

Содержание

Предисловие	10
Об этой книге	12
ЧАСТЬ 1. ВВЕДЕНИЕ	15
Глава 1. Начнем сначала	16
1.1. Смарт-карты	16
1.1.1. Краткая история	17
1.1.2. Преимущества смарт-карт	17
1.1.3. Области применения	18
1.2. Проблемы разработки приложений для смарт-карт	20
1.3. Использование технологии Java для смарт-карт	21
1.3.1. Преимущества технологии Java Card	21
1.3.2. Краткая история технологии Java Card	22
Глава 2. Введение в мир смарт-карт	24
2.1. Обзор технологии смарт-карт	24
2.2. Основные типы смарт-карт	25
2.2.1. Сравнение карт памяти и карт со встроенным микропроцессором	25
2.2.2. Сравнение контактных и бесконтактных карт	26
2.3. Аппаратные средства смарт-карт	27
2.3.1. Контакты смарт-карты	27
2.3.2. Центральный процессор смарт-карты	28
2.3.3. Сопроцессоры смарт-карт	28
2.3.4. Система памяти смарт-карт	28
2.4. Взаимодействие со смарт-картами	30
2.4.1. Устройства считывания карт и хост-приложения	30
2.4.2. Модель взаимодействия со смарт-картами	30
2.4.3. Протокол APDU	31
2.4.4. Протокол TPDU	33
2.4.5. Сообщение ATR	33
2.5. Операционные системы смарт-карт	33
2.5.1. Файловые системы смарт-карт	34
2.5.2. Корневой каталог (Master File)	35
2.5.3. Каталог (Dedicated File)	35
2.5.4. Файл данных (Elementary File)	35
2.6. Программное обеспечение смарт-карт	35
2.7. Стандарты и спецификации смарт-карт	36
2.7.1. Стандарты ISO 7816	37
2.7.2. Стандарты GSM	37
2.7.3. Спецификация EMV	37
2.7.4. Спецификации Open Platform	38
2.7.5. Интегрированная среда OpenCard Framework	38
2.7.6. Спецификации PC/SC	38
ЧАСТЬ 2. ТЕХНОЛОГИЯ JAVA CARD	39
Глава 3. Обзор технологии Java Card	40
3.1. Обзор архитектуры	40
3.2. Подмножество языка Java Card	41
3.3. Виртуальная машина Java Card	42

3.3.1. Файлы CAP и экспортные файлы	42
3.3.2. Конвертер Java Card	43
3.3.3. Интерпретатор Java Card	44
3.4. Инсталлятор Java Card и внешняя инсталляционная программа	45
3.5. Среда исполнения Java Card	46
3.5.1. Время существования JCRE	47
3.5.2. Как работает JCRE во время сеанса связи с устройством считывания	48
3.5.3. Дополнительные возможности среды исполнения Java Card	49
3.6. API, поддерживаемые платформой Java Card	49
3.6.1. Пакет <code>java.lang</code>	50
3.6.2. Пакет <code>javacard.framework</code>	50
3.6.3. Пакет <code>javacard.security</code>	51
3.6.4. Пакет <code>javacardx.crypto</code>	51
3.7. Апплеты Java Card	52
3.8. Соглашение о присваивании имен пакетам и апплетам	52
3.9. Процесс разработки апплета	53
3.10. Инсталляция апплета	55
3.10.1. Апплеты ROM	55
3.10.2. Предопределенные и загружаемые апплеты	55
3.10.3. Инсталляция загружаемых апплетов	56
3.10.4. Обработка ошибок в процессе инсталляции апплета	57
3.10.5. Ограничения инсталляции	57
Глава 4. Объекты Java Card	58
4.1. Модель памяти Java Card	58
4.2. Постоянные объекты	59
4.3. Временные объекты	60
4.3.1. Свойства временных объектов	60
4.3.2. Типы временных объектов	61
4.3.3. Создание временных объектов	62
4.3.4. Запросы к временным объектам	62
4.4. Кратко о создании и удалении объектов	63
Глава 5. Атомарность и транзакции	64
5.1. Атомарность	64
5.2. Атомарное обновление блока данных в массиве	65
5.3. Транзакции	65
5.3.1. Фиксация транзакции	66
5.3.2. Прерывание транзакции	66
5.3.3. Вложенные транзакции	67
5.3.4. Размер буфера транзакций	67
5.3.5. <code>TransactionException</code>	68
5.3.6. Изменение значений локальных переменных и временных объектов в процессе выполнения транзакции	68
Глава 6. Исключения Java Card и их обработка	71
6.1. Исключения в пакете <code>java.lang</code>	71
6.2. Исключения Java Card	72
6.2.1. Код причины исключений Java Card	73
6.2.2. Возбуждение исключений в Java Card	73
6.2.3. <code>ISOException</code>	74
6.2.4. <code>UserException</code>	75

