ГОСТ Р МЭК 821—2000

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

МАГИСТРАЛЬ МИКРОПРОЦЕССОРНЫХ СИСТЕМ ДЛЯ ОБМЕНА ИНФОРМАЦИЕЙ РАЗРЯДНОСТЬЮ ОТ 1 ДО 4 БАЙТОВ (МАГИСТРАЛЬ VME)

Издание официальное

151-2001
1

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИИ

БЗ 5—98/875

ГОССТАНДАРТ РОССИИ
Москва

Предисловие

1 РАЗРАБОТАН Нижегородским научно-исследовательским приборостроительным институтом «Кварц» (ННИПИ «Кварц») и Научно-исследовательским институтом стандартизации и унификации (НИИСУ)

ВНЕСЕН Всероссийским научно-исследовательским институтом «Эталон»

2 ПРИНЯТ И ВВЕДЕН В ДЕЙСТВИЕ Постановлением Госстандарта России от 4 апреля 2000 г. № 90-ст

3 Настоящий стандарт содержит полный аутентичный текст международного стандарта МЭК 821-91 «Магистраль микропроцессорных систем для обмена информацией разрядностью от 1 до 4 байтов (Магистраль VME)»

4 ВВЕДЕН ВПЕРВЫЕ

© ИПК Издательство стандартов, 2000

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Госстандарта России

Содержание

1 Область применения . . . 1
1.1 Назначение стандарта магистрали VME . . . 1
1.2 Элементы интерфейсной системы . . . 1
1.2.1 Основные определения . . . 1
1.2.1.1 Термины, используемые для описания механических конструкций магистрали VME . . . 1
1.2.1.2 Термины, используемые для описания функциональной структуры магистрали VME . . . 2
1.2.1.3 Типы циклов магистрали VME . . . 4
1.2.2 Основная структура магистрали VME . . . 5
1.3 Диаграммы, используемые в стандарте магистрали VME . . . 5
1.4 Терминология . . . 7
1.4.1 Состояния сигнальных линий . . . 8
1.4.2 Использование звездочки (*) . . . 8
1.5 Технические требования к протоколу . . . 8
1.5.1 Взаимосвязанные сигналы магистрали . . . 9
1.5.2 Широковещательные сигналы магистрали . . . 10
1.6 Примеры функционирования систем и пояснения . . . 10
2 Шина пересылки данных магистрали VME . . . 10
2.1 Введение . . . 10
2.2 Линии шины пересылки данных . . . 12
2.2.1 Линии адресации . . . 12
2.2.2 Линии модификатора адреса . . . 13
2.2.3 Линии данных . . . 16
2.2.4 Линии управления шины пересылки данных . . . 16
2.2.4.1 Линия AS* . . . 16
2.2.4.2 Линии DS0* и DS1* . . . 16
2.2.4.3 Линия DTACK* . . . 17
2.2.4.4 Линия BERR* . . . 17
2.2.4.5 Линия WRITE* . . . 17
2.3 Функциональные блоки шины пересылки данных . . . 17
2.3.1 Задатчик . . . 17
2.3.2 Исполнитель . . . 19
2.3.3 Шинный таймер . . . 20
2.3.4 Адресный монитор . . . 21
2.3.5 Режимы адресации . . . 22
2.3.6 Основные функциональные возможности пересылки данных . . . 23
2.3.7 Возможности блочной пересылки . . . 25
2.3.8 Функциональная возможность Чтение—Модификация—Запись . . . 27
2.3.9 Возможности невыровненных пересылок . . . 28
2.3.10 Возможность использования цикла Только Адрес . . . 30
2.3.11 Взаимодействие между функциональными блоками шины пересылки данных . . . 30
2.4 Типичные примеры работы . . . 31
2.4.1 Типичные циклы пересылки данных . . . 32
2.4.2 Конвейерная адресация . . . 36
2.5 Получение права на использование шины пересылки данных . . . 37
2.6 Правила и замечания по временны́м соотношениям сигналов шины пересылки данных . . . 38
3 Шина арбитража . . . 72
3.1 Основные принципы арбитража шины . . . 72
3.1.1 Типы арбитража . . . 72
3.2 Линии шины арбитража . . . 74
3.2.1 Линии запроса и предоставления шины . . . 75
3.2.2 Линия сигнала занятости шины (BBSY*) . . . 75
3.2.3 Линия очистки шины (BCLR*) . . . 75

3.3 Функциональные блоки . . . 76
3.3.1 Арбитр . . . 78
3.3.2 Запросчик . . . 79
3.3.3 Задатчик шины пересылки данных . . . 80
3.3.3.1 Освобождение шины пересылки данных . . . 80
3.3.3.2 Получение права на использование шины пересылки данных . . . 81
3.3.3.3 Дополнительная информация . . . 81
3.4 Типичные примеры работы . . . 81
3.4.1 Арбитраж запросов шины двух различных уровней . . . 81
3.4.2 Арбитраж двух запросов шины на одной линии запроса . . . 85
3.5 Условия возникновения гонок между сигналами запроса задатчика и предоставления шины арбитра . . . 89
4 Шина приоритетных прерываний . . . 91
4.1 Введение . . . 91
4.1.1 Системы прерываний с одним обработчиком . . . 91
4.1.2 Распределенные системы прерываний . . . 91
4.2 Линии шины приоритетных прерываний . . . 91
4.2.1 Линии запроса прерывания . . . 91
4.2.2 Линия подтверждения прерывания . . . 93
4.2.3 Цепочка подтверждения прерывания (IACKIN*/IACKOUT*) . . . 93
4.3 Функциональные блоки шины приоритетных прерываний. Общее описание . . . 94
4.3.1 Обработчики прерываний . . . 94
4.3.2 Прерыватель и его функции . . . 97
4.3.3 Формирователь цепочки подтверждения прерывания . . . 99
4.3.4 Возможности обработки прерывания . . . 100
4.3.5 Возможности запроса прерывания . . . 100
4.3.6 Возможности пересылки информации статуса/идентификации . . . 100
4.3.7 Возможность снятия запросов прерываний . . . 101
4.3.8 Взаимодействие между функциональными блоками шины приоритетных прерываний 103
4.4 Типичные примеры работы . . . 104
4.4.1 Работа с одним обработчиком прерываний . . . 105
4.4.2 Работа распределенных систем прерывания . . . 105
4.4.2.1 Распределенные системы прерывания с семью обработчиками прерываний . 105
4.4.2.2 Распределенные системы прерывания с двумя—шестью обработчиками . . 105
4.4.3 Типичный пример работы системы прерывания с одним обработчиком . . . 105
4.4.4 Пример назначения приоритетов двум прерываниям в распределенной системе прерывания . . . 110
4.5 Условия возникновения гонок . . . 112
4.6 Правила и замечания по временным соотношениям сигналов шины приоритетных прерываний . . . 113
5 Служебная шина . . . 132
5.1 Введение . . . 132
5.2 Сигнальные линии служебной шины . . . 132
5.3 Функциональные блоки служебной шины . . . 132
5.3.1 Формирователь системного тактового сигнала . . . 132
5.3.2 Формирователь тактового сигнала последовательной магистрали . . . 132
5.3.3 Блок контроля питания . . . 132
5.4 Инициализация и диагностика системы . . . 135
5.5 Контакты питания . . . 137
5.6 Резервная линия . . . 138
6 Электрические характеристики магистрали VME . . . 138
6.1 Введение . . . 138
6.2 Распределение питания . . . 139
6.2.1 Технические требования к источникам питания постоянного тока . . . 139
6.2.2 Электрические параметры гнездовых и штыревых соединителей . . . 139

6.3 Электрические характеристики сигналов . . . 140
6.4 Требования к характеристикам формирователей и приемников сигналов магистрали . . . 140
6.4.1 Определение типов формирователей магистрали . . . 141
6.4.2 Требования к характеристикам формирователей и нагрузок для всех линий . . . 141
6.4.2.1 Требования к характеристикам формирователей с тремя состояниями и нагрузок для сильноточных линий AS*, DS0*, DS1* . . . 141
6.4.2.2 Требования к характеристикам формирователей с тремя состояниями и нагрузок для стандартных линий A01—A31, D00—D31, AM0—AM5, IACK*, LWORD*, WRITE* . . . 142
6.4.2.3 Требования к характеристикам формирователей с двумя состояниями и нагрузок для сильноточных линий SERCLK, SYSCLK, BCLR* . . . 142
6.4.2.4 Требования к характеристикам формирователей с двумя состояниями и нагрузок для стандартных линий BG0OUT*—BG3OUT*/BG0IN*—BG3IN*, IACKOUT*/IACKIN* . . . 143
6.4.2.5 Требования к характеристикам формирователей с открытым коллектором и нагрузок для линий BR0*—BR3*, BBSY*, IRQ1*—IRQ7*, DTACK*, BERR*, SYSFAIL*, SYSRESET*, ACFAIL* и IACK* . . . 143
6.5 Межсоединения сигнальных линий объединительной платы . . . 144
6.5.1 Оконечные нагрузки . . . 144
6.5.2 Волновое сопротивление . . . 145
6.5.3 Дополнительные сведения . . . 147
6.6 Сигналы, определяемые пользователем . . . 148
6.7 Формирователи и оконечные нагрузки сигнальных линий . . . 148
7 Требования к механическим конструкциям . . . 149
7.1 Введение . . . 149
7.2 Печатные платы модулей магистрали VME . . . 150
7.2.1 Платы модулей одинарной высоты . . . 150
7.2.2 Платы модулей двойной высоты . . . 151
7.2.3 Соединители печатных плат модулей . . . 151
7.2.4 Компоновка модуля . . . 151
7.2.5 Ширина модуля . . . 152
7.2.6 Коробление печатной платы модуля, длина выводов и высота элементов . . . 152
7.3 Передние панели . . . 152
7.3.1 Рукоятки . . . 152
7.3.2 Крепление передних панелей . . . 153
7.3.3 Размеры передней панели . . . 153
7.3.4 Панели-заглушки . . . 153
7.3.5 Эжекторы и инжекторы модулей . . . 154
7.4 Объединительные платы . . . 154
7.4.1 Требования к размерам объединительной платы . . . 154
7.4.2 Оконечные нагрузки сигнальных линий . . . 155
7.5 Сборка каркасов магистрали VME . . . 155
7.5.1 Каркасы и ширина гнезд . . . 155
7.5.2 Размеры каркаса . . . 155
7.6 Соединители объединительных плат и модулей магистрали VME . . . 175
7.6.1 Распределение сигналов по контактам соединителей J1/P1 . . . 175
7.6.2 Распределение сигналов по контактам соединителей J2/P2 . . . 175

Приложение A Словарь специальных терминов, используемых для описания магистрали VME . 177
Приложение B Описание сигналов на выходах соединителей магистрали VME . . . 180
Приложение C Использование линий SERCLK и SERDAT* . . . 183
Приложение D Метастабильность и ресинхронизация . . . 185
Приложение E Допустимые поднаборы возможностей . . . 198
Приложение F Библиография . . . 203

Перечень рисунков

1.1 Элементы системы, определенные настоящим стандартом 2
1.2 Функциональные блоки и шины, определяемые настоящим стандартом 6
1.3 Система обозначений, применяемых при изображении временнóй диаграммы сигналов . . 9
2.1 Схема взаимодействия шины пересылки данных с функциональными блоками . . . 11
2.2 Схема взаимодействия задатчика с магистралью 18
2.3 Схема взаимодействия исполнителя с магистралью 19
2.4 Схема взаимодействия шинного таймера с магистралью 20
2.5 Схема взаимодействия адресного монитора с магистралью 21
2.6 Четыре способа размещения в памяти 32 разрядов данных 28
2.7 Четыре способа размещения в памяти 16 разрядов данных 29
2.8 Пример однобайтового цикла считывания 32
2.9 Пример двухбайтового цикла записи 34
2.10 Пример четырехбайтового цикла записи 35
2.11 Последовательность смены задатчика шины пересылки данных 37
2.12 Временнáя диаграмма широковещательной пересылки адреса. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР (Все типы циклов) 53
2.13 Временнáя диаграмма широковещательной пересылки адреса. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Пересылка одного четного байта; пересылка одного нечетного байта; двухбайтовые пересылки; четырехбайтовые пересылки; невыровненные пересылки) 54
2.14 Временнáя диаграмма широковещательной пересылки адреса. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Однобайтовые блочные пересылки; двухбайтовые блочные пересылки; четырехбайтовые блочные пересылки) 55
2.15 Временнáя диаграмма широковещательной пересылки адреса. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Однобайтовые циклы Чтение—Модификация—Запись; двухбайтовые циклы Чтение—Модификация—Запись; четырехбайтовые циклы Чтение—Модификация—Запись) 56
2.16 Временнáя диаграмма пересылки данных. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Считывание байта (0); считывание байта (1); считывание байта (2); считывание байта (3); считывание байтов (0—2); считывание байтов (1—3); однобайтовое блочное считывание) 57
2.17 Временнáя диаграмма пересылки данных. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Считывание байтов (0—1); считывание байтов (2—3); считывание байтов (0—3); считывание байтов (1-2); двухбайтовое блочное считывание; четырехбайтовое блочное считывание) 59
2.18 Временнáя диаграмма пересылки данных. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Запись байта (0); запись байта (1); запись байта (2); запись байта (3); запись байтов (0—2); запись байтов (1—3); однобайтовая блочная запись) 61
2.19 Временнáя диаграмма пересылки данных. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Запись байтов (0—1); запись байтов (2—3); запись байтов (0—3); запись байтов (1—2); двухбайтовая блочная запись; четырехбайтовая блочная запись) 63
2.20 Временнáя диаграмма пересылки данных. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Однобайтовый цикл Чтение—Модификация—Запись) 65
2.21 Временнáя диаграмма пересылки данных. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Двухбайтовые циклы Чтение—Модификация—Запись; четырехбайтовые циклы Чтение—Модификация—Запись) 66
2.22 Временнáя диаграмма адресного строба между циклами 67
2.23 Временнáя диаграмма стробов данных между циклами. За циклом, в котором оба строба данных устанавливаются низкими, следует цикл, в котором один или оба строба устанавливаются низкими 68
2.24 Временнáя диаграмма стробов данных между циклами. За циклом, в котором один из стробов данных устанавливается низким, следует цикл, в котором один или оба строба устанавливаются низкими 69
2.25 Временнáя диаграмма пересылки данных. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и ШИННЫЙ ТАЙМЕР. (Цикл с тайм-аутом) 70

2.26 Временна́я диаграмма передачи управления шиной пересылки данных. ЗАДАТЧИК . . . 71
3.1 Схема взаимодействия шины арбитража с функциональными блоками 73
3.2 Иллюстрация последовательных цепочек предоставления шины 74
3.3 Схема взаимодействия арбитра с магистралью VME 76
3.4 Схема взаимодействия запросчика с магистралью VME 77
3.5 Алгоритм арбитража. Два запросчика, два уровня запросов 82
3.6 Диаграмма последовательности арбитража. Два запросчика, два уровня запросов 85
3.7 Алгоритм арбитража. Два запросчика, один и тот же уровень запроса 86
3.8 Диаграмма последовательности арбитража. Два запросчика, один и тот же уровень запросов 90
4.1 Схема взаимодействия шины приоритетных прерываний с функциональными блоками . . 92
4.2 Структурная схема системы прерывания с одним обработчиком 93
4.3 Структурная схема распределенной системы прерывания 94
4.4 Последовательная цепочка IACKIN*/IACKOUT* 95
4.5 Схема взаимодействия обработчика прерываний с магистралью VME 96
4.6 Схема взаимодействия прерывателя с магистралью VME 98
4.7 Схема взаимодействия формирователя цепочки подтверждения прерывания с магистралью VME 99
4.8 Освобождение линий запроса прерывания прерывателями типов ROAK и RORA . . . 102
4.9 Формирователь цепочки подтверждения прерывания и прерыватель в одном и том же модуле 103
4.10 Два прерывателя в одном и том же модуле 104
4.11 Три фазы последовательности обработки прерывания 104
4.12 Два обработчика прерываний, каждый из которых контролирует одну линию запроса прерывания 106
4.13 Два обработчика прерываний, каждый из которых контролирует несколько линий запроса прерывания 106
4.14 Алгоритм работы системы прерывания с одним обработчиком 107
4.15 Алгоритм работы распределенной системы прерывания с двумя обработчиками прерываний 111
4.16 Временна́я диаграмма выбора прерывателя. ОБРАБОТЧИК ПРЕРЫВАНИЙ и ПРЕРЫВАТЕЛЬ. (Одно-, двух- и четырехбайтовые циклы подтверждения прерывания) 123
4.17 Временна́я диаграмма выбора прерывателя. ФОРМИРОВАТЕЛЬ ЦЕПОЧКИ ПОДТВЕРЖДЕНИЯ ПРЕРЫВАНИЯ. (Одно-, двух- и четырехбайтовые циклы подтверждения прерывания) 124
4.18 Временна́я диаграмма выбора прерывателя. Участвующий ПРЕРЫВАТЕЛЬ. (Одно-, двух- и четырехбайтовые циклы подтверждения прерывания) 125
4.19 Временна́я диаграмма выбора прерывателя. Отвечающий ПРЕРЫВАТЕЛЬ. (Одно-, двух- и четырехбайтовые циклы подтверждения прерывания) 126
4.20 Временна́я диаграмма пересылки информации статуса/идентификации. ОБРАБОТЧИК ПРЕРЫВАНИЙ. (Однобайтовый цикл подтверждения прерывания) 127
4.21 Временна́я диаграмма пересылки информации статуса/идентификации. ОБРАБОТЧИК ПРЕРЫВАНИЙ. (Двух- и четырехбайтовый циклы подтверждения прерывания) 128
4.22 Временна́я диаграмма пересылки информации статуса/идентификации. Отвечающий ПРЕРЫВАТЕЛЬ. (Однобайтовый цикл подтверждения прерывания) 129
4.23 Временна́я диаграмма пересылки информации статуса/идентификации. Отвечающий ПРЕРЫВАТЕЛЬ. (Двух- и четырехбайтовый циклы подтверждения прерывания) 130
4.24 Временна́я диаграмма работы цепочки подтверждения прерывания между циклами ФОРМИРОВАТЕЛЬ ЦЕПОЧКИ ПОДТВЕРЖДЕНИЯ ПРЕРЫВАНИЯ, отвечающий ПРЕРЫВАТЕЛЬ и участвующий ПРЕРЫВАТЕЛЬ 131
5.1 Схема взаимодействия служебной шины с функциональными блоками 133
5.2 Временна́я диаграмма работы формирователя системного тактового сигнала 134
5.3 Схема взаимодействия блока контроля питания с устройствами магистрали VME 134
5.4 Временна́я диаграмма работы блока контроля питания в случае отказа питания 135
5.5 Временна́я диаграмма работы блока контроля питания при включении системы 135
5.6 Временна́я диаграмма сигналов SYSRESET* и SYSFAIL* 137
5.7 Оценка значения тока для контактов питания 138
6.1 Уровни логических сигналов 140

6.2 Стандартные оконечные нагрузки магистрали . . . 145
6.3 Поперечное сечение микрополосковой сигнальной линии объединительной платы . . . 146
6.4 Зависимость волнового сопротивления Z_0 от ширины сигнальной линии . . . 146
6.5 Зависимость погонной емкости C_0 от ширины сигнальной линии . . . 147
7.1 Каркас с модулями разной высоты . . . 156
7.2 Основные размеры печатной платы одинарной высоты . . . 157
7.3 Основные размеры печатной платы двойной высоты . . . 158
7.4 Расположение соединителей на платах одинарной и двойной высоты . . . 159
7.5 Поперечный разрез печатной платы, соединителя, объединительной платы и передней панели . . . 160
7.6 Высота ЭРЭ, длина выводов, коробление печатных плат . . . 161
7.7 Передняя панель одинарной высоты, одинарной ширины . . . 162
7.8 Передняя панель двойной высоты, одинарной ширины . . . 163
7.9 Крепежные угольники передней панели и размеры модуля одинарной высоты . . . 164
7.10 Крепежные угольники передней панели и размеры модуля двойной высоты . . . 165
7.11 Панель-заглушка одинарной высоты . . . 166
7.12 Панель-заглушка двойной высоты . . . 167
7.13 Габаритные и присоединительные размеры объединительных плат J1 и J2 . . . 168
7.14 Детальные размеры объединительных плат J1 и J2 . . . 169
7.15 Габаритные и присоединительные размеры комбинированной объединительной платы J1/J2 170
7.16 Детальные размеры комбинированной объединительной платы J1/J2 . . . 171
7.17 Расположение оконечных нагрузок вне объединительной платы (вид на объединительную плату сверху) . . . 172
7.18 Расположение оконечных нагрузок непосредственно на объединительной плате (вид на объединительную плату сверху) . . . 173
7.19 Каркас на 21 гнездо . . . 173
7.20 Направляющие модулей . . . 174

Перечень таблиц

2.1 Категории байтовых ячеек . . . 12
2.2 Использование сигналов DS0*, DS1*, A01 и LWORD* для выбора байтовых ячеек . . . 12
2.3 Коды модификатора адреса . . . 14
2.4 Использование линий данных для доступа к байтовым ячейкам . . . 16
2.5 Задатчики. Правила и разрешения для контроля и возбуждения сигнальных линий, отмеченных на рисунке 2.2 пунктиром . . . 18
2.6 Исполнители. Правила и разрешения для контроля и возбуждения сигнальных линий, отмеченных на рисунке 2.3 пунктиром . . . 19
2.7 Использование мнемонического обозначения BTO(x) для указания тайм-аута шинных таймеров . . . 20
2.8 Адресный монитор. Правила и разрешения для контроля сигнальных линий, отмеченных на рисунке 2.5 пунктиром . . . 21
2.9 Мнемонические обозначения режимов адресации . . . 22
2.10 Мнемонические обозначения основных возможностей пересылки данных . . . 24
2.11 Мнемоническое обозначение возможностей блочных пересылок . . . 26
2.12 Мнемоническое обозначение возможностей цикла Чтение—Модификация—Запись . . . 27
2.13 Пересылка 32 разрядов данных с использованием кратнобайтовых циклов . . . 28
2.14 Пересылка 16 разрядов данных с использованием кратнобайтовых циклов . . . 29
2.15 Мнемоническое обозначение возможности невыровненных пересылок . . . 30
2.16 Мнемоническое обозначение возможности цикла Только Адрес . . . 30
2.17 Перечень временных диаграмм, определяющих работу задатчиков, отвечающих исполнителей и адресных мониторов (временные параметры указаны в таблице 2.22) . . . 39
2.18 Определение мнемонических обозначений, употребляемых в таблицах 2.19—2.21 . . . 40

2.19 Использование адресных линий для выбора четырехбайтовой группы 41
2.20 Использование DS0*, DS1*, A01 и LWORD* во время различных циклов 42
2.21 Использование линий данных для пересылки данных 43
2.22 Значения параметров временны́х соотношений для задатчика, исполнителя и адресного монитора 45
2.23 Значения параметров временны́х соотношений для шинного таймера 46
2.24 Задатчик. Правила и замечания по временны́м соотношениям 46
2.25 Исполнитель. Правила и замечания по временны́м соотношениям 49
2.26 Адресный монитор. Замечания по временны́м соотношениям 52
2.27 Шинный таймер. Правила по временны́м соотношениям 53
3.1 Арбитры. Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных на рисунке 3.3 пунктиром 77
3.2 Запросчики. Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных на рисунке 3.4 пунктиром 78
4.1 Обработчик прерываний. Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных на рисунке 4.5 пунктиром 96
4.2 Прерыватели. Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных на рисунке 4.6 пунктиром 99
4.3 Использование мнемонического обозначения IH( ) для описания возможностей обработки прерываний 100
4.4 Использование мнемонического обозначения I( ) для описания возможностей формирования запроса прерывания 100
4.5 Мнемонические обозначения для описания возможностей по пересылке информации статуса/идентификации 100
4.6 Мнемонические обозначения для описания возможностей освобождения линии запроса прерывания 102
4.7 Трехразрядный код подтверждения прерывания 110
4.8 Перечень временны́х диаграмм, определяющих работу обработчика прерываний и прерывателя 113
4.9 Перечень временны́х диаграмм, определяющих работу формирователя цепочки подтверждения прерывания 114
4.10 Перечень временны́х диаграмм, определяющих работу участвующего прерывателя . . . 114
4.11 Перечень временны́х диаграмм, определяющих работу отвечающего прерывателя . . . 114
4.12 Определение мнемонических обозначений, используемых в таблицах 4.13—4.15 114
4.13 Использование линий A01—A03 и IACK* во время циклов подтверждения прерывания . . 115
4.14 Использование линий DS1*, DS0*, LWORD* и WRITE* во время циклов подтверждения прерывания 115
4.15 Использование линий данных D00—D31 для пересылки информации статуса/идентификации 116
4.16 Значения параметров временны́х соотношений для обработчиков прерываний, прерывателей и формирователей цепочки подтверждения прерывания 116
4.17 Обработчик прерываний. Правила и замечания по временны́м соотношениям 117
4.18 Прерыватель. Правила и замечания по временны́м соотношениям 119
4.19 Формирователь цепочки подтверждения прерывания. Правила и замечания по временны́м соотношениям 122
5.1 Правила возбуждения функциональными блоками сигналов в процессе включения и выключения питания 136
6.1 Технические характеристики питающих напряжений магистрали 139
6.2 Требования к характеристикам формирователей и приемников сигналов магистрали . . 141
6.3 Сводная таблица типов формирователей магистрали 148
7.1 Распределение сигналов по контактам соединителей J1/P1 175
7.2 Распределение сигналов по контактам соединителей J2/P2 176

Введение

Настоящий стандарт определяет магистраль, реализуемую на объединительной плате (объединительных платах) и обладающую высокими техническими характеристиками. Магистраль предназначена для использования в микрокомпьютерных системах, в которых задействованы один или несколько микропроцессоров.

Основой стандарта являются «Общие технические требования на магистраль VME», изданные группой изготовителей VME-магистральных изделий в августе 1982 г. Магистраль VME состоит из следующих четырех субмагистралей (далее — шин): шины пересылки данных, шины приоритетных прерываний, шины арбитража и служебной шины.

Шина пересылки данных выполняет 8-, 16- и 32-разрядные пересылки данных. Линии данных и адресов не мультиплексируются. Протоколы пересылки являются асинхронными. Пересылки полностью квитируются. Шина приоритетных прерываний обеспечивает прерывания в реальном времени. Передача управления магистралью выполняется шиной арбитража, которая позволяет реализовать алгоритмы кругового и приоритетного арбитража. Служебная шина обеспечивает синхронизацию в процессе включения и выключения питания. Требования к механическим конструкциям модулей, объединительных плат, каркасов и оболочек основаны на соответствующих требованиях Публикации МЭК [1], [2].

Перечень нормативной документации, на которую делаются ссылки в настоящем стандарте, приведен в ссылочных материалах.

Настоящий стандарт соответствует международному стандарту МЭК 821 (второе издание, 1991 г.).

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

МАГИСТРАЛЬ МИКРОПРОЦЕССОРНЫХ СИСТЕМ ДЛЯ ОБМЕНА ИНФОРМАЦИЕЙ РАЗРЯДНОСТЬЮ ОТ 1 ДО 4 БАЙТОВ (МАГИСТРАЛЬ VME)

IEC 821 VME bus.
Microprocessor system bus for 1 byte to 4 byte data

Дата введения 2001—01—01

1 Область применения

1.1 Назначение стандарта магистрали VME

Настоящий стандарт устанавливает требования к интерфейсной системе, используемой для взаимного соединения устройств обработки, запоминания данных и управления периферией (в дальнейшем — устройств) в единый аппаратный комплекс, и предназначен для решения следующих задач:

- обеспечение обмена информацией между устройствами по магистрали без вмешательства во внутренние операции других устройств, сопряженных с этой магистралью;
- определение электрических и конструктивных характеристик системы, необходимых для проектирования устройств, которые будут иметь надежную и однозначную связь с другими устройствами, сопряженными с магистралью;
- установление протоколов, которые однозначно определяют порядок взаимодействия между магистралью и сопряженными с ней устройствами;
- введение терминов и определений, которые употребляются для описания системного протокола;
- предоставление существенной свободы проектирования, позволяющей разработчику оптимизировать стоимость и/или рабочие характеристики, не влияющие на системную совместимость ;
- предоставление системы, в которой рабочие характеристики в основном ограничиваются устройствами, а не интерфейсной системой.

1.2 Элементы интерфейсной системы

1.2.1 Основные определения

Структура магистрали VME может быть описана с двух точек зрения: механической конструкции и функциональной структуры.

Конструктивные технические требования содержат данные о физических размерах каркасов, объединительных плат, передних панелей, модулей и т.д.

Функциональные технические требования содержат данные о принципах работы магистрали, функциональных блоков, задействованных в каждой операции, и правилах, которые определяют их поведение.

Данный раздел содержит не претендующие на строгость определения некоторых основных терминов, употребляемых для описания структурной схемы и механических конструкций магистрали VME.

1.2.1.1 *Термины, используемые для описания механических конструкций магистрали VME*

объединительная плата (backplane): Печатная плата с 96-контактными соединителями и печатными сигнальными проводниками, которые соединяют соответствующие контакты этих соединителей. Некоторые системы магистрали VME имеют единственную печатную плату — объединительную

Издание официальное
★

плату J1. Она содержит печатные сигнальные проводники, требуемые для выполнения основных операций. Другие системы магистрали VME имеют также дополнительную печатную плату — объединительную плату J2. Она содержит дополнительные 96-контактные соединители и печатные сигнальные проводники, требуемые для пересылок данных и адресов повышенной разрядности. Наконец, третьи имеют комбинированную печатную плату — объединительную плату J1/J2, которая содержит печатные сигнальные проводники и соединители объединительных плат J1 и J2.

модуль (board): Печатная плата с набором электрорадиоэлементов и одним или двумя 96-контактными соединителями, которые могут сочленяться с соединителями объединительной платы.

гнездо (slot): Место перед объединительной платой (платами), куда помещается модуль для сочленения с соединителем (соединителями) этой платы. Если система имеет обе платы J1 и J2 или комбинированную объединительную плату J1/J2, то каждое гнездо имеет два 96-контактных соединителя. Если система имеет только одну плату J1, то каждое гнездо имеет один 96-контактный соединитель.

каркас (subrack): Жесткая рамочная несущая конструкция, обеспечивающая механическую поддержку модулей, вставляемых в объединительную плату, надлежащее сочленение соединителей, отсутствие контакта между соседними модулями, распределение охлаждающих воздушных потоков в системе и невозможность нарушения контакта между вставленными модулями и объединительной платой вследствие вибрации или ударов.

1.2.1.2 *Термины, используемые для описания функциональной структуры магистрали VME*

На рисунке 1.1 приведена упрощенная схема функциональной структуры системы, содержащая сигнальные линии, интерфейсную логику объединительной платы и функциональные блоки.

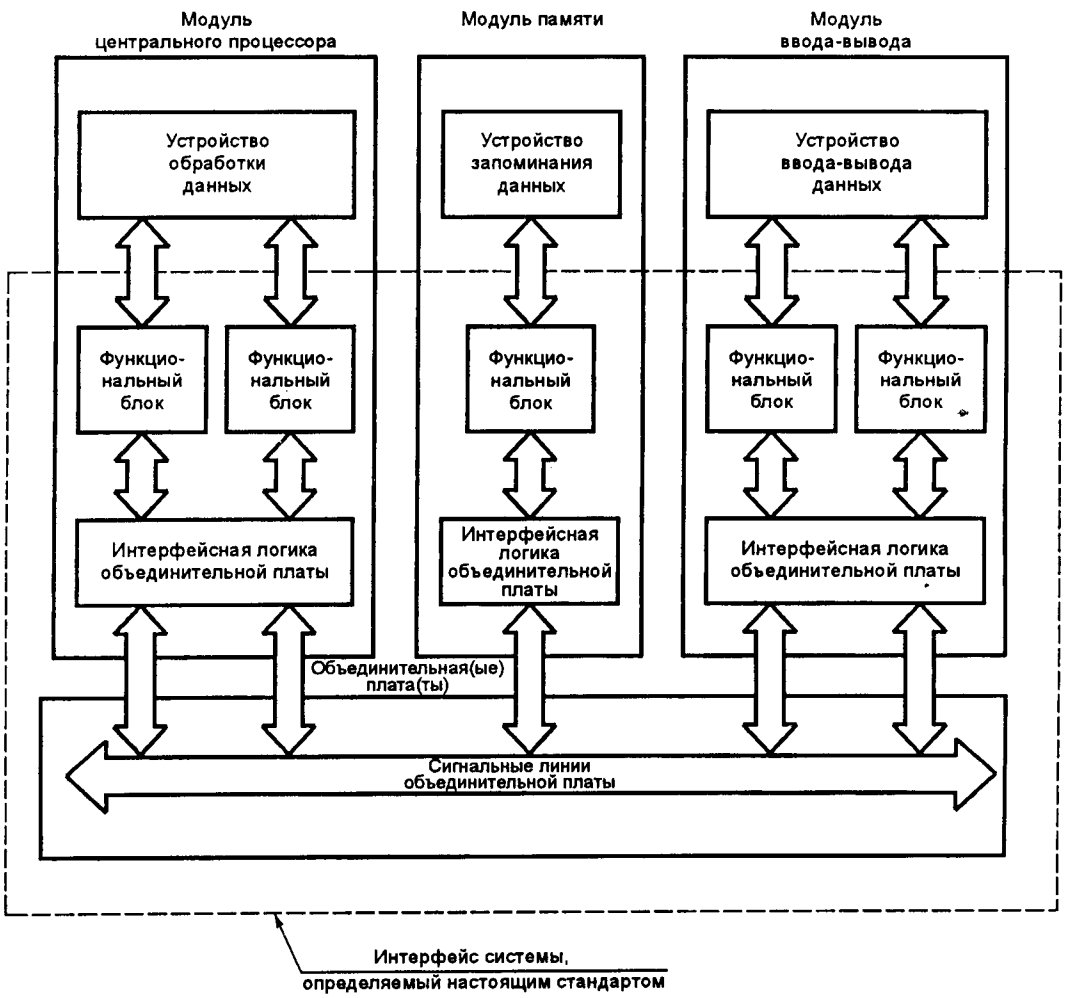


Рисунок 1.1 — Элементы системы, определенные настоящим стандартом

интерфейсная логика объединительной платы магистрали VME (backplane interface logic): Определенная логика, учитывающая характеристики объединительной платы: полное сопротивление ее сигнальных линий, время распространения сигналов, значения оконечных нагрузок и т. д. Настоящий стандарт устанавливает правила проектирования такой логики с учетом максимальной длины объединительной платы и максимального количества гнезд для модулей.

функциональный блок (functional module): Совокупность электронных схем, расположенных на одном модуле магистрали VME и совместно выполняющих какую-то определенную задачу.

шина пересылки данных (data tansfer bus): Одна из четырех шин объединительной платы. Позволяет задатчикам устанавливать направление пересылок двоичных данных между ними и исполнителями.

цикл шины пересылки данных (data transfer bus cycle): Последовательность перепадов напряжения логических уровней на сигнальных линиях шины пересылки данных, которая реализует пересылку адреса или адреса и данных между задатчиком и исполнителем. Цикл шины пересылки данных подразделяется на две части:

- широковещательная пересылка адреса;
- пересылка данных, если она предусмотрена.

Предусмотрено 34 типа циклов шины пересылки данных. Они определены ниже в данном разделе.

задатчик (master): Функциональный блок, который инициирует циклы шины пересылки данных для пересылки данных между ним и исполнителем.

исполнитель (slave): Функциональный блок, который обнаруживает циклы шины пересылки данных, инициируемые задатчиком, и, когда эти циклы предусматривают его (исполнителя) участие, осуществляет пересылки данных между собой и задатчиком.

адресный монитор (location monitor): Функциональный блок, контролирующий пересылки по шине пересылки данных с целью обнаружения обращений к ячейкам, которые ему предписано контролировать. Если происходит обращение к одной из этих ячеек, адресный монитор вырабатывает внутримодульный сигнал.

шинный таймер (bus timer): Функциональный блок, который выполняет отсчет времени каждой пересылки по шине пересылки данных и завершает цикл шины, если это время превышает разумные пределы. Без этого блока может возникнуть ситуация, при которой задатчик попытается выполнить операцию пересылки данных в/из какой-то несуществующей ячейки исполнителя и неопределенно долго ждать результаты. Шинный таймер предотвратит эту задержку завершением цикла.

шина приоритетных прерываний (priority interrupt bus): Одна из четырех шин объединительной платы магистрали VME, которая позволяет прерывателям посылать запросы прерывания обработчикам прерываний.

прерыватель (interrupter): Функциональный блок, формирующий запрос прерывания на шине приоритетных прерываний, а затем предоставляющий информацию статуса/идентификации по требованию обработчика прерываний.

обработчик прерываний (interrupt handler): Функциональный блок, обнаруживающий запросы прерывания, которые формируются прерывателями, и отвечающий на эти запросы требованием о предоставлении ему информации статуса/идентификации.

последовательная цепочка (далее — цепочка) (daisy—chain): Определенный тип сигнальной линии магистрали VME, используемой для распространения уровня сигнала от модуля к модулю, начиная с первого гнезда и кончая последним гнездом. Магистраль VME имеет четыре цепочки предоставления шины и одну цепочку подтверждения прерывания.

