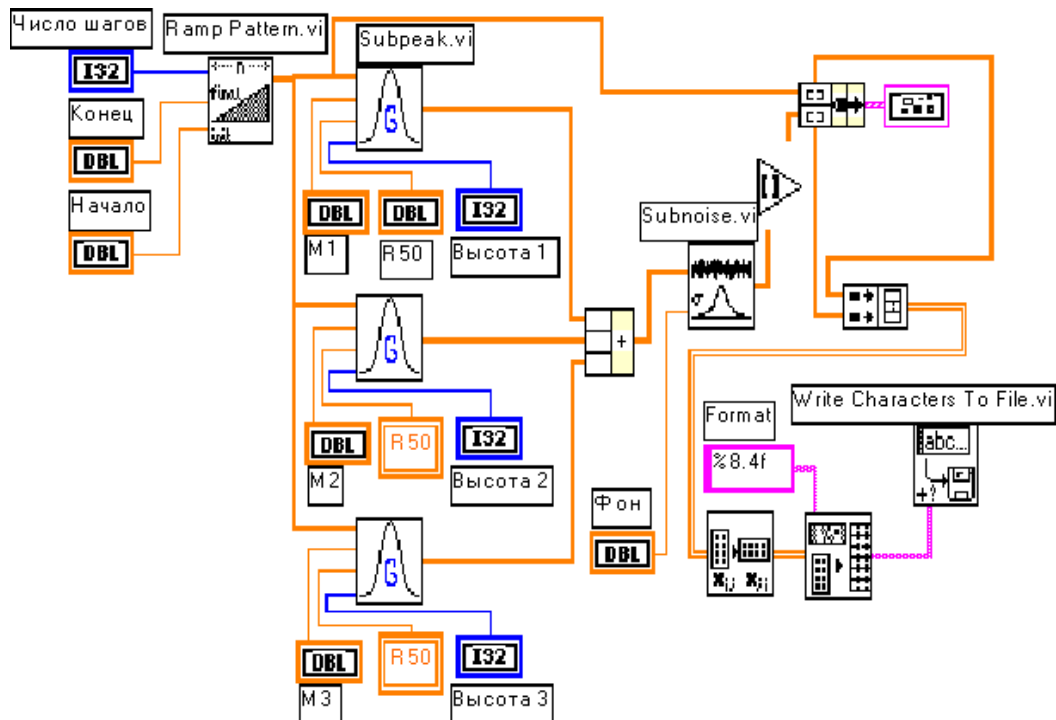


Рис. 10

ПОДПРОГРАММА (SUBVI) SUBPEAK

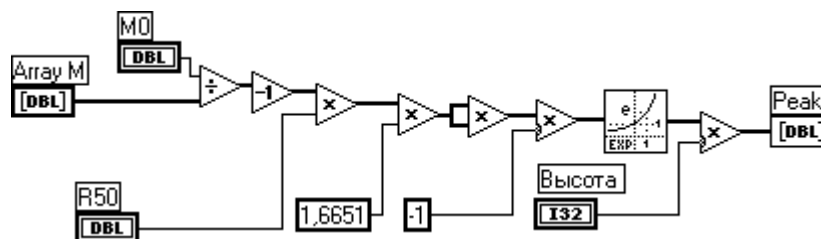


Рис. 11

ПОДПРОГРАММА (SUBVI) SUBNOISE

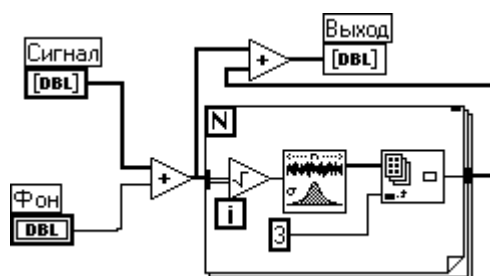


Рис. 12

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 33. ОПРЕДЕЛЕНИЕ ПАРАМЕТРОВ ПИКОВ МАСС-СПЕКТРА

Основные параметры пиков масс-спектра и точность их определения могут быть найдены нелинейным методом наименьших квадратов (МНК). Определения только положения и высоты пика, как показывает опыт, недостаточно, так как третий параметр — разрешающая способность — содержит информацию о возможном наложении пиков, а также, как будет видно из дальнейшего, о других особенностях пиков: узкие пики отраженных ионов, широкие пики метастабильных ионов и так далее.

Кроме того, разрешающая способность представляет собой некоторый аналог аппаратной функции и будучи определена по сильным пикам, может быть использована как известный параметр при обработке слабых зашумленных пиков. Последнее позволяет уменьшить число подбираемых при обработке МНК параметров и таким образом улучшить сходимость процедуры. Подобная информация может быть извлечена из определения фонового ионного тока в окрестности обрабатываемого пика. При обработке пика в пакете TableCurve пик