Глава 7. Апплеты Java Card	76
7.1. Обзор апплетов	76
7.1.1. Инсталляция и выполнение апплетов	76
7.1.2. Взаимодействие хост-системы и апплетов	77
7.2. Класс <code>javacard.framework.Applet</code>	77
7.3. Метод <code>install</code>	78
7.3.1. Создание объектов в конструкторе апплета	80
7.3.2. Регистрация экземпляра апплета в JCRE	80
7.3.3. Обработка параметров инсталляции	81
7.3.4. Дополнительная инициализация апплетов	83
7.4. Метод <code>select</code>	83
7.4.1. Формат и обработка команды <code>SELECT APDU</code>	84
7.4.2. Апплет по умолчанию	85
7.5. Метод <code>deselect</code>	85
7.6. Метод <code>process</code>	86
7.7. Другие методы класса <code>javacard.framework.Applet</code>	86
Глава 8. Работа с пакетами данных APDU	88
8.1. Класс APDU	88
8.1.1. Объект APDU	89
8.1.2. Размер буфера APDU	89
8.2. Интерфейс ISO7816	89
8.3. Работа с пакетами APDU в апплетах	90
8.3.1. Получение ссылки на буфер APDU	90
8.3.2. Проверка заголовка команды APDU	90
8.3.3. Извлечение данных из команды APDU	91
8.3.3.1. Получение больших блоков данных	92
8.3.4. Обработка команды APDU и генерация ответных данных	94
8.3.5. Отправка ответных данных APDU	94
8.3.5.1. Отправка данных из других местоположений	96
8.3.5.2. Отправка больших ответных блоков данных	97
8.3.6. Возвращение слова состояния	98
8.4. Способы обработки команд APDU, зависящие от протокола	99
8.4.1. Метод <code>getProtocol</code>	100
8.4.2. Метод <code>getInBlockSize</code>	100
8.4.3. Метод <code>getOutBlockSize</code>	101
8.4.4. Метод <code>setOutgoingNoChaining</code>	102
8.4.5. Метод <code>getNAD</code>	102
8.4.6. Метод <code>waitExtension</code>	102
8.5. Выводы	103
Глава 9. Брандмауэр апплетов и совместное использование объектов	105
9.1. Брандмауэр апплетов	105
9.1.1. Контексты	106
9.1.2. Владение объектами	107
9.1.3. Доступ к объектам	107
9.1.4. Контексты и доступ к временным массивам	108
9.1.5. Статические поля и методы	108
9.2. Использование объектов разными контекстами	109
9.2.1. Переключение контекстов	109
9.2.2. Привилегии JCRE	110
9.2.3. Объекты—точки входа в JCRE	110