формирователь последовательной цепочки подтверждения прерывания (IACK (interrupt acknowledge) daisy—chain driver): Функциональный блок, устанавливающий в активное состояние цепочку подтверждения прерывания всякий раз, когда обработчик прерываний подтверждает запрос прерывания. Такая цепочка гарантирует пересылку информации о статусе/идентификации по шине пересылки данных только от одного прерывателя, даже если запрос прерывания сформирован несколькими прерывателями.

шина арбитража (arbritration bus): Одна из четырех шин объединительной платы магистрали VME. Позволяет одному арбитру и нескольким запросчикам координировать порядок использования шины пересылки данных.

запросчик (requester): Функциональный блок, находящийся на одной плате с задатчиком или обработчиком прерываний и запрашивающий право на использование шины пересылки данных всякий раз, когда это потребуется его задатчику или обработчику прерываний.

арбитр (arbiter): Функциональный блок, который принимает запросы на использование шины от запросчиков и предоставляет управление шиной пересылки данных одновременно только одному запросчику.

служебная шина (utility bus): Одна из четырех шин объединительной платы магистрали VME, по которой передаются периодические системные синхросигналы и сигналы, координирующие последовательность действий системы при включении и выключении питания.

формирователь системного тактового сигнала (system clock driver): Функциональный блок, подающий на служебную шину синхронизирующий сигнал частотой 16 МГц.

формирователь тактового сигнала последовательной магистрали (serial clock driver): Функциональный блок, подающий периодический тактовый сигнал, который синхронизирует работу магистрали МЭК 823 [5]. Хотя стандарт магистрали VME определяет формирователь тактового сигнала последовательной пересылки и в нем предусматриваются две сигнальные линии на объединительной плате, используемые магистралью МЭК 823, протокол последней абсолютно не зависит от магистрали VME.

Технические требования к временным параметрам для формирователя тактового сигнала последовательной магистрали приведены в приложении С.

блок контроля питания (power monitor module): Функциональный блок, контролирующий состояние первичного источника питания системы магистрали VME и сигнализирующий о выходе параметров питания за пределы, гарантирующие надежную работу системы. Поскольку большинство систем запитываются от источника переменного тока, блок контроля питания обычно проектируется с возможностью обнаружения состояния обесточивания сети переменного тока.

модуль системного контроллера (system controller board): Модуль, размещаемый в гнезде 1 объединительной платы магистрали VME и имеющий в своем составе формирователь системного тактового сигнала, арбитр, формирователь цепочки подтверждения прерывания и шинный таймер. Некоторые контроллеры содержат также формирователь тактового сигнала последовательной пересылки либо блок контроля питания, либо то и другое.

1.2.1.3 *Типы циклов магистрали VME*

цикл считывания (read cycle): Цикл шины пересылки данных, используемый для пересылки одного, двух, трех или четырех байтов от исполнителя задатчику. Цикл начинается, когда задатчик выполняет широковещательную пересылку адреса и модификатора адреса. Каждый исполнитель принимает адрес и модификатор адреса и проверяет, не он ли должен отвечать в этом цикле считывания. Если это так, он извлекает данные из своей внутренней памяти, помещает их на шину пересылки данных и подтверждает пересылку. После этого задатчик завершает цикл.

цикл записи (write cycle): Цикл шины пересылки данных, используемый для пересылки одного, двух, трех или четырех байтов от задатчика исполнителю. Цикл начинается, когда задатчик выполняет широковещательную пересылку адреса и модификатора адреса и помещает данные на шину пересылки данных. Каждый исполнитель принимает адрес и модификатор адреса и проверяет, не он ли должен отвечать в этом цикле. Если это так, он запоминает эти данные и затем подтверждает пересылку. После этого задатчик завершает цикл.

цикл блочного считывания (block read cycle): Цикл шины пересылки данных, используемый для пересылки блока размером от 1 до 256 байтов от исполнителя задатчику. Эта пересылка выполняется с использованием ряда последовательных одно-, двух- или четырехбайтовых пересылок. После начала блочной пересылки задатчик не освобождает шину пересылки данных до тех пор, пока им не будут считаны все байты. Цикл блочного считывания отличается от ряда из нескольких обычных циклов считывания тем, что задатчик только один раз (в начале цикла) выполняет широковещательную пересылку адреса и модификатора адреса. Затем при каждой пересылке исполнитель наращивает этот адрес, чтобы данные для следующей пересылки извлекались из следующей ячейки.

цикл блочной записи (block write cycle): Цикл шины пересылки данных, используемый для пересылки блока размером от 1 до 256 байтов от задатчика исполнителю. Эта пересылка выполняется с использованием ряда последовательных одно-, двух- или четырехбайтовых пересылок данных. Задатчик не освобождает шину пересылки данных до тех пор, пока не будут переданы все байты. Цикл блочной записи отличается от ряда из нескольких обычных циклов записи тем, что задатчик только один раз (в начале цикла) выполняет широковещательную пересылку адреса и модификатора адреса. Затем исполнитель наращивает этот адрес при каждой пересылке, чтобы данные следующей пересылки записывались в следующую ячейку.

цикл Чтение—Модификация—Запись (read—modify—write cycle): Цикл шины пересылки данных, используемый задатчиком для обращения к ячейке исполнителя как в режиме считывания, так и в режиме записи при запрещении доступа к этой ячейке со стороны других задатчиков. Этот цикл очень полезен для мультипроцессорных систем, в которых определенные ячейки памяти используются для выполнения семафорных функций.

цикл Только Адрес (address—only cycle): Цикл шины пересылки данных, состоящий только из широковещательной пересылки адреса без пересылки данных. Исполнители не подтверждают такие циклы, а задатчики завершают такой цикл, не ожидая подтверждения.

цикл подтверждения прерывания (interrupt acknowledge cycle): Цикл шины пересылки данных, инициируемый обработчиком прерываний, который выполняет считывание информации статуса/идентификации от прерывателя. Обработчик прерываний вырабатывает этот цикл всякий раз, когда обнаруживает запрос прерывания от прерывателя, а шина пересылки данных находится под его управлением.

1.2.2 Основная структура магистрали VME

Интерфейсная система магистрали VME состоит из интерфейсной логики объединительной платы, четырех групп сигнальных линий, называемых шинами, и набора функциональных блоков, которые могут быть сконфигурированы так, как это необходимо. Функциональные блоки взаимодействуют между собой, используя сигнальные линии объединительной платы.

Функциональные блоки, определенные настоящим стандартом, служат средством описания протокола магистрали и их не следует рассматривать как ограничение при проектировании интерфейсной логики. Например, разработчик может спроектировать логику, взаимодействующую с магистралью VME описанным способом, но использующую другие внутримодульные сигналы или контролирующую дополнительно другие сигналы магистрали VME. Модули магистрали VME можно проектировать из различных комбинаций функциональных блоков, определенных настоящим стандартом.

Функциональная структура магистрали VME подразделяется на четыре категории. Каждая состоит из шины и подсоединенных к ней функциональных блоков, выполняющих совместно свои конкретные задачи. Функциональные блоки и шины магистрали изображены на рисунке 1.2. Каждая категория функциональной структуры кратко описана ниже.

Шина пересылки данных. Устройства пересылают данные по шине пересылки данных, состоящей из магистральных линий данных, адреса и соответствующих управляющих сигналов. Функциональные блоки (задатчики, исполнители, прерыватели и обработчики прерываний) используют шину пересылки данных для обмена данными между собой. В этом процессе им оказывают содействие два других блока: шинный таймер и формирователь последовательной цепочки подтверждения прерывания.

Шина арбитража. Поскольку системы магистрали могут конфигурироваться с несколькими задатчиками или обработчиками прерываний, предусмотрен механизм, который упорядоченным способом выполняет передачу управления шиной пересылки данных от одного блока к другому и гарантирует соблюдение правила: в любой заданный момент времени только один из них управляет шиной пересылки данных. Передачу управления координируют функциональные блоки шины арбитража (запросчики и арбитр).

Шина приоритетных прерываний. Функциональная возможность приоритетных прерываний магистрали VME является средством, с помощью которого устройства могут запрашивать обслуживание от обработчиков прерываний. Эти запросы прерываний могут быть подразделены по приоритету максимально на семь уровней. Прерыватели и обработчики прерываний используют сигнальные линии шины приоритетных прерываний.

Служебная шина. Служебная шина обеспечивает следующие функции: передачу периодических тактовых сигналов, выполнение операций инициализации и обнаружение отказов. Шина состоит из двух линий тактовых сигналов, линии сигнала системного сброса, линии сигнала системного отказа, линии сигнала отказа сети переменного тока и линии данных последовательной магистрали пересылки.

1.3 Диаграммы, используемые в стандарте магистрали VME

Для более наглядного описания протоколов магистрали VME используются три типа диаграмм, перечисленных ниже.

Временны́е диаграммы. Изображают временны́е соотношения между изменениями сигналов. Указанные временны́е параметры имеют минимальные и/или максимальные значения. Некоторые времен-

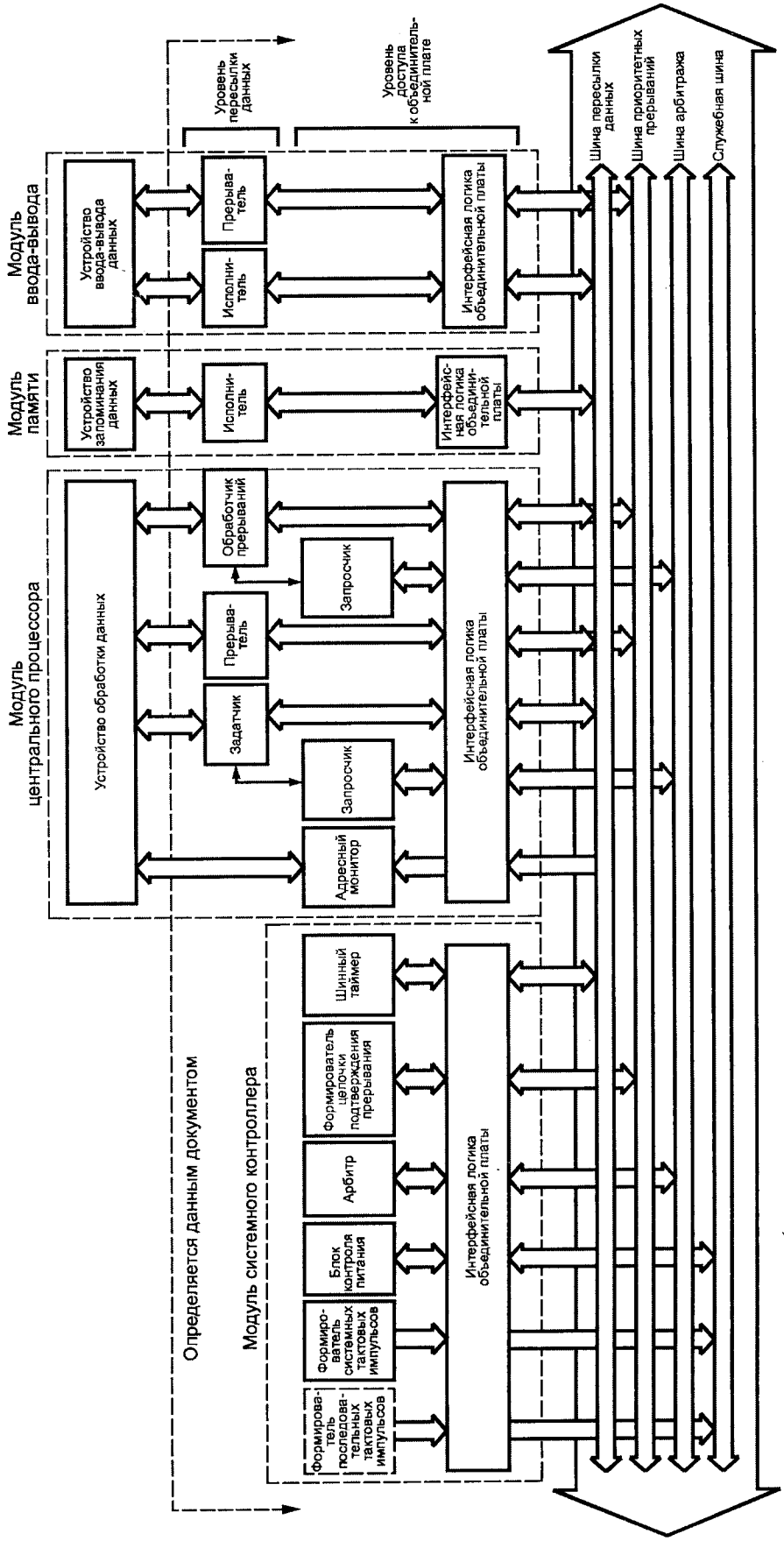


Рисунок 1.2 — Функциональные блоки и шины, определяемые настоящим стандартом

ны́е параметры определяют поведение интерфейсной логики объединительной платы, другие — поведение функциональных блоков в их взаимосвязи.

Диаграммы последовательностей. Аналогичны временно́й диаграмме, но показывают лишь временные соотношения взаимосвязи функциональных блоков. Они показывают последовательность событий, но не определяют соответствующие им временны́е параметры. Например, диаграмма последовательности может показать, что блок А не сможет сформировать изменение состояния сигнала В до тех пор, пока он не обнаружит, что блок С изменил состояние сигнала D.

Схемы последовательностей. Показывают порядок следования событий, по мере того как они происходят во время работы магистрали VME. События формулируются словами и являются результатом взаимодействия двух или более функциональных блоков. Схемы последовательностей дают описание операций магистрали в последовательном порядке и одновременно показывают, как именно взаимодействуют функциональные блоки.

1.4 Терминология

Чтобы избежать путаницы и четко определить требования, нуждающиеся в согласованности, многие абзацы настоящего стандарта озаглавлены ключевыми словами, которые указывают на тип содержащейся в них информации. Это следующие слова:

- ***Правило***
- **Рекомендация**
- **Предложение**
- **Разрешение**
- **Замечание**

В тексте за ключевым словом следуют числа, разделенные точкой и обозначающие: первое — номер раздела стандарта, второе — индивидуальный номер данного типа информации в разделе.

Любой текст, не имеющий в качестве заголовка перечисленных ключевых слов, является описательной частью структуры интерфейсной системы или ее работы. Он имеет описательную или повествовательную форму. Ниже приводятся указания по использованию ключевых слов.

Правило. *Правила образуют основу данного стандарта и могут быть представлены в виде текстов, рисунков, таблиц или чертежей. Все правила ДОЛЖНЫ СОБЛЮДАТЬСЯ неукоснительно с целью обеспечения совместимости изделий магистрали VME. Правила характеризуются употреблением императивных оборотов с использованием модального глагола долженствования (ДОЛЖЕН, НЕ ДОЛЖЕН) и смыслового глагола в неопределенной форме, выделяемых в тексте прописными буквами и употребляемых в этом виде исключительно для формулирования правил.*

Рекомендация. Разработчику следует выполнять все рекомендации, содержащиеся в стандарте. Игнорирование рекомендаций может привести к возникновению тупиковых ситуаций или ухудшению характеристик системы. Магистраль VME разработана для реализации высокопроизводительных систем, но можно спроектировать систему, формально соответствующую всем правилам, но имеющую очень низкую производительность. Во многих случаях разработчику нужно обладать определенным опытом проектирования модулей, которые обеспечивали бы наивысшую производительность. Рекомендации основываются именно на таком опыте и дают разработчику соответствующую информацию для его освоения.

Предложение. Предложение содержит совет, который является полезным, но не имеет первостепенной важности. Прежде чем отвергнуть этот совет, разработчику предлагается его рассмотреть. Без приобретенного опыта некоторые решения при проектировании принимать очень сложно. Предложения имеют цель помочь разработчику в приобретении такого опыта. Некоторые предложения касаются проектирования модулей с возможностью их несложной переконфигурации для совместной работы с другим модулем или упрощения отладки системы и т.д.

Разрешение. В некоторых случаях правила не содержат специальных запретов по методам проектирования, однако разработчик может оказаться в затруднении, решая вопрос о правомерности применения этих методов и вероятности возникновения при этом каких-либо неожиданных проблем. Разрешение убеждает разработчика, что какой-то определенный подход является приемлемым и не вызовет никаких ошибок. Для формулирования разрешений употребляется модальный глагол МОЧЬ (МОЖЕТ, МОГУТ, МОЖНО), выделяемый в тексте разрешений прописными буквами. Глагол МОЧЬ в таком виде употребляется исключительно для этой цели.

Замечание. Замечания не содержат каких-то конкретных советов. Обычно они являются естественным продолжением только что рассмотренных вопросов. Они разъясняют смысл некоторых

правил и обращают внимание на те стороны, которые без этих разъяснений могут оказаться упущенными. Они содержат также обоснование введения определенных правил, чтобы разработчик понимал, по какой причине эти правила должны соблюдаться.

1.4.1 Состояния сигнальных линий

Настоящий стандарт устанавливает протоколы магистрали, описание которых дается в терминах логических уровней сигналов и их переходов с одного уровня на другой на линиях магистрали. Предполагается, что сигнал на любой линии всегда имеет один из двух уровней или находится в состоянии перехода между этими уровнями. При употреблении термина ВЫСОКИЙ подразумевается высокий уровень напряжения схем транзисторно-транзисторной логики (ТТЛ), термином НИЗКИЙ обозначается низкий уровень напряжения схем ТТЛ. Если напряжение изменяется между этими уровнями, то сигнал на линии находится в состоянии перехода (информация по пороговым напряжениям магистрали VME содержится в разделе 6).

На линии возможны два вида переходов сигнала, которые называются перепадами. Переход сигнала с низкого уровня на высокий называется положительным перепадом. Переход сигнала с высокого уровня на низкий называется отрицательным перепадом.

Техническими требованиями некоторых магистралей для этих перепадов устанавливаются максимальные или минимальные времена нарастания и спада сигнала. Однако проблема состоит в том, что разработчики модулей имеют мало возможностей контролировать эти времена. Если объединительная плата имеет большую нагрузку, времена нарастания и спада будут длительными. Если загрузка невелика, эти времена могут быть короткими. Даже если разработчики знают величины максимальной и минимальной нагрузок, все же потребуется время для лабораторных экспериментов, чтобы определить, какие формирователи сигналов обеспечат необходимые времена нарастания и спада. Эти времена являются результатом сложных взаимодействий ряда параметров, включающих волновое сопротивление сигнальной линии объединительной платы, величины ее оконечной нагрузки, внутреннее сопротивление источника формирователей и емкостную нагрузку сигнальной линии. Для выбора оптимального сочетания всех этих величин разработчику модулей необходимо изучить теорию линий передачи, а также некоторые специфические параметры формирователей и приемников, которые в документации большинства фирм-изготовителей обычно отсутствуют.

Учитывая изложенное, настоящий стандарт не устанавливает времена нарастания и спада. Вместо этого он определяет электрические характеристики формирователей и приемников и дает советы по проектированию объединительной платы. Он также информирует разработчиков, каким образом нагрузка магистрали для наихудшего случая повлияет на задержку распространения сигналов этих формирователей, чтобы еще до начала проектирования модуля они могли быть уверены в том, что требуемые временные соотношения будут соблюдены. Пользуясь этими рекомендациями, разработчики добьются надежной работы своего модуля при его взаимодействии с другими модулями магистрали VME для наихудших условий эксплуатации.

1.4.2 Использование звездочки (*)

В конце мнемонических обозначений сигналов при необходимости используется символ звездочки (*). В этих случаях звездочка несет следующий смысл:

- звездочка в конце мнемонического обозначения сигнала, для которого имеет значение его уровень, показывает, что такой сигнал является истинным или достоверным, когда он установлен низким;

- звездочка в конце мнемонического обозначения сигнала, для которого имеет значение его перепад, показывает, что действия, инициируемые таким сигналом, происходят при его переходе с высокого на низкий логический уровень.

Замечание 1.1. Звездочка не применяется для асинхронно работающих сигналов SYSCLK (system clock) и SERCLK (serial clock). Между сигналами на этих линиях и другими сигналами магистрали фиксированных фазовых соотношений не устанавливается.

1.5 Технические требования к протоколу

Протокол магистрали имеет два уровня. Нижний уровень, называемый уровнем доступа к объединительной плате, состоит из интерфейсной логики объединительной платы, функциональных блоков служебной шины и функциональных блоков шины арбитража. Второй уровень, называемый уровнем пересылки данных, состоит из функциональных блоков шины пересылки данных и функциональных блоков шины приоритетных прерываний. Подразделение на уровни показано на рисунке 1.2.

Замечание 1.2. Сигнальные линии, используемые функциональными блоками уровня пересылки данных, образуют особый класс, поскольку они возбуждаются в разные моменты времени разными функциональными блоками. Эти линии возбуждаются формирователями, которые в каждом функциональном блоке могут включаться и выключаться в зависимости от сигналов, формируемых на уровне доступа к объединительной плате. Очень важен строгий контроль за моментами времени включения и выключения формирователей, чтобы два формирователя не могли одновременно попытаться установить на одной и той же сигнальной линии два различных логических уровня. При определении моментов времени их включения и выключения используется специальная система обозначений в виде временных диаграмм (см. рисунок 1.3).

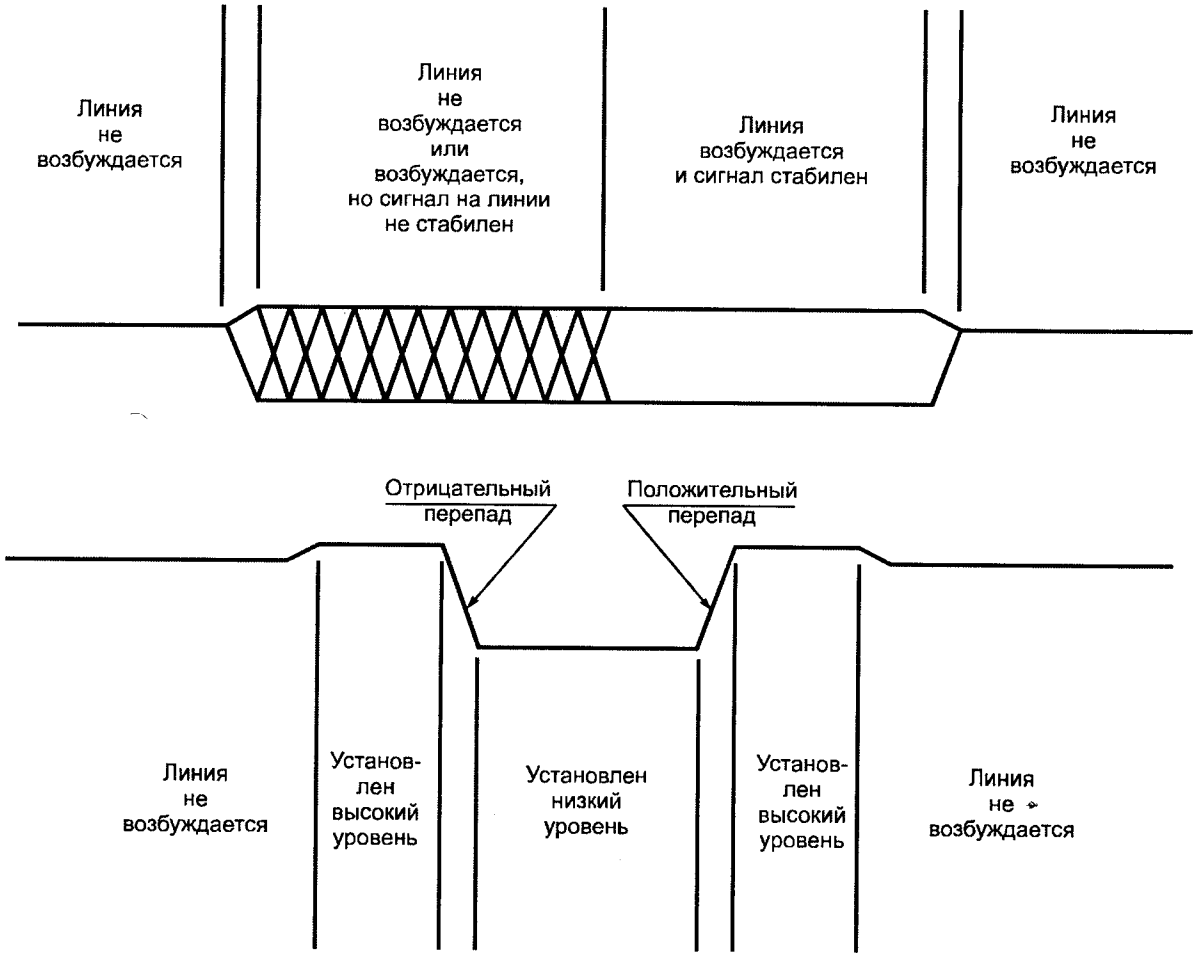


Рисунок 1.3 - Система обозначений, применяемых при изображении временной диаграммы сигналов

В магистрали используются два основных вида протоколов:

- протоколы замкнутого цикла;
- протоколы открытого цикла.

Протоколы замкнутого цикла используют взаимосвязанные сигналы магистрали, а протоколы открытого цикла - широковещательные сигналы магистрали.

1.5.1 Взаимосвязанные сигналы магистрали

Взаимосвязанный сигнал магистрали посылается одним конкретным функциональным блоком в другой конкретный функциональный блок. Принимающий функциональный блок подтверждает получение сигнала. Взаимосвязь между двумя функциональными блоками длится до тех пор, пока сигнал не будет подтвержден.

Например, прерыватель может послать запрос прерывания, на который позже поступает ответ в виде сигнала подтверждения прерывания. Никаких временных ограничений при этом не устанавливается. Прерыватель не снимает запрос до тех пор, пока обработчик прерываний не подтвердит его.

Взаимосвязанные сигналы магистрали координируют исполнение внутренних функций системы в противоположность взаимодействию системы с внешними стимулирующими сигналами. Каждый взаимосвязанный сигнал имеет в рамках системы функциональный блок источника и функциональный блок назначения.

Особо важными взаимосвязанными сигналами являются строб адреса и стробы данных. Они взаимосвязаны с сигналами подтверждения пересылки данных и ошибки шины и координируют пересылку адресов и данных, которые являются основой всего информационного потока между функциональными блоками на уровне пересылки данных.

1.5.2 Широковещательные сигналы магистрали

Функциональный блок формирует широковещательный сигнал в ответ на какое-то событие. Протокол для подтверждения широковещательного сигнала не определен. Вместо этого предусмотрен механизм, обеспечивающий широковещательную передачу сигнала в течение минимально заданного времени, которое достаточно для его обнаружения всеми соответствующими функциональными блоками. Широковещательные сигналы могут быть сформированы в любой момент времени, независимо от любых других операций, выполняемых магистралью. Каждый из них посылается по своей специальной сигнальной линии. Примерами могут служить линии сигналов системного сброса и отказа сети питания переменного тока. Сигналы по этим линиям посылаются не какому-то конкретному функциональному блоку, а сообщают о возникновении особых условий всем функциональным блокам.

1.6 Примеры функционирования систем и пояснения

Технические требования на протокол магистрали содержат подробное описание поведения различных функциональных блоков. В них рассмотрено, каким способом любой функциональный блок отвечает на какой-то сигнал без указания на происхождение сигнала. Из-за такой процедуры описания технические требования на протокол не дают полной картины. Чтобы помочь пользователю разобраться, настоящий стандарт содержит примеры типовых операций магистрали. В каждом таком примере приведена возможная последовательность событий, хотя возможны и другие последовательности. Существует опасность, что одновариантность последовательности событий в приведенных примерах будет восприниматься как единственно законный порядок операций, выполняемых магистралью. Во избежание такого неправильного восприятия все примеры приведены в повествовательной форме. Такое изложение противопоставлено строгому императивному стилю формулирования правил, обязательных для соблюдения соответствия настоящему стандарту.

2 Шина пересылки данных магистрали VME

2.1 Введение

Магистраль VME включает в себя высокоскоростную асинхронную параллельную шину пересылки данных. На рисунке 2.1 показана типичная система магистрали VME, содержащая все функциональные блоки шины пересылки данных. Они выделены жирными линиями.

Задатчики используют шину пересылки данных для выбора ячеек памяти исполнителей, а также для пересылки данных в эти ячейки или из них. Некоторые задатчики и исполнители используют все линии шины пересылки данных, другие — только какое-то их подмножество.

Адресные мониторы отслеживают пересылку данных между задатчиком и исполнителями и вырабатывают внутримодульный сигнал всякий раз, когда осуществляется обращение к одной из байтовых ячеек, которую он контролирует. Например, он может просигнализировать своему внутримодульному процессору посредством запроса прерывания. В такой конфигурации, где процессорный модуль А производит запись в ячейку глобальной памяти магистрали VME, контролируемую адресным монитором процессора В, работа процессора В будет прервана.

После того, как задатчик инициирует цикл пересылки данных, он не завершит этот цикл до тех пор, пока не дождется ответа от отвечающего исполнителя. Асинхронный характер магистрали позволяет исполнителю тратить на ответ столько времени, сколько потребуется. Если исполнитель не в состоянии дать ответа из-за какой-то неисправности или при случайном обращении задатчика к ячейке, где нет исполнителя, вмешивается шинный таймер, позволяя циклу завершиться.

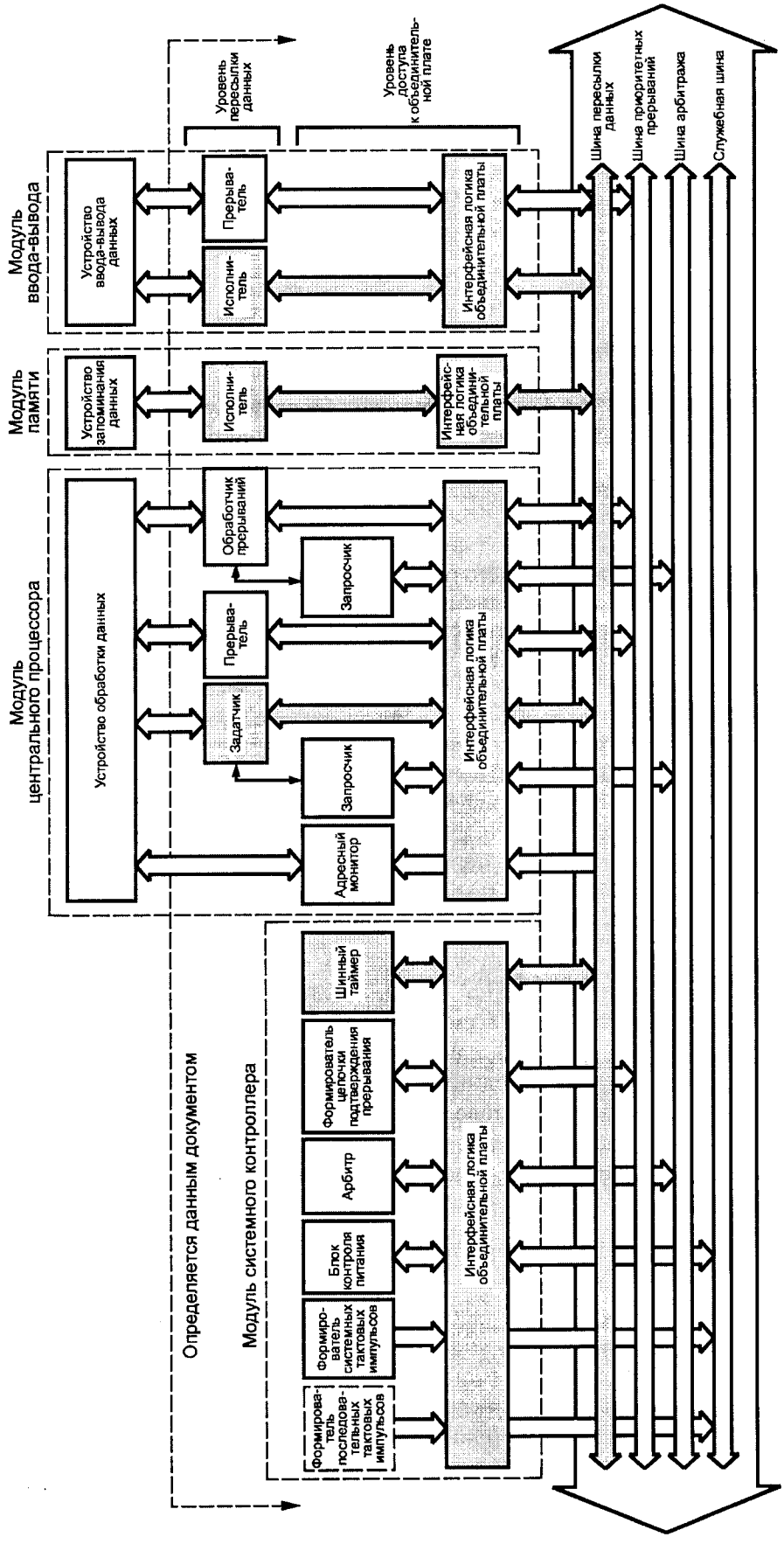


Рисунок 2.1 — Схема взаимодействия шины пересылки данных с функциональными блоками

2.2 Линии шины пересылки данных

Линии шины пересылки данных подразделяются на три группы:

-линии адресации: A01—A31 (address lines 01—31),
AM0—AM5 (address modifier lines 0—5),
DS0* (data strobe 0*),
DS1* (data strobe 1*),
LWORD* (long word*);

- линии данных: D00—D31 (data lines 00—31);

- линии управления: AS* (address strobe),
DS0*,
DS1*,
BERR* (bus error),
DTACK* (data acknowledge*),
WRITE*.

Замечание 2.1. Два строба данных (DS0* и DS1*) выполняют двоякую функцию:

- их логические уровни используются для выбора байта (байтов), к которому (ым) осуществляется обращение;

- их перепады используются также в качестве синхросигналов для координирования пересылки данных между задатчиком и исполнителем.

2.2.1 Линии адресации

Наименьшей адресуемой единицей запоминающего устройства является байтовая ячейка. Каждой байтовой ячейке присваивается уникальный двоичный адрес. Каждая байтовая ячейка может быть отнесена к одной из четырех категорий в соответствии со значением двух младших разрядов ее адреса (см. таблицу 2.1).

Таблица 2.1 — Категории байтовых ячеек

Категория	Адрес байта
БАЙТ (0)	XXXXXX.....XXXXXX00
БАЙТ (1)	XXXXXX.....XXXXXX01
БАЙТ (2)	XXXXXX.....XXXXXX10
БАЙТ (3)	XXXXXX.....XXXXXX11

Набор байтовых ячеек, адреса которых различаются только двумя младшими разрядами, называется четырехбайтовой группой или группой БАЙТЫ (0—3). За один цикл шины пересылки данных можно одновременно обратиться к части или ко всем байтам четырехбайтовой группы.

Задатчики используют адресные линии A02—A31 для выбора четырехбайтовой группы, к которой будет происходить обращение. Четыре дополнительные линии (DS1*, DS0*, A01 и LWORD*), в свою очередь, выбирают, к какой байтовой ячейке (ячейкам) в пределах этой четырехбайтовой группы происходит обращение во время пересылки данных. Используя эти четыре линии, задатчик может обращаться к одной, двум, трем или четырем байтовым ячейкам одновременно, как показано в таблице 2.2.

Таблица 2.2 — Использование сигналов DS0*, DS1*, A01 и LWORD* для выбора байтовых ячеек

Выбираемые байтовые ячейки	DS1*	DS0*	A01	LWORD*
Однобайтовый доступ				
БАЙТ (0)	Низкий	Высокий	Низкий	Высокий
БАЙТ (1)	Высокий	Низкий	Низкий	Высокий
БАЙТ (2)	Низкий	Высокий	Высокий	Высокий
БАЙТ (3)	Высокий	Низкий	Высокий	Высокий

Окончание таблицы 2.2

Выбираемые байтовые ячейки		DS1*	DS0*	A01	LWORD*
Двухбайтовый доступ					
БАЙТЫ (0—1)	низкий	Низкий	Низкий	Высокий	
БАЙТЫ (1—2)	низкий	Низкий	Высокий	Низкий	
БАЙТЫ (2—3)	низкий	Низкий	Высокий	Высокий	
Трехбайтовый доступ					
БАЙТЫ (0—2)	низкий	Высокий	Низкий	Низкий	
БАЙТЫ (1—3)	высокий	Низкий	Низкий	Низкий	
Четырехбайтовый доступ					
БАЙТЫ (0—3)	низкий	Низкий	Низкий	Низкий	

Замечание 2.2. При установке обоих стробов данных низкими один строб данных может установиться низким несколько позднее другого. В этом случае уровни сигналов, указанные в таблице 2.2, подразумевают окончательно установившиеся уровни.

Замечание 2.3. Для данных уровней четырех сигнальных линий, указанных в таблице 2.2, возможны 16 комбинаций. Из этих 16 две комбинации являются запрещенными и не используются (см. правило 2.1).

Правило 2.1. *Задатчики НЕ ДОЛЖНЫ ФОРМИРОВАТЬ циклов шины пересылки данных в случаях, когда окончательные уровни сигналов DS0*, DS1*, A01 и LWORD* составляют одну из следующих запрещенных комбинаций:*

*DS1**	*DS0**	*A01*	*LWORD**
высокий	*низкий*	*высокий*	*низкий*
низкий	*высокий*	*высокий*	*низкий*

Разрешение 2.1. Когда задатчик обращается к байтовым ячейкам БАЙТЫ (1—2) (см. таблицу 2.2), он МОЖЕТ сформировать на короткое время в виде переходных состояний одну из двух комбинаций, приведенных в правиле 2.1 (т.е. в то время, как один строб данных перейдет на низкий логический уровень, а другой на него не перейдет).