ПРИМЕР ОБРАБОТКИ ПИКА M=292

FitStdErr=21029.291 Fstat=1072.3422

a=-61655.056 b=556504.21 c=994.96187 d=292.03514

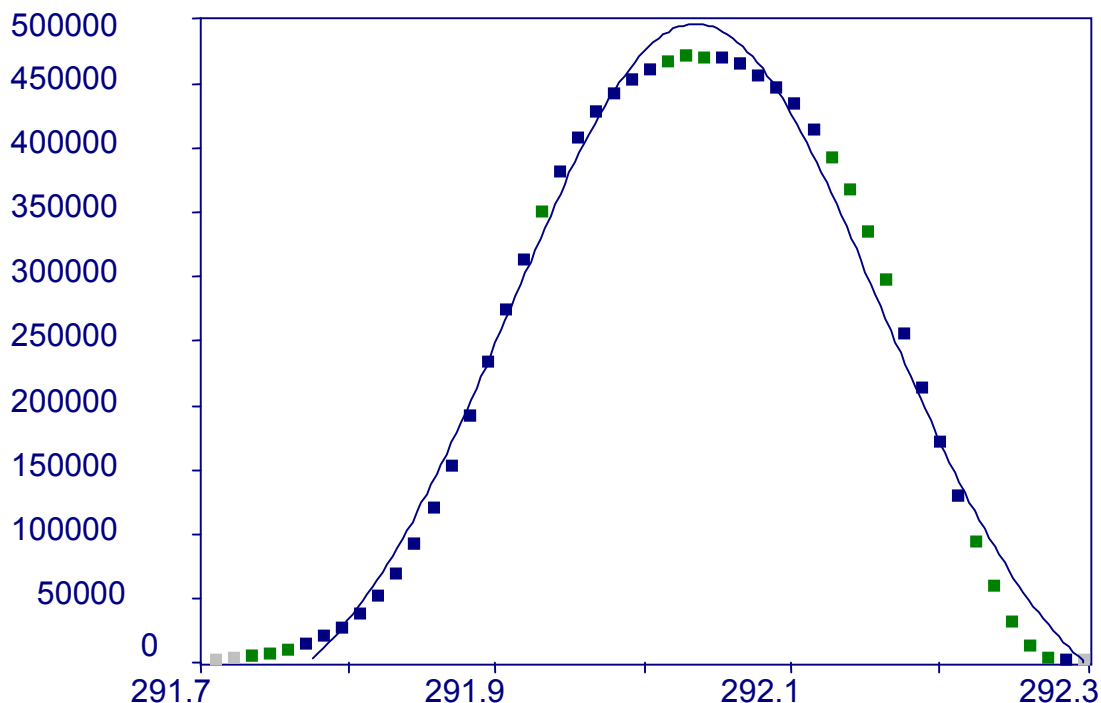


Рис. 13

вырезается из масс-спектра и обрабатывается нелинейным МНК по заданной функциональной зависимости с указанием начальных приближенных параметров. Пример обработки пика $M = 292$ приведен на рис. 3.8.

Обработка проведена по кривой Гаусса

$$Y = A + B \exp \left[-1.6651 \cdot C \cdot \left(\frac{D}{x} - 1 \right) \right]^2.$$

Видны систематические отклонения формы пика, требующие или увеличения числа подбираемых параметров, или применения методики обработки с использованием аппаратной функции.

Часть подробных результатов обработки пика $M = 292$ приведена в табл. 3.1.

В таблице табл. 3.1 кроме параметров (Parm) a , b , c , d приведены ошибки определения этих параметров (*Std Error*), величина стандартного отклонения (*Fit Std Err*), обратная величина относительной ошибки (*t-value*), а также доверительный интервал для доверительной вероятности 95% (*95% Confidence Limits*).

РЕЗУЛЬТАТЫ ОБРАБОТКИ ПИКА M=292

Rank 1 Eqn 8001 [UDF 1] y=(a,b,c,d)

r ² Coef Det	DF	Adj r ²	Fit Std Err	F-val
0.9871127161	0.9858554201	21029.290833	1072.3421721	

Parm	Value	Std Error	t-value	95% Confidence Limits	
a	-61655.0559	15398.49274	-4.00396694	-92725.1867	-30584.9251
b	556504.2133	14304.79295	38.90333926	527640.8826	585367.5440
c	994.9618699	33.08943721	30.06886649	928.1960403	1061.727700
d	292.0351360	0.001592102	183427.4387	292.0319235	292.0383484

Area Xmin-Xmax Area Precision

134626.87152 9.252565e-14

Function min X-Value Function max X-Value

-31209.35024 291.73496454 494849.13909 292.03510397

1st Deriv min X-Value 1st Deriv max X-Value

-2705684.220 292.15982052 2710307.4861 291.91045643

2nd Deriv min X-Value 2nd Deriv max X-Value

-3.58196e+07 292.03503117 1.612445e+07 291.81925170

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 34. КАЛИБРОВКА ШКАЛЫ МАСС

Основной причиной смещения шкалы масс с течением времени является, по-видимому, нестабильность тока электромагнита, максимальная величина которой достигает 10^{-4} за час. Это соответствует на массе 300 уходу шкалы масс

$$\Delta M = 2M\Delta B/B = 2,300 \cdot 10^{-4} = 60 \text{ mmi}$$

в течение часа.

С целью учета ухода магнитного поля масс-спектр гексафторида урана измерялся без остановки развертки магнитного поля в течение 2÷3 часов (шаг по шкале масс 10 импульсов, время интегрирования 1 с, диапазон масс 230÷360).

Корректировка шкалы масс проводилась с использованием реперных масс осколочных ионов UF₆.