9.2.4. Глобальные массивы	111
9.2.5. Интерфейсный механизм совместного использования объектов	112
9.2.5.1. Интерфейс совместного использования	112
9.2.5.2. Объект интерфейса совместного использования	112
9.2.5.3. Принципы действия интерфейсного механизма совместного использования объектов	113
9.2.5.4. Пример совместного использования объекта разными апплетами	114
9.2.5.5. Создание объекта интерфейса совместного использования	115
9.2.5.6. Запрос объекта интерфейса совместного использования	116
9.2.5.7. Применение объекта интерфейса совместного использования	117
9.2.5.8. Переключения контекстов в процессе совместного использования объектов	119
9.2.5.9. Типы параметров и результатов методов интерфейса совместного использования	120
9.2.5.10. Аутентификация клиентского апплета	121
9.2.5.11. Метод <code>getPreviousContextAID</code>	124
9.2.5.12. Выводы	125
Глава 10. Программирование криптографических функций	127
10.1. Введение в криптографию	127
10.1.1. Шифрование и расшифровка	128
10.1.2. Дайджест сообщения	130
10.1.3. Цифровая подпись	131
10.1.4. Случайные данные	132
10.2. Использование криптографии в приложениях для смарт-карт	133
10.2.1. Обеспечение безопасности приложений	133
10.2.2. Использование смарт-карт в качестве защищенного средства идентификации	134
10.2.3. Выводы	134
10.3. Криптографические API, поддерживаемые платформой Java Card	135
10.3.1. Принципы разработки	135
10.3.2. Архитектура	135
10.3.3. Структура пакетов	137
10.4. Примеры программ	138
10.4.1. Вычисление дайджеста сообщения	138
10.4.2. Создание криптографического ключа	140
10.4.3. Создание и проверка цифровой подписи	142
10.4.4. Шифрование и расшифровка данных	144
10.4.5. Генерация случайных данных	145
Глава 11. Безопасность платформы Java Card	147
11.1. Средства безопасности платформы Java Card	147
11.1.1. Средства безопасности языка Java	147
11.1.2. Дополнительные средства обеспечения безопасности платформы Java Card	148
11.2. Механизмы безопасности платформы Java Card	149
11.2.1. Проверка в процессе компиляции	149
11.2.2. Проверка файлов классов и проверка соответствия подмножеству языка	150
11.2.3. Проверка файлов CAP и экспортных файлов	151

11.2.4. Проверка при инсталляции	153
11.2.5. Криптографическая защита цепочки создания апплетов	155
11.2.6. Обеспечение безопасности во время выполнения	155
11.2.7. Криптографическая поддержка платформы Java Card	157
11.3. Защита апплетов	157
ЧАСТЬ 3. РУКОВОДСТВО ДЛЯ ПРОГРАММИСТА И ПРИЕМЫ ПРОГРАММИРОВАНИЯ	159
Глава 12. Пошаговое руководство по разработке апплета	160
12.1. Проектирование апплета	160
12.1.1. Определение функций апплета	160
12.1.2. Определение AID для апплета	161
12.1.3. Определение структуры классов и функций методов апплета	161
12.1.4. Определение интерфейса между апплетом и его хост-приложением	162
12.1.4.1. SELECT APDU	163
12.1.4.2. VERIFY APDU	164
12.1.4.3. CREDIT APDU	164
12.1.4.4. DEBIT APDU	164
12.1.4.5. GET BALANCE APDU	165
12.2. Разработка кода апплета	165
12.2.1. Пример кода апплета «электронный кошелек»	165
12.2.2. Реализация контроля за ошибками в апплетах	170
12.3. Что дальше?	171
Глава 13. Оптимизация апплетов	172
13.1. Общая оптимизация дизайна апплета	172
13.2. Быстродействие апплета	173
13.3. Вызов методов	173
13.4. Создание объектов в апплетах	173
13.5. Повторное использование объектов	174
13.6. Устранение избыточного кода	175
13.7. Доступ к массивам	176
13.8. Сравнение операторов switch и if-else	177
13.9. Арифметические операторы	179
13.10. Оптимизация переменных в апплетах	179
Глава 14. Работа с типом данных int	181
14.1. 32-разрядные арифметические операции	181
14.2. Размеры и индексы массива	191
14.3. Хранение и вычисление данных типа int	191
14.4. Выводы	195
ЧАСТЬ 4. ПРИЛОЖЕНИЯ	197
Приложение А. Подмножество языка Java Card	198
Приложение В. Интерфейс прикладного программирования Java Card 2.1	205
Глоссарий	338
Библиография	341