Замечание 2.4. Всякий раз, когда задатчик устанавливает LWORD* низким и A01 высоким, он устанавливает оба строба данных низкими (любая другая комбинация является запрещенной). Проектировщики модулей могут воспользоваться этим обстоятельством для упрощения логической схемы исполнителей.

Разрешение 2.2. С целью упрощения требуемой схемы исполнители, отвечающие на какие-либо циклы, в которых происходит обращение к байтовым ячейкам БАЙТЫ (1—2) (см. таблицу 2.2), МОГУТ быть спроектированы без логики, отличающей эти циклы от двух запрещенных циклов, указанных в правиле 2.1.

2.2.2 Линии модификатора адреса

Имеется шесть линий модификатора адреса. Они позволяют задатчику отправлять дополнительную двоичную информацию исполнителю во время циклов шины пересылки данных. В таблице 2.3 указаны все 64 возможных кода модификатора адреса, которые подразделены на три класса.

а) Определяемые стандартом, которые включают:

- коды модификатора адреса короткой адресации, указывающие, что для выбора группы БАЙТЫ (0—3) используются адресные линии A02—A15;
- коды модификатора адреса стандартной адресации, указывающие, что для выбора группы БАЙТЫ (0—3) используются адресные линии A02—A23;
- коды модификатора адреса расширенной адресации, указывающие, что для выбора группы БАЙТЫ (0—3) используются адресные линии A02—A31.

б) Зарезервированные коды.

в) Коды, определяемые пользователем.

Таблица 2.3 — **Коды модификатора адреса**

Шестнадцатеричный код	Линии модификатора адреса 5	4	3	2	1	0	Функция
3F	H	H	H	H	H	H	Стандартная супервизорная блочная пересылка
3E	H	H	H	H	H	L	Стандартный супервизорный доступ к программе
3D	H	H	H	H	L	H	Стандартный супервизорный доступ к данным
3C	H	H	H	H	L	L	Зарезервирован
3B	H	H	H	L	H	H	Стандартная непривилегированная блочная пересылка
3A	H	H	H	L	H	L	Стандартный непривилегированный доступ к программе
39	H	H	H	L	L	H	Стандартный непривилегированный доступ к данным
38	H	H	H	L	L	L	Зарезервирован
37	H	H	L	H	H	H	Зарезервирован
36	H	H	L	H	H	L	Зарезервирован
35	H	H	L	H	L	H	Зарезервирован
34	H	H	L	H	L	L	Зарезервирован
33	H	H	L	L	H	H	Зарезервирован
32	H	H	L	L	H	L	Зарезервирован
31	H	H	L	L	L	H	Зарезервирован
30	H	H	L	L	L	L	Зарезервирован
2F	H	L	H	H	H	H	Зарезервирован
2E	H	L	H	H	H	L	Зарезервирован
2D	H	L	H	H	L	H	Короткий супервизорный доступ
2C	H	L	H	H	L	L	Зарезервирован
2B	H	L	H	L	H	H	Зарезервирован
2A	H	L	H	L	H	L	Зарезервирован
29	H	L	H	L	L	H	Короткий непривилегированный доступ
28	H	L	H	L	L	L	Зарезервирован
27	H	L	L	H	H	H	Зарезервирован
26	H	L	L	H	H	L	Зарезервирован
25	H	L	L	H	L	H	Зарезервирован
24	H	L	L	H	L	L	Зарезервирован
23	H	L	L	L	H	H	Зарезервирован
22	H	L	L	L	H	L	Зарезервирован
21	H	L	L	L	L	H	Зарезервирован
20	H	L	L	L	L	L	Зарезервирован
1F	L	H	H	H	H	H	Зарезервирован
1E	L	H	H	H	H	L	Определяемая пользователем
1D	L	H	H	H	L	H	Определяемая пользователем
1C	L	H	H	H	L	L	Определяемая пользователем
1B	L	H	H	L	H	H	Определяемая пользователем
1A	L	H	H	L	H	L	Определяемая пользователем
19	L	H	H	L	L	H	Определяемая пользователем
18	L	H	H	L	L	L	Определяемая пользователем
17	L	H	L	H	H	H	Определяемая пользователем
16	L	H	L	H	H	L	Определяемая пользователем
15	L	H	L	H	L	H	Определяемая пользователем
14	L	H	L	H	L	L	Определяемая пользователем
13	L	H	L	L	H	H	Определяемая пользователем
12	L	H	L	L	H	L	Определяемая пользователем
11	L	H	L	L	L	H	Определяемая пользователем
10	L	H	L	L	L	L	Определяемая пользователем
0F	L	L	H	H	H	H	Расширенная супервизорная блочная пересылка
0E	L	L	H	H	H	L	Расширенный супервизорный доступ к программе
0D	L	L	H	H	L	H	Расширенный супервизорный доступ к данным
0C	L	L	H	H	L	L	Зарезервирован
0B	L	L	H	L	H	H	Расширенная непривилегированная блочная пересылка
0A	L	L	H	L	H	L	Расширенный непривилегированный доступ к программе
09	L	L	H	L	L	H	Расширенный непривилегированный доступ к данным

Окончание таблицы 2.3

Шестнадцатеричный код	Линии модификатора адреса 5	4	3	2	1	0	Функция
08	L	L	H	L	L	L	Зарезервирован
07	L	L	L	H	H	H	Зарезервирован
06	L	L	L	H	H	L	Зарезервирован
05	L	L	L	H	L	H	Зарезервирован
04	L	L	L	H	L	L	Зарезервирован
03	L	L	L	L	H	H	Зарезервирован
02	L	L	L	L	H	L	Зарезервирован
01	L	L	L	L	L	H	Зарезервирован
00	L	L	L	L	L	L	Зарезервирован
L — низкий логический уровень сигнала; H — высокий логический уровень сигнала.							

Правило 2.2. *Коды, указанные в таблице 2.3, за исключением кодов, определяемых пользователем, НЕ ДОЛЖНЫ ИСПОЛЬЗОВАТЬСЯ в иных целях, чем это указанно.*

Правило 2.3. *Исполнители НЕ ДОЛЖНЫ ОТВЕЧАТЬ на зарезервированные коды модификатора адреса.*

Замечание 2.5. Зарезервированные коды адресного модификатора предназначены для расширения в будущем функциональных возможностей системы. Если исполнители отвечают на такие коды, это может привести к их несовместимости с тем их конкретным использованием, которое будет определено позднее.

Разрешение 2.3. Коды, определяемые пользователем, МОГУТ использоваться для любых целей, которые сочтут нужными фирмы-изготовители и пользователи модулей (переключение страниц памяти, защита памяти, идентификация задатчика или задачи, привилегированный доступ к ресурсам и т.д.).

Рекомендация 2.1. Чтобы позволить пользователям приспособить для своих нужд определяемые ими коды модификатора адреса, необходимо дешифровать их удобным способом в модулях исполнителей. Далее пользователи могут конфигурировать модуль, выполняя любое декодирование, которое необходимо их системе.

Замечание 2.6. Удобным средством декодирования кодов модификатора адреса являются устанавливаемые в панельки на плату программируемые устройства.

Предложение 2.1. В тех случаях, когда исполнители содержат программируемые элементы (например, ППЗУ или ПЛМ), устанавливаемые в панельки на платы, предлагается эти элементы программировать так, чтобы указанные типы исполнителей отвечали на следующие коды модификатора адреса:

исполнители A16 (address lines A01—A15) с функцией
D08(O) (data lines 00—07, odd) ,
D08(EO) (data lines D00—D07/D08—D15 even/odd),
D16 (data lines D00—D15) или
D32 (data lines D00—D31) : 29, 2D;

исполнители A24 (address lines A01—A23) с функцией
D08(O), D08(EO),
D16 или D32 : 39, 3A, 3D и 3E;

исполнители A32 (address lines A01—A31) с функцией
D08(O), D08(EO),
D16 или D32 : 09, 0A, 0D и 0E;

исполнители A24 с функцией BLT (block transfer) : 3B, 3F;

исполнители A32 с функцией BLT : 0B, 0F.

Мнемонические обозначения A16, A24 и A32 определены в таблице 2.9, а D08(O), D08(EO), D16, D32 и BLT — в таблицах 2.10 и 2.11.

2.2.3 Линии данных

Системы можно построить с двумя конфигурациями объединительной платы; первая обладает 16 линиями данных (D00—D15), вторая — 32 линиями данных (D00—D31). Конфигурация объединительной платы, имеющей 16 линий данных, позволяет задатчику обращаться одновременно только к двум байтовым ячейкам, тогда как объединительная плата с 32 линиями данных позволяет задатчику одновременно обращаться к четырем байтовым ячейкам. Когда задатчик выбрал одну, две, три или четыре байтовые ячейки в соответствии с методом, приведенным в 2.2.1, он может пересылать данные между собой и этими ячейками по линиям данных. Таблица 2.4 показывает, как используются линии данных для доступа к байтовым ячейкам.

Таблица 2.4 — Использование линий данных для доступа к байтовым ячейкам

Доступ к байтовым ячейкам	D24—D31	D16—D23	D08—D15	D00—D07
БАЙТ (0)			БАЙТ (0)	
БАЙТ (1)				БАЙТ (1)
БАЙТ (2)			БАЙТ (2)	
БАЙТ (3)				БАЙТ (3)
БАЙТЫ (0—1)			БАЙТ (0)	БАЙТ (1)
БАЙТЫ (1—2)		БАЙТ (1)	БАЙТ (2)	
БАЙТЫ (2—3)			БАЙТ (2)	БАЙТ (3)
БАЙТЫ (0—2)	БАЙТ (0)	БАЙТ (1)	БАЙТ (2)	
БАЙТЫ (1—3)		БАЙТ (1)	БАЙТ (2)	БАЙТ (3)
БАЙТЫ (0—3)	БАЙТ (0)	БАЙТ (1)	БАЙТ (2)	БАЙТ (3)

2.2.4 Линии управления шины пересылки данных

Для управления пересылкой данных по шине используются следующие сигнальные линии:

AS* — линия адресного строба (далее — линия AS*)

DS0* — линия строба данных ноль (далее — линия DS0*)

DS1* — линия строба данных один (далее — линия DS1*)

BERR* — линия сигнала ошибки магистрали (далее — линия BERR*)

DTACK* — линия сигнала подтверждения пересылки данных (далее — линия DTACK*)

WRITE* — линия сигнала считывания/записи (далее — линия WRITE*)

2.2.4.1 *Линия AS**

Отрицательный перепад сигнала на линии AS* информирует всех исполнителей, что адрес стабилен и может быть принят.

2.2.4.2 *Линии DS0* и DS1**

Кроме своих функций выбора байтовых ячеек для пересылки данных, как это описано в 2.2.1, стробы данных выполняют также дополнительные функции. При циклах записи первый отрицательный перепад строба данных указывает, что задатчик поместил на шину данных достоверные данные. При циклах считывания первый положительный перепад сообщает исполнителю, что тот может снять свои данные с шины данных.

Замечание 2.7. Как указано в 2.6, задатчикам не разрешается устанавливать низким ни тот, ни другой строб данных до установки низким AS*. Однако из-за того обстоятельства, что линия AS* может быть загружена на объединительной плате значительно больше, чем линии стробов данных, исполнители и адресные мониторы могут обнаружить отрицательный перепад на линиях стробов данных до того, как они обнаружат отрицательный перепад на линии AS*.

Разрешение 2.5. Исполнители, не имеющие функциональной возможности блочной пересылки (см. 2.3.7), и адресные мониторы МОГУТ быть спроектированы для захвата адреса, когда они обнаруживают отрицательный перепад на любой линии строба данных, а не по отрицательному перепаду AS*.

Замечание 2.8. Исполнителям и адресным мониторам, которые захватывают адрес по отрицательному перепаду строба (стробов) данных, нет необходимости контролировать линию AS*.

Замечание 2.9. Чтобы полностью воспользоваться преимуществами конвейерной адресации (см. 2.4.2) или выполнять циклы блочного считывания и записи, исполнителю следует захватывать адрес по отрицательному перепаду AS*.

2.2.4.3 *Линия DTACK**

Исполнитель устанавливает DTACK* низким, чтобы показать, что он успешно принял данные в цикле записи. В цикле считывания исполнитель устанавливает DTACK* низким, чтобы показать, что он поместил данные на линии данных.

2.2.4.4 *Линия BERR**

BERR* устанавливается исполнителем или шинным таймером низким, чтобы показать задатчику, что пересылка данных оказалась безуспешной. Например, если задатчик пытается осуществить запись в ячейку ПЗУ, отвечающий исполнитель может установить BERR* низким. Если задатчик пытается осуществить обращение к ячейке, которая не предоставлена ни одним из исполнителей, то после какого-то заданного временного интервала (тайм-аута) шинный таймер устанавливает BERR* низким.

Предложение 2.2. Предлагается проектировать исполнители так, чтобы они отвечали отрицательным перепадом на линии BERR* при обнаружении неисправимой ошибки в данных, которые они выбирают из своего внутреннего запоминающего устройства во время цикла считывания.

2.2.4.5 *Линия WRITE**

Для сигнала на линии WRITE*, который стробируется отрицательным перепадом первого поступающего строба данных, главное значение имеет его логический уровень. Он используется задатчиком, чтобы показать направление операций пересылки данных. Если WRITE* устанавливается низким, направление пересылки данных — от задатчика к исполнителю. Если WRITE* устанавливается высоким, направление пересылки данных — от исполнителя к задатчику.

2.3 Функциональные блоки шины пересылки данных

Кроме цикла Только Адрес, протокол шины пересылки данных определяет еще 33 различных типа циклов, которые используются для пересылки данных. Каждый из этих 34 циклов может использоваться в любом из трех режимов адресации:

- короткий режим (16 разрядов);
- стандартный режим (24 разряда);
- расширенный режим (32 разряда).

Функциональные возможности задатчиков, исполнителей и адресных мониторов описаны с помощью мнемонических обозначений, которые указывают, какие типы циклов эти блоки могут, соответственно, формировать, принимать или контролировать. Эти мнемонические обозначения описаны в 2.3.5 — 2.3.10.

В 2.3.1 — 2.3.4 описаны схемы взаимодействия с магистралью четырех типов функциональных блоков шины пересылки данных: задатчика, исполнителя, адресного монитора и шинного таймера.

Правило 2.7. *Линии выходных сигналов, показанные на рисунках 2.2— 2.5 сплошными линиями, ДОЛЖНЫ ВОЗБУЖДАТЬСЯ соответствующим функциональным блоком, если он не устанавливает на них высокий уровень постоянно.*

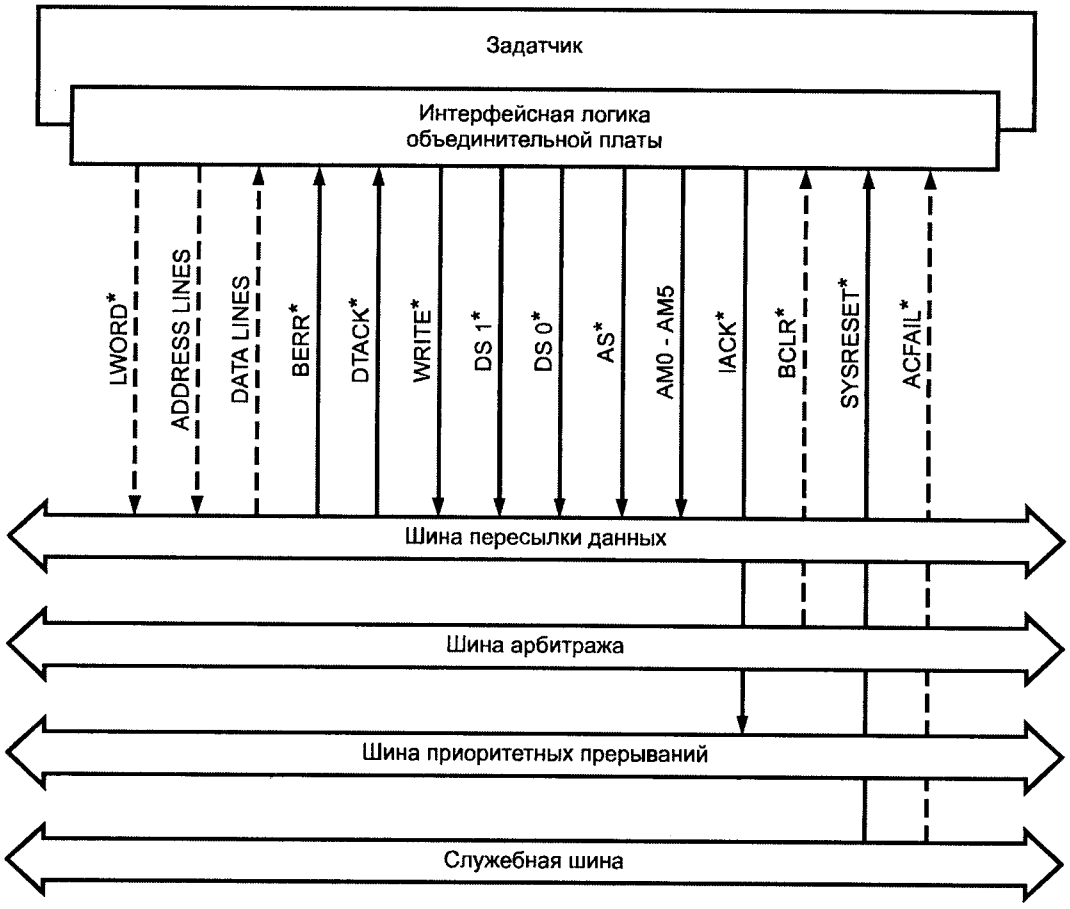
Замечание 2.11. Если линии выходных сигналов не возбуждены, то установка на них сигнала высокого уровня гарантируется оконечными нагрузками объединительной платы.

Правило 2.8. *Линии входных сигналов, показанные на рисунках 2.2— 2.5 сплошными линиями, ДОЛЖНЫ КОНТРОЛИРОВАТЬСЯ и на их сигналы ДОЛЖНЫ ДАВАТЬСЯ соответствующие ответы.*

Замечание 2.12. Правила и разрешения для возбуждения и контроля сигнальных линий, показанных на рисунках 2.2, 2.3 и 2.5 пунктиром, приведены в таблицах 2.5, 2.6 и 2.8.

2.3.1 Задатчик

Схема взаимодействия задатчика с магистралью приведена на рисунке 2.2. Пунктиром показаны сигнальные линии, использование которых меняется в зависимости от типа задатчиков. Таблица 2.5 устанавливает требования для различных типов задатчиков по возбуждению и контролю этих линий. Прочие требования, описывающие, как различные типы задатчиков возбуждают адресные линии, линии данных и линии LWORD*, DS0*, DS1* и A01, приведены в таблицах 2.19 — 2.21.



IACK* — interrupt acknowledge;
BCLR* — bus clear;
SYSRESET* — system reset;
ACFAIL* — AC failure

Рисунок 2.2 — Схема взаимодействия задатчика с магистралью

Таблица 2.5 — Задатчики. Правила и разрешения для контроля и возбуждения сигнальных линий, отмеченных на рисунке 2.2 пунктиром

Тип задатчика	Правила и разрешения
D08(EO) и D16	*ДОЛЖНЫ КОНТРОЛИРОВАТЬ и ВОЗБУЖДАТЬ линии D00—D15* МОГУТ возбуждать или МОГУТ НЕ возбуждать линию LWORD* МОГУТ возбуждать или контролировать либо МОГУТ НЕ возбуждать или НЕ контролировать линии D16—D31
D32	*ДОЛЖЕН ВОЗБУЖДАТЬ линию LWORD** *ДОЛЖЕН КОНТРОЛИРОВАТЬ и ВОЗБУЖДАТЬ линии D00—D31*
A16	*ДОЛЖЕН ВОЗБУЖДАТЬ линии A01—A15* МОЖЕТ возбуждать или МОЖЕТ НЕ возбуждать линии A16—A31
A24	*ДОЛЖЕН ВОЗБУЖДАТЬ линии A01—A23* МОЖЕТ возбуждать или МОЖЕТ НЕ возбуждать линии A24—A31
A32	*ДОЛЖЕН ВОЗБУЖДАТЬ линии A01—A31*
Все	МОГУТ контролировать или МОГУТ НЕ контролировать линии BCLR* или ACFAIL* (см. разделы 3 и 5)

Примечания
1 Мнемонические обозначения D08(EO), D16 и D32 определены в таблице 2.10.
2 Мнемонические обозначения A16, A24 и A32 определены в таблице 2.9.

2.3.2 Исполнитель

Схема взаимодействия исполнителя с магистралью приведена на рисунке 2.3. Пунктиром показаны сигнальные линии, использование которых меняется в зависимости от различных типов исполнителей. Таблица 2.6 устанавливает требования по возбуждению и контролю этих линий для различных типов исполнителей. Прочая информация о том, как различные типы исполнителей возбуждают линии данных, приведена в таблице 2.21.

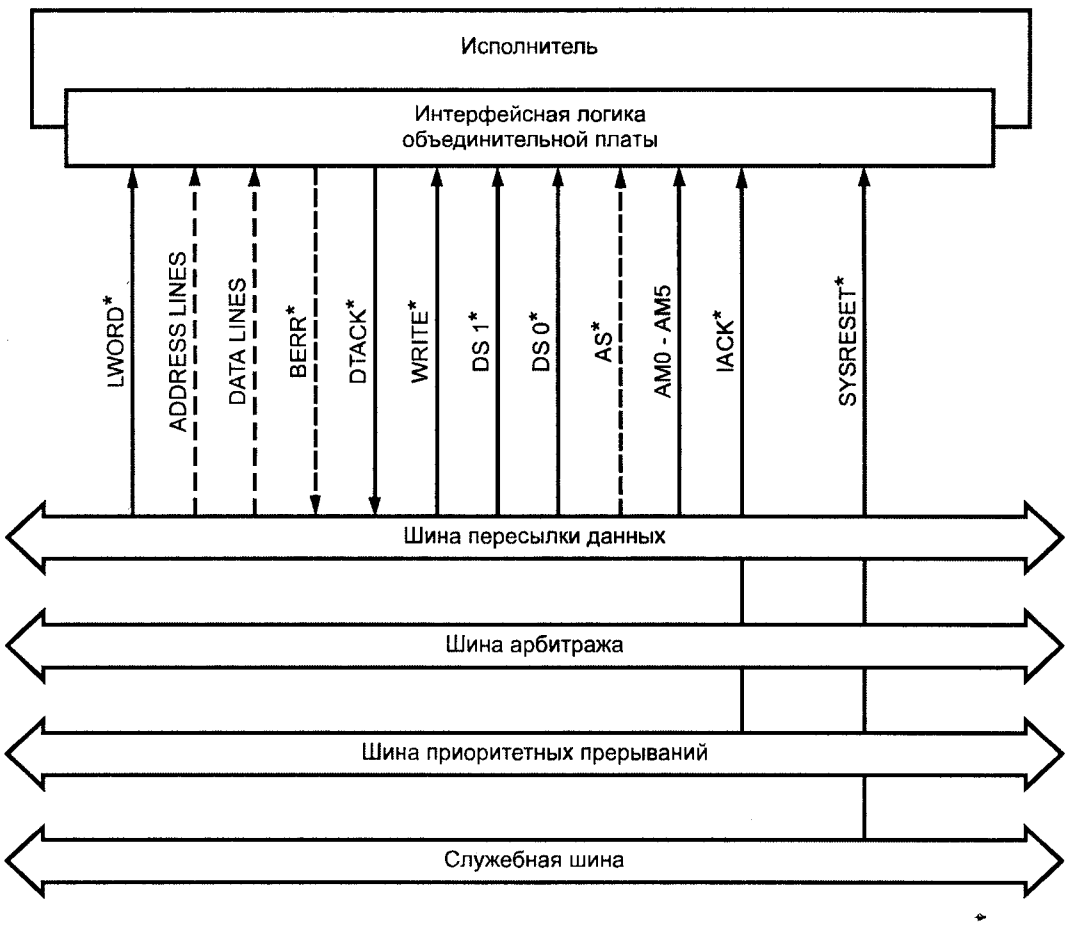


Рисунок 2.3 — Схема взаимодействия исполнителя с магистралью

Таблица 2.6 — Исполнители. Правила и разрешения для контроля и возбуждения сигнальных линий, отмеченных на рисунке 2.3 пунктиром

Тип исполнителя	Правила и разрешения
D08(O)	*ДОЛЖЕН КОНТРОЛИРОВАТЬ и ВОЗБУЖДАТЬ линии D00—D07*
D08(EO) и D16	*ДОЛЖЕН КОНТРОЛИРОВАТЬ и ВОЗБУЖДАТЬ линии D00—D15* МОГУТ контролировать или возбуждать либо МОГУТ НЕ контролировать или НЕ возбуждать линии D16—D31 МОГУТ контролировать или МОГУТ НЕ контролировать линию AS*
D32	*ДОЛЖЕН КОНТРОЛИРОВАТЬ и ВОЗБУЖДАТЬ линии D00—D31* МОЖЕТ контролировать или МОЖЕТ НЕ контролировать линию AS*
BLT	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линию AS**
A16	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линии A01—A15* МОЖЕТ контролировать или МОЖЕТ НЕ контролировать линии A16—A31

Окончание таблицы 2.6

Тип исполнителя	Правила и разрешения
A24	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линии A01—A23* МОЖЕТ контролировать или МОЖЕТ НЕ контролировать линии A24—A31
A32 Все	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линии A01—A31* МОГУТ возбуждать или МОГУТ НЕ возбуждать линию BERR*

Примечания
1 Мнемонические обозначения D08(O), D08(EO), D16 и D32 определены в таблице 2.10.
2 Мнемоническое обозначение BLT определено в таблице 2.11.
3 Мнемонические обозначения A16, A24 и A32 определены в таблице 2.9.

2.3.3 Шинный таймер

Схема взаимодействия шинного таймера с магистралью приведена на рисунке 2.4. Шинные таймеры могут быть спроектированы для установки BERR* низким по истечении различных временных интервалов (тайм-аута). В таблице 2.7 показан порядок использования мнемоники BTO(x) (bus timeout (x)) для описания различных типов шинных таймеров.

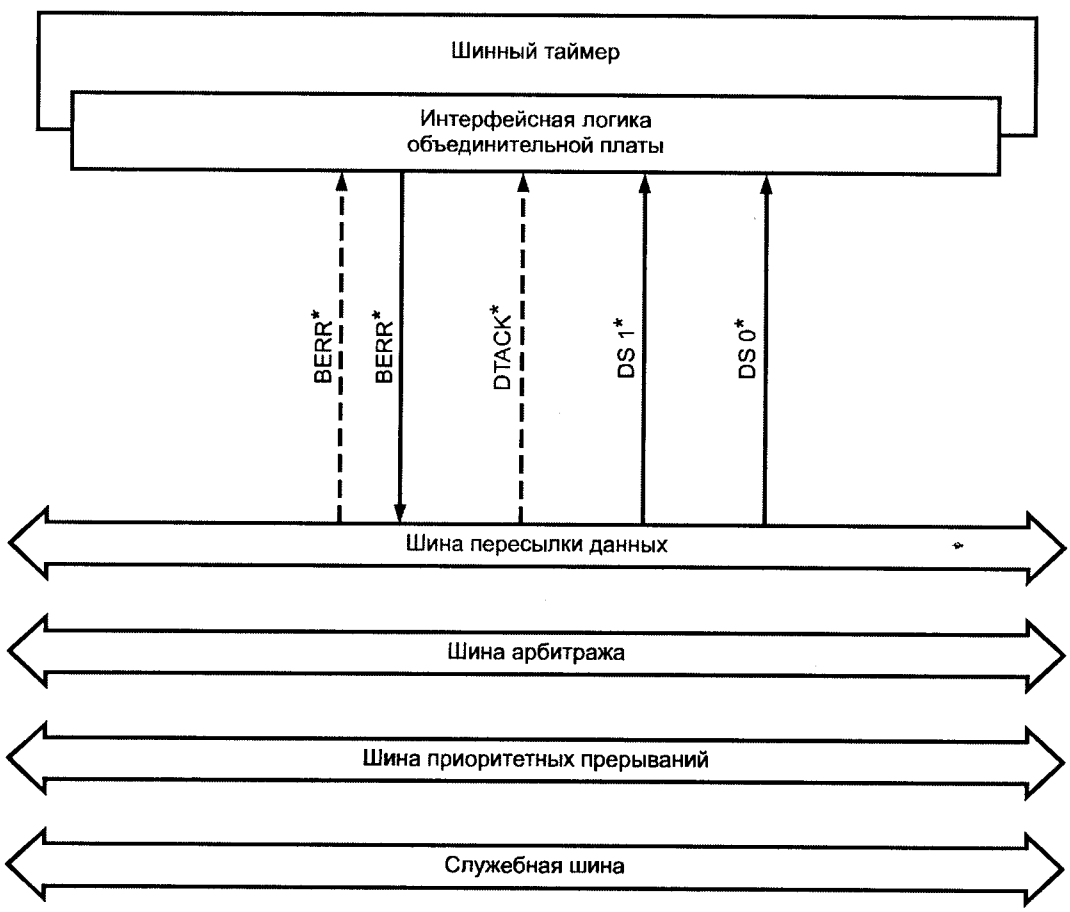


Рисунок 2.4 — Схема взаимодействия шинного таймера с магистралью

Таблица 2.7 — Использование мнемонического обозначения BTO(x) для указания тайм-аута шинных таймеров

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
BTO(x)	Шинный таймер	Устанавливает BERR* низким, когда первый строб данных остается низким в течение временного интервала, превышающего *x* мкс

Замечание 2.13. Обозначенные на рисунке 2.4 пунктиром линии DTACK* и BERR* показывают, что шинный таймер можно реализовать в двух вариантах:

а) устанавливать BERR* низким, когда первый строб данных остается низким в течение времени, превышающего тайм-аут, независимо от уровней на линиях DTACK* и BERR*;

б) устанавливать BERR* низким, когда первый строб данных остается низким в течение времени, превышающего тайм-аут, но только когда обе линии DTACK* и BERR* имеют высокий уровень в момент окончания тайм-аута.

2.3.4 Адресный монитор

Схема взаимодействия адресного монитора с магистралью показана на рисунке 2.5. Пунктиром показаны сигнальные линии, использование которых меняется в зависимости от различных типов адресных мониторов. Таблица 2.8 устанавливает требования по контролю этих линий для различных типов адресных мониторов.

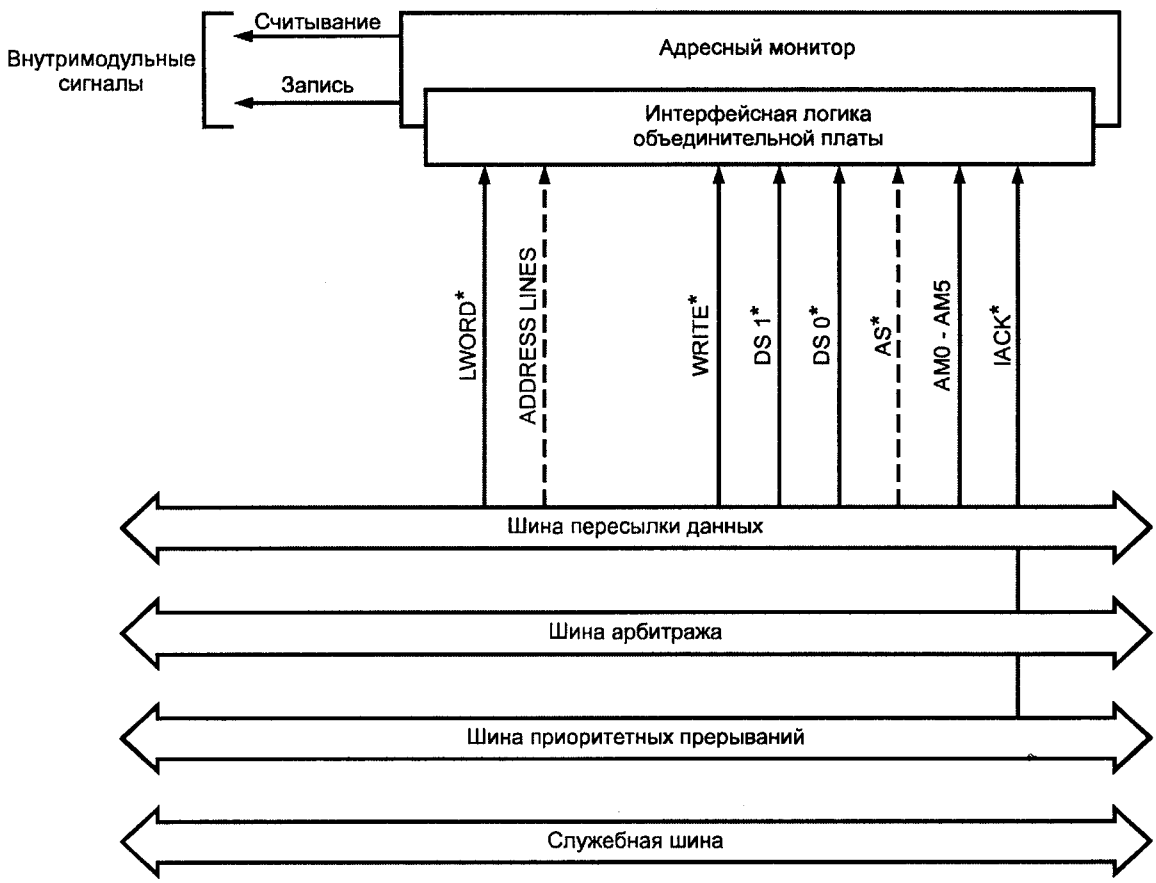


Рисунок 2.5 — Схема взаимодействия адресного монитора с магистралью

Таблица 2.8 — Адресный монитор. Правила и разрешения для контроля сигнальных линий, отмеченных на рисунке 2.5 пунктиром

Тип адресного монитора	Правила и разрешения
A16	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линии A01—A15* МОЖЕТ контролировать или МОЖЕТ НЕ контролировать линии A16—A31
A24	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линии A01—A23* МОЖЕТ контролировать или МОЖЕТ НЕ контролировать линии A24—A31
A32	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линии A01—A31*
Все	МОГУТ контролировать или МОГУТ НЕ контролировать линию AS*
Примечание — Мнемонические обозначения A16, A24 и A32 определены в таблице 2.9.	

2.3.5 Режимы адресации

Задатчики широковещательно передают адрес по шине пересылки данных в начале каждого цикла. Этот адрес может быть 16-разрядным, 24-разрядным или 32-разрядным в зависимости от функциональных возможностей задатчика, который выполняет его широковещательную передачу. 16-разрядные адреса называются короткими адресами, 24-разрядные — стандартными, а 32-разрядные — расширенными. В табл. 2.9 приведены различные мнемонические обозначения, употребляемые для описания возможностей адресации, и показано, как они описывают возможности задатчиков, исполнителей и адресных мониторов.

Таблица 2.9 — Мнемонические обозначения режимов адресации

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
A16	Задатчик	Может формировать циклы с короткими (16-разрядными) адресами
	Исполнитель	Может воспринимать циклы с короткими (16-разрядными) адресами
	Адресный монитор	Может контролировать циклы с короткими (16-разрядными) адресами
A24	Задатчик	Может формировать циклы со стандартными (24-разрядными) адресами
	Исполнитель	Может воспринимать циклы со стандартными (24-разрядными) адресами
	Адресный монитор	Может контролировать циклы со стандартными (24-разрядными) адресами
A32	Задатчик	Может формировать циклы с расширенными (32-разрядными) адресами
	Исполнитель	Может воспринимать циклы с расширенными (32-разрядными) адресами
	Адресный монитор	Может контролировать циклы с расширенными (32-разрядными) адресами

С каждым адресом задатчик широковещательно посылает код модификатора адреса (AM), чтобы проинформировать исполнителя, какой адрес (короткий, стандартный или расширенный) передается.

Короткие адреса вырабатываются задатчиками A16 и принимаются исполнителями A16. Стандартные адреса вырабатываются задатчиками A24 и принимаются исполнителями A24. Расширенные адреса вырабатываются задатчиками A32 и принимаются исполнителями A32.

Режим короткой адресации в основном предназначен для обращения к устройствам ввода/вывода. Он позволяет проектировать исполнители A16 с минимальной логикой, так как им нет необходимости декодировать большое количество адресных линий. Хотя модули ввода/вывода можно спроектировать для декодирования стандартных и расширенных адресов, режим короткой адресации обычно делает такой подход излишним.

Режимы стандартной и расширенной адресации предназначены в основном для обращения к памяти, хотя нет никаких запретов против проектирования модулей ввода/вывода, в которых эти режимы были бы реализованы. Стандартный и расширенный режимы адресации позволяют значительно увеличить количество адресуемых ячеек.

Правило 2.9. *Исполнители ДОЛЖНЫ ДЕКОДИРОВАТЬ все линии модификатора адреса.*

Замечание 2.14. Возможность декодирования всех линий модификатора адреса позволяет исполнителю различать короткие, стандартные и расширенные адреса.