Перед записью масс-спектра выполняется начальная установка шкалы масс, с тем чтобы даже без обработки масс-спектра можно было определять по-

ОПРЕДЕЛЕНИЕ ПОПРАВКИ К ШКАЛЕ МАСС

$r^2=0.9986296$ DF Adj $r^2=0.99780736$ FitStdErr=0.0026521285 Fstat=2186.1
 $a=1.4674244$ $b=-0.011565704$
 $c=2.1649535e-05$

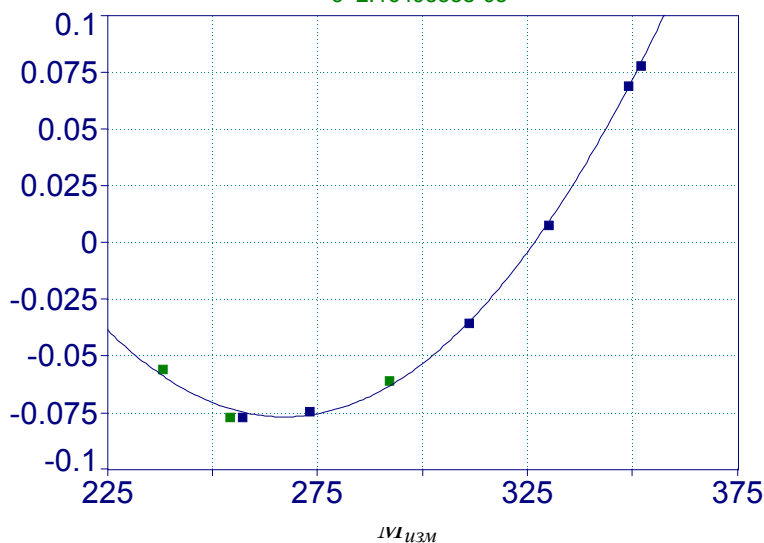


Рис. 14

ложение отдельных линий. Для этого измеряем положение двух крайних реперных линий M_1^* и M_2^* , табличные массы которых M_1 и M_2 . Измерения проводятся при первоначально установленных коэффициенте шкалы масс k^* и сдвиге шкалы масс M_0^* .

Поскольку искомые (правильные) значения коэффициента шкалы масс k и сдвига шкалы масс M_0 должны при том же самом числе импульсов регулировки N давать табличные значения масс, можем записать:

$$\begin{aligned} M_1^* &= k^* N_1^2 + M_0^* \\ M_2^* &= k^* N_2^2 + M_0^* \\ M_1 &= k N_1^2 + M_0 \\ M_2 &= k N_2^2 + M_0, \end{aligned} \quad (22)$$

откуда

$$\begin{aligned} M_1^* - M_2^* &= k^* (N_1^2 - N_2^2) \\ M_1 - M_2 &= k (N_1^2 - N_2^2), \end{aligned}$$

и

$$k = k^* \frac{M_1 - M_2}{M_1^* - M_2^*}. \quad (23)$$

Из 2-го и 4-го уравнений (22)

$$\frac{M_2^* - M_0^*}{M_2 - M_0} = \frac{M_1^* - M_2^*}{M_1 - M_2},$$

откуда

$$M_0 = \frac{M_2 M_1^* - M_1 M_2^*}{M_1^* - M_2^*} + \frac{M_0^* (M_1 - M_2)}{M_1^* - M_2^*}.$$

После записи масс-спектра положение вершины пиков $M_{изм}$ определялись аппроксимацией параболой точек спектра в 10% интервале от высоты пика. Функция зависимости поправок $\Delta M = M_{табл} - M_{изм}$, где $M_{табл}$ - табличное значение массы, обрабатывалась по квадратичной формуле и параметры этой зависимости использовались для окончательной корректировки шкалы масс масс-спектра. Пример обработки приведен на рис. 14.

4. АНАЛИЗ ПРИЧИН ВОЗНИКНОВЕНИЯ ФОНОВОГО ИОННОГО ТОКА

При проведении количественного масс-спектрометрического анализа предел обнаружения примеси, наряду с чувствительностью прибора, определяется уровнем фонового ионного тока (ФИТ) [5, стр. 186]. Установление причин возникновения этого тока позволит определить пути его снижения.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 35. ИЗМЕРЕНИЯ С ЗАДЕРЖИВАЮЩИМ ПОТЕНЦИАЛОМ

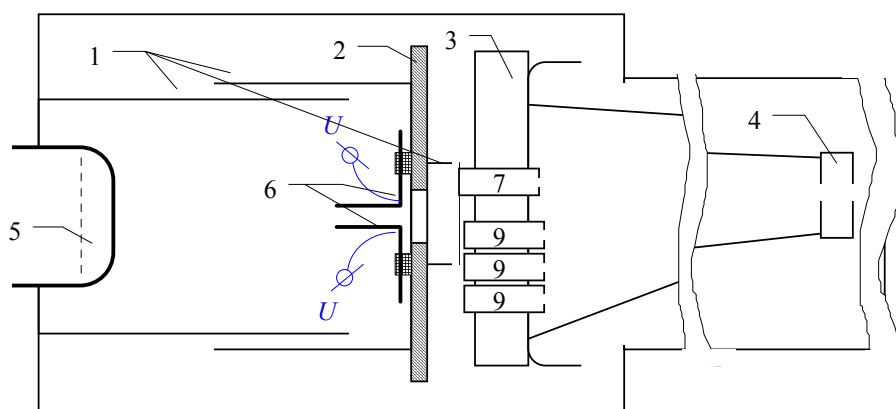
Одной из основных причин создания ФИТ в литературе [6, стр. 80] считается рассеяние ионов на молекулах остаточного газа и стенках вакуумной камеры. В предположении, что такие ионы имеют пониженную энергию, были проведены измерения с задерживающим потенциалом. С этой целью в приемнике ионов перед ВЭУ был установлен плоский конденсатор с зазором 5 мм и площадью пластин $15 \times 15 \text{ мм}^2$. Пластины конденсатора были подключены к вакуумным вводам каналов 2 и 4 измерителей ионных токов. Схема модификации приведена на рис. 4.1.