Замечание 2.15. Кроме указанных здесь трех режимов адресации, существует четвертый режим, который используется при циклах подтверждения прерывания (см. раздел 4). Эти циклы подтверждения прерывания можно отличить от циклов пересылки данных тем, что в первом случае сигнальная линия IACK* имеет низкий логический уровень, а во втором — высокий.

Правило 2.10. *Всякий раз, когда задатчик широковещательно передает адрес по линиям адреса, он ДОЛЖЕН ОБЕСПЕЧИТЬ установку IACK* высоким.*

Разрешение 2.7. Задатчик МОЖЕТ либо установить IACK* высоким во время широковещательной передачи адреса, либо оставить линию IACK* невозбужденной (в последнем случае на ней установят высокий уровень оконечные нагрузки шины).

Правило 2.11. *Когда IACK* находится в низком состоянии, исполнители НЕ ДОЛЖНЫ ОТВЕЧАТЬ на циклы шины пересылки данных.*

Правило 2.61. *Задатчики A32 ДОЛЖНЫ ОБЛАДАТЬ функциональными возможностями A24 и A16.*

Правило 2.62. *Задатчики A24 ДОЛЖНЫ ОБЛАДАТЬ функциональной возможностью A16.*

Предложение 2.6. Не следует полагать, что приведенные выше правила известны читателям документации, содержащей технические характеристики конкретных изделий. Поэтому в документации на изделия, являющиеся задатчиками A32, предлагается указывать, что они обладают функциональными возможностями A32, A24 и A16, а на изделия, являющиеся задатчиками A24, — указывать, что они обладают функциональными возможностями A24 и A16.

2.3.6 Основные функциональные возможности пересылки данных

Существуют четыре основных возможности пересылки данных по шине пересылки данных: D08(O) (только нечетные байты), D08(EO) (четные и нечетные байты), D16 и D32. Эти возможности удобны для сопряжения с шиной процессоров и периферийных устройств различных типов.

8-разрядные процессоры можно сопрягать с шиной в качестве задатчиков D08(EO). 16-разрядные процессоры — в качестве задатчиков D16. Исполнитель D16 целесообразно использовать для сопряжения с шиной пересылки данных 16-разрядных устройств памяти или исполнителей 16-разрядных устройств ввода/вывода.

Многие выпускаемые промышленностью интегральные схемы (ИС) периферийных устройств имеют только 8-разрядные регистры. Несмотря на то, что эти ИС имеют в своем составе несколько таких регистров, они не могут предоставлять одновременно содержимое двух регистров, когда задатчик D16 делает попытки обратиться к двум соседним ячейкам в цикле двухбайтового считывания. Такие 8-разрядные ИС периферийных устройств можно сопрягать с шиной пересылки данных в качестве исполнителей D08(O), которые обеспечивают прием и передачу содержимых БАЙТА (1) или БАЙТА (3) и отвечают только на обращения к одному нечетному байту. Это позволяет упростить логику сопряжения исполнителя D08(O), поскольку обращения к одному нечетному байту всегда происходят с использованием линий D00—D07.

Правило 2.63. *Задатчики D32, исполнители D32 и адресные мониторы D32 ДОЛЖНЫ ОБЛАДАТЬ функциональными возможностями D08(EO) и D16.*

Правило 2.64. *Задатчики D16, исполнители D16 и адресные мониторы D16 ДОЛЖНЫ ОБЛАДАТЬ функциональной возможностью D08 (EO).*

Предложение 2.7. Не следует полагать, что приведенные выше правила известны читателям документации, содержащей технические характеристики конкретных изделий. Поэтому в документации на изделия, обладающие функциональной возможностью D32, предлагается указывать, что они обладают функциональными возможностями D32, D16 и D08(EO), а на изделия, обладающие функциональной возможностью D16, — указывать, что они обладают функциональными возможностями D16 и D08(EO).

Правило 2.4. *Исполнители D16 НЕ ДОЛЖНЫ ОТВЕЧАТЬ установкой DTACK* низким во время исполнения циклов, которые запрашивают обращение к байтовым ячейкам БАЙТЫ (1—2), БАЙТЫ (0—2), БАЙТЫ (1—3) или БАЙТЫ (0—3).*

Правило 2.5. *Исполнители D08(EO) НЕ ДОЛЖНЫ ОТВЕЧАТЬ установкой DTACK* низким во время исполнения циклов, которые запрашивают обращение к байтовым ячейкам БАЙТЫ (0—1), БАЙТЫ (1—2), БАЙТЫ (2—3), БАЙТЫ (0—2), БАЙТЫ (1—3) или БАЙТЫ (0—3).*

Правило 2.65. *Исполнители D08(O) НЕ ДОЛЖНЫ ОТВЕЧАТЬ отрицательным перепадом сигнала на линии DTACK* во время исполнения циклов, которые запрашивают обращение к байтовым ячейкам БАЙТ (0), БАЙТ(2), БАЙТЫ(0—1), БАЙТЫ(1—2), БАЙТЫ(2—3), БАЙТЫ(0—2), БАЙТЫ(1—3) или БАЙТЫ(0—3).*

Предложение 2.8. Предлагается проектировать исполнители так, чтобы они отвечали отрицательным перепадом на линии BERR* при возникновении следующих ситуаций:

а) когда у исполнителя D08(O), D08(EO) или D16 запрашивается выполнение 4-байтового цикла;

б) когда у исполнителя D08(O) или D08(EO) запрашивается выполнение 2-байтового цикла;

в) когда у исполнителя D08(O), D08(EO) или D16 запрашивается выполнение невыровненной пересылки (т. е. 3-байтовая пересылка или 2-байтовая пересылка БАЙТОВ (1—2)).

В таблице 2.10 приведены различные мнемонические обозначения, употребляемые для описания основных возможностей пересылки данных, и смысловое содержание каждого из них при описании возможностей задатчиков, исполнителей и адресных мониторов.

Таблица 2.10 — Мнемонические обозначения основных возможностей пересылки данных

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
D08(O)	Исполнитель	Может воспринимать следующие циклы: - однобайтовые циклы считывания: СЧИТЫВАНИЕ БАЙТА (1) СЧИТЫВАНИЕ БАЙТА (3) - однобайтовые циклы записи: ЗАПИСЬ БАЙТА (1) ЗАПИСЬ БАЙТА (3)
D08(EO)	Задатчик Исполнитель Адресный монитор	Может формировать следующие циклы: Может воспринимать следующие циклы: Может контролировать следующие циклы: - однобайтовые циклы считывания: СЧИТЫВАНИЕ БАЙТА (0) СЧИТЫВАНИЕ БАЙТА (1) СЧИТЫВАНИЕ БАЙТА (2) СЧИТЫВАНИЕ БАЙТА (3) - однобайтовые циклы записи: ЗАПИСЬ БАЙТА (0) ЗАПИСЬ БАЙТА (1) ЗАПИСЬ БАЙТА (2) ЗАПИСЬ БАЙТА (3)
D16	Задатчик Исполнитель Адресный монитор	Может формировать следующие циклы: Может воспринимать следующие циклы: Может контролировать следующие циклы: - двухбайтовые циклы считывания: СЧИТЫВАНИЕ БАЙТОВ (0—1) СЧИТЫВАНИЕ БАЙТОВ (2—3) - двухбайтовые циклы записи: ЗАПИСЬ БАЙТОВ (0—1) ЗАПИСЬ БАЙТОВ (2—3)
D32	Задатчик Исполнитель Адресный монитор	Может формировать следующие циклы: Может воспринимать следующие циклы: Может контролировать следующие циклы: - четырехбайтовые циклы считывания: СЧИТЫВАНИЕ БАЙТОВ (0—3) - четырехбайтовые циклы записи: ЗАПИСЬ БАЙТОВ (0—3)

Замечание 2.16. Может показаться логичным определить исполнители только четных байтов, которые бы отвечали ячейкам памяти с четными байтами, смежным с ячейками для исполнителей D08(O). Однако этого нельзя делать по той причине, что существует только одна линия DTACK*. Если задатчик выбрал бы одновременно как четнобайтовую, так и нечетнобайтовую ячейки при выполнении двухбайтовой пересылки, то оба исполнителя в один и тот же момент времени возбуждали бы линию DTACK* и задатчик не смог бы узнать, оба ли модуля подтвердили обращение.

Замечание 2.17. Поскольку исполнители D08(O) отвечают только на нечетнобайтовые адреса, они не обеспечивают работу с адресами четных байтов памяти. Исполнители D08(O) удобны только для регистров ввода/вывода, статуса или управления, тогда как для работы с запоминающими устройствами удобны исполнители D08(EO), D16 и D32.

2.3.7 Возможности блочной пересылки

Задатчики часто обращаются к нескольким ячейкам памяти в порядке возрастания адресов. В этом случае очень удобными являются циклы блочной пересылки. Они позволяют задатчику выдавать один адрес, а затем осуществлять ввод/вывод данных в/из соответствующей ячейки и следующих в возрастающем порядке адресов ячеек, не выставляя каждый раз новый адрес.

Когда задатчик инициирует цикл блочной пересылки, отвечающий исполнитель фиксирует адрес во внутримодульном счетчике. После завершения первой пересылки данных (т. е. установки высокими стробов данных) задатчик не позволяет устанавливаться адресному стробу высоким. Вместо этого он каждый раз устанавливает строб(ы) данных низким(и) в ответ на подтверждения пересылок данных от исполнителя и пересылает данные в/из следующих в порядке возрастания адресов ячеек памяти.

Для обращения к следующей (следующим) ячейке (ячейкам) памяти исполнитель осуществляет приращение внутримодульного счетчика, который формирует адрес при каждом переходе строба (стробов) данных с одного уровня на другой.

Замечание 2.18. Реализация возможности циклов блочной пересылки неограниченной длины усложняет проектирование схем модулей памяти. В частности, все исполнители блочных пересылок (и тот, который отвечает, и те, которые не отвечают) обязаны фиксировать начальный адрес, а затем осуществлять приращение адресного счетчика на каждую пересылку по шине. В таком случае все исполнители должны декодировать получившийся в результате приращения адрес, чтобы определить, не пересекает ли он границу адресного пространства конкретного модуля. Хотя проектирование такого модуля вполне возможно, но упомянутая процедура декодирования адресов обычно ограничивает времена обращения исполнителя. Для упрощения проектирования схем таких исполнителей и сокращения времени обращения настоящий стандарт устанавливает правило 2.12.

Правило 2.12. *Циклы блочной пересылки НЕ ДОЛЖНЫ ПЕРЕСЕКАТЬ ни одной 256-байтовой границы.*

Замечание 2.19. Правило 2.12 ограничивает длину блочных пересылок 256 байтами. В процессе такой блочной пересылки изменяются только состояния адресных линий A01—A07. Это упрощает проектирование исполнителей блочных пересылок. В этом случае старшие адресные линии приходится декодировать только один раз в начале цикла блочной пересылки, позволяя сократить время обращения при всех последующих пересылках данных.

Замечание 2.20. В некоторых случаях может возникнуть необходимость переслать большой блок данных, который пересекает одну или несколько 256-байтовых границ. Если при проектировании аппаратной части модуля, выполняющей блочную пересылку, предусмотрена возможность распознавать достижение 256-байтовой границы, модуль может мгновенно установить линию AS* высокой, а затем инициировать следующую блочную пересылку без вмешательства системной программы.

Цикл блочного считывания очень напоминает последовательность циклов считывания. Аналогично, цикл блочной записи очень напоминает последовательность циклов записи. Разница заключается в том, что при блочных считывании и записи задатчик широковещательно передает только начальный адрес, а адресный строб удерживается низким во время всех блочных пересылок данных.

Замечание 2.21. Управление шиной пересылки данных не может быть передано в процессе циклов блочных пересылок, поскольку в процессе этих пересылок на линии AS* удерживается низкий уровень, а управление шиной пересылки данных может передаваться, если на линии AS* удерживается высокий уровень.

Правило 2.66. *Исполнители, обладающие возможностью блочных пересылок, ДОЛЖНЫ КОНТРОЛИРОВАТЬ линию AS*, а также ДОЛЖНЫ ЗАХВАТЫВАТЬ адресную информацию, когда они обнаруживают отрицательный перепад сигнала на линии AS*.*

Замечание 2.86. В процессе однобайтовых блочных пересылок одновременно пересылаются 8 разрядов данных по линиям D00—D07 или D08—D15. Один из примеров такой пересылки приведен ниже.

	D08—D15	D00—D07
Первая пересылка данных		БАЙТ (1)
	БАЙТ (2)	
		БАЙТ (3)
	БАЙТ (0)	
		БАЙТ (1)
	БАЙТ (2)	
Последняя пересылка данных		БАЙТ (3)

Замечание 2.87. В процессе двухбайтовых блочных пересылок одновременно пересылаются 16 разрядов данных по линиям D00—D15. Один из примеров такой пересылки приведен ниже.

	D08—D15	D00—D07
Первая пересылка данных	БАЙТ (2)	БАЙТ (3)
	БАЙТ (0)	БАЙТ (1)
	БАЙТ (2)	БАЙТ (3)
	БАЙТ (0)	БАЙТ (1)
	БАЙТ (2)	БАЙТ (3)
Последняя пересылка данных	БАЙТ (0)	БАЙТ (1)

Таблица 2.11 содержит мнемоническое обозначение, употребляемое для описания возможностей блочной пересылки, и смысловое содержание применительно к задатчикам, исполнителям и адресным мониторам.

Таблица 2.11 — Мнемоническое обозначение возможностей блочных пересылок

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
BLT	Задатчик D08(EO) Исполнитель D08(EO) Адресный монитор D08 (EO)	Может формировать следующие циклы: Может воспринимать следующие циклы: Может контролировать следующие циклы: - циклы блочного считывания: ОДНОБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ - циклы блочной записи: ОДНОБАЙТОВАЯ БЛОЧНАЯ ЗАПИСЬ
	Задатчик D16 Исполнитель D16 Адресный монитор D16	Может формировать следующие циклы: Может воспринимать следующие циклы: Может контролировать следующие циклы: - циклы блочного считывания: ДВУХБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ - циклы блочной записи: ДВУХБАЙТОВАЯ БЛОЧНАЯ ЗАПИСЬ
	Задатчик D32 Исполнитель D32 Адресный монитор D32	Может формировать следующие циклы: Может воспринимать следующие циклы: Может контролировать следующие циклы: - циклы блочного считывания: ЧЕТЫРЕХБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ - циклы блочной записи: ЧЕТЫРЕХБАЙТОВАЯ БЛОЧНАЯ ЗАПИСЬ

2.3.8 Функциональная возможность Чтение—Модификация—Запись

В мультипроцессорных системах, совместно использующих ресурсы, такие как устройства памяти и ввода/вывода, нужен алгоритм по распределению этих ресурсов. Одна из очень важных целей этого алгоритма распределения состоит в том, чтобы гарантировать невозможность одновременного использования какого-либо ресурса двумя задачами. Лучше всего эта проблема может быть проиллюстрирована следующим примером.

Два процессора в мультипроцессорной системе совместно используют какой-то общий ресурс (например, принтер). В любой момент времени ресурс может быть использован только одним процессором. Этот ресурс распределяется каким-либо разрядом в памяти, т.е. когда разряд установлен, то ресурс занят, когда он очищен, ресурс свободен. Для получения ресурса в свое распоряжение процессор А считывает разряд и проверяет его содержимое, чтобы определить, очищен ли он. Если разряд очищен, процессор А устанавливает его, чтобы заблокировать использование принтера процессором В. Эта операция состоит из двух пересылок данных: считывание разряда для его проверки и записи для его установки. Однако может возникнуть проблема, если между этими двумя пересылками шина будет предоставлена процессору В. В этом случае процессор В может тоже установить, что упомянутый разряд очищен, и сделать вывод, что ресурс свободен. Далее оба будут устанавливать этот разряд в следующий свободный цикл и попытаются использовать ресурс.

Этот конфликт разрешается введением цикла Чтение—Модификация—Запись, который предотвращает передачу управления шиной пересылки данных между фазами считывания и записи этого цикла. Этот цикл очень напоминает совокупность двух циклов: цикла считывания и непосредственно следующего за ним цикла записи. Отличие заключается в том, что в цикле Чтение—Модификация—Запись адресный строб удерживается низким в процессе обеих пересылок. Это гарантирует невозможность передачи управления шиной пересылки данных во время цикла Чтение—Модификация—Запись в противоположность случаю с совокупностью двух циклов, так как передача управления возможна только в случае, когда адресный строб является высоким.

Таблица 2.12 содержит мнемоническое обозначение, употребляемое для описания возможностей цикла Чтение—Модификация—Запись (read—modify—write—RMW), и смысловое содержание применительно к задатчикам, исполнителям и адресным мониторам.

Таблица 2.12 — Мнемоническое обозначение возможностей цикла Чтение—Модификация—Запись

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
RMW	Исполнитель D08(O)	Может воспринимать следующие циклы: - нечетные однобайтовые циклы RMW: ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА(1) ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА(3)
	Задатчик D08(EO) Исполнитель D08(EO) Адресный монитор D08(EO)	Может формировать следующие циклы: Может воспринимать следующие циклы: Может контролировать следующие циклы: - однобайтовые циклы RMW: ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА(0) ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА(1) ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА(2) ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА(3)
	Задатчик D16 Исполнитель D16 Адресный монитор D16	Может формировать следующие циклы: Может воспринимать следующие циклы: Может контролировать следующие циклы: - двухбайтовые циклы RMW: ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ(0—1) ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ(2—3)
	Задатчик D32 Исполнитель D32 Адресный монитор D32	Может формировать следующие циклы: Может воспринимать следующие циклы: Может контролировать следующие циклы: - четырехбайтовые циклы RMW: ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ(0—3)

2.3.9 Возможности невыровненных пересылок

Некоторые 32-разрядные микропроцессоры размещают и выбирают данные невыровненным способом. Например, 32-разрядная величина может быть размещена четырьмя различными способами, как показано на рисунке 2.6.

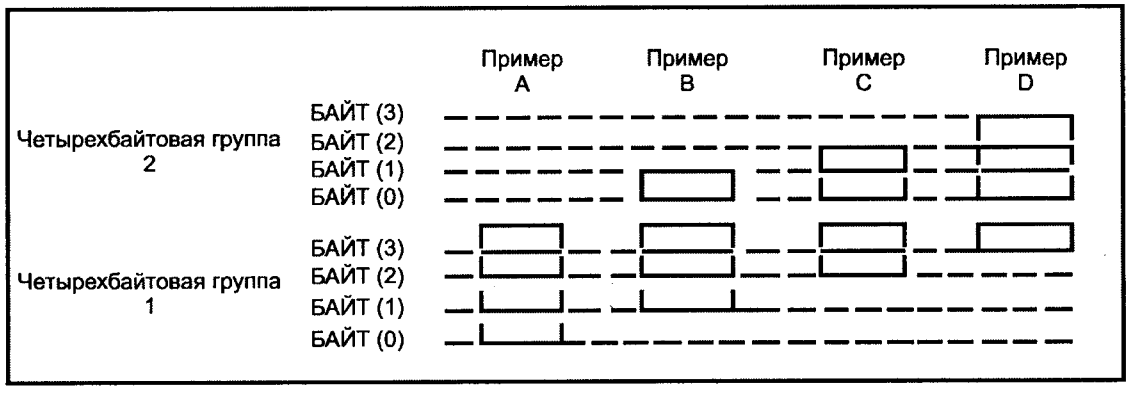


Рисунок 2.6 — Четыре способа размещения в памяти 32 разрядов данных

Задатчик может пересылать 32 разряда данных, используя несколько различных последовательностей циклов шины пересылки данных. Например, он может пересылать данные по одному байту, используя четыре однобайтовых пересылки данных. Однако задатчик может выполнить пересылку значительно быстрее, используя одну из последовательностей циклов, которые указаны в таблице 2.13.

Таблица 2.13 — Пересылка 32 разрядов данных с использованием кратнобайтовых циклов

Пример (см. рисунок 2.6)	Последовательности циклов	Используемые линии данных	Байтовые ячейки, к которым осуществляется доступ (см. рисунок 2.6)
A	Четырехбайтовая пересылка	D00—D31	Группа 1, БАЙТЫ (0—3)
B	Однобайтовая пересылка	D00—D07	Группа 1, БАЙТ (1)
	Двухбайтовая пересылка	D00—D15	Группа 1, БАЙТЫ (2—3)
	Однобайтовая пересылка	D08—D15	Группа 2, БАЙТ (0)
	или		
	Трехбайтовая пересылка	D00—D23	Группа 1, БАЙТЫ (1—3)
	Однобайтовая пересылка	D08—D15	Группа 2, БАЙТ (0)
C	Двухбайтовая пересылка	D00—D15	Группа 1, БАЙТЫ (2—3)
	Двухбайтовая пересылка	D00—D15	Группа 2, БАЙТЫ (0—1)
D	Однобайтовая пересылка	D00—D07	Группа 1, БАЙТ (3)
	Двухбайтовая пересылка	D00—D15	Группа 2, БАЙТЫ (0—1)
	Однобайтовая пересылка	D08—D15	Группа 2, БАЙТ (2)
	или		
	Однобайтовая пересылка	D00—D07	Группа 1, БАЙТ (3)
	Трехбайтовая пересылка	D08—D31	Группа 2, БАЙТЫ (0—2)

Замечание 2.22. Последовательности, указанные в таблице 2.13, являются типичными для задатчика, выполняющего обращение к байтовым ячейкам в возрастающем порядке их адресов. Протокол магистрали VME этого требования не содержит.

Как указано в таблице 2.13, каждая из этих 32-разрядных пересылок может быть выполнена комбинацией однобайтовых и двухбайтовых пересылок. Однако в примерах B и D при выполнении этой процедуры требуются три цикла шины. По этой причине протокол шины пересылки данных

предусматривает также два цикла трехбайтовых пересылок. В комбинации с однобайтовым циклом эти трехбайтовые циклы позволяют размещать данные, как это указано в примерах B и D, всего за два цикла шины.

Некоторые 32-разрядные микропроцессоры одновременно размещают и выбирают по 16 разрядов данных невыровненным способом, как показано на рисунке 2.7.

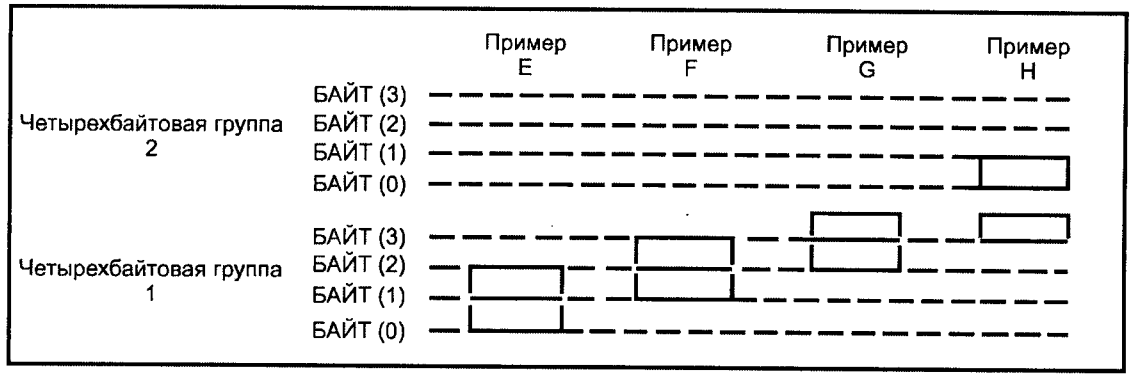


Рисунок 2.7 — Четыре способа размещения в памяти 16 разрядов данных

Задатчик может выполнять пересылки 16 разрядов данных, используя несколько различных последовательностей циклов шины пересылки данных, как указано в таблице 2.14.

Таблица 2.14 — Пересылка 16 разрядов данных с использованием кратнобайтовых циклов

Пример (см. рисунок 2.7)	Последовательности циклов	Используемые линии данных	Байтовые ячейки, к которым осуществляется доступ (см. рисунок 2.7)
E	Двухбайтовая пересылка	D00—D15	Группа 1, БАЙТЫ (0—1)
F	Однобайтовая пересылка Однобайтовая пересылка или Двухбайтовая пересылка	D00—D07 D08—D15 D08—D23	Группа 1, БАЙТ (1) Группа 1, БАЙТ (2) Группа 1, БАЙТЫ (1—2)
G	Двухбайтовая пересылка	D00—D15	Группа 1, БАЙТЫ (2—3)
H	Однобайтовая пересылка Однобайтовая пересылка	D00—D07 D08—D15	Группа 1, БАЙТ (3) Группа 2, БАЙТ (0)

Замечание 2.23. Последовательности, указанные в таблице 2.14, являются типичными для задатчика, который выполняет доступы к байтовым ячейкам в возрастающем порядке номеров. Протокол магистрали VME этого не требует.

Как видно из таблицы 2.14, 16-разрядная пересылка в примере F может выполняться двумя однобайтовыми пересылками. Однако для этого потребуются два цикла шины. По этой причине протокол шины пересылки данных предусматривает также цикл двухбайтовой пересылки, позволяющий размещать данные только за один цикл шины, как показано в примере F.

Замечание 2.24. Поскольку в невыровненных пересылках используются все 32 линии данных, эти пересылки могут выполняться только задатчиками D32 и исполнителями D32.

Правило 2.67. *Исполнители и адресные мониторы D32 ДОЛЖНЫ ОБЛАДАТЬ функциональной возможностью невыровненных пересылок.*

Правило 2.6. *Исполнители D08(O), D08(EO) и D16 НЕ ДОЛЖНЫ ОТВЕЧАТЬ установкой DTACK* низким во время цикла, в котором выполняется доступ к байтовым ячейкам БАЙТЫ(1—2), БАЙТЫ(0—2) или БАЙТЫ(1—3).*

Таблица 2.15 объясняет смысл использования мнемонического обозначения UAT (unaligned transfer — невыровненные пересылки) применительно к задатчикам.

Таблица 2.15 — Мнемоническое обозначение возможности невыровненных пересылок

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
UAT	Задатчик D32	Может формировать следующие циклы: - трехбайтовые циклы считывания: СЧИТЫВАНИЕ БАЙТОВ(0—2) СЧИТЫВАНИЕ БАЙТОВ(1—3) - трехбайтовые циклы записи: ЗАПИСЬ БАЙТОВ(0—2) ЗАПИСЬ БАЙТОВ(1—3) - двухбайтовый цикл считывания: СЧИТЫВАНИЕ БАЙТОВ(1—2) - двухбайтовый цикл записи: ЗАПИСЬ БАЙТОВ(1—2)

2.3.10 Возможность использования цикла Только Адрес

Цикл Только Адрес является единственным циклом на шине пересылки данных, который для пересылки данных не используется. Он начинается как обычный цикл шины пересылки данных с установки в достоверное состояние адреса, кода модификатора адреса, IACK* и LWORD* и установки в низкое состояние AS* после времени установления. Однако при этом стробы данных никогда не устанавливаются низкими. После удержания в стабильном состоянии сигналов различных линий, стробируемых AS*, в течение заданного минимального периода задатчик завершает цикл, не ожидая установки в низкое состояние DTACK* или BERR* (цикл Только Адрес является также единственным циклом шины пересылки данных, которому для завершения не требуется никакого ответа). Таблица 2.16 объясняет смысл использования мнемонического обозначения ADO (Address only — Только Адрес) применительно к задатчикам.

Таблица 2.16 — Мнемоническое обозначение возможности цикла Только Адрес

Мнемоническое содержание	Функциональный блок	Смысловое содержание
ADO	Задатчик	Может формировать циклы Только Адрес

Замечание 2.25. Циклы Только Адрес могут использоваться для увеличения производительности модулей магистрали VME, позволяя модулю центрального процессора выполнять широковещательную передачу адреса до того, как этот процессор определит, что данный адрес выбрал или не выбрал исполнителя на шине. Широковещательная передача адреса таким способом позволяет исполнителям магистрали VME декодировать адрес одновременно с модулем центрального процессора.

Правило 2.68. *Все исполнители ДОЛЖНЫ БЫТЬ СПРОЕКТИРОВАНЫ так, чтобы циклы Только Адрес (ADO) выполнялись без потери данных или сбоев в работе.*

2.3.11 Взаимодействие между функциональными блоками шины пересылки данных

Пересылки данных происходят между задатчиками и исполнителями. Задатчик — это блок, управляющий пересылкой. Исполнитель, который распознает адрес пересылки как свой собственный, является отвечающим исполнителем, а все другие — неотвечающими исполнителями.

После инициирования цикла пересылки данных задатчик ждет ответа от отвечающего исполнителя. Когда задатчик обнаруживает этот ответ, он устанавливает свои стробы данных и адресный строб высокими, заканчивая данный цикл. Исполнитель отвечает освобождением линии ответа.

Замечание 2.26. Хотя временные соотношения при пересылке адреса и данных в значительной степени независимы, имеются два исключения:

а) прежде чем установить низкими DS0* или DS1*, задатчик устанавливает низким AS*;

б) исполнитель подтверждает как AS*, так и DS0* и DS1* либо сигналом DTACK*, либо сигналом BERR*.

Правило 2.13. *Если исполнитель отвечает на цикл пересылки данных, то он ДОЛЖЕН УСТАНОВИТЬ низким либо DTACK*, либо BERR*, но НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ эти сигналы низкими одновременно.*

Замечание 2.27. Вследствие возможной разницы во времени распространения сигналов по магистрали, вызванной неодинаковой загруженностью линий AS*, DS0* и DS1*, отрицательный перепад DS0* и DS1* может быть обнаружен исполнителем несколько раньше, чем отрицательный перепад сигнала AS*.

Замечание 2.28. Сигнал WRITE* устанавливается высоким для распознавания циклов считывания и низким для распознавания циклов записи прежде, чем первый строб данных устанавливается низким, и сохраняет неизменным любое свое состояние до тех пор, пока оба строба данных не установятся высокими.

Правило 2.14. *Перед началом управления линиями данных задатчик ДОЛЖЕН УБЕДИТЬСЯ, что предыдущий отвечающий исполнитель прекратил управление этими линиями. С этой целью задатчик ДОЛЖЕН ПРОВЕРИТЬ, являются ли DTACK* и BERR* высокими, прежде чем установить строб(ы) данных низким(и) при любом типе цикла и прежде чем начать управление любыми линиями данных в цикле записи.*

Правило 2.15. *Прежде чем разрешить установку DTACK* высоким в конце цикла считывания, отвечающий исполнитель ДОЛЖЕН ОСВОБОДИТЬ линии данных.*

Правило 2.16. *Во время циклов считывания отвечающий исполнитель ДОЛЖЕН УДЕРЖИВАТЬ достоверные данные до тех пор, пока не обнаружит высоким первый строб данных.*

Предложение 2.3. Для достижения оптимальных рабочих характеристик предлагается разрабатывать задатчики таким образом, чтобы они устанавливали DS0* и DS1* высокими как можно быстрее, после того как DTACK* или BERR* станет низким. Предлагается также разрабатывать исполнители так, чтобы они освобождали линии данных и линию DTACK* как можно быстрее после обнаружения ими высокого уровня сигналов на линиях DS0* и DS1*. Это позволяет обеспечить максимальную скорость пересылки данных по магистрали.

Замечание 2.29. Адресная информация на линиях адреса может измениться вскоре после того, как установится низким DTACK* или BERR*, и перед тем, как задатчик установит высокими DS0* и DS1* или AS*.

Третий тип функционального блока — адресный монитор. Он контролирует пересылки данных и генерирует либо один из двух, либо оба внутримодульных сигнала всякий раз, когда происходит обращение к контролируемой им байтовой ячейке. Если таким обращением является цикл записи, то формируется внутримодульный сигнал ЗАПИСЬ. Если происходит цикл считывания, то формируется внутримодульный сигнал ЧТЕНИЕ. Если выполняется цикл Чтение—Модификация—Запись, то формируются оба сигнала.

Если выполнение цикла затягивается, то вмешивается четвертый функциональный блок, шинный таймер, устанавливая BERR* низким и заканчивая, таким образом, квитирование пересылки данных и разрешая шине возобновить работу.

Правило 2.17. *Имеется строгая взаимосвязь между положительными и отрицательными перепадами DS0*/DS1* и уровнями DTACK*/BERR*. Прежде чем установить низким DS0* или DS1*, задатчик ДОЛЖЕН УБЕДИТЬСЯ, что DTACK* и BERR* оба высокие. Когда же задатчик установил низким DS0* или DS1*, он НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ их высокими и ЗАКАНЧИВАТЬ пересылку до тех пор, пока не обнаружит низкий уровень DTACK* или BERR*.*

Замечание 2.30. Модуль со встроенным процессором, который управляет пересылками данных между собой и другими модулями магистрали VME, содержит функциональный блок задатчика. Если этот же модуль имеет еще и память, доступную со стороны магистрали, то он содержит также и функциональный блок исполнителя. Интеллектуальный периферийный контроллер может получать команды через интерфейс исполнителя от процессора общего назначения. Затем, исполняя эти команды, он может вести себя как задатчик при обращении к глобальной памяти магистрали VME.

2.4 Типичные примеры работы

Задатчики инициируют пересылки по шине пересылки данных. Адресуемый исполнитель затем подтверждает пересылку. После получения подтверждения пересылки данных задатчик заканчивает цикл. Асинхронный характер шины пересылки данных позволяет исполнителю устанавливать время, затрачиваемое на пересылку.

До начала каких-либо пересылок задатчику должно быть предоставлено исключительное право управления шиной пересылки данных. Это гарантирует, что несколько задатчиков не будут пытаться одновременно использовать шину пересылки данных. Задатчики получают право управления шиной пересылки данных с помощью функциональных блоков и сигнальных линий шины арбитража (подробнее см. в 2.5). При данном рассмотрении предполагается, что задатчик уже получил право и взял на себя управление шиной пересылки данных.

2.4.1 Типичные циклы пересылки данных

На рисунке 2.8 показан типичный цикл однобайтового считывания. Для начала пересылки задатчик устанавливает на линиях адреса требуемый адрес и код модификатора адреса. Так как данный пример является циклом считывания БАЙТА (1), задатчик устанавливает LWORD* высоким и A01 — низким. Цикл подтверждения прерывания здесь отсутствует, поэтому IACK* низким не устанавливается. Затем задатчик в течение определенного времени установления делает паузу, позволяя стабилизировать информацию на линиях адреса и линиях модификатора адреса, и устанавливает низким AS* для последующего декодирования адреса исполнителями.

Задатчик | **Исполнитель**

Адресовать исполнителю

Выставить адрес
Выставить модификатор адреса
Установить LWORD* высоким
Установить IACK* высоким
Установить AS* низким

Указать направление пересылки данных

Установить WRITE* высоким

Обработать адрес

Принять адрес
Принять модификатор адреса
Принять LWORD* высокого уровня
Принять IACK* высокого уровня
Принять AS* низкого уровня
ЕСЛИ адрес совпадает с адресом данного исполнителя,
ТО выбрать внутримодульное устройство

Указать разрядность данных

Подождать, пока DTACK* и BERR* не станут высокими (это свидетельствует, что предшествующий исполнитель больше не управляет линиями данных)
Установить DS0* низким и DS1* высоким

Вызвать данные

Принять WRITE* высокого уровня
Считать данные из выбранного устройства
Принять DS1* высокого уровня
Принять DS0* низкого уровня
Поместить данные на D00—D07

Ответить задатчику

Установить DTACK* низким

(см. лист 2)

Рисунок 2.8, лист 1 — Пример однобайтового цикла считывания

Задатчик **Исполнитель**

Получить данные

Принять данные с D00—D07
Принять DTACK* низкого уровня

Завершить цикл

ЕСЛИ этот цикл последний, ТО
Освободить линии адреса
Освободить линии модификатора адреса
Освободить LWORD*
Освободить IACK*
КОНЕЦ_ЕСЛИ
Установить DS0* высоким
Установить AS* высоким

Закончить завершение цикла

ЕСЛИ этот цикл последний, ТО
Освободить DS0* и DS1*
Освободить AS*;
В ПРОТИВНОМ СЛУЧАЕ перейти к состоянию **«Адресовать исполнитель»**
КОНЕЦ_ЕСЛИ

Закончить ответ задатчику

Принять AS* и DS0* высокого уровня
Освободить D00—D07

Подтвердить завершение цикла

Освободить DTACK*

Рисунок 2.8, лист 2

Каждый исполнитель, анализируя уровни сигналов на линиях адреса, модификатора адреса и IACK*, определяет, должен ли он отвечать. В то время, когда все это происходит, задатчик удерживает WRITE* высоким, что указывает на предстоящую операцию считывания. Кроме того, задатчик проверяет, находятся ли DTACK* и BERR* в высоком состоянии, убеждаясь в том, что исполнитель из предыдущего цикла уже больше не управляет линиями данных. Если это так, задатчик устанавливает DS0* низким, оставляя DS1* высоким.

Затем отвечающий исполнитель определяет, к какой четырехбайтовой группе и к какому байту внутри этой группы происходит обращение, и начинает пересылку. После извлечения данных из внутренней памяти и помещения их на линии D00—D07 исполнитель сигнализирует задатчику, устанавливая DTACK* низким. После этого исполнитель удерживает DTACK* низким и сохраняет достоверные данные до тех пор, пока задатчик удерживает низким DS0*.