Оценим возможное отклонение иона с энергией $eU_{уск}$ в поле плоского конденсатора. Время пролета конденсатора с размером пластин l в направлении движения ионов:

$$t = \frac{l}{V},$$

где V - скорость движения ионов,

СХЕМА УСТАНОВКИ ПЛОСКОГО КОНДЕНСАТОРА ДЛЯ
СОЗДАНИЯ ЗАДЕРЖИВАЮЩЕГО ПОТЕНЦИАЛА



1 - экран; 2 - основание конденсатора; 3 - основание коллекторов; 4 - вынесенная щель; 5 - ВЭУ; 6 - пластины конденсатора; 7 - подвижный коллектор (цилиндр Фарадея); 8 - неподвижные коллекторы.

Рис. 15

$$V^2 = \frac{2U_{\text{уск}} e}{m},$$

отсюда

$$t^2 = \frac{l^2 m}{2U_{\text{уск}} e}.$$

Отклоняющая сила, действующая на ион

$$F = eE = e \frac{U_{\text{откл}}}{d},$$

где E - напряженность электрического поля в конденсаторе; $U_{\text{откл}}$ - напряжение на пластинах конденсатора; d - зазор конденсатора.

Создаваемое отклоняющей силой ускорение

$$a = \frac{eU_{\text{откл}}}{md},$$

и величина отклонения

$$S = \frac{eU_{\text{откл}} t^2}{2md}.$$

Поэтому

$$S = \frac{l^2 U_{откл}}{4d U_{уск}}.$$

При $U_{откл} \approx 0,5U_{уск}$, $l = 15 \text{ мм}$, $d = 5 \text{ мм}$, $S \approx 6 \text{ мм}$, что приведет к запира-нию луча. При оценке принято, что средний потенциал на пластинах конденса-тора равен 0 и не приводит к ускорению (замедлению) движения частиц. Таким образом, источник напряжения на $2,5 \text{ кВ}$ позволит уменьшить ионный ток до 0. Следует отметить, что в наших измерениях запираение луча за счет его отклоне-ния не использовалось, и к пластинам прикладывалось только задерживающее, положительное и одинаковое для обеих пластин, напряжение.

При проведении измерений с задерживающим потенциалом было бы удобно располагать ионами, о которых заведомо известно, что они отражены от поверхности. Такие ионы, отраженные от стенки приемника канала 5, были об-наружены при анализе формы пиков, находящихся в непосредственной близо-сти от интенсивных пиков осколочных ионов $^{238}\text{UF}_n$. Характерным признаком пиков отраженных ионов является их малая ширина ($R_{50} \approx 3000$), связанная, по-видимому, с их диафрагмированием на пути к ВЭУ, и смена положения пика при перемещении приемника канала 5 микрометрическим винтом.

На рис. 16 совмещены 4 масс-спектра, соответствующие разным поло-жениям приемника канала 5. Смещение пиков происходит в сторону уменьше-

ИЗМЕНЕНИЕ ПОЛОЖЕНИЯ ПИКА ОТРАЖЕННЫХ ИОНОВ ПРИ
ИЗМЕНЕНИИ ПОЛОЖЕНИЯ ПРИЕМНИКА КАНАЛА 5

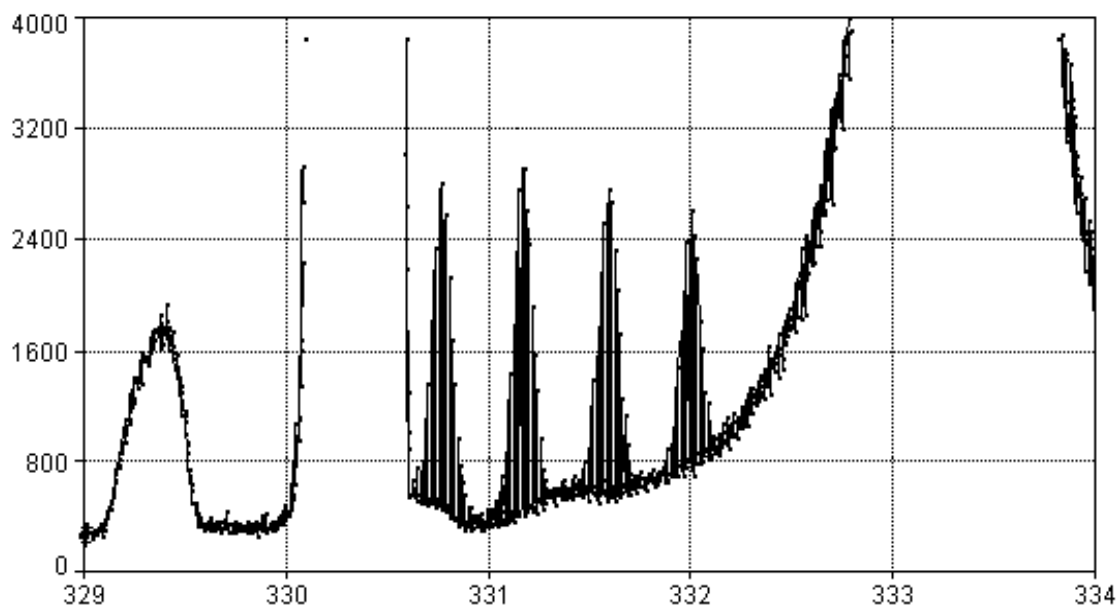
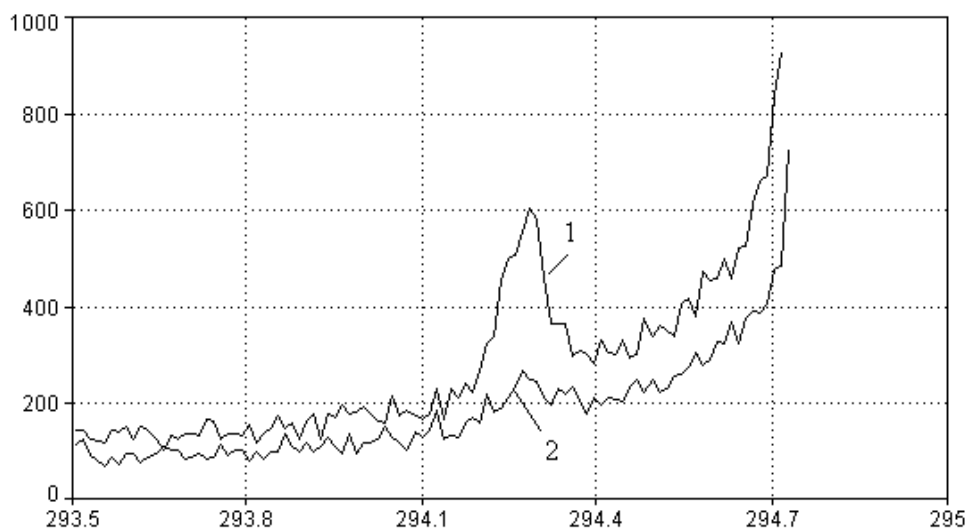


Рис. 16

ВЛИЯНИЕ ЗАДЕРЖИВАЮЩЕГО ПОТЕНЦИАЛА НА ВЫСОТУ
ПИКА ОТРАЖЕННЫХ ИОНОВ



1 - задерживающий потенциал 0 В. 2 - задерживающий потенциал 2,5 кВ.

Рис. 17

ния массы при удалении подвижного приемника канала 5 от неподвижных приемников 4, 3, 2 каналов и соответствует дисперсии прибора 1,25 мм/ае.м. Оценочные измерения показали, что задерживающий потенциал, оказывает сильное влияние на интенсивность пика отраженных ионов. При ускоряющем напряжении +5 кВ увеличение задерживающего потенциала с 0 до 2,5 кВ уменьшает высоту пика отраженных ионов в 7 раз (рис. 4.3).

В то же время измерения показали, что высота остальных пиков и величина ФИТ изменяются при увеличении задерживающего потенциала лишь на 15÷20% (это видно и из рис. 17). Таким образом, основная составляющая ФИТ имеет иную природу нежели рассеяние ионов на стенках вакуумной камеры.

ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9].

36. МЕТАСТАБИЛЬНЫЕ ИОНЫ

Метастабильные ионы присутствуют в масс-спектре UF_6 (рис. **Ошибка! Источник ссылки не найден.**) в виде уширенных пиков вблизи масс $M = 258, 277, 296, 315$ или в виде локального, в диапазоне 5÷6 единиц массы, повышения ФИТ, как например вблизи массы $M = 261$. Положение метастабильных пиков отмечено на рис. 4.4 стрелками.