Когда задатчик принимает сигнал DTACK* низкого уровня, он захватывает данные на D00—D07, освобождает линии адреса и устанавливает DS0* и AS* высокими. Исполнитель отвечает освобождением линий D00—D07, а также линии DTACK*, на которой после этого устанавливается высокий уровень сигнала.

Замечание 2.31. Задатчик на рисунке 2.8 освобождает все линии шины пересылки данных в конце пересылки. Этого не требуется до тех пор, пока запросчик задатчика не освободит BBSY* (bus busy) во время пересылки данных, как описано в 2.5.

Алгоритмы двухбайтовых и четырехбайтовых циклов пересылки данных очень схожи с однобайтовым циклом. Они приведены на рисунках 2.9 и 2.10.

Задатчик | **Исполнитель**

Адресовать исполнителю

Выставить адрес
Выставить модификатор адреса
Установить LWORD* высоким
Установить IACK* высоким
Установить AS* низким

Обработать адрес

Принять адрес
Принять модификатор адреса
Принять LWORD* высокого уровня
Принять IACK* высокого уровня
Принять AS* низкого уровня
ЕСЛИ адрес совпадает с адресом данного исполнителя,
ТО выбрать внутримодульное устройство

Указать направление пересылки данных

Установить WRITE* низким

Указать разрядность данных

Подождать, пока DTACK* и BERR* не станут высокими (это свидетельствует, что предшествующий исполнитель больше не управляет линиями данных)
Поместить данные на D00—D15
Установить DS0* и DS1* низкими

Запомнить данные

Принять WRITE* низкого уровня
Принять DS1* низкого уровня
Принять DS0* низкого уровня
Захватить данные с D00—D15
Записать данные в выбранное устройство

Ответить задатчику

Установить DTACK* низким

Завершить цикл

Принять DTACK* низкого уровня
ЕСЛИ этот цикл последний, ТО
    Освободить линии адреса
    Освободить линии модификатор адреса
    Освободить линии данных
    Освободить LWORD*
    Освободить IACK*
КОНЕЦ_ЕСЛИ
Установить DS0* и DS1* высокими
Установить AS* высоким

(см. лист 2)

Рисунок 2.9, лист 1 — Пример двухбайтового цикла записи

Задатчик

Закончить завершение

Если этот цикл последний, то
Освободить DS0* и DS1*
Освободить AS*
Или в противном случае перейти к состоянию **«Адресовать исполнитель»**
КОНЕЦ_ЕСЛИ

Исполнитель

Подтвердить завершение цикла

Принять AS*, DS0* и DS1* высокого уровня
Освободить DTACK*

Рисунок 2.9, лист 2

Задатчик

Адресовать исполнителю

Выставить адрес
Выставить модификатор адреса
Установить LWORD* низким
Установить IACK* высоким
Установить AS* низким

Указать направление пересылки данных

Установить WRITE* низким

Указать разрядность данных

Подождать, пока DTACK*
и BERR* не станут высокими
(показывает, что предшествующий исполнитель больше не управляет линиями данных)
Поместить данные на D00—D31
Установить DS0* и DS1* низкими

Исполнитель

Обработать адрес

Принять адрес
Принять модификатор адреса
Принять LWORD* низкого уровня
Принять IACK* высокого уровня
Принять AS* низкого уровня
ЕСЛИ адрес совпадает с адресом данного исполнителя,
ТО выбрать внутримодульное устройство

Запомнить данные

Принять WRITE* низкого уровня
Принять DS0* и DS1* низкого уровня
Захватить данные с D00—D31
Записать данные в выбранное устройство

(см. лист 2)

Рисунок 2.10, лист 1— Пример четырехбайтового цикла записи

Задатчик | **Исполнитель**

Ответить задатчику

Установить DTACK* низким

Завершить цикл

Принять DTACK* низкого уровня
ЕСЛИ этот цикл последний, ТО
Освободить линии адреса
Освободить линии модификатора адреса
Освободить линии данных
Освободить LWORD*
Освободить IACK*
КОНЕЦ_ЕСЛИ
Установить DS0* и DS1* высокими
Установить AS* высоким

Закончить завершение цикла

ЕСЛИ этот цикл последний, ТО
Освободить DS0* и DS1*
Освободить AS*
Или в противном случае перейти к состоянию **«Адресовать исполнитель»**
КОНЕЦ_ЕСЛИ

Подтвердить завершение цикла

Принять AS*, DS0* и DS1* высокого уровня
Освободить DTACK*

Рисунок 2.10, лист 2

2.4.2 Конвейерная адресация

Поскольку для адреса и данных определены отдельные стробы, задатчики могут осуществлять широковещательную пересылку адреса для следующего цикла в то время, как еще происходит пересылка данных для предыдущего цикла. Это получило название конвейерной адресации.

Разрешение 2.8. Как только задатчик обнаружит, что отвечающий исполнитель установил DTACK* или BERR* низким, он МОЖЕТ изменить адрес и, после удержания AS* высоким в течение минимального времени, снова установить AS* низким.

Например, когда исполнитель в цикле считывания установит DTACK* или BEER* низким, задатчику разрешается поместить новый адрес на адресную шину адреса во время считывания данных. Это равносильно перекрытию текущего цикла со следующим и увеличению производительности магистрали VME.

Правило 2.18. *Все исполнители ДОЛЖНЫ БЫТЬ СПРОЕКТИРОВАНЫ с возможностью конвейерной адресации без потерь данных или ошибочных операций.*

Ниже предлагаются две конструкции, которые выполняют конвейерную адресацию.

Замечание 2.32. Отвечающий исполнитель может распознать свой адрес и очень быстро ответить по линии DTACK* или BERR*. Поскольку задатчику разрешено снимать адрес после того, как отвечающий исполнитель установит DTACK* или BERR* низким, неотвечающие исполнители могут оказаться не в состоянии декодировать адресную информацию до того, как задатчик снимет ее с шины.

Замечание 2.33. Поскольку задатчик может осуществлять широковещательную пересылку нового адреса в процессе окончания предыдущего цикла, разработчики исполнителей должны обеспечить,

чтобы следующая установка AS* не нарушала достоверность первого адреса, если он все еще необходим внутримодульной логике для сохранения данных на шине.

Предложение 2.4. Предлагается разрабатывать исполнители, захватывающие адресную информацию по отрицательному перепаду AS*.

Замечание 2.34. Задатчик может установить AS* низким для нового цикла до того, как он установит DS0* или DS1* высоким от предыдущего цикла. В связи с этим при перекрытии циклов может быть такой период, когда AS* для нового цикла совпадет, по крайней мере, с одним из DS0* или DS1* от предыдущего цикла.

Предложение 2.5. Предлагается разрабатывать исполнители таким образом, чтобы они осуществляли пересылку данных с шины или на шину в момент, когда один или оба строба данных DS0* и DS1* были низкими, а DTACK* и BERR* — оба высокими, а не тогда, когда одновременно удерживается низкий уровень AS* и одного или обоих DS0* и DS1*.

Разрешение 2.9. Задатчики МОГУТ конструироваться без возможности работы в режиме конвейерной адресации (например, они МОГУТ ждать освобождения отвечающим исполнителем линий DTACK* и BERR*, прежде чем установить AS* низким для следующего цикла).

2.5 Получение права на использование шины пересылки данных

Правило 2.19. *Прежде чем пересылать какие-либо данные по шине, задатчик ДОЛЖЕН ПОЛУЧИТЬ разрешение на ее использование, как описано в разделе 3.*

Шина пересылки данных может потребоваться нескольким задатчикам одновременно. Процесс, определяющий, какой задатчик сможет использовать шину пересылки данных, называется арбитражем и рассматривается в разделе 3. Поскольку арбитраж тесно связан с работой шины пересылки данных, он кратко описывается и здесь.

На рисунке 2.11 представлены два примера, показывающие возможные последовательности событий, когда задатчик (названный «задатчик А») заканчивает использование шины пересылки данных и разрешает арбитраж.

Пример 1. Арбитраж во время последней пересылки данных

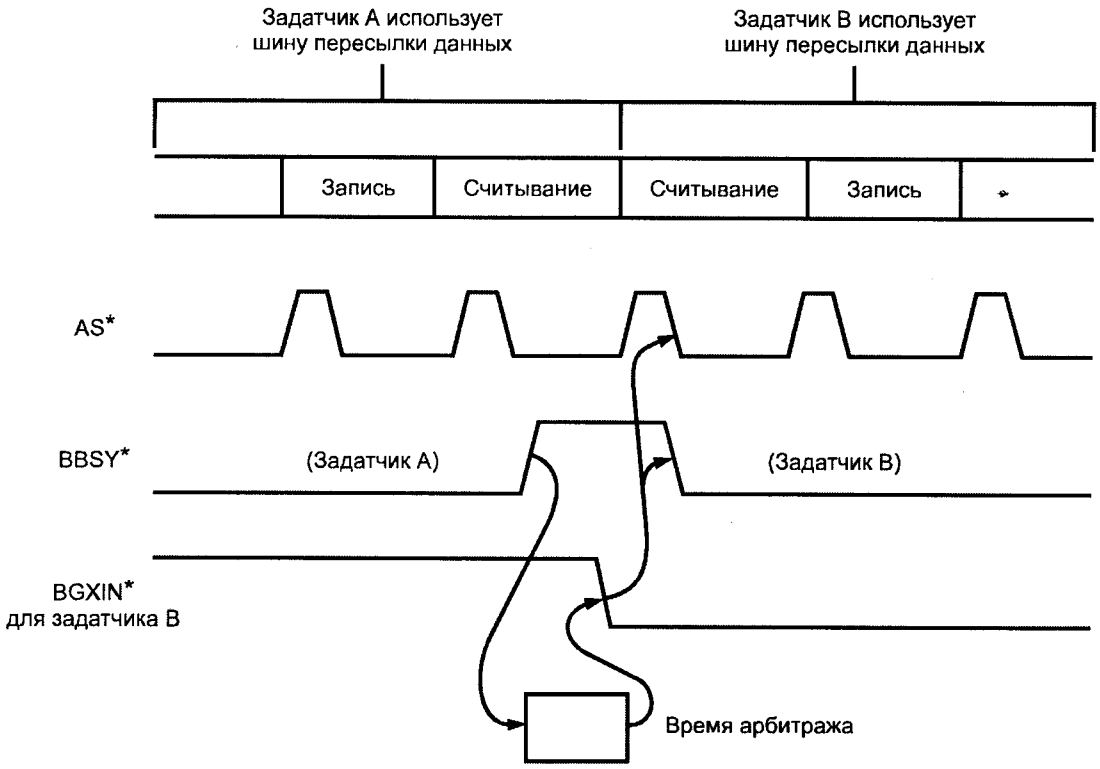


Рисунок 2.11, лист 1 — Последовательность смены задатчика шины пересылки данных

Пример 2. Арбитраж после последней пересылки данных

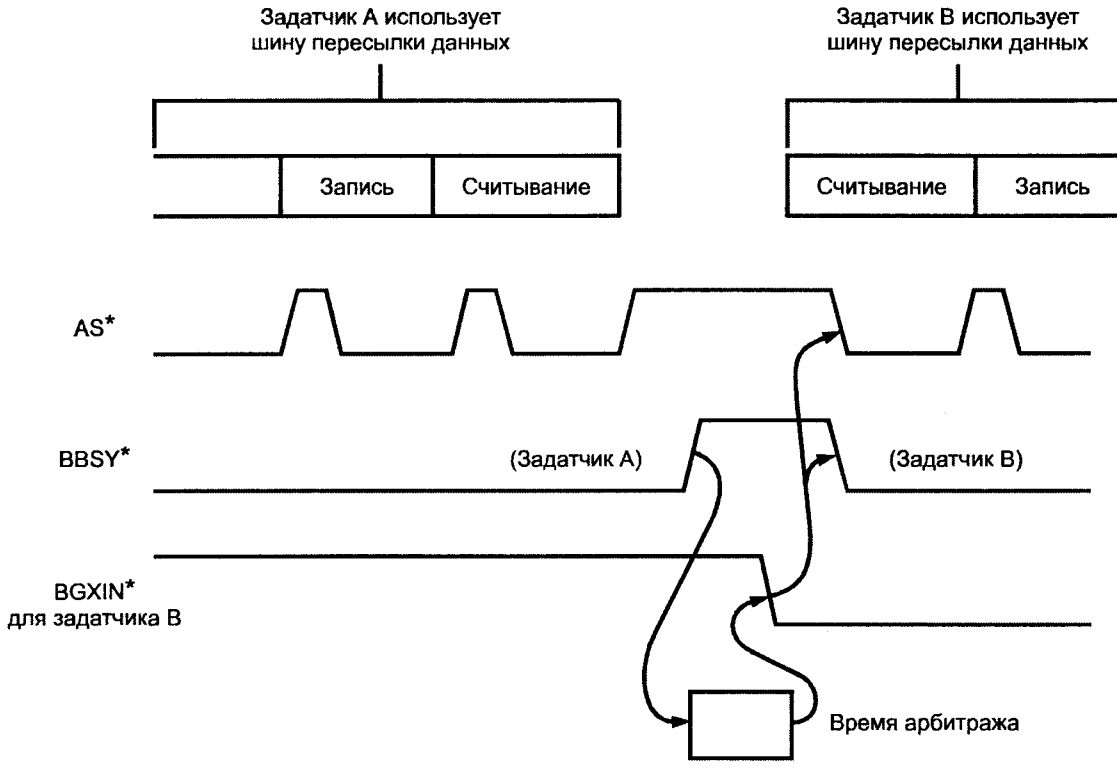


Рисунок 2.11, лист 2

В примере 1 задатчик А еще в процессе своей последней пересылки показывает, что шина пересылки данных ему больше не нужна. Он делает это с помощью своего запросчика, который освобождает сигнальную линию занятости шины (BBSY*). Так как задатчик А заранее уведомляет о скором освобождении шины пересылки данных, арбитраж осуществляется во время последней пересылки. Арбитраж заканчивается, и задатчику В предоставляется разрешение на использование шины пересылки данных до того, как задатчик А закончит свой цикл, но он ждет, пока задатчик А не освободит линию AS*. Это гарантирует, что задатчик В не начнет управление шиной пересылки данных прежде, чем задатчик А не закончит свою последнюю пересылку.

В примере 2 задатчик А ожидает завершения последней пересылки данных (то есть, когда линия AS* будет освобождена), прежде чем освободить BBSY*. В этом случае шина пересылки данных не используется во время выполнения арбитража. Задатчику В затем предоставляется право на использование шины, и, поскольку AS* уже высокий, он начнет немедленно использовать шину пересылки данных.

Правило 2.20. *Как только запросчик задатчика освобождает линию BBSY*, этот задатчик НЕ ДОЛЖЕН ПЕРЕКЛЮЧАТЬ AS* с высокого уровня на низкий, то есть НЕ ДОЛЖЕН НАЧИНАТЬ новый цикл до получения его запросчиком нового разрешения на использование шины.*

2.6 Правила и замечания по временны́м соотношениям сигналов шины пересылки данных

Данный подраздел содержит правила и замечания по временны́м соотношениям, которые определяют поведение задатчиков и исполнителей. Эта информация о временны́х соотношениях дается в виде рисунков и таблиц.

Таблица 2.17 содержит перечень временны́х диаграмм, которые определяют работу задатчика, исполнителя и адресного монитора.

Таблица 2.18 вводит мнемонические обозначения, употребляемые в данном подразделе.

Таблицы 2.19—2.21 определяют использование сигналов шины пересылки данных.

Таблицы 2.22—2.27 устанавливают значения параметров временны́х соотношений сигналов шины пересылки данных (ссылочные номера, употребляемые в таблицах 2.24—2.27, соответствуют номерам параметров временны́х соотношений в таблицах 2.22 и 2.23).

Таблица 2.17 — Перечень временны́х диаграмм, определяющих работу задатчиков, отвечающих исполнителей и адресных мониторов (временны́е параметры указаны в таблице 2.22)

Мнемоническое обозначение	Тип цикла пересылки	Номер рисунка временно́й диаграммы	
		широковещательной пересылки адреса	пересылки данных
ADO	Только Адрес	2.12	Отсутствует
D08 (EO)	Пересылки одного четного байта:		
	СЧИТЫВАНИЕ БАЙТА (0)	2.12, 2.13	2.16
	СЧИТЫВАНИЕ БАЙТА (2)	2.12, 2.13	2.16
	ЗАПИСЬ БАЙТА (0)	2.12, 2.13	2.17
	ЗАПИСЬ БАЙТА (2)	2.12, 2.13	2.17
D08 (EO) или D08 (O)	Пересылки одного нечетного байта:		
	СЧИТЫВАНИЕ БАЙТА (1)	2.12, 2.13	2.16
	СЧИТЫВАНИЕ БАЙТА (3)	2.12, 2.13	2.16
	ЗАПИСЬ БАЙТА (1)	2.12, 2.13	2.18
	ЗАПИСЬ БАЙТА (3)	2.12, 2.13	2.18
D16	Двухбайтовые пересылки:		
	СЧИТЫВАНИЕ БАЙТОВ (0—1)	2.12, 2.13	2.17
	СЧИТЫВАНИЕ БАЙТОВ (2—3)	2.12, 2.13	2.17
	ЗАПИСЬ БАЙТОВ (0—1)	2.12, 2.13	2.19
	ЗАПИСЬ БАЙТОВ (2—3)	2.12, 2.13	2.19
D32	Четырехбайтовые пересылки:		
	СЧИТЫВАНИЕ БАЙТОВ (0—3)	2.12, 2.13	2.17
	ЗАПИСЬ БАЙТОВ (0—3)	2.12, 2.13	2.19
D08(EO): BLT	Однобайтовые блочные пересылки:		
	ОДНОБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ	2.12, 2.14	2.16
	ОДНОБАЙТОВАЯ БЛОЧНАЯ ЗАПИСЬ	2.12, 2.14	2.18
D16:BLT	Двухбайтовые блочные пересылки:		
	ДВУХБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ	2.12, 2.14	2.17
	ДВУХБАЙТОВАЯ БЛОЧНАЯ ЗАПИСЬ	2.12, 2.14	2.19
D32:BLT	Четырехбайтовые блочные пересылки:		
	ЧЕТЫРЕХБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ	2.12, 2.14	2.17
	ЧЕТЫРЕХБАЙТОВАЯ БЛОЧНАЯ ЗАПИСЬ	2.12, 2.14	2.19
D08(EO): RMW	Однобайтовые пересылки Чтения—Модификации—Записи:		
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (0)	2.12, 2.15	2.20
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (1)	2.12, 2.15	2.20
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (2)	2.12, 2.15	2.20
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (3)	2.12, 2.15	2.20

Окончание таблицы 2.17

Мнемоническое обозначение	Тип цикла пересылки	Номер рисунка временной диаграммы	
		широковещательной пересылки адреса	пересылки данных
D16:RMW	Двухбайтовые пересылки Чтения—Модификации—Записи:		
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ (0—1)	2.12, 2.15	2.21
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ (2—3)	2.12, 2.15	2.21
D32:RMW	Четырехбайтовые пересылки Чтения—Модификации—Записи:		
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ (0—3)	2.13, 2.15	2.21
D32:UAT	Невыровненные пересылки:		
	СЧИТЫВАНИЕ БАЙТОВ (0—2)	2.12, 2.13	2.16
	СЧИТЫВАНИЕ БАЙТОВ (1—3)	2.12, 2.13	2.16
	СЧИТЫВАНИЕ БАЙТОВ (1—2)	2.12, 2.13	2.17
	ЗАПИСЬ БАЙТОВ (0—2)	2.12, 2.13	2.18
	ЗАПИСЬ БАЙТОВ (1—3)	2.12, 2.13	2.18
	ЗАПИСЬ БАЙТОВ (1—2)	2.12, 2.13	2.19

Таблицы 2.19-2.21 показывают, как различные сигнальные линии шины пересылки данных используются для широковещательной пересылки адресов и для пересылки данных. Эти таблицы связаны с различными временными диаграммами, которые следуют далее. Для обеспечения компактности этих таблиц употребляются мнемонические обозначения, описывающие, когда и в какое состояние устанавливаются различные сигнальные линии. Эти мнемонические обозначения определены в таблице 2.18. Во второй графе (Описание) этой таблицы наряду с русским приведен и английский эквивалент, объясняющий образование мнемонических обозначений в первой графе.

Таблица 2.18 — Определение мнемонических обозначений, употребляемых в таблицах 2.19—2.21

Мнемоническое обозначение	Описание	Комментарий
DVBM	Driven Valid By Master — устанавливаются задатчиком достоверными	***Правило 2.21.*** *Задатчик ДОЛЖЕН УСТАНАВЛИВАТЬ на линиях, обозначаемых DVBM, достоверный уровень*
DLBM	Driven Low By Master — устанавливаются задатчиком низкими	***Правило 2.22.*** *Задатчик ДОЛЖЕН УСТАНАВЛИВАТЬ на линиях, обозначаемых DLBM, низкий уровень*
DHBM	Driven High By Master — устанавливаются задатчиком высокими	***Правило 2.23.*** *Задатчик ДОЛЖЕН УСТАНАВЛИВАТЬ на линиях, обозначаемых DHBM, высокий уровень*
dhbm?	driven high by master — устанавливаются задатчиком высокими?	**Разрешение 2.10.** Задатчик МОЖЕТ устанавливать на линиях, обозначаемых dhbm?, высокий уровень

Окончание таблицы 2.18

Мнемоническое обозначение	Описание	Комментарий
		Правило 2.24. *Задатчик НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ на линиях, обозначаемых dhbm?, низкий уровень*
dxbm?	driven by master — управляются задатчиком?	**Разрешение 2.11.** Задатчик МОЖЕТ управлять линиями, обозначаемыми xbm?, либо он МОЖЕТ прекратить управление этими линиями (Если линиями dxbm? управляют, они не несут никакой достоверной информации)
DVBS	Driven Valid By Slave — устанавливаются исполнителем достоверными	***Правило 2.25.*** *Отвечающий исполнитель ДОЛЖЕН УСТАНАВЛИВАТЬ на линиях, обозначаемых DVBS, достоверный уровень*
dxbs?	driven by slave — управляются исполнителем?	**Разрешение 2.12.** Отвечающий исполнитель МОЖЕТ управлять линиями, обозначаемыми dxbs?, либо он МОЖЕТ прекратить управление этими линиями. (Если линиями dxbs? управляют, они не несут никакой достоверной информации)
DVBB	Driven Valid By Both Slave and Master — устанавливаются достоверными как исполнителем, так и задатчиком	***Правило 2.26.*** *Во время фазы считывания цикла Чтение—Модификация—Запись, отвечающий исполнитель ДОЛЖЕН УСТАНАВЛИВАТЬ на линиях, обозначаемых DVBB, достоверные данные. Во время фазы записи цикла Чтение—Модификация—Запись задатчик ДОЛЖЕН УСТАНАВЛИВАТЬ на линиях, обозначаемых DVBB, достоверные данные*
dxbb?	driven by both slove and master — управляются как исполнителем, так и задатчиком?	**Разрешение 2.13.** Во время фазы считывания цикла Чтение—Модификация—Запись отвечающий исполнитель МОЖЕТ управлять линиями, обозначаемыми dxbb?, либо он МОЖЕТ прекратить управление этими линиями. Во время фазы записи цикла Чтение—Модификация—Запись задатчик МОЖЕТ управлять линиями, обозначаемыми dxbb?, либо он МОЖЕТ прекратить управление этими линиями. (Если линиями, обозначаемыми dxbb?, управляют, они не несут никакой достоверной информации).

Таблица 2.19 — Использование адресных линий для выбора четырехбайтовой группы

Мнемоническое обозначение	Режим адресации	A02—A15 (см. примечание)	A16—A23	A24—A31	IACK*
A16	КОРОТКАЯ	DVBM	dxbm?	dxbm?	dhbm?
A24	СТАНДАРТНАЯ	DVBM	DVBM	dxbm?	dhbm?
A32	РАСШИРЕННАЯ	DVBM	DVBM	DVBM	dhbm?

Примечание — Линия A01 вместе с LWORD*, DS0* и DS1* используется для определения, к каким из четырех байтов внутри четырехбайтовой группы производится обращение (см. таблицу 2.20).

Таблица 2.20 — Использование DS0*, DS1*, A01 и LWORD* во время различных циклов

Мнемоническое обозначение	Тип цикла	DS1*	DS0*	A01	LWORD*
ADO	Только адрес (Adress Only)	dhbm?	dhbm?	dxbm?	dxbm?
D08(EO)	Пересылки одного четного байта:				
	СЧИТЫВАНИЕ или ЗАПИСЬ БАЙТА (0)	DLBM	dhbm?	DLBM	dhbm?
	СЧИТЫВАНИЕ или ЗАПИСЬ БАЙТА (2)	DLBM	dhbm?	DHBM	dhbm?
D08(EO) или D08(O)	Пересылки одного нечетного байта:				
	СЧИТЫВАНИЕ или ЗАПИСЬ БАЙТА(1)	dhbm?	DLBM	DLBM	dhbm?
	СЧИТЫВАНИЕ или ЗАПИСЬ БАЙТА (3)	dhbm?	DLBM	DHBM	dhbm?
D16	Двухбайтовые пересылки:				
	СЧИТЫВАНИЕ или ЗАПИСЬ БАЙТОВ (0—1)	DLBM	DLBM	DLBM	dhbm?
	СЧИТЫВАНИЕ или ЗАПИСЬ БАЙТОВ (2—3)	DLBM	DLBM	DHBM	dhbm?
D32	Четырехбайтовые пересылки:				
	СЧИТЫВАНИЕ или ЗАПИСЬ БАЙТОВ (0—3)	DLBM	DLBM	DLBM	DLBM
D08(EO): BLT	Однобайтовые блочные пересылки:				
	ОДНОБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ или ЗАПИСЬ	См. примечание 1			dhbm?
D16:BLT	Двухбайтовые блочные пересылки: ДВУХБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ или ЗАПИСЬ	DLBM	DLBM	Примечание 2	dhbm?
D32:BLT	Четырехбайтовые блочные пересылки:				
	ЧЕТЫРЕХБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ или ЗАПИСЬ	DLBM	DLBM	DLBM	DLBM
D08(EO): RMW	Однобайтовые пересылки Чтения—Модификации—Записи:				
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (0)	DLBM	dhbm?	DLBM	dhbm?
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (1)	dhbm?	DLBM	DLBM	dhbm?
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (2)	DLBM	dhbm?	DHBM	dhbm?
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (3)	dhbm?	DLBM	DHBM	dhbm?
D16:RMW	Двухбайтовые пересылки Чтения—Модификации—Записи:				
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ (0—1)	DLBM	DLBM	DLBM	dhbm?
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ (2—3)	DLBM	DLBM	DHBM	dhbm?

Окончание таблицы 2.20

Мнемоническое обозначение	Тип цикла	DS1*	DS0*	A01	LWORD*
D32:RMW	Четырехбайтовые пересылки Чтения—Модификации—Записи:				
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ (0—3)	DLBM	DLBM	DLBM	DLBM
D32:UAT	Невыровненные пересылки:				
	СЧИТЫВАНИЕ или ЗАПИСЬ БАЙТОВ (0—2)	DLBM	dhbm?	DLBM	DLBM
	СЧИТЫВАНИЕ или ЗАПИСЬ БАЙТОВ (1—3)	dhbm?	DLBM	DLBM	DLBM
	СЧИТЫВАНИЕ или ЗАПИСЬ БАЙТОВ (1—2)	DLBM	DLBM	DHBM	DLBM

Примечания

1 Во время однобайтовых блочных пересылок два строба данных устанавливаются низкими попеременно. При первой пересылке низким может быть установлен любой из стробов данных. Если первой адресуемой байтовой ячейкой является БАЙТ(0) или БАЙТ(2), то задатчик сначала устанавливает низким DS1*. Если первой адресуемой байтовой ячейкой является БАЙТ(1) или БАЙТ(3), то он сначала устанавливает низким DS0*. Уровень на линии A01 является достоверным только при первой пересылке данных (т.е. до первой установки исполнителем низким DTACK* или BERR*) и может устанавливаться низким или высоким в зависимости от того, с какого байта начинается однобайтовая блочная пересылка. Если первой байтовой ячейкой является БАЙТ(0) или БАЙТ(1), то задатчик устанавливает на A01 низкий уровень. Если первой байтовой ячейкой является БАЙТ(2) или БАЙТ (3), то задатчик устанавливает на A01 высокий уровень.

Ниже приводится пример использования DS0*, DS1*, A01 и LWORD* во время цикла однобайтовой блочной пересылки, которая начинается с БАЙТА (2):

		DS1*	DS0*	A01	LWORD*
Первая пересылка данных	БАЙТ (2)	DLBM	DHBM	DHBM	dhbm?
↓	БАЙТ (3)	DHBM	DLBM	dxbm?	dxbm?
	БАЙТ (0)	DLBM	DHBM	dxbm?	dxbm?
	БАЙТ (1)	DHBM	DLBM	dxbm?	dxbm?
Последняя пересылка данных	БАЙТ (2)	DLBM	DHBM	dxbm?	dxbm?

2 Во время двухбайтовой блочной пересылки данных уровень на A01 является достоверным только при первой пересылке данных (т.е. до первой установки исполнителем низким DTACK* или BERR*) и устанавливается низким или высоким в зависимости от того, с какой двухбайтовой группы начинается двухбайтовая блочная пересылка. Если первой двухбайтовой группой являются БАЙТЫ (0—1), то задатчик устанавливает на A01 низкий уровень. Если первой двухбайтовой группой являются БАЙТЫ(2—3), то задатчик устанавливает на A01 высокий уровень.

Таблица 2.21 — Использование линий данных для пересылки данных

Мнемоническое обозначение	Тип цикла	D24—D31	D16—D23	D08—D15	D00—D07
ADO	Только адрес	dxbm?	dxbm?	dxbm?	dxbm?
D08(EO)	Пересылки одного четного байта:				
	СЧИТЫВАНИЕ БАЙТА (0)	dxbs?	dxbs?	DVBS	dxbs?
	СЧИТЫВАНИЕ БАЙТА (2)	dxbs?	dxbs?	DVBS	dxbs?

Продолжение таблицы 2.21

Мнемоническое обозначение	Тип цикла	D24—D31	D16—D23	D08—D15	D00—D07
	ЗАПИСЬ БАЙТА (0)	dxbm?	dxbm?	DVBM	dxbm?
	ЗАПИСЬ БАЙТА (2)	dxbm?	dxbm?	DVBM	dxbm?
D08(EO) или D08(O)	Пересылки одного нечетного байта:				
	СЧИТЫВАНИЕ БАЙТА (1)	dxbs?	dxbs?	dxbs?	DVBS
	СЧИТЫВАНИЕ БАЙТА (3)	dxbs?	dxbs?	dxbs?	DVBS
	ЗАПИСЬ БАЙТА (1)	dxbm?	dxbm?	dxbm?	DVBM
	ЗАПИСЬ БАЙТА (3)	dxbm?	dxbm?	dxbm?	DVBM
D16	Двухбайтовые пересылки:				
	СЧИТЫВАНИЕ БАЙТОВ (0—1)	dxbs?	dxbs?	DVBS	DVBS
	СЧИТЫВАНИЕ БАЙТОВ (2—3)	dxbs?	dxbs?	DVBS	DVBS
	ЗАПИСЬ БАЙТОВ (0—1)	dxbm?	dxbm?	DVBM	DVBM
	ЗАПИСЬ БАЙТОВ (2—3)	dxbm?	dxbm?	DVBM	DVBM
D32	Четырехбайтовые пересылки:				
	СЧИТЫВАНИЕ БАЙТОВ (0—3)	DVBS	DVBS	DVBS	DVBS
	ЗАПИСЬ БАЙТОВ (0—3)	DVBM	DVBM	DVBM	DVBM
D08(EO): BLT	Однобайтовые блочные пересылки:				
	ОДНОБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ	dxbs?	dxbs?	См. примечание	
	ОДНОБАЙТОВАЯ БЛОЧНАЯ ЗАПИСЬ	dxbm?	dxbm?	См. примечание	
D16:BLT	Двухбайтовые блочные пересылки:				
	ДВУХБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ	dxbs?	dxbs?	DVBS	DVBS
	ДВУХБАЙТОВАЯ БЛОЧНАЯ ЗАПИСЬ	dxbm?	dxbm?	DVBM	DVBM
D32:BLT	Четырехбайтовые блочные пересылки:				
	ЧЕТЫРЕХБАЙТОВОЕ БЛОЧНОЕ СЧИТЫВАНИЕ	DVBS	DVBS	DVBS	DVBS
	ЧЕТЫРЕХБАЙТОВАЯ БЛОЧНАЯ ЗАПИСЬ	DVBM	DVBM	DVBM	DVBM
D08(EO): RMW	Однобайтовые пересылки Чтения—Модификации—Записи:				
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (0)	dxbb?	dxbb?	DVBB	dxbb?
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (1)	dxbb?	dxbb?	dxbb?	DVBB

Окончание таблицы 2.21

Мнемоническое обозначение	Тип цикла	D24—D31	D16—D23	D08—D15	D00—D07
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (2)	dxbb?	dxbb?	DVBB	dxbb?
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТА (3)	dxbb?	dxbb?	dxbb?	DVBB
D16:RMW	Двухбайтовые пересылки Чтения—Модификации—Записи:				
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ (0—1)	dxbb?	dxbb?	DVBB	DVBB
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ (2—3)	dxbb?	dxbb?	DVBB	DVBB
D32:RMW	Четырехбайтовые пересылки Чтения—Модификации—записи:				
	ЧТЕНИЕ—МОДИФИКАЦИЯ—ЗАПИСЬ БАЙТОВ (0—3)	DVBB	DVBB	DVBB	DVBB
D32:UAT	Невыровненные пересылки:				
	СЧИТЫВАНИЕ БАЙТОВ (0—2)	DVBS	DVBS	DVBS	dxbs?
	СЧИТЫВАНИЕ БАЙТОВ (1—3)	dxbs?	DVBS	DVBS	DVBS
	СЧИТЫВАНИЕ БАЙТОВ (1—2)	dxbs?	DVBS	DVBS	dxbs?
	ЗАПИСЬ БАЙТОВ (0—2)	DVBM	DVBM	DVBM	dxbm?
	ЗАПИСЬ БАЙТОВ (1—3)	dxbm?	DVBM	DVBM	DVBM
	ЗАПИСЬ БАЙТОВ (1—2)	dxbm?	DVBM	DVBM	dxbm?

Примечание — Во время однобайтовых блочных пересылок данные передаются восемью разрядами одновременно по линиям D00—D07 или D08—D15. Пример однобайтового блочного считывания приведен ниже:

	D08—D15	D00—D07
Первая пересылка данных	DVBS	dxbs?
↓	dxbs?	DVBS
	DVBS	dxbs?
	dxbs?	DVBS
	DVBS	dxbs?
	dxbs?	DVBS
Последняя пересылка данных	DVBS	dxbs?

Таблица 2.22 — Значения параметров временных соотношений для задатчика, исполнителя и адресного монитора

Номер параметра	Значение параметра, нс					
	Задатчик (см. также таблицу 2.24)		Исполнитель (см. также таблицу 2.25)		Адресный монитор (см. также таблицу 2.26)	
	мин.	макс.	мин.	макс.	мин.	макс.
1	0	—	—	—	—	—
2	0	—	—	—	—	—
3	60	—	—	—	—	—

Окончание таблицы 2.22

Номер параметра	Значение параметра, нс					
	Задатчик (см. также таблицу 2.24)		Исполнитель (см. также таблицу 2.25)		Адресный монитор (см. также таблицу 2.26)	
	мин.	макс.	мин.	макс.	мин.	макс.
4	35	—	10	—	10	—
5	40	—	30	—	30	—
6	0	—	0	—	—	—
7	0	—	0	—	—	—
8	35	—	10	—	—	—
9	0	—	0	—	—	—
10	0	—	—10	—	—10	—
11	40	—	30	—	30	—
12	35	—	10	—	10	—
13	—	10	—	20	—	20
14	0	—	0	—	—	—
15	0	—	0	—	—	—
16	0	—	0	—	—	—
17	40	—	30	—	30	—
18	0	—	0	—	—	—
19	40	—	30	—	30	—
20	0	—	0	—	—	—
21	0	—	0	—	—	—
22	0	—	0	—	—	—
23	10	—	0	—	0	—
24A	0	—	—	—	—	—
24B	0	—	—	—	—	—
25	—	25	—	—	—	—
26	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
31	0	—	0	—	—	—
32	—	—	10	—	10	—
33	—	—	30	—	30	—

Примечание — *T* — значение тайм-аута в микросекундах.

Таблица 2.23 — Значения параметров временны́х соотношений для шинного таймера

Номер параметра	Значение параметра	
	мин.	макс.
28	*T*	2*T*
30	0	-

Примечание — *T* — значение тайм-аута в микросекундах. См. также таблицу 2.27.

Таблица 2.24 — Задатчик. Правила и замечания по временны́м соотношениям

Номер параметра (см.таблицу 2.22)	Правила и замечания
1	***Правило 2.27.*** *Получая управление шиной пересылки данных, задатчик НЕ ДОЛЖЕН УПРАВЛЯТЬ ни одной из линий IACK*, AM0—AM5, A01—A31, LWORD*, D00—D31, WRITE*, DS0*, DS1* или AS*, пока предыдущий задатчик не позволяет сигналу на линии AS* превысить напряжение низкого уровня*

Продолжение таблицы 2.24

Номер параметра (см.таблицу 2.22)	Правила и замечания
	Замечание 2.35. Порядок предоставления права использования шины пересылки данных запросчику задатчика определен в разделе 3
2	***Правило 2.28.*** *Получая управление шиной пересылки данных, задатчик НЕ ДОЛЖЕН УПРАВЛЯТЬ ни одной из линий IACK*, AM0—AM5, A01—A31, LWORD*, D00—D31, WRITE*, DS0*, DS1* или AS* до предоставления шины его запросчику*
	Замечание 2.36. Порядок предоставления права использования шины пересылки данных запросчику задатчика определен в разделе 3
3	***Правило 2.29.*** *Получая управление шиной пересылки данных, задатчик НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ AS* низким до истечения указанного времени после того, как предыдущий задатчик позволит сигналу на линии AS* превысить напряжение низкого уровня*
	Замечание 2.37. Правило 2.29 обеспечивает соблюдение для исполнителей временного параметра 5 при смене задатчика шины пересылки данных
4	***Правило 2.30.*** *Задатчик НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ AS* низким до истечения указанного минимального времени удержания высокого уровня IACK* и достоверных уровней сигналов на требуемых линиях из A01—A31, AM0—AM5 и LWORD**
	Замечание 2.38. Таблица 19 указывает конкретные линии из A01—A31, которыми должен управлять задатчик. Использование задатчиком линий AM0—AM5 определено в таблице 3, а линии LWORD* — в таблице 20
5	***Правило 2.31.*** *Если задатчик использует шину пересылки данных в течение двух последовательных циклов, то он НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ AS* низким до истечения указанного минимального времени удержания высокого уровня этого сигнала*
6	***Правило 2.32.*** *После цикла считывания задатчик НЕ ДОЛЖЕН УПРАВЛЯТЬ ни одной из линий D00—D31 до установления высокими обоих сигналов DTACK* и BERR**
7	***Правило 2.33.*** *Во время циклов считывания задатчик НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DSA* низким, пока он не освободит все линии D00—D31*
8	***Правило 2.34.*** *Во время циклов записи задатчик НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DSA* низким до истечения указанного минимального времени удержания достоверных уровней сигналов на требуемых линиях из D00—D31*
	Замечание 2.39. Конкретные линии из D00—D31, которыми должен управлять задатчик, указаны в таблице 21
9	***Правило 2.35.*** *Задатчик НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DSA* низким до установления высокими обоих сигналов DTACK* и BERR**
10	***Правило 2.36.*** *Задатчик НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DSA* низким, пока не установит низким AS**
11	***Правило 2.37.*** *Задатчик НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DSA* низким до истечения указанного минимального времени одновременного удержания высокими DS0* и DS1**
12	***Правило 2.38.*** *Задатчик НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DSA* низким до истечения указанного минимального времени удержания достоверного уровня сигнала на линии WRITE**
13	***Правило 2.39.*** *Во время циклов, в которых задатчик переводит в низкое состояние оба сигнала DS0* и DS1*, он ДОЛЖЕН УСТАНАВЛИВАТЬ DSB* низким в пределах указанного максимального времени после того, как установит низким DSA**

Продолжение таблицы 2.24

Номер параметра (см.таблицу 2.22)	Правила и замечания
	Замечание 2.40. Данное правило не распространяется на пересылки, в которых устанавливается низким один из сигналов DS0* или DS1*, но не оба вместе
14	***Правило 2.40.*** *Во время всех циклов пересылки данных, за исключением циклов Чтения-Модификации-Записи, задатчик ДОЛЖЕН СОХРАНЯТЬ достоверные уровни сигналов на линиях A01—A31 и соответствующий уровень LWORD* до тех пор, пока не обнаружит первый отрицательный перепад DTACK* или BERR** **Замечание 2.41.** Во время всех циклов пересылки данных, за исключением блочных пересылок и циклов Чтения-Модификации -Записи, произойдет только один отрицательный перепад DTACK* или BERR*
15	***Правило 2.41.*** *Во время циклов Чтения—Модификации—Записи задатчик ДОЛЖЕН СОХРАНЯТЬ достоверные уровни сигналов на линиях A01—A31 и соответствующий уровень LWORD* до тех пор, пока не обнаружит второй отрицательный перепад DTACK* или BERR**
16	***Правило 2.42.*** *Во время всех циклов пересылки данных задатчик ДОЛЖЕН СОХРАНЯТЬ достоверные уровни кода модификатора адреса и ОБЕСПЕЧИВАТЬ высокий уровень сигналов на линии IACK* до тех пор, пока не обнаружит последний отрицательный перепад DTACK* или BERR** **Замечание 2.42.** Во время всех циклов пересылки данных, за исключением блочных пересылок и циклов Чтения—Модификации—Записи, произойдет только один отрицательный перепад DTACK* или BERR*
17	***Правило 2.43.*** *Задатчик НЕ ДОЛЖЕН ИЗМЕНЯТЬ уровни сигналов на линиях IACK*, A01—A31, AM0—AM5 или LWORD* в течение указанного минимального времени после того, как он установит низким AS**
18	***Правило 2.44.*** *Во время всех циклов пересылки данных задатчик ДОЛЖЕН УДЕРЖИВАТЬ AS* низким до тех пор, пока не обнаружит последний отрицательный перепад DTACK* или BERR** **Замечание 2.43.** Во время всех циклов пересылки данных, за исключением блочных пересылок и циклов Чтения—Модификации—Записи, произойдет только один отрицательный перепад DTACK* или BERR*
19	***Правило 2.45.*** *Задатчик ДОЛЖЕН УДЕРЖИВАТЬ AS* низким в течение указанного минимального времени*
20	***Правило 2.46.*** *Установив DSA* низким, задатчик ДОЛЖЕН УДЕРЖИВАТЬ его низким до обнаружения низкого уровня DTACK* или BERR** **Замечание 2.44.** Во время всех циклов пересылки данных, за исключением блочных пересылок и циклов Чтения—Модификации—Записи, произойдет только один отрицательный перепад DTACK* или BERR*
21	***Правило 2.47.*** *Установив DSB* низким, задатчик ДОЛЖЕН УДЕРЖИВАТЬ его низким до обнаружения низкого уровня DTACK* или BERR** **Замечание 2.45.** Во время всех циклов пересылки данных, за исключением блочных пересылок и циклов Чтения—Модификации—Записи, произойдет только один отрицательный перепад DTACK* или BERR*
22	***Правило 2.48.*** *Установив DSA* низким во время циклов записи, задатчик НЕ ДОЛЖЕН ИЗМЕНЯТЬ уровни сигналов ни на одной из линий D00-D31 до обнаружения низкого уровня DTACK* или BERR** **Замечание 2.46.** Во время всех циклов пересылки данных, за исключением блочных пересылок и циклов Чтения—Модификации—Записи, произойдет только один отрицательный перепад DTACK* или BERR*
23	***Правило 2.49.*** *Установив DSA* низким, задатчик НЕ ДОЛЖЕН ИЗМЕНЯТЬ уровень сигнала на линии WRITE* до истечения указанного минимального времени после установки высокими обоих сигналов DS0* и DS1** **Замечание 2.47.** Во время всех циклов пересылки данных, за исключением блочных пересылок и циклов Чтения—Модификации—Записи, произойдет только один отрицательный перепад DTACK* или BERR*
24А	***Правило 2.50.*** *Если задатчик устанавливает AS* высоким, либо освобождает линию этого сигнала после того, как его запросчик освободит BBSY*, то этот задатчик ДОЛЖЕН ОСВОБОДИТЬ линии IACK*, AM0—AM5, A01—A31, LWORD*, D00—D31, WRITE*, DS0* и DS1* прежде, чем позволит сигналу AS* превысить напряжение низкого уровня*

Окончание таблицы 2.24

Номер параметра (см.таблицу 2.22)	Правила и замечания
	Замечание 2.43. Процесс освобождения линии BBSY* запросчиком задатчика описан в разделе 3
24B	***Правило 2.51.*** *Если задатчик устанавливает AS* высоким либо освобождает линию этого сигнала прежде, чем его запросчик освободит BBSY*, то этот задатчик ДОЛЖЕН ОСВОБОДИТЬ линии AS*, IACK*, AM0—AM5, A01—A31, LWORD*, D00—D31, WRITE*, DS0* и DS1* прежде, чем позволит своему запросчику освободить BBSY**
	Замечание 2.44. Процесс освобождения линии BBSY* запросчиком задатчика описан в разделе 3
25	***Правило 2.52.*** *Если задатчик устанавливает AS* высоким либо освобождает линию этого сигнала после того, как его запросчик освободит BBSY*, то этот задатчик ДОЛЖЕН ОСВОБОДИТЬ AS* в пределах указанного времени после того, как позволит этому сигналу превысить напряжение низкого уровня*
	Замечание 2.45. Процесс освобождения линии BBSY* запросчиком задатчика описан в разделе 3
26	**Замечание 2.46.** Временно́й параметр 26 гарантирует, что во время циклов считывания линии данных не будут возбуждаться, пока задатчик не установит DSA* низким
27	**Замечание 2.47.** Во время циклов считывания задатчику гарантируется, что после установки DTACK* низким уровни сигналов на линиях данных будут достоверными в пределах указанного времени. Данное замечание не распространяется на циклы, в которых исполнитель устанавливает низким BERR*, а не DTACK*
28	**Замечание 2.48.** Задатчику гарантируется, что ни DTACK*, ни BERR* не установятся низкими, пока не истечет указанное минимальное время после того, как он установит DSA* низким. Шинный таймер гарантирует задатчику, что если DTACK* не переходит в низкое состояние по истечении тайм-аута и в пределах удвоенного тайм-аута, то шинный таймер установит BERR* низким
29	**Замечание 2.49.** Во время циклов считывания задатчику гарантируется, что на линиях данных будут сохраняться достоверные данные, пока он не установит DSA* высоким
30	**Замечание 2.50.** Временно́й параметр 30 гарантирует, что ни DTACK*, ни BERR* не установятся высокими до тех пор, пока задатчик не установит высокими оба сигнала DS0* и DS1*
31	**Замечание 2.51.** Во время циклов считывания задатчику гарантируется, что линии данных будут освобождены к тому моменту, когда DTACK* и BERR* установятся высокими

Таблица 2.25 — Исполнитель. Правила и замечания по временны́м соотношениям

Номер параметра (см.таблицу 2.22)	Правила и замечания
4	**Замечание 2.52.** Всем исполнителям гарантируется, что на линиях IACK*, A01—A31, AM0—AM5 и LWORD* будут выдержаны достоверные уровни сигналов в течение указанного минимального времени до обнаружения этими исполнителями отрицательного перепада AS*

Продолжение таблицы 2.25

Номер параметра (см.таблицу 2.22)	Правила и замечания
5	**Замечание 2.53.** Всем исполнителям гарантируется, что между циклами пересылки данных сигнал AS* будет выдержан высоким в течение указанного минимального времени
6	**Замечание 2.54.** Во время циклов считывания отвечающему исполнителю гарантируется, что ни одна из линий D00—D31 не будет возбуждаться никаким другим функциональным блоком до тех пор, пока этот исполнитель не освободит линии DTACK* и BERR*, установив на них сигналы высокого уровня
7	**Замечание 2.55.** Во время циклов считывания отвечающему исполнителю гарантируется, что линии данных будут освобождены всеми другими функциональными блоками к моменту, когда DSA* установится низким
8	**Замечание 2.56.** Во время циклов записи отвечающему исполнителю гарантируется, что на линиях данных будут выдержаны достоверные уровни сигналов в течение указанного минимального времени к моменту, когда этот исполнитель обнаружит DSA* низким
9	**Замечание 2.57.** Отвечающему исполнителю гарантируется, что ни DS0*, ни DS1* не будут установлены низкими до установления высокими DTACK* и BERR* от предыдущего цикла
10	**Замечание 2.58.** Из-за разницы во времени распространения сигналов по магистрали исполнители на шине пересылки данных могут обнаружить отрицательный перепад DSA* прежде, чем обнаружат отрицательный перепад AS*. Однако исполнителям гарантируется, что отрицательный перепад DSA* не будет опережать отрицательный перепад AS* более чем на абсолютную величину указанного времени
11	**Замечание 2.59.** Исполнителям гарантируется, что между последовательными пересылками данных сигналы DS0* и DS1* будут одновременно выдержаны высокими в течение указанного минимального времени
12	**Замечание 2.60.** Исполнителям гарантируется, что на линии WRITE* достоверный уровень сигнала будет выдержан в течение указанного минимального времени до возникновения отрицательного перепада DSA*
13	**Замечание 2.61.** Если задатчик устанавливает низкими оба сигнала DS0* и DS1*, то отвечающему исполнителю гарантируется, что DSB* будет установлен низким в пределах указанного максимального времени после того, как установится низким DSA*
14	**Замечание 2.62.** Во время всех циклов пересылки данных, за исключением циклов Чтения—Модификации—Записи, отвечающему исполнителю гарантируется, что на линиях A01—A31 и LWORD* будут удерживаться достоверные уровни сигналов до первой установки этим исполнителем низким DTACK* или BERR* при условии, что он сделает ее в пределах тайм-аута шины
15	**Замечание 2.63.** Во время всех циклов Чтения—Модификации—Записи, отвечающему исполнителю гарантируется, что на линиях A01—A31 и LWORD* будут удерживаться достоверные уровни сигналов до второй установки этим исполнителем низким DTACK* или BERR* при условии, что он сделает ее в пределах тайм-аута шины
16	**Замечание 2.64.** Отвечающему исполнителю гарантируется, что на линиях IACK* и AM0—AM5 будут удерживаться достоверные уровни сигналов до последней установки этим исполнителем низким DTACK* или BERR* при условии, что он сделает ее в пределах тайм-аута шины
17	**Замечание 2.65.** Исполнителям гарантируется, что на линиях IACK*, A01—A31, AM0—AM5 и LWORD* будут удерживаться достоверные уровни сигналов в течение указанного минимального времени после отрицательного перепада AS*. При исполнении циклов Только Адрес это время гарантируется задатчиком. При исполнении всех других циклов это время получается из значений временных параметров 10, 14, 16 и 28

Окончание таблицы 2.25

Номер параметра (см.таблицу 2.22)	Правила и замечания
18	**Замечание 2.66.** Отвечающему исполнителю гарантируется, что AS* будет оставаться низким до установки этим исполнителем низким DTACK* или BERR* при условии, что он сделает ее в пределах тайм-аута шины
19	**Замечание 2.67.** Исполнителям гарантируется, что AS* будет оставаться низким в течение указанного минимального времени
20	**Замечание 2.68.** Отвечающему исполнителю гарантируется, что как только DSA* установится низким, он останется в этом состоянии до установки этим исполнителем низким DTACK* или BERR* при условии, что он сделает ее в пределах тайм-аута шины
21	**Замечание 2.69.** Отвечающему исполнителю гарантируется, что как только DSB* установится низким, он останется в этом состоянии до установки этим исполнителем низким DTACK* или BERR* при условии, что он сделает ее в пределах тайм-аута шины
22	**Замечание 2.70.** Во время циклов записи отвечающему исполнителю гарантируется, что на линиях данных будут удерживаться достоверные уровни сигналов до установки этим исполнителем низким DTACK* или BERR* при условии, что он сделает ее в пределах тайм-аута шины
23	**Замечание 2.71.** Отвечающему исполнителю гарантируется, что на линии WRITE* будет сохраняться достоверный уровень сигнала до установки высокими обоих стробов данных
26	***Правило 2.53.*** *Во время циклов считывания отвечающий исполнитель НЕ ДОЛЖЕН ВОЗБУЖДАТЬ линии данных до установки DSA* низким*
27	***Правило 2.54.*** *Во время циклов считывания отвечающий исполнитель НЕ ДОЛЖЕН ВОЗБУЖДАТЬ линию DTACK*, пока не установит достоверные данные на линиях данных* **Замечание 2.72.** Правило 2.54 не распространяется на циклы, в которых отвечающий исполнитель устанавливает низким BERR*, а не DTACK*
28	***Правило 2.55.*** *Отвечающий исполнитель ДОЛЖЕН ОЖИДАТЬ в течение указанного минимального времени после установки DSA* низким, прежде чем установить низким DTACK* или BERR**
29	***Правило 2.56.*** *Установив DTACK* низким во время циклов считывания, отвечающий исполнитель НЕ ДОЛЖЕН ИЗМЕНЯТЬ уровней сигналов на линиях D00—D31 до установки DSA* высоким*
30	***Правило 2.57.*** *Установив сигнал на линии DTACK* или BERR* низким, отвечающий исполнитель НЕ ДОЛЖЕН ОСВОБОЖДАТЬ ее, пока не обнаружит высокими оба сигнала DS0* и DS1**
31	***Правило 2.58.*** *Во время циклов считывания отвечающий исполнитель ДОЛЖЕН ОСВОБОДИТЬ все линии D00—D31, прежде чем освободить DTACK* или BERR* переводом его в высокое состояние*
32	**Замечание 2.73.** Исполнителям гарантируется, что на линиях IACK*, LWORD*, A01—A31 и AM0—AM5 достоверные уровни сигналов будут выдержаны в течение указанного минимального времени до обнаружения исполнителями отрицательного перепада DSA*. Это время получается из значений временных параметров 4 и 10
33	**Замечание 2.74.** Во время циклов пересылки данных исполнителям гарантируется, что либо DS0*, либо DS1* будет оставаться низким, как минимум, в течение указанного времени. Это время получается из значения временного параметра 28, определяющего для отвечающего исполнителя требуемое минимальное время ожидания до установки низким DTACK* или BERR*

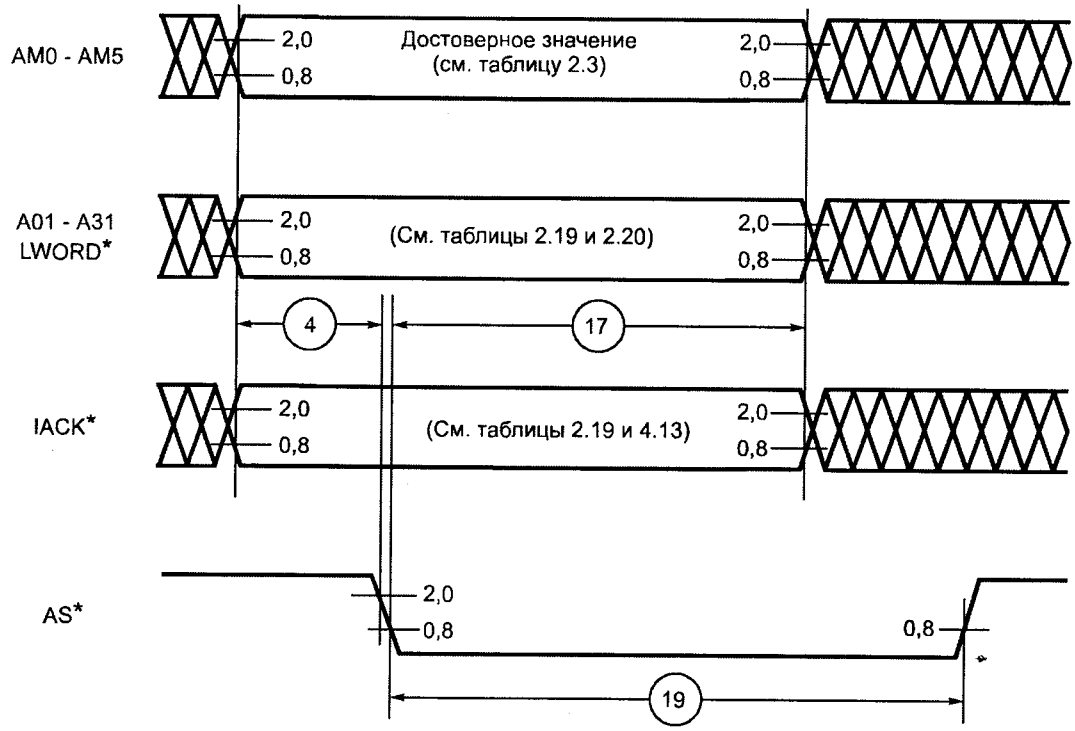
Таблица 2.26 — Адресный монитор. Замечания по временным соотношениям

Номер параметра (см. таблицу 2.22)	Замечания
4	**Замечание 2.75.** Адресным мониторам гарантируется, что на линиях IACK*, A01—A31, AM0—AM5 и LWORD* достоверные уровни сигналов будут выдержаны в течение указанного минимального времени до обнаружения этими адресными мониторами отрицательного перепада AS*
5	**Замечание 2.76.** Адресным мониторам гарантируется, что между циклами пересылки данных AS* будет выдержан высоким в течение указанного минимального времени
10	**Замечание 2.77.** Из-за разницы во времени распространения сигналов по магистрали адресные мониторы на шине пересылки данных могут обнаружить отрицательный перепад DSA* прежде, чем обнаружат отрицательный перепад AS*. Однако адресным мониторам гарантируется, что отрицательный перепад сигнала на линии DSA* не будет опережать отрицательный перепад сигнала на линии AS* более чем на указанное время
11	**Замечание 2.78.** Адресным мониторам гарантируется, что между последовательными пересылками данных оба сигнала DS0 и DS1* будут одновременно выдержаны высокими в течение указанного минимального времени
12	**Замечание 2.79.** Адресным мониторам гарантируется, что на линии WRITE* достоверный уровень сигнала будет выдержан в течение указанного минимального времени до возникновения отрицательного перепада DSA*
13	**Замечание 2.80.** Если задатчик устанавливает низкими оба сигнала DS0* и DS1*, то адресным мониторам гарантируется, что DSB* будет установлен низким в пределах указанного максимального времени после того, как установится низким DSA*
17	**Замечание 2.81.** Адресным мониторам гарантируется, что на линиях IACK*, A01—A31, AM0—AM5 и LWORD* будут удерживаться достоверные уровни сигналов в течение указанного минимального времени после отрицательного перепада AS*. При исполнении циклов Только Адрес это время гарантируется задатчиком. При исполнении всех других циклов это время получается из значений временны́х параметров 10, 14, 16 и 28
19	**Замечание 2.82.** Адресным мониторам гарантируется, что AS* будет оставаться низким в течение указанного минимального времени
23	**Замечание 2.83.** Адресным мониторам гарантируется, что на линии WRITE* будет сохраняться достоверный уровень сигнала до установки высокими стробов данных DS0* и DS1*
32	**Замечание 2.84.** Адресным мониторам гарантируется, что на линиях IACK*, LWORD*, A01—A31 и AM0—AM5 достоверные уровни сигналов будут выдержаны в течение указанного минимального времени до обнаружения этими мониторами отрицательного перепада DSA*
33	**Замечание 2.85.** Во время циклов пересылки данных адресным мониторам гарантируется, что DS0* и/или DS1* будут оставаться низкими, по меньшей мере, в течение указанного минимального времени. Это время получается из временно́го параметра 28, определяющего для отвечающего исполнителя требуемое минимальное время ожидания до установки низким DTACK* или BERR*

Таблица 2.27 — Шинный таймер. Правила по временны́м соотношениям

Номер параметра (см. таблицу 2.23)	Правила
28	***Правило 2.59.*** *После того, как установится DSA* низким, шинный таймер ДОЛЖЕН ОЖИДАТЬ, как минимум, в течение тайм-аута, но не более чем удвоенное время тайм-аута, прежде чем установить низким BERR**
30	***Правило 2.60.*** *Установив сигнал на линии BERR* низким, шинный таймер НЕ ДОЛЖЕН ОСВОБОЖДАТЬ ее, пока не обнаружит высокими оба сигнала DS0* и DS1**

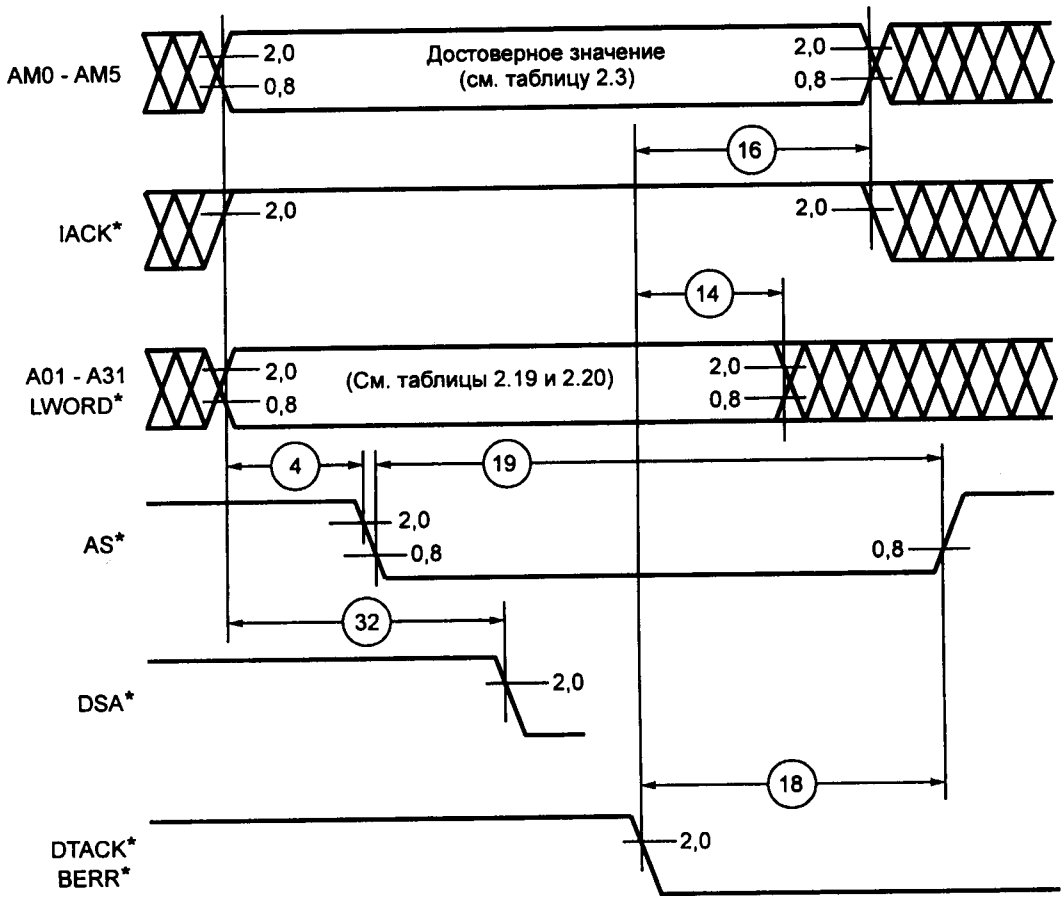
Рисунки 2.12—2.15 содержат временны́е диаграммы к правилам и замечаниям по временны́м соотношениям, связанным с широковещательной пересылкой адреса.



Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
4	35	—	10	—	10	—
17	40	—	30	—	30	—
19	40	—	30	—	30	—
Примечание — Все значения параметров в наносекундах.						

Рисунок 2.12 — Временна́я диаграмма широковещательной пересылки адреса. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Все типы циклов)

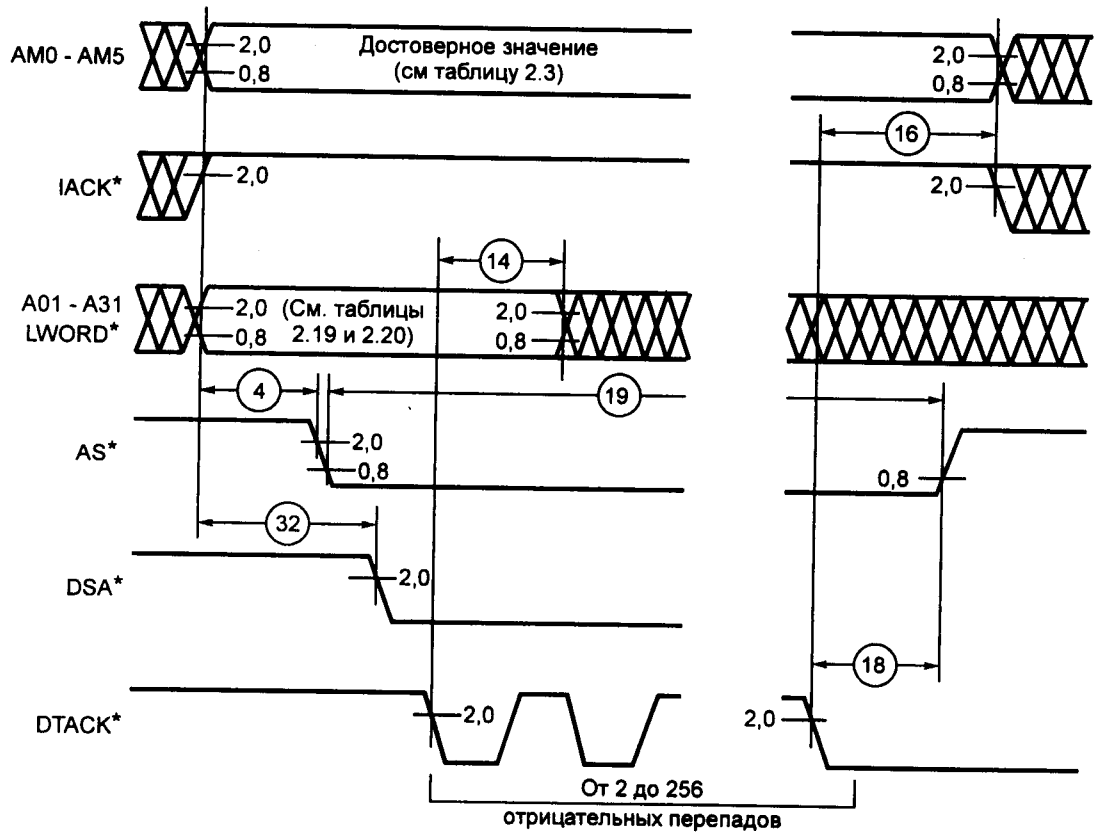


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
4	35	—	10	—	10	—
14	0	—	0	—	—	—
16	0	—	0	—	—	—
18	0	—	0	—	—	—
19	40	—	30	—	30	—
32	—	—	10	—	10	—

Примечание — Все значения параметров в наносекундах.

Рисунок 2.13 — Временна́я диаграмма широковещательной пересылки адреса.
ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР.
(Пересылка одного четного байта; пересылка одного нечетного байта; двухбайтовые пересылки; четырехбайтовые пересылки; невыровненные пересылки)

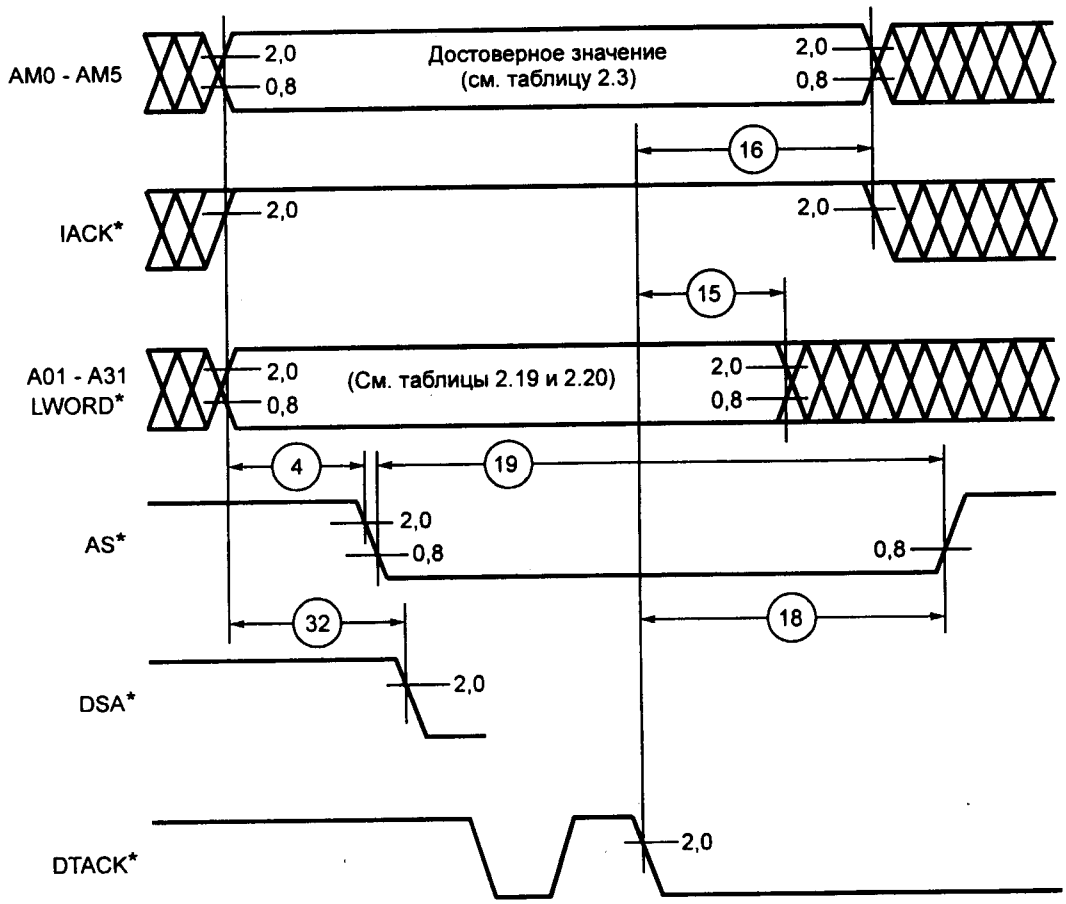


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
4	35	—	10	—	10	—
14	0	—	0	—	—	—
16	0	—	0	—	—	—
18	0	—	0	—	—	—
19	40	—	30	—	30	—
32	—	—	10	—	10	—

Примечание — Все значения параметров в наносекундах.

Рисунок 2.14 — Временна́я диаграмма широковещательной пересылки адреса. ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР. (Однобайтовые блочные пересылки; двухбайтовые блочные пересылки; четырехбайтовые блочные пересылки)



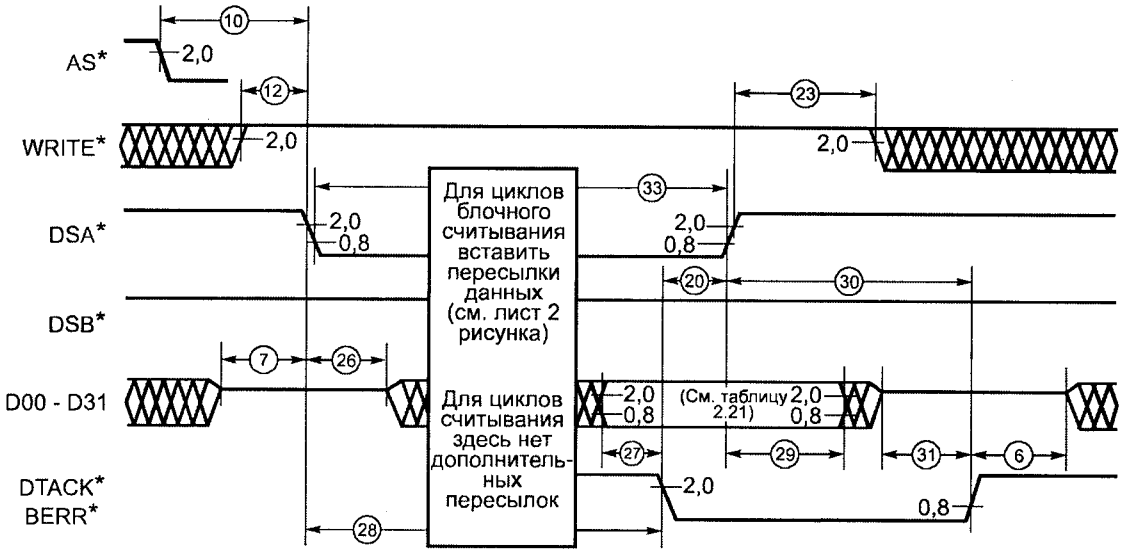
Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
4	35	—	10	—	10	—
15	0	—	0	—	—	—
16	0	—	0	—	—	—
18	0	—	0	—	—	—
19	40	—	30	—	30	—
32	—	—	10	—	10	—

Примечание — Все значения параметров в наносекундах.

Рисунок 2.15 — Временна́я диаграмма широковещательной пересылки адреса.
ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР.
(Однобайтовые циклы Чтение—Модификация—Запись; двухбайтовые циклы Чтение—Модификация—Запись; четырехбайтовые циклы Чтение—Модификация—Запись)

Рисунки 2.16—2.21 содержат временны́е диаграммы к правилам и замечаниям по временны́м соотношениям для задатчиков, исполнителей и адресных мониторов в части цикла, связанной с пересылкой данных.