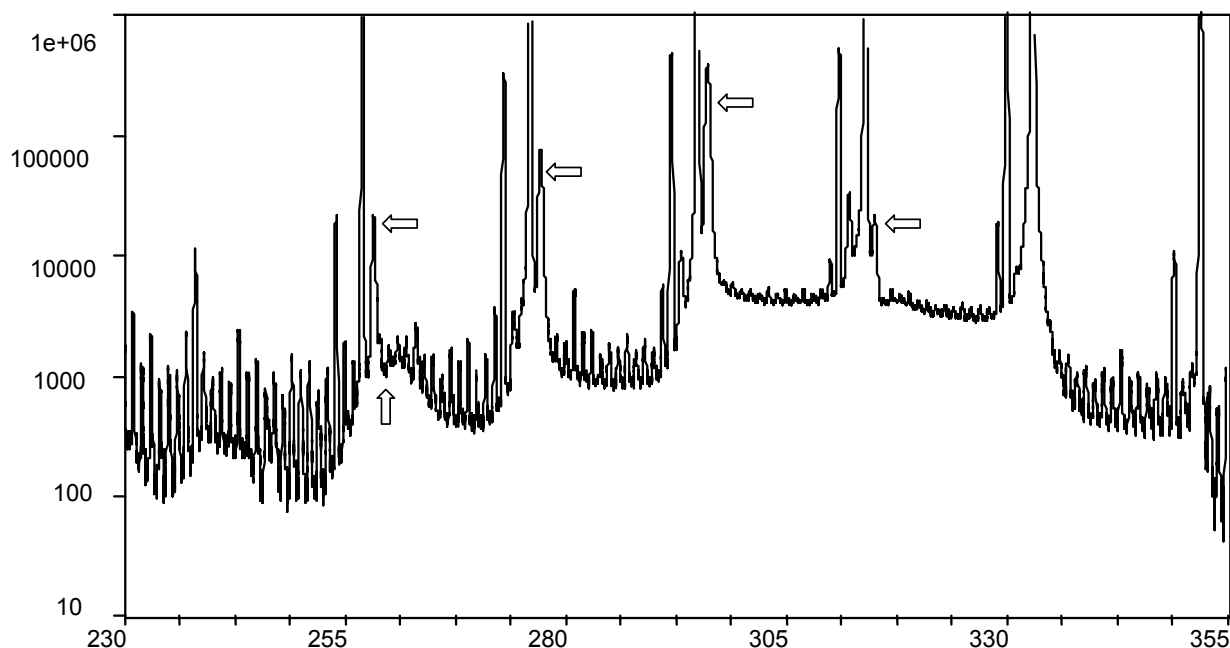
МЕТАСТАБИЛЬНЫЕ ИОНЫ В МАСС-СПЕКТРЕ UF₆

Рис. 18

Происхождение метастабильных ионов объясняется тем, что некоторая доля $\sim 0,1\%$ ионов, в том числе и осколочных, образуется в источнике ионов в возбужденном состоянии с временем жизни соизмеримым с временем пролета ионов от источника до приемника ионов.

Скорость движения ионов $^{238}\text{UF}_5$ ($M = 333$) при ускоряющем напряжении $U_{\text{уск}} = 5 \text{ кВ}$ составляет

$$V = \sqrt{\frac{2eU_{\text{уск}}}{m}} = 5.5 \cdot 10^4 \frac{\text{м}}{\text{с}}.$$

Поскольку длина пути l ионов в масс-спектрометре равна $\approx 1 \text{ м}$, то время t движения ионов от источника до приемника ионов равно

$$t = \frac{l}{V} = 18 \text{ мкс}.$$

На практике термин “метастабильные ионы” часто употребляется в отношении ионов, образующихся при распаде действительно метастабильных ионов, распадающихся между источником и коллектором [6, стр. 147]. В то же время используются термины родительские ионы и дочерние ионы [7, стр. 33].

Метастабильный ион с массой M_1 распадаясь по схеме

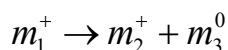


Таблица 8

ПОЛОЖЕНИЕ ДОЧЕРНЕГО ИОНА $^{238}\text{UF}_4^+$ ($M = 314$) В МАСС-СПЕКТРЕ В ЗАВИСИМОСТИ ОТ МЕСТА РАСПАДА МЕТАСТАБИЛЬНОГО ИОНА $^{238}\text{UF}_5^+$

Место распада $333^+ \rightarrow 314^+ + 19^0$	Место на спектре
Ионизационная камера	314
Ионизационная камера — выходная щель источника ионов	314÷296
Выходная щель источник ионов — вход в магнитное поле	296
Магнитное поле	296÷333
Выход из магнитного поля	333

изменяет свою энергию и как следствие кривизну траектории в магнитном поле.

Если ион распадается в ионизационной камере, то образующийся при распаде ион m_2^+ пройдет полное значение ускоряющего напряжения и будет зарегистрирован со своей действительной массой m_2 . Если ион распадается после выхода из источника ионов, но до магнитного поля, то как показывает рассмотрение [Ошибка! Закладка не определена., стр.147], он будет зарегистрирован с кажущейся массой, определяемой уравнением

$$m^* = \frac{m_2^2}{m_1}. \quad (24)$$

При распаде родительского иона после магнитного поля ион m_2^+ будет зарегистрирован как ион с массой m_1 . Наконец при распаде метастабильного иона m_1 в магнитном поле ионы m_2^+ будут распределены в масс-спектре, в соответствии с местом распада, между массами m^* и m_1 . Результаты данного рассмотрения приведены в таблице для наиболее интенсивного метастабильного пика $M = 296$, соответствующего распаду $^{238}\text{UF}_5^+ \rightarrow ^{238}\text{UF}_4^+ + \text{F}^0$ ($333^+ \rightarrow 314^+ + 19$).

Для подтверждения схемы образования пиков, отмеченных на рис. 18, положение этих пиков было сравнено с кажущимися массами, рассчитанными по соотношению (24). Результаты расчета и ошибки определения масс приведены в табл. 4.2

Высокая степень $\approx 0,01$ аем = 10 мти совпадения рассчитанных и измеренных масс свидетельствует о том, что данные пики действительно являются пиками метастабильных дочерних ионов, образующихся из родительских ионов $^{238}\text{UF}_6^+$, $^{238}\text{UF}_5^+$, $^{238}\text{UF}_4^+$, $^{238}\text{UF}_3^+$.

Таблица 9

СРАВНЕНИЕ ИЗМЕРЕННЫХ И РАССЧИТАННЫХ (КАЖУЩИХСЯ) МАСС ДОЧЕРНИХ
МЕТАСТАБИЛЬНЫХ ИОНОВ

$M_{изм}, аем$	$M_{расч}, аем$	$M_{изм} - M_{расч}, mmi$	Ошибка, mmi
258,2846	258,2725	12,0	±1,5
277,2079	277,1969	11,0	±1,6
296,1417	296,1298	11,9	±1,8
315,0728	315,0697	3,1	±1,7

Существование метастабильных ионов объясняет ранее отмеченное [2, стр. 16] резкое увеличение (8÷10 раз) фонового ионного тока при переходе через 295 массу (рис. 18). Такое увеличение происходит на самом деле при переходе через 296 массу и обусловлено, как следует из табл. 8, распадом наиболее многочисленных родительских метастабильных ионов $^{238}\text{UF}_5^+$ в магнитном поле. На масс-спектре рис. 18 также можно проследить снижение фонового ионного тока в интервале масс 296÷330, связанное с этим распадом. Особенность распада метастабильных ионов в источнике ионов заключается в том, что по мере смещения точки распада к выходной щели ионы попадают в область меньших масс. Это приводит к замедлению снижения ФИТ (при увеличении массы в масс-спектре) в диапазоне масс 296÷310.

Рассмотрение возможности распада



дает возможность предположить существование пика метастабильных (дочерних) ионов с кажущейся массой 261,3. Такой пик как уже указывалось присутствует в масс-спектре рис. **Ошибка! Источник ссылки не найден.**

**ПРОПУЩЕН НОМЕР ПРЕДЫДУЩЕГО УРОВНЯ: 1. ГЛАВА! ВСТАВЬТЕ
ЕГО И НАЖМИТЕ КЛАВИШИ: [CTRL]+A, [F9]. 37. ПУТИ
УМЕНЬШЕНИЯ ФОНОВОГО ИОННОГО ТОКА**

Поскольку ФИТ при анализе UF_6 в основном обусловлен метастабильными ионами, то наиболее эффективно снижать ФИТ уменьшая число метастабильных ионов. Схема образования метастабильных ионов (дочерних) может быть записана в соответствии с [8, стр.253]:

- 1) Образование молекулярного иона;
- 2) Образование осколочных ионов (диссоциация), часть из которых находится в метастабильном состоянии (родительские метастабильные ионы);

- 3) Распад родительских метастабильных ионов с образованием дочерних метастабильных ионов.

Поэтому можно использовать следующие пути уменьшения числа метастабильных ионов.

- 1) Увеличение времени пребывания ионов в ионизационной камере (это время составляет значительную часть продолжительности жизни ионов до их попадания на коллектор). Время пребывания в ионизационной камере может быть увеличено уменьшением потенциала выталкивающего электрода [8, стр. 261] или, предположительно, повышением потенциала вытягивающего электрода.
- 2) Снижение ионизирующего напряжения, поскольку при этом будет уменьшена доля осколочных ионов и, возможно, будет уменьшаться вероятность образования осколочных ионов в метастабильном состоянии.
- 3) Уменьшение щели приемника ионов. Широкие пики метастабильных ионов будут при этом уменьшаться более интенсивно, чем другие (узкие) пики в масс-спектре.
- 4) Использование метода задерживающего потенциала при величине потенциала, близкой к ускоряющему напряжению. В литературе [8, стр.445] упоминается о использовании электрода для подавления метастабильных ионов, значительно понижающего энергию ионов, достигающих коллектора.

При реализации некоторых вышеперечисленных направлений было рассмотрено влияние величины ионизирующего напряжения (U_i) на высоту дочерних метастабильных пиков 277 и 296, определяющих фон в диапазонах масс 277-290 и 297-329, соответственно. В табл. 4.3 приведены данные измерений зависимости величины пиков 330, 311, 296, 277, а также их отношений, от величины U_i . Пики 330 и 311 выбраны как пропорциональные по высоте родительским пикам 333 и 314.

Величина фокусирующего напряжения при этих измерениях не менялась. Из данных, приведенных в табл. 10, видно, что при снижении напряжения ионизации уменьшается отношение пиков 330/311 — уменьшается доля осколочных ионов, часть из которых составляют родительские метастабильные ионы, создающие фон в диапазоне масс 277÷291. Наиболее существенный результат заключается в том, что при снижении ионизирующего напряжения до 30 В резко снижается вероятность образования осколочных ионов 314 в метастабильном состоянии (последняя колонка табл. 10). Некоторый вклад в снижение величины пика $M=277$ может дать “неоптимальность” времени жизни родительских метастабильных ионов с массой 314, поскольку, как отмечалось выше, как малые так и большие времена жизни родительских ионов уменьшают величину соответствующих им метастабильных пиков.

Для более детального рассмотрения влияния напряжения ионизации U_i на величину отдельных пиков и ФИТ были записаны и сравнены масс-спектры при ионизирующих напряжениях 30,1 В и 50,0 В (рис. 6). Сравнение подтверждает сделанные по данным табл. 10 выводы о снижении при уменьшении U_i доли осколочных ионов и уменьшение вероятности их образования в метастабильном состоянии. Например, метастабильные дочерние ионы с массой 258 в масс-спектре при $U_i=30,1$ В отсутствуют. Это позволяет более четко увидеть широкий метастабильный пик с положением максимума на массе $M=261$.

При $U_i=30,1$ В и узкой выходной щели масс-спектрометра (входной щели приемника ионов) для стандартного образца с содержанием ^{235}U 0,72% зарегистрированы величина пика $M=330$, составляющая 417250 ионов в секунду, и фон в области 285 пика 20 ионов в секунду. Предел обнаружения примеси, обусловленный статическим шумом, рассчитанный по формуле [3, с.12]

$$\Delta C_n = \frac{\sqrt{2N_\phi}}{N_c} C,$$

Таблица 10

ВЛИЯНИЕ ВЕЛИЧИНЫ ИОНИЗИРУЮЩЕГО НАПРЯЖЕНИЯ НА ВЕЛИЧИНЫ РОДИТЕЛЬСКИХ И ДОЧЕРНИХ МЕТАСТАБИЛЬНЫХ ИОНОВ

U_i , В	Пик 330, 10^3 ионов/с	Пик 311, 10^3 ионов/с	Пик 296, 10^3 ионов/с	Пик 277, 10^3 ионов/с	Отноше- ние пиков 330/311	Отноше- ние пиков 330/296	Отноше- ние пиков 311/277
40,1	970	302	185	38,8	3,2	5,2	7,8
35,0	838	215	134	17,9	3,9	6,2	12,0
30,0	512	80	53	3,0	6,4	9,7	27

составил

$$\frac{\sqrt{2 \cdot 20}}{4,2 \cdot 10^5} \cdot 0,0072 = 1,1 \cdot 10^{-7}.$$

При измерении содержания WF_6 это значение предела обнаружения снижается с учетом коэффициента относительной чувствительности, равного 0,7 в 1,5 раза и может быть дополнительно уменьшено при увеличении длительности измерений. Анализ микропримесей с данным пределом обнаружения может быть реализован либо при отсутствии в масс-спектре фоновых линий, либо при достаточно высокой разрешающей способности масс-спектрометра.

ВЫВОДЫ

- 1) Разработан и протестирован драйвер МИ-1201 АГМ для Windows NT, включающий в себя и виртуальный драйвер DOS. Драйвер позволяет работать с масс-спектрометром в ОС Windows NT и запускать в ОС Windows NT старые программы для ОС DOS.
- 2) Предложена математическая модель пика масс-спектра и реализован алгоритм обработки пика методом наименьших квадратов.
- 3) Предложена методика математической обработки масс-спектра с вычислением методом наименьших квадратов положения, высоты и формы отдельных пиков, а также ошибок определения этих величин. Погрешность определения положения отдельных пиков составила ± 2 *mtu* в диапазоне масс 238÷352.
- 4) Использование метода задерживающего потенциала показало, что отраженные от поверхности ионы могут быть подавлены до 15% своего начального уровня задерживающим напряжением 2,5 *kB*, тогда как остальные пики и фон рассеянных ионов изменяются незначительно.
- 5) Показано, что основной вклад в фоновый ионный ток дают дочерние метастабильные ионы, порождаемые интенсивными пиками осколочных ионов гексафторида урана.
- 6) Снижение ионизирующего напряжения, уменьшающего долю осколочных ионов, а также вероятность образования осколочных ионов в метастабильном состоянии позволило снизить предел обнаружения примесей в диапазоне масс 279÷285 до $1,1 \cdot 10^{-5}$ %.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Анализ поверхности методами оже- и рентгеновской фотоэлектронной спектроскопии / Пер. с английского. Под ред. Д. Бриггса, М.П. Сиха-М.: Мир, 1987. 600 с.
2. Александров О.Е., Атанов В.Е., Калинин Б.А. Разработка методики прямого масс-спектрометрического анализа микропримесей на уровне $10^{-5}\%$ в технологических газовых смесях: Отчет по научно-исследовательской работе. Екатеринбург: УГТУ, 1999. с. 21.
3. Александров О.Е., Атанов В.Е., Калинин Б.А. Разработка аппаратно-программных средств повышения чувствительности масс-спектрометрического анализа: Отчет по научно-исследовательской работе. Екатеринбург: УГТУ, 2000г.
4. Щиголев Б.М. Математическая обработка наблюдений. М.:Наука, 1969. 344 с.
5. Чепмен Дж. Практическая органическая масс-спектрометрия. / Пер. с англ.- М.: Мир.,1988.-216 с.
6. Сысоев А.А., Чупахин М.С. Ββεδενθє β μῶρρ-ροєκσπξμεσπθῶ. М.: Атомиздат, 1977. 304 с.
7. Полякова А.А. Молекулярный масс-спектральный анализ органических соединений. М.:Химия, 1983. 248с.
8. Бейнон Дж. Масс-спектрометрия и ее применение в органической химии. / Пер.с англ. М.: Химия, 1983. 248с.