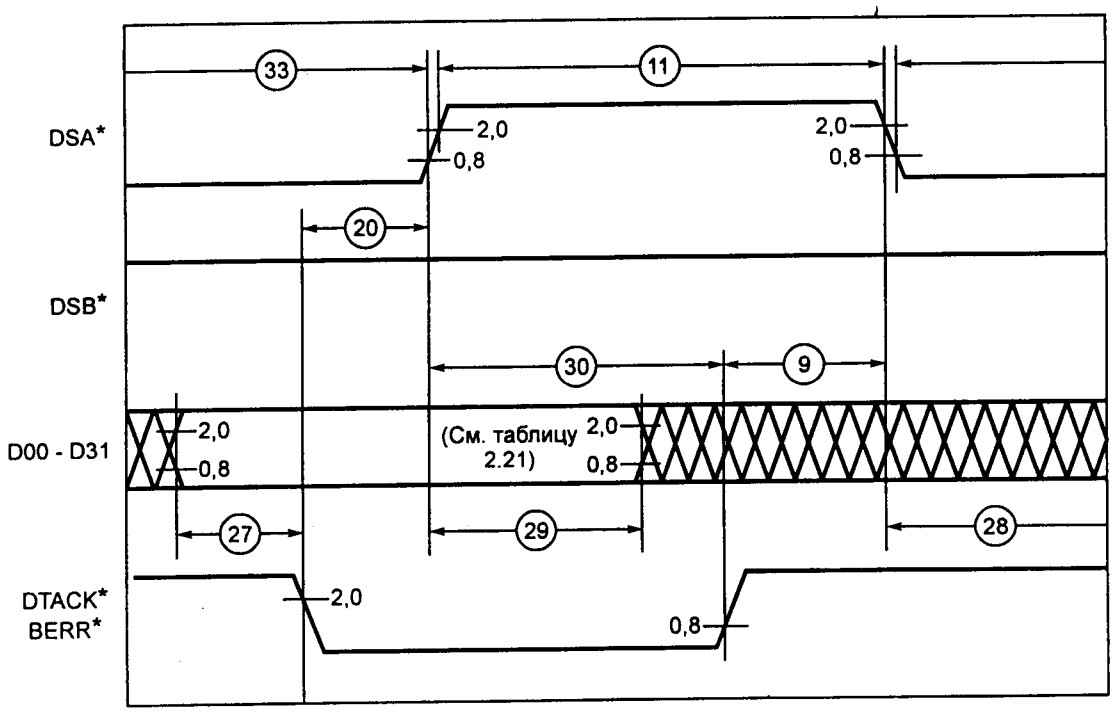


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
4	0	—	0	—	—	—
7	0	—	0	—	—	—
10	0	—	—10	—	—10	—
12	35	—	10	—	10	—
20	0	—	0	—	—	—
23	10	—	0	—	0	—
26	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
31	0	—	0	—	—	—
33	—	—	30	—	30	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.16, лист 1 — Временна́я диаграмма пересылки данных.
ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР.
(Считывание байта (0); считывание байта (1); считывание байта (2); считывание байта (3); считывание байтов (0—2); считывание байтов (1—3); однобайтовое блочное считывание)

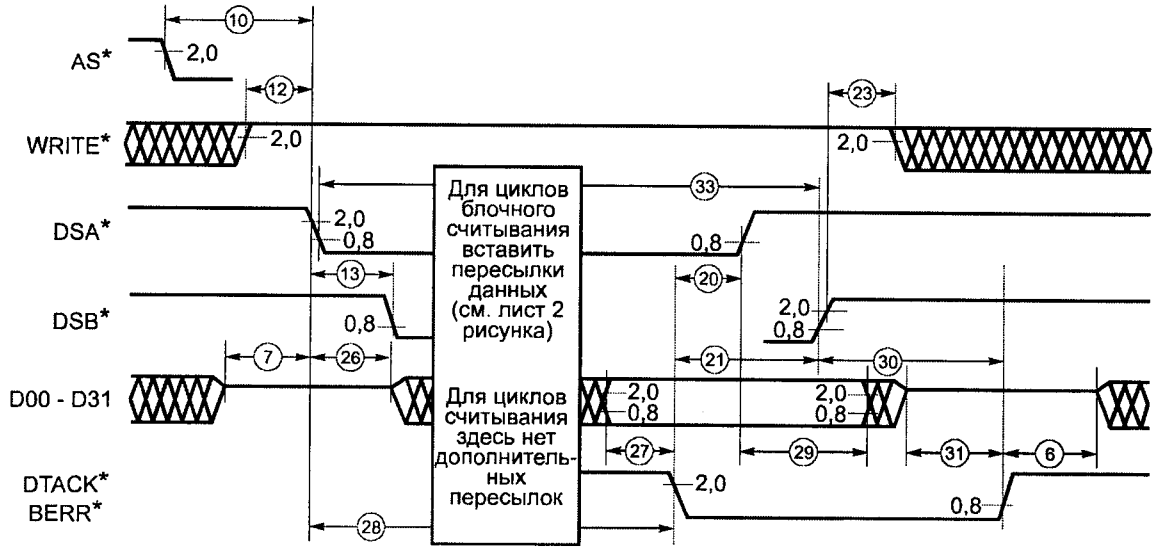


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
9	0	—	0	—	—	—
11	40	—	30	—	30	—
20	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
33	—	—	30	—	30	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.16, лист 2

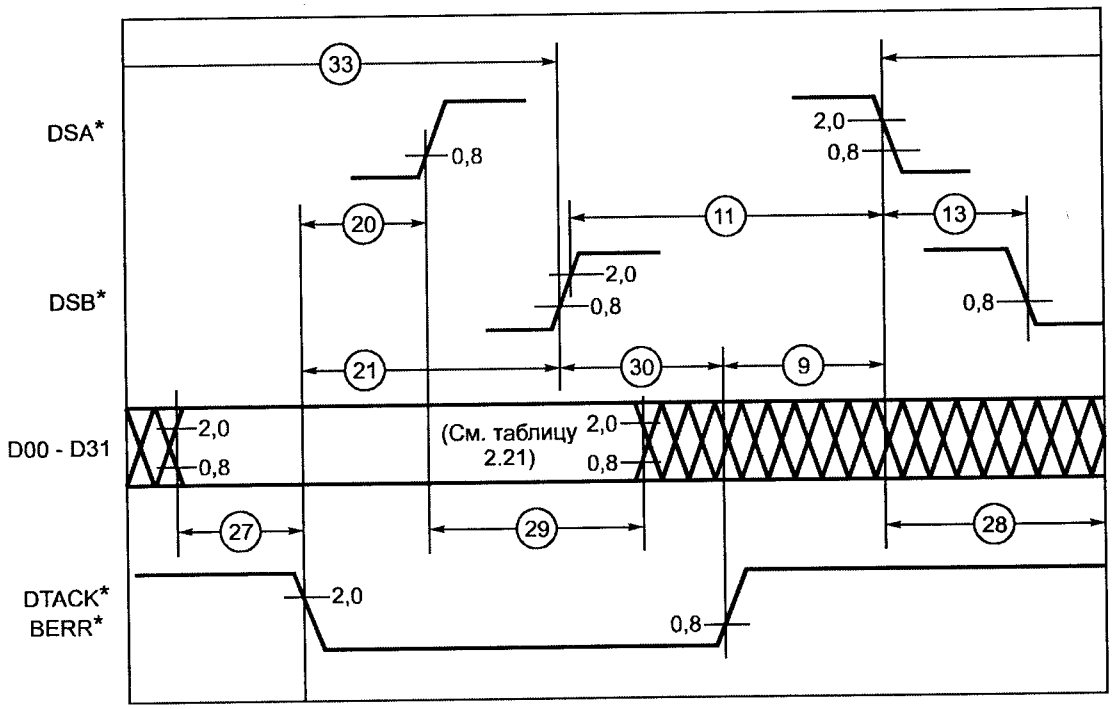


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
6	0	—	0	—	—	—
7	0	—	0	—	—	—
10	0	—	—10	—	—10	—
12	35	—	10	—	10	—
13	—	10	—	20	—	20
20	0	—	0	—	—	—
21	0	—	0	—	—	—
23	10	—	0	—	0	—
26	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
31	0	—	0	—	—	—
33	—	—	30	—	30	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.17, лист 1 — Временна́я диаграмма пересылки данных.
ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР.
(Считывание байтов (0—1); считывание байтов (2—3); считывание байтов (0—3); считывание байтов (1—2); двухбайтовое блочное считывание; четырехбайтовое блочное считывание)

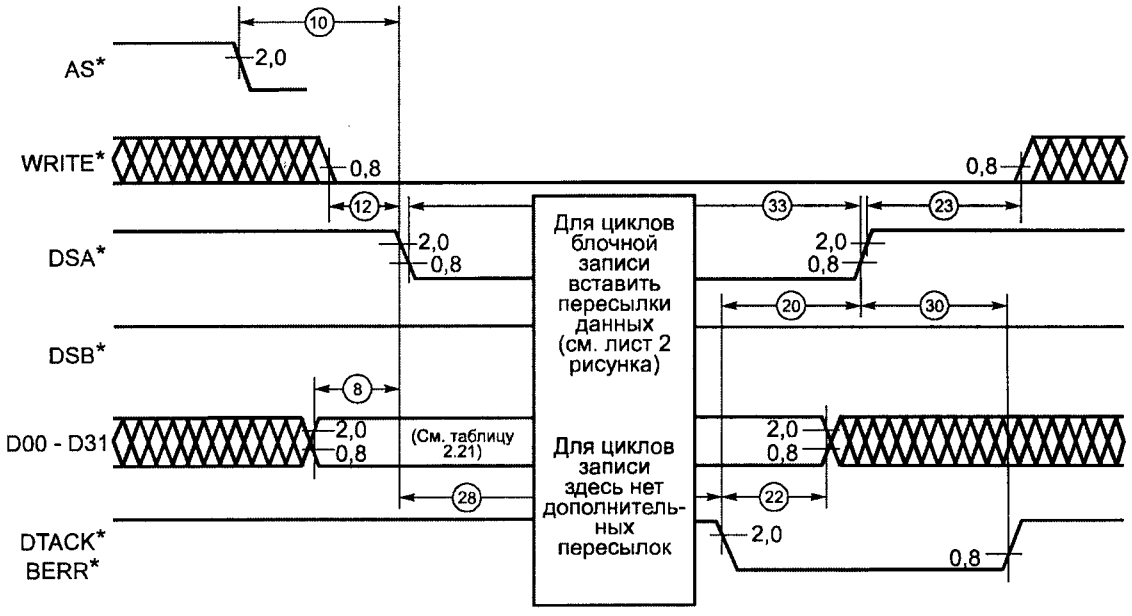


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
9	0	—	0	—	—	—
11	40	—	30	—	30	—
13	—	10	—	20	—	20
20	0	—	0	—	—	—
21	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
33	—	—	30	—	30	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.17, лист 2

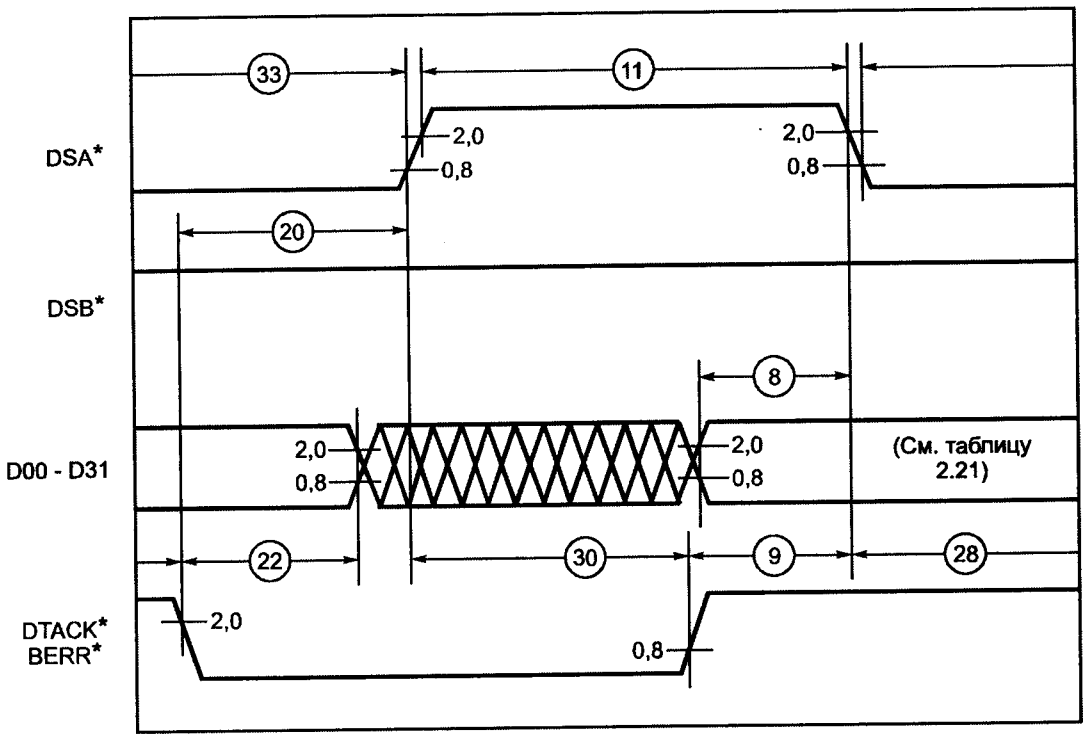


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
8	35	—	10	—	—	—
10	0	—	—10	—	—10	—
12	35	—	10	—	10	—
20	0	—	0	—	—	—
22	0	—	0	—	—	—
23	10	—	0	—	0	—
28	30	2*T*	30	—	—	—
30	0	—	0	—	—	—
33	—	—	30	—	30	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.18, лист 1 — Временна́я диаграмма пересылки данных.
ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР.
(Запись байта (0); запись байта (1); запись байта (2); запись байта (3); запись байтов (0—2); запись байтов (1—3); однобайтовая блочная запись)

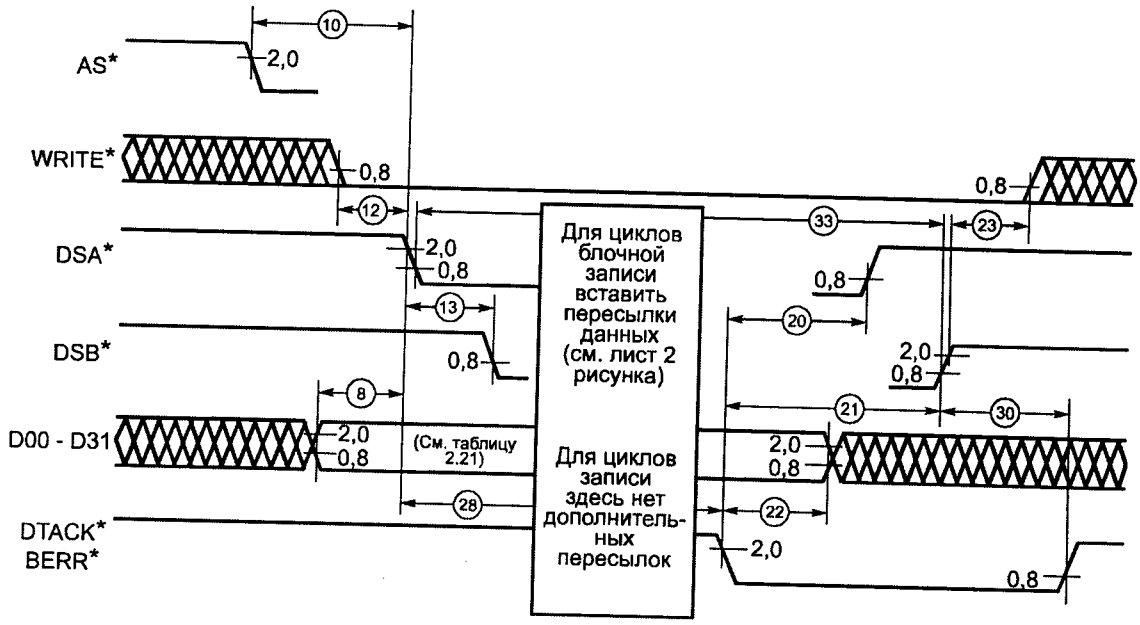


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
8	35	—	10	—	—	—
9	0	—	0	—	—	—
11	40	—	30	—	30	—
20	0	—	0	—	—	—
22	0	—	0	—	—	—
28	30	2*T*	30	—	—	—
30	0	—	0	—	—	—
33	—	—	30	—	30	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.18, лист 2

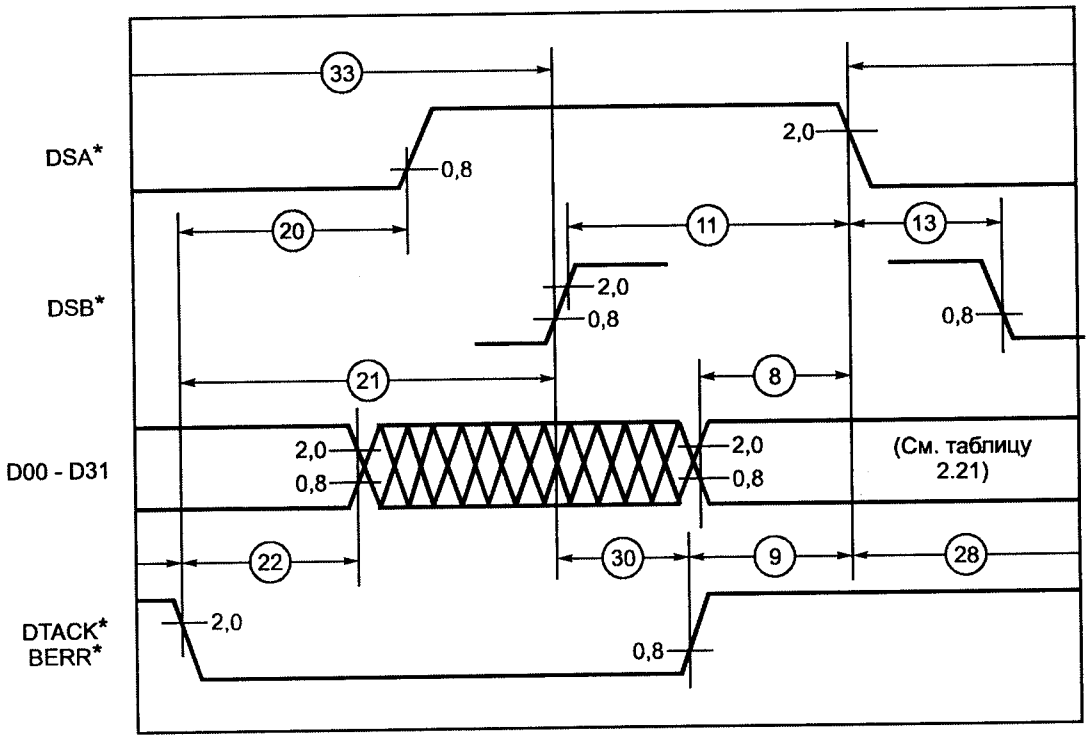


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
8	35	—	10	—	—	—
10	0	—	—10	—	—10	—
12	35	—	10	—	10	—
13	—	10	—	20	—	20
20	0	—	0	—	—	—
21	0	—	0	—	—	—
22	0	—	0	—	—	—
23	10	—	0	—	0	—
28	30	2*T*	30	—	—	—
30	0	—	0	—	—	—
33	—	—	30	—	30	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.19, лист 1 — Временна́я диаграмма пересылки данных.
ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР.
(Запись байтов (0—1); запись байтов (2—3); запись байтов (0—3); запись байтов (1—2); двухбайтовая блочная запись; четырехбайтовая блочная запись)

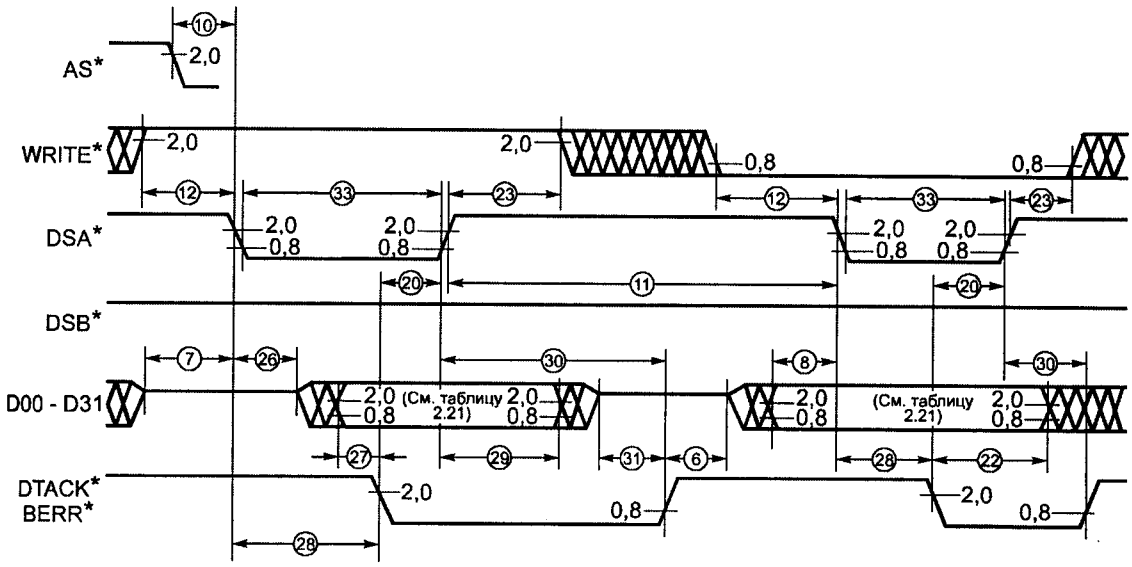


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
8	35	—	10	—	—	—
9	0	—	0	—	—	—
11	40	—	30	—	30	—
13	—	10	—	20	—	20
20	0	—	0	—	—	—
21	0	—	0	—	—	—
22	0	—	0	—	—	—
28	30	2*T*	30	—	—	—
30	0	—	0	—	—	—
33	—	—	30	—	30	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.19, лист 2

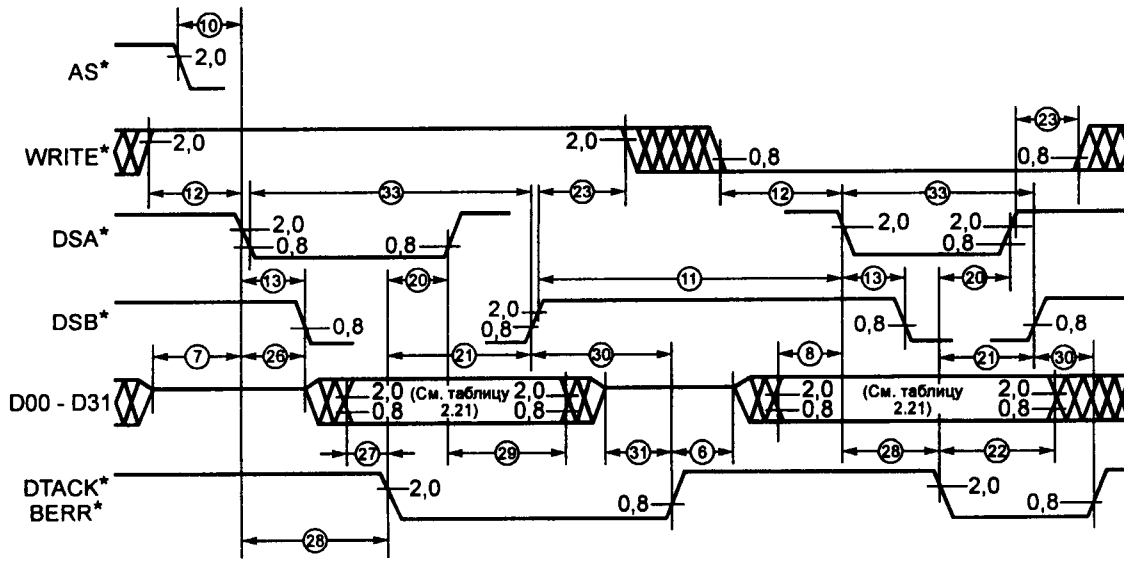


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
6	0	—	0	—	—	—
7	0	—	0	—	—	—
8	35	—	10	—	—	—
10	0	—	—10	—	—10	—
11	40	—	30	—	30	—
12	35	—	10	—	10	—
20	0	—	0	—	—	—
22	0	—	0	—	—	—
23	10	—	0	—	0	—
26	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
31	0	—	0	—	—	—
33	—	—	30	—	30	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.20 — Временна́я диаграмма пересылки данных.
ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР.
(Однобайтовый цикл Чтение—Модификация—Запись)



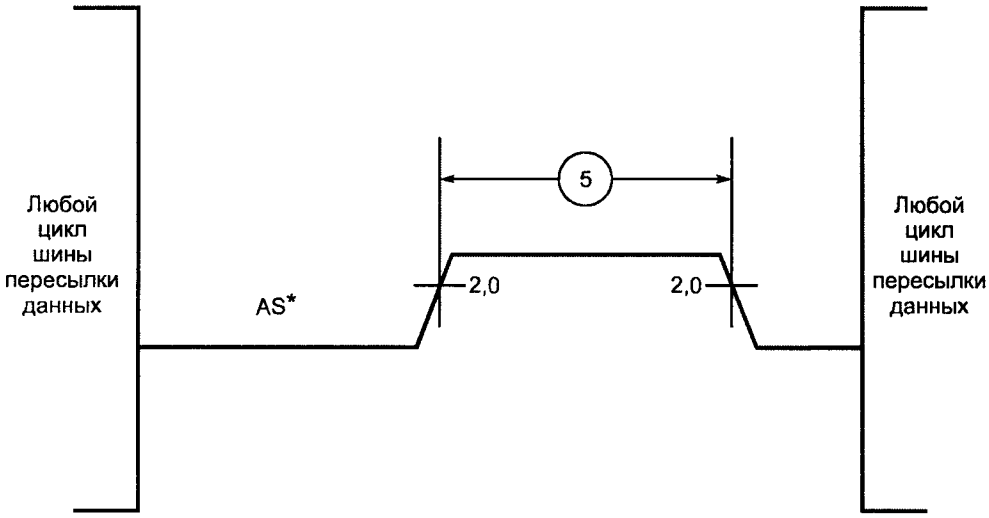
Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
6	0	—	0	—	—	—
7	0	—	0	—	—	—
8	35	—	10	—	—	—
10	0	—	—10	—	—10	—
11	40	—	30	—	30	—
12	35	—	10	—	10	—
13	—	10	—	20	—	20
20	0	—	0	—	—	—
21	0	—	0	—	—	—
22	0	—	0	—	—	—
23	10	—	0	—	0	—
26	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
31	0	—	0	—	—	—
33	—	—	30	—	30	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.21 — Временна́я диаграмма пересылки данных.
ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и АДРЕСНЫЙ МОНИТОР.
(Двухбайтовые циклы Чтение—Модификация—Запись; четырехбайтовые циклы Чтение—Модификация—Запись)

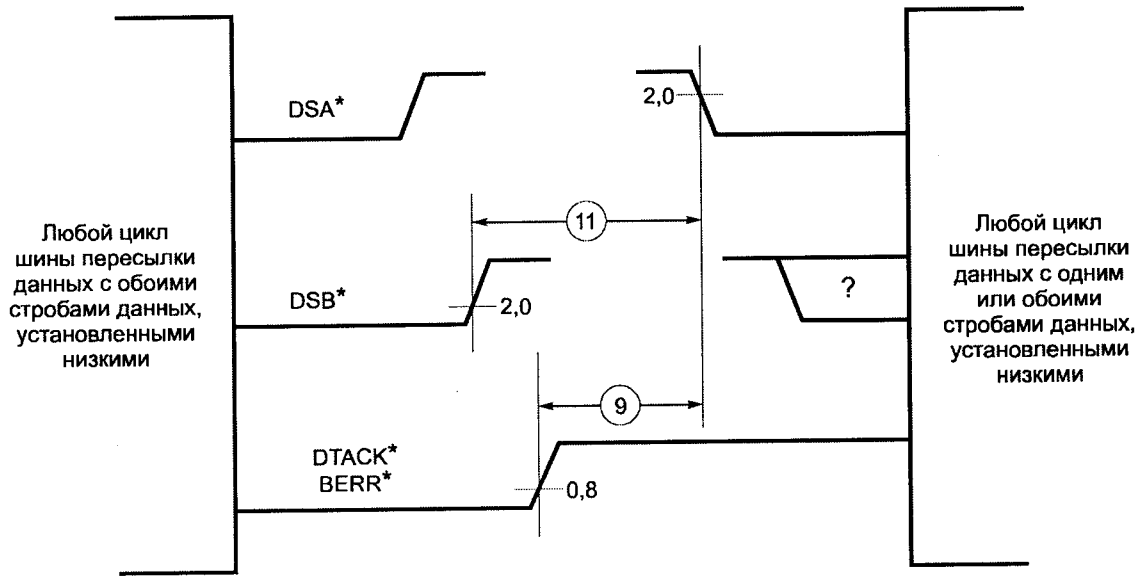
Рисунки 2.22—2.24 содержат временны́е диаграммы к правилам и замечаниям по временны́м соотношениям для задатчиков и исполнителей между циклами шины пересылки данных.



Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
5	40	—	30	—	30	—
Примечание — Все значения параметров в наносекундах.						

Рисунок 2.22 — Временна́я диаграмма адресного строба между циклами

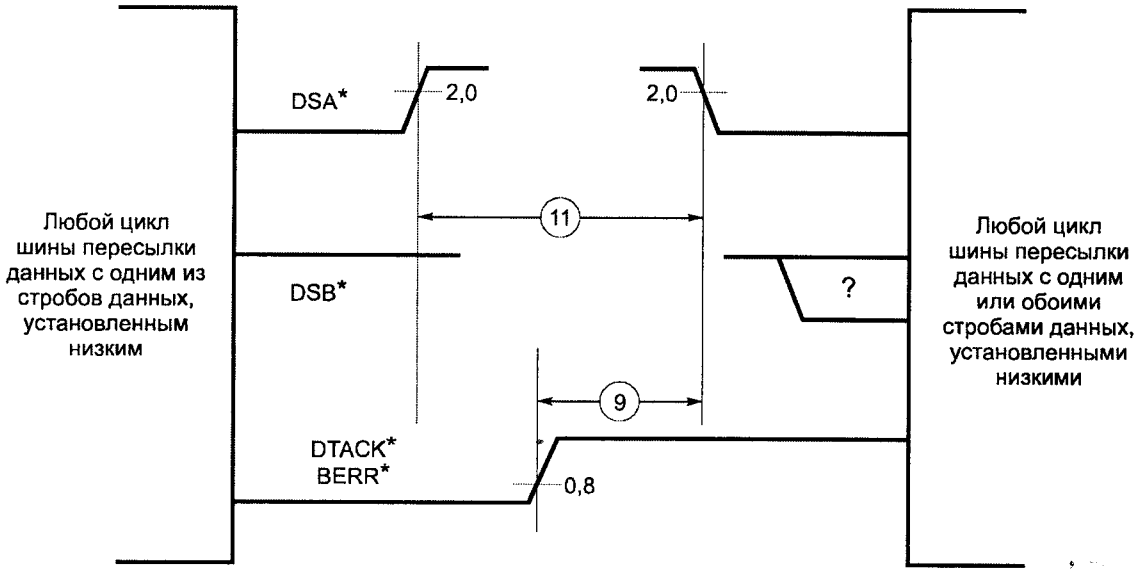


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
9	0	—	0	—	—	—
11	40	—	30	—	30	—

Примечание — Все значения параметров в наносекундах.

Рисунок 2.23 — Временна́я диаграмма стробов данных между циклами. За циклом, в котором оба строба данных устанавливаются низкими, следует цикл, в котором один или оба строба устанавливаются низкими



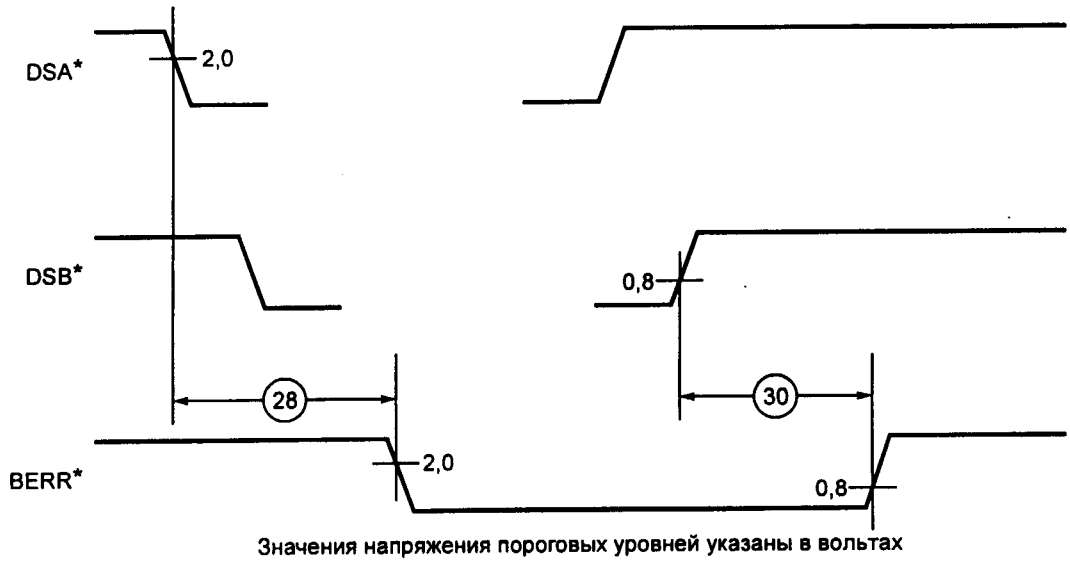
Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
9	0	—	0	—	—	—
11	40	—	30	—	30	—

Примечание — Все значения параметров в наносекундах.

Рисунок 2.24 — Временна́я диаграмма стробов данных между циклами. За циклом, в котором один из стробов данных устанавливается низким, следует цикл, в котором один или оба строба устанавливаются низкими

Рисунок 2.25 содержит временны́е диаграммы для задатчика, исполнителя и шинного таймера во время цикла тайм-аута.

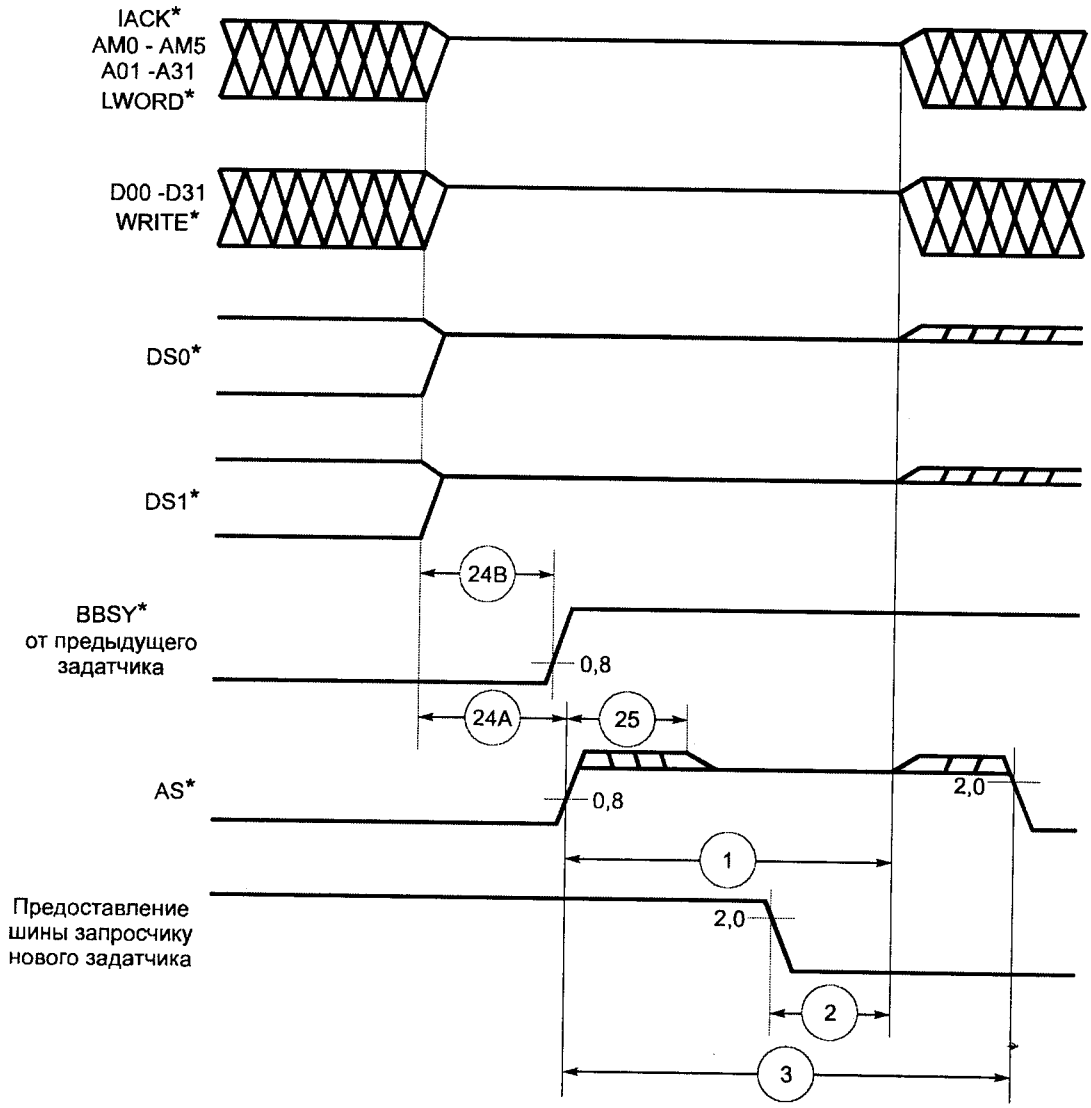


Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
28	30	2*T*	30	—	—	—
30	0	—	0	—	—	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 2.25 — Временна́я диаграмма пересылки данных.
ЗАДАТЧИК, ИСПОЛНИТЕЛЬ и ШИННЫЙ ТАЙМЕР.
(Цикл с тайм-аутом)

Рисунок 2.26 содержит временны́е диаграммы во время передачи управления шиной пересылки данных.



Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Задатчик		Исполнитель		Адресный монитор	
	мин.	макс.	мин.	макс.	мин.	макс.
1	0	—	—	—	—	—
2	0	—	—	—	—	—
3	60	—	—	—	—	—
24A	0	—	—	—	—	—
24B	0	—	—	—	—	—
25	—	25	—	—	—	—

Примечание — Все значения параметров в наносекундах.

Рисунок 2.26 — Временна́я диаграмма передачи управления шиной пересылки данных. ЗАДАТЧИК

Для выполнения требований по временны́м соотношениям разработчики модулей должны принимать в расчет значения задержек распространения сигналов между шинными формирователями и приемниками, используемыми в их модулях, для наихудших случаев. Задержки распространения формирователей зависят от их выходных нагрузок, однако в технической документации изготовителей не всегда дается достаточно информации для расчета задержек распространения при различных нагрузках. Чтобы помочь разработчику модулей, в разделе 6 содержатся некоторые предложения.

Замечания указывают временны́е соотношения для перепадов сигналов, поступающих с линий. Этим соотношениям можно доверять, если не нарушены правила нагрузки объединительной платы, указанные в разделе 6. Правила для оконечных нагрузок в разделе 6 гарантируют, что соответствующие временны́е параметры для сигнальных линий после их освобождения соблюдаются.

Обычно для каждого правила по временны́м соотношениям имеется соответствующее замечание. Однако время, гарантируемое в замечании, может отличаться от времени, определяемого правилом. Например, внимательное рассмотрение временны́х диаграмм показывает, что от задатчика требуется обеспечить время установления адреса и данных 35 нс, а исполнителю гарантируется только 10 нс. Это связано с тем, что шинные формирователи адреса и данных не всегда способны полностью обеспечить переход уровней на сигнальных линиях объединительной платы с низкого на высокий через пороговую область, пока этот переход не распространится до конца объединительной платы и обратно. С другой стороны, отрицательные перепады стробов адреса и данных обычно пересекают пороговое значение 0,8 В, не ожидая отражения. В результате время установления сигнала у исполнителя равно времени установления сигнала у задатчика, уменьшенному на удвоенное время распространения сигнала по магистрали.

Для описания временны́х соотношений стробов данных используется специальная система обозначений. Два строба данных (DS0* и DS1*) не всегда переходят с уровня на уровень одновременно. Во временны́х диаграммах обозначение DSA* употребляется для представления строба данных, первым сформировавшего свой перепад (будь то DS0* или DS1*). Обозначение DSB* употребляется для представления строба данных, сформировавшего свой перепад вторым (будь то DS0* или DS1*). Прерванная линия, изображенная на рисунках при установившихся состояниях стробов данных, показывает, что строб данных, осуществивший первым отрицательный перепад, может быть не тем, который первым сформировал свой положительный перепад. Иными словами, DSA* может представлять DS0* на своем отрицательном перепаде и DS1* — на положительном.

В тексте настоящего стандарта употребляются термины: линия DSA*, линия DSB*, строб DSA*, строб DSB* или просто DSA*, DSB*, хотя из приведенных выше пояснений очевидно, что эти линии и сигналы являются виртуальными. Употребление этих терминов оправдано тем, что они реально отражают временны́е соотношения соответствующих сигналов DS0* и DS1* при исполнении всех циклов пересылки данных и значительно упрощают формулировки текста и приведенные на рисунках временны́е диаграммы.

3 Шина арбитража

3.1 Основные принципы арбитража шины

По мере удешевления микропроцессоров становится все более экономически выгодно проектировать системы с несколькими процессорами, совместно использующими общие ресурсы.

Наиболее важным из этих общих ресурсов является шина пересылки данных, через которую осуществляется доступ ко всем другим общим ресурсам. Поэтому любая многопроцессорная система должна содержать эффективный механизм предоставления шины пересылки данных. Так как скорость предоставления шины имеет первостепенное значение, технические средства схемы предоставления шины выбираются исключительно из условия целесообразности. Магистраль VME реализует механизм предоставления шины с помощью подсистемы арбитража (см. рисунок 3.1)

Подсистема арбитража решает следующие задачи:

а) предотвращает одновременное использование шины двумя задатчиками;

б) устанавливает очередность обслуживания запросов от различных задатчиков с целью оптимального использования шины.

3.1.1. Типы арбитража

Когда несколько модулей одновременно запрашивают разрешение на использование шины пересылки данных, подсистема арбитража обнаруживает эти запросы и предоставляет шину каждый раз

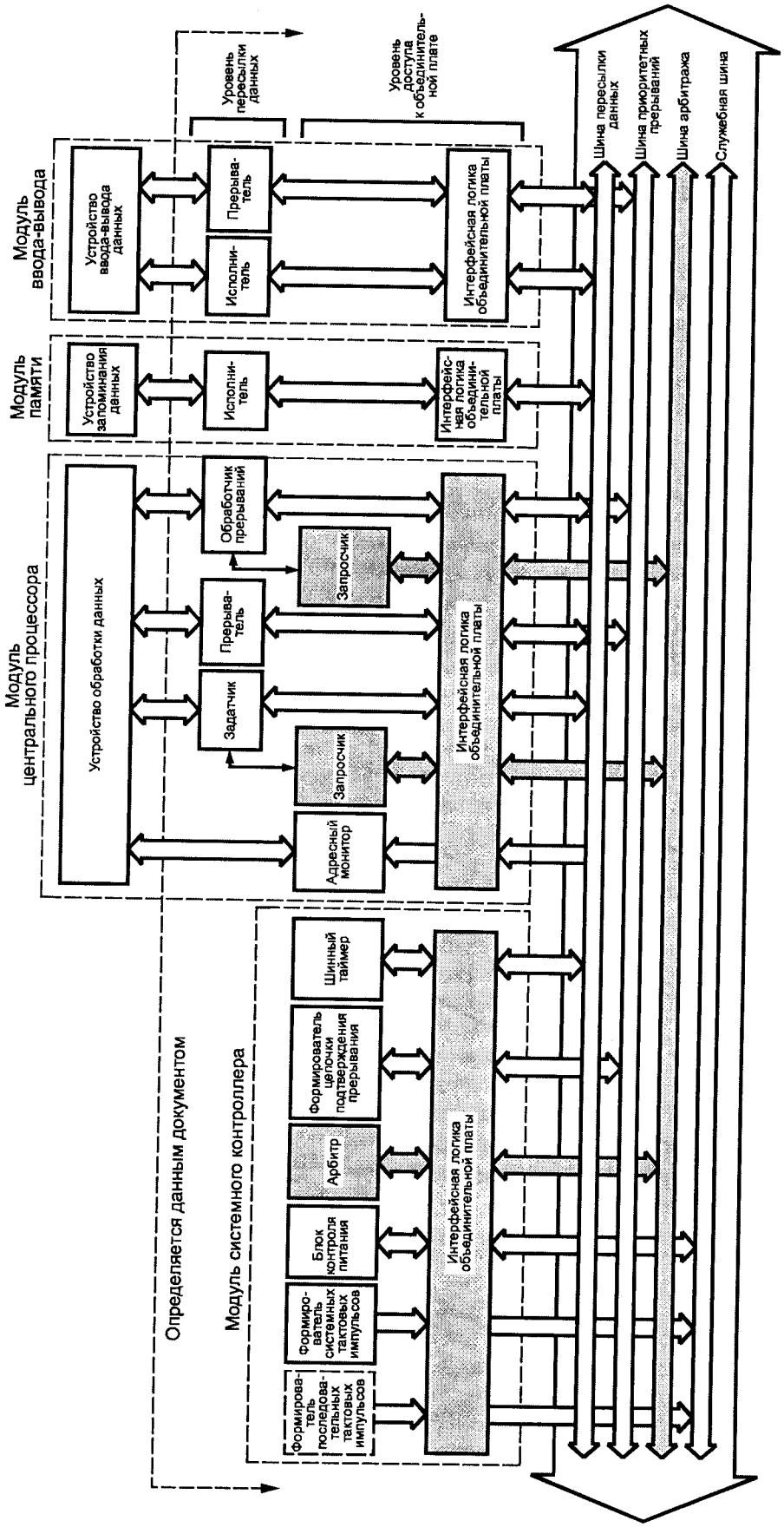


Рисунок 3.1 — Схема взаимодействия шины арбитража с функциональными блоками

только одному из модулей. В зависимости от того, какой алгоритм установления очередности обслуживания запросов используется, принимается решение о том, какому из модулей предоставить шину в первую очередь.

Несмотря на то, что существует множество различных алгоритмов, для магистрали VME определены три таких алгоритма: приоритетный, круговой и одноуровневый.

Приоритетный арбитраж предоставляет шину в соответствии с фиксированной приоритетной схемой, в которой каждой из четырех линий запроса шины (bus request line — BR) установлен приоритет от наивысшего (BR3*) до низшего (BR0*).

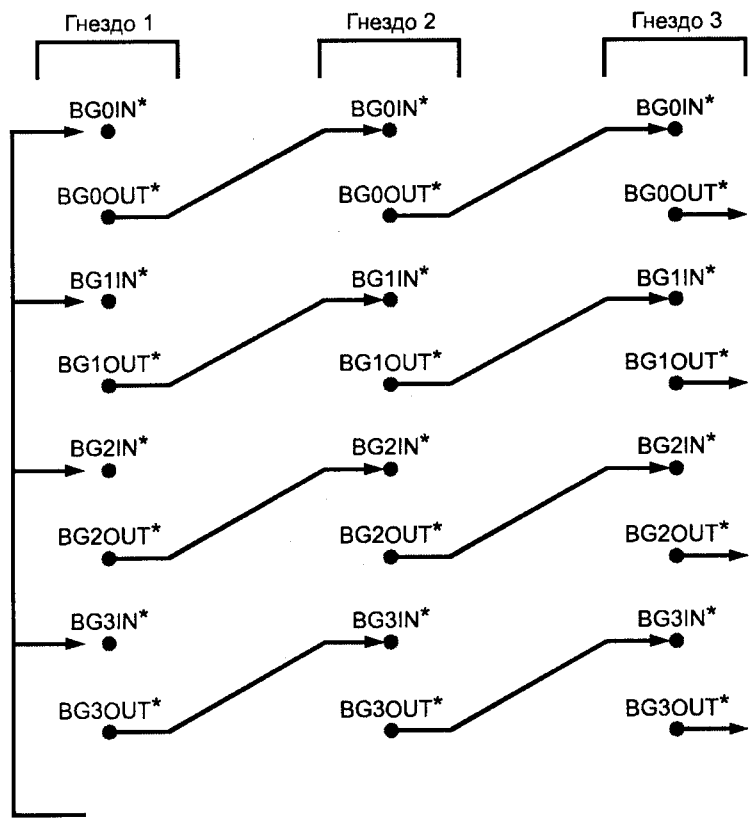
Круговой арбитраж предоставляет шину на основе принципа циклического приоритета. Если шина предоставлена запросчику по линии «BR(n)*» запроса шины, то наивысший приоритет для следующего арбитража назначается для линии «BR(n — 1)*» запроса шины.

Одноуровневый арбитраж воспринимает запросы только по линии BR3* и для арбитража этих запросов использует соответствующую BR3* последовательную цепочку предоставления шины.

Разрешение 3.1. Кроме приоритетного, кругового или одноуровневого алгоритмов арбитража, МОГУТ использоваться другие алгоритмы установления очередности обслуживания запросов. Например, может быть выбран алгоритм работы арбитража, присваивающий наивысший приоритет линии BR3*, но предоставляющий шину по запросам BR0*—BR2* на основе кругового арбитража.

3.2 Линии шины арбитража

Шина арбитража состоит из шести магистральных линий и четырех последовательных цепочек. Для этих последовательных цепочек требуются специальные наименования сигналов. Сигналы, поступающие на каждый модуль, называются входными сигналами предоставления шины (Bus Grant IN, сокращенно BGxIN*), а сигналы, выходящие с модулей, — выходными сигналами предоставления шины (Bus Grant OUT — BGxOUT*). Линии, которые выходят из гнезда n как BGxOUT*, поступают на гнездо $n + 1$ как BGxIN*. Это показано на рисунке 3.2.



Примечание — Входные линии предоставления шины гнезда 1 возбуждаются арбитром, который обычно размещается в модуле, вставляемом в гнездо 1.

Рисунок 3.2 — Иллюстрация последовательных цепочек предоставления шины

Замечание 3.1. В тексте этого раздела мнемонические обозначения BRx*, BGxIN* и BGxOUT* употребляются для описания сигналов линий запроса и предоставления шины, где x принимает какое-либо значение от ноля до трех.

В подсистеме арбитража магистрали VME функциональный блок запросчика возбуждает следующие линии:

одну линию запроса шины (из BR0*—BR3*);

одну линию выходного сигнала предоставления шины (из BG0OUT*—BG3OUT*);

линию сигнала занятости шины (BBSY*).

Правило 3.1. *Если модуль не формирует запросы шины на каких-либо уровнях запроса шины, то он ДОЛЖЕН ПЕРЕДАТЬ сигналы цепочек для этих уровней со своих входных линий BGxIN* на свои выходные линии BGxOUT*.*

Разрешение 3.2. Распространение сигналов по неиспользуемым цепочкам предоставления шины МОЖЕТ быть обеспечено с помощью перемычек или логических схем. Последний метод позволяет выбирать уровень запроса программным способом, в то время как первый приводит к более быстрому распространению сигнала по цепочке.

Данным стандартом определены три типа арбитров:

приоритетный (PRIoritized — PRI);

круговой (Round—Robin Select — RRS);

одноуровневый (SinGle Level — SGL).

Работа этих трех типов арбитров описана в 3.3.

Арбитр PRI возбуждает следующие линии:

линию очистки шины (Bus CLeaR line — BCLR*),

четыре линии предоставления шины (BG0IN*—BG3IN*) гнезда 1.

Арбитр RRS возбуждает четыре линии BGxIN гнезда 1 и дополнительно может возбуждать линию BCLR*.

Арбитр SGL возбуждает только линию BG3IN* гнезда 1.

Подразумевается, что с подсистемой арбитража во время процессов включения и выключения питания связаны также две дополнительные линии SYSRESET* и ACFAIL*. Хотя их влияние на подсистему арбитража и рассматривается в данном разделе, более подробно эти линии рассмотрены в разделе 5.

3.2.1 Линии запроса и предоставления шины

Линии запроса шины используются каждым запросчиком для запроса на использование шины пересылки данных. Линии предоставления шины позволяют арбитру разрешить ее использование. Это разрешение дается установкой в низкое состояние сигнала в цепочке предоставления шины. Этот отрицательный перепад распространяется по цепочке, обычно проходя при этом через несколько модулей. Если модуль никогда не использует данный уровень запроса/предоставления шины, сигнал проходит через модуль дальше. Если модуль использует какой-либо уровень "x" запроса/предоставления шины, дальнейшее распространение соответствующего сигнала BGxIN* определяется модулем. Если запросчик модуля в текущий момент запрашивает разрешение на использование шины пересылки данных на этом уровне, модуль не пропустит отрицательный перепад на свой выход BGxOUT*. В противном случае, отрицательный перепад передается на выход BGxOUT*.

Правило 3.2. *Если гнездо объединительной платы не занято модулем, а далее по цепочке имеются другие модули, то для пропускания сигнала по цепочке в пустое гнездо ДОЛЖНЫ УСТАНАВЛИВАТЬСЯ перемычки.*

Замечание 3.2. Требования к конструкции объединительной платы в разделе 7 содержат описание средств, обеспечивающих установку перемычек в каждом гнезде.

Правило 3.3. *Арбитр ДОЛЖЕН РАСПОЛАГАТЬСЯ в гнезде 1.*

3.2.2 Линия сигнала занятости шины (BBSY*)

Когда запросчику предоставлено право управления шиной пересылки данных через цепочку предоставления шины, он устанавливает BBSY* низким. После этого он управляет шиной пересылки данных до тех пор, пока не освободит BBSY*. Освобождение BBSY* позволяет арбитру предоставить шину пересылки данных какому-либо другому запросчику.

3.2.3 Линия очистки шины (BCLR*)

Арбитр PRI устанавливает BCLR* низким, чтобы информировать задатчик, который в текущий момент управляет шиной пересылки данных, о наличии запроса шины более высокого приоритета.

От этого задатчика не требуется уступить шину в пределах заранее предписанного времени. Он может продолжить пересылку данных до подходящего момента, когда возможна остановка, а затем разрешает своему запросчику освободить BBSY*.

Разрешение 3.3. Хотя арбитру RRS не требуется возбуждать линию BCLR*, он МОЖЕТ делать это.

Предложение 3.1. Если арбитр RRS устанавливает BCLR* низким, предлагается, чтобы он делал это всякий раз, когда появляется запрос по любой из линий запроса шины, кроме линии, по которой состоялся запрос на текущее предоставление шины.

3.3 Функциональные блоки

Подсистема арбитража состоит из нескольких функциональных блоков:

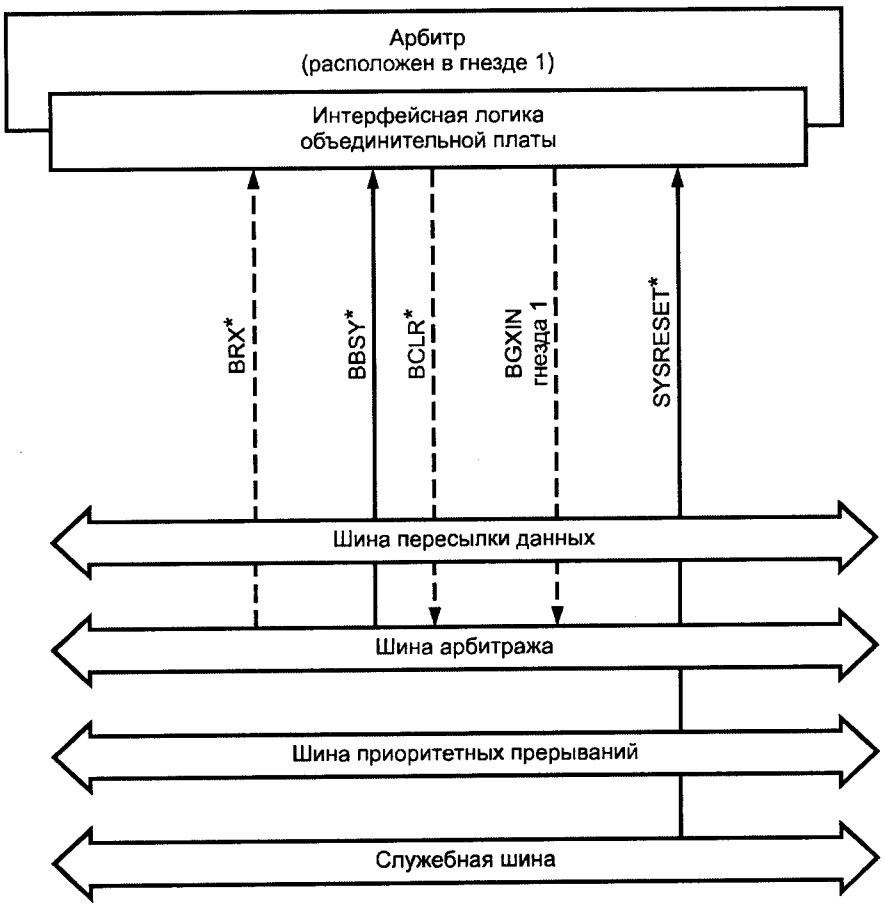
- одного арбитра;
- одного или нескольких запросчиков.

На рисунках 3.3 и 3.4 приведены схемы взаимодействия этих двух типов функциональных блоков с магистралью VME.

Правило 3.4. *Линии выходных сигналов, показанные на рисунках 3.3 и 3.4 сплошными линиями, ДОЛЖНЫ ВОЗБУЖДАТЬСЯ соответствующим функциональным блоком, если он не устанавливает на них высокий уровень постоянно.*

Правило 3.5. *Линии входных сигналов, показанные на рисунках 3.3 и 3.4 сплошными линиями, ДОЛЖНЫ КОНТРОЛИРОВАТЬСЯ и на их сигналы ДОЛЖНЫ ДАВАТЬСЯ соответствующие ответы.*

Замечание 3.3. Правила и разрешения по возбуждению и контролю сигнальных линий, изображенных на рисунках 3.3 и 3.4 пунктиром, приведены в таблицах 3.1, 3.2.

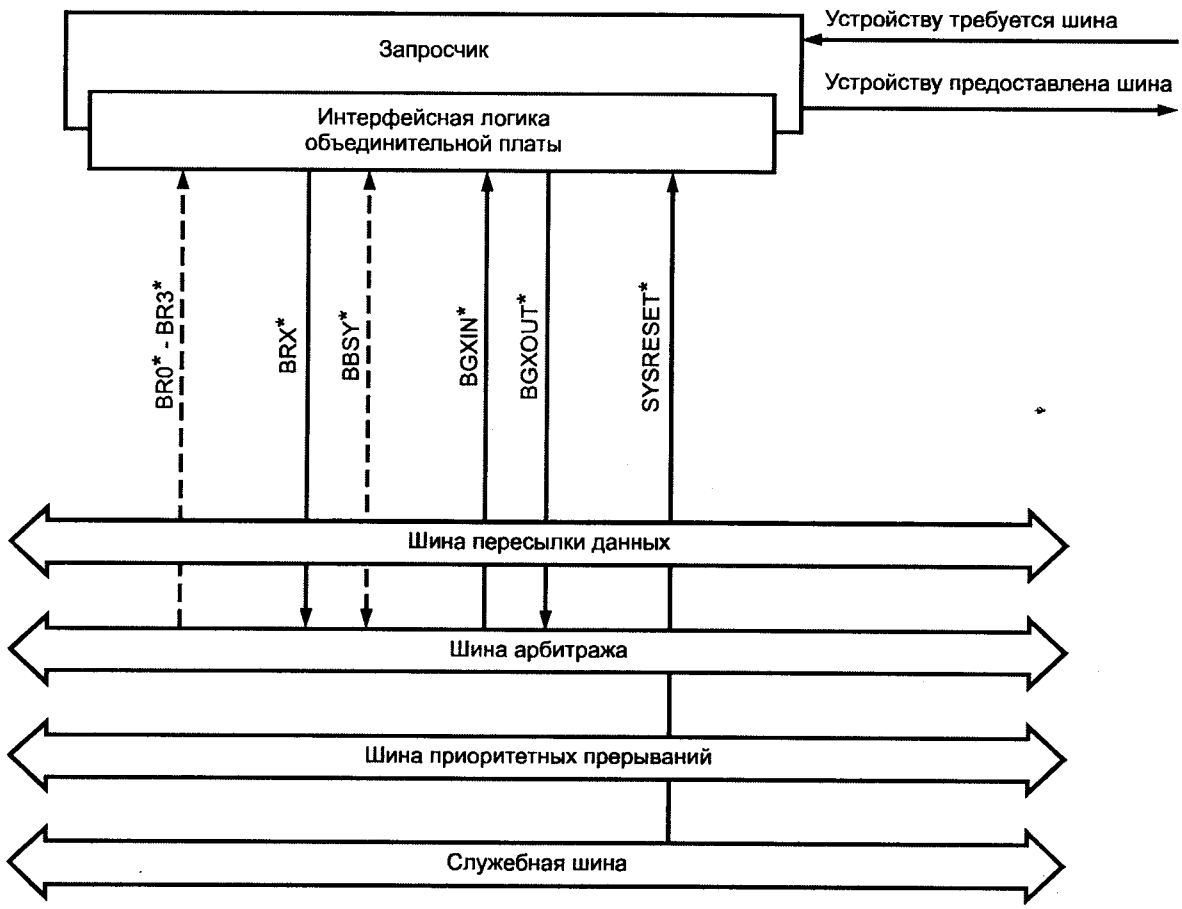


Примечание — Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных пунктиром, приведены в таблице 3.1.

Рисунок 3.3 — Схема взаимодействия арбитра с магистралью VME

Таблица 3.1 — Арбитры. Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных на рисунке 3.3 пунктиром

Тип арбитра	Правила и разрешения
SGL	*ДОЛЖЕН ВОЗБУЖДАТЬ линию BG3IN* гнезда 1* *ДОЛЖЕН ГАРАНТИРОВАТЬ высокий уровень сигнала на линиях BG0IN*—BG2IN* гнезда 1* *ДОЛЖЕН КОНТРОЛИРОВАТЬ линию BR3** МОЖЕТ возбуждать или МОЖЕТ не возбуждать линию BCLR* или линии BG0IN*—BG2IN* гнезда 1 МОЖЕТ контролировать или МОЖЕТ не контролировать линии BR0*—BR2*
RRS	*ДОЛЖЕН ВОЗБУЖДАТЬ линии BG0IN*—BG3IN* гнезда 1* *ДОЛЖЕН КОНТРОЛИРОВАТЬ линии BR0*—BR3** МОЖЕТ возбуждать или МОЖЕТ не возбуждать линию BCLR*
PRI	*ДОЛЖЕН ВОЗБУЖДАТЬ линии BG0IN*—BG3IN* гнезда 1 и BCLR** *ДОЛЖЕН КОНТРОЛИРОВАТЬ линии BR0*—BR3**



Примечание — Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных пунктиром, приведены в таблице 3.2.

Рисунок 3.4 — Схема взаимодействия запросчика с магистралью VME

Таблица 3.2 — Запросчики. Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных на рисунке 3.4 пунктиром

Тип запросчика	Правила и разрешения
RWD	МОЖЕТ контролировать или МОЖЕТ не контролировать линии BRO*—BR3* МОЖЕТ контролировать или МОЖЕТ не контролировать линию BBSY*
ROR	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линии BR0*—BR3** МОЖЕТ контролировать или МОЖЕТ не контролировать линию BBSY*
FAIR	*ДОЛЖЕН КОНТРОЛИРОВАТЬ ту линию запроса шины, которую использует* *ДОЛЖЕН КОНТРОЛИРОВАТЬ линию BBSY**

Замечание 3.4. Если линия выходного сигнала не возбуждается, оконечные нагрузки объединительной платы гарантируют установление на ней напряжения высокого уровня.

Замечание 3.5. Хотя линии сигналов SYSRESET* и ACFAIL* не определены как часть шины арбитража, в данном случае они важны, поскольку задатчики, работающие с запросчиком, отвечают на сигналы этих линий (эти сигнальные линии возбуждаются блоком контроля питания, который рассмотрен в разделе 5).

3.3.1 Арбитр

Арбитр — это функциональный блок, который принимает решение о том, какому из запросчиков предоставить управление шиной пересылки данных, если несколько из них запрашивают шину одновременно. Существует множество возможных алгоритмов, которые могли бы использоваться для принятия такого решения. В настоящем стандарте определены три типа арбитров: приоритетный (PRI), круговой (RRS), одноуровневый (SGL).

Арбитр отвечает на поступающие запросы шины и предоставляет шину пересылки данных соответствующему запросчику, используя одну из линий предоставления шины.

Если арбитр обнаруживает на линии BBSY* высокий уровень, а затем один или несколько запросов шины, он предоставляет шину в соответствии с запросом, имеющим наивысший приоритет.

Сигналы на линиях запроса шины могут быть в состоянии перехода из высокого состояния в низкое в тот самый момент, когда арбитр фиксирует их состояние. Если запоминание состояния сигнала на линии производится во время такого перехода, то выходной сигнал устройства, запоминающего это состояние, может быть нестабильным в течение некоторого времени. Это явление иногда определяют как метастабильность. В приложении D приведена примерная схема арбитра, в которой влияние этого явления устранено.

Когда запросчик принимает сигнал предоставления шины, он устанавливает BBSY* низким и сообщает своему внутримодульному задатчику или обработчику прерываний о том, что ему предоставлена шина пересылки данных. После того как задатчик или обработчик прерываний закончит использование шины пересылки данных, этот запросчик освобождает линию BBSY*. Получившийся в результате положительный перепад сигнала BBSY* разрешает арбитру сформировать другой сигнал предоставления шины на основе анализа состояния уровней сигналов линий запросов шины в данный момент.

Помимо арбитража, обеспечиваемого арбитром, предусмотрен еще вторичный уровень арбитража с помощью цепочек предоставления шины. За счет последовательности этих цепочек запросчики, использующие совместно одну общую линию запросов, распределяются по приоритетам в зависимости от номера гнезда. Запросчик, ближайший к гнезду 1, имеет наивысший приоритет.

Арбитр SGL отвечает только на запросы шины по линии BR3*, возлагая функцию выполнения арбитража на цепочку BG3IN*/BG3OUT*.

Арбитр PRI устанавливает приоритеты четырех линий запроса шины от BR0* (низший) до BR3* (высший) и отвечает соответствующим сигналом BG0IN*—BG3IN*. Кроме того, при появлении запроса более высокого уровня, арбитр PRI сообщает об этом задатчику, управляющему шиной в данный момент, установкой BCLR* низким.

Для наглядности работы арбитра RRS рассмотрим механический переключатель с приводом от шагового двигателя. В каждом положении переключатель соединяет линию запроса шины с соответствующей линией предоставления шины.

Если шина занята, переключатель останавливается на текущем уровне. При освобождении шины переключатель переходит на одну позицию ниже (т. е. с BR(n)* на BR(n—1)*) и проверяет наличие запроса. Он продолжает этот процесс сканирования, пока не обнаружит запрос, после чего посылает сигнал предоставления шины по соответствующей линии.

Разрешение 3.4. Арбитр МОЖЕТ быть спроектирован со встроенным механизмом тайм-аута, который заставляет его переназначать предоставление шины, если BBSY* не устанавливается запросчиком низким в пределах заранее заданного времени.

Замечание 3.6. Тайм-аут, используемый арбитром в соответствии с разрешением 3.4, должен быть больше суммы максимального времени задержки распространения сигнала по цепочке предоставления шины и времени, которое требуется самому медленному запросчику для формирования сигнала BBSY*.

Предложение 3.4. Следует указывать максимальное время задержки распространения от BGxIN* до BGxOUT* в технической документации на изделия. Также следует указывать максимальное время, которое потребуется запросчику, чтобы сформировать сигнал BBSY*. Это позволит пользователям определить требуемое значение тайм-аута процесса арбитража.

Правило 3.6. *За исключением ситуации тайм-аута, когда отсутствуют ответы от запросчиков, арбитр, предоставив шину какому-либо запросчику, НЕ ДОЛЖЕН ФОРМИРОВАТЬ новый сигнал предоставления шины, пока этот запросчик не сгенерирует положительный перепад BBSY* (запросчик генерирует положительный перепад, устанавливая сигнал на линии BBSY* низким, а затем освобождая ее).*

Замечание 3.7. Если арбитр фиксирует состояния сигналов на линиях запроса шины до положительного перепада BBSY*, он может предоставить шину запросчику, который уже снял свой запрос.

3.3.2 Запросчик

Каждый запросчик в системе выполняет следующие функции:

- контролирует сигнал «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА» своего внутримодульного задатчика или обработчика прерываний и формирует сигнал запроса шины, если требуется шина пересылки данных;
- если он обнаруживает низкий уровень сигнала на своей линии BGxIN* и его внутримодульному задатчику или обработчику прерываний шина пересылки данных не требуется, он пропускает этот низкий уровень на свою линию BGxOUT*;
- если он обнаруживает низкий уровень сигнала на своей линии BGxIN* и его внутримодульному задатчику или обработчику прерываний требуется шина пересылки данных, он генерирует внутримодульный сигнал «УСТРОЙСТВУ ПРЕДОСТАВЛЕНА ШИНА», чтобы показать, что шина доступна, и устанавливает сигнал BBSY* низким.

В настоящем стандарте определены три типа запросчиков:

запросчик, освобождающий шину после выполнения пересылки (Release When Done Requester — RWD);

запросчик, освобождающий шину по запросу (Release On Request Requester — ROR);

паритетный запросчик (FAIR Requester — FAIR).

Запросчик RWD освобождает линию BBSY*, когда его задатчик или обработчик прерываний устанавливает ложным внутримодульный сигнал «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА».

Запросчик ROR не освобождает линию BBSY*, когда его внутримодульный сигнал «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА» становится ложным, пока какой-либо другой запросчик на шине не установит низким сигнал на одной из линий запроса шины. Запросчик ROR контролирует четыре линии запроса шины и освобождает линию BBSY*, если только появляется другой запрос шины. Запросчики ROR уменьшают количество операций арбитража, инициируемых тем задатчиком, который выполняет большую часть от всех пересылок по шине.

Пока система имеет не более четырех задатчиков или обработчиков прерываний (один на каждую линию запроса), алгоритм арбитража RRS обеспечивает паритетность (т.е., запросы более высокого уровня не будут препятствовать ни одному задатчику обращаться к шине неопределенно долго). В системах, имеющих более четырех задатчиков или обработчиков прерываний, паритетность может быть обеспечена запросчиками FAIR. После того, как запросчику FAIR была предоставлена шина, он воздерживается от генерирования новых запросов шины до тех пор, пока имеются любые активные запросы шины, ожидающие обслуживания на его уровне запроса.

Замечание 3.17. Для обеспечения паритетности при конфигурировании систем с более чем четырьмя задатчиками или обработчиками прерываний, все запросчики в системе должны быть типа FAIR. Если один или несколько запросчиков будут другого типа, паритетность все еще может быть

обеспечена, если общая суммарная интенсивность пересылок по шине, инициируемая запросчиками, отличными от FAIR, не превышает возможностей шины.

Предположим, что входной сигнал запросчика «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА» истинен и что этот запросчик установил низким сигнал на своей линии BRx*. Если запросчик получает сигнал предоставления шины, он выполняет три следующих действия:

- устанавливает BBSY* низким;
- освобождает свою линию BRx*, переводя сигнал на ней в высокое состояние;
- устанавливает истинным внутримодульный сигнал «УСТРОЙСТВУ ПРЕДОСТАВЛЕНА ШИНА», разрешая своему задатчику или обработчику прерываний инициировать пересылки по шине.

Указанные события могут произойти в любом порядке. Возможно даже, что задатчик или обработчик прерываний не воспользуется шиной в ответ на данное конкретное предоставление. В любом случае действуют следующие правила:

Правило 3.7. *В ответ на отрицательный перепад сигнала на линии BGxIN* запросчик ДОЛЖЕН УСТАНАВЛИВАТЬ сигнал BBSY* низким и УДЕРЖИВАТЬ его в этом состоянии не менее 90 нс.*

Правило 3.8. *В ответ на отрицательный перепад сигнала на линии BGxIN* запросчик ДОЛЖЕН ОСВОБОДИТЬ линию BRx*, переводя сигнал на ней в высокое состояние.*

Правило 3.9. *В ответ на отрицательный перепад сигнала на линии BGxIN* запросчик ДОЛЖЕН УДЕРЖИВАТЬ BBSY* низким не менее 30 нс после освобождения им линии BRx*.*

Замечание 3.8. Задержка 30 нс между положительным перепадом сигнала на линии BRx* и положительным перепадом сигнала на линии BBSY* гарантирует, что арбитр не будет ошибочно интерпретировать старый запрос шины как новый и формировать другой сигнал предоставления шины.

Правило 3.10. *Запросчик ДОЛЖЕН УДЕРЖИВАТЬ сигнал BBSY* низким, пока его входной сигнал BGxIN* не установится высоким.*

Замечание 3.9. Правило 3.10 гарантирует, что переход сигнала BBSY* в низкое состояние будет обнаружен арбитром и что сигнал во всех звеньях цепочки предоставления шины возвратится в высокое состояние, готовясь к следующему арбитражу.

Разрешение 3.5. Если запросчик имеет нерассмотренный запрос шины и если он обнаруживает, что какой-либо другой запросчик устанавливает низким BBSY*, то он МОЖЕТ снять свой запрос, освобождая свою линию BRx* переводом сигнала на ней в высокое состояние.

Правило 3.11. *Если запросчик снимает запрос, не получив предварительно сигнала предоставления шины, то, прежде чем сделать это, он ДОЛЖЕН ОЖИДАТЬ, пока BBSY* не установится низким, и ДОЛЖЕН СДЕЛАТЬ это не более чем через 50 нс после установления BBSY* низким.*

Предложение 3.2. Предлагается проектировать запросчики таким образом, чтобы они пропускали сигнал по цепочке предоставления шины как можно быстрее после получения сигнала предоставления шины. Это улучшит эксплуатационные характеристики системы.

Правило 3.14. *Прежде чем сформировать запрос шины, запросчик FAIR ДОЛЖЕН ОПРЕДЕЛИТЬ состояние своей линии запроса шины, когда обнаружит BBSY* высоким.*

3.3.3 Задатчик шины пересылки данных

3.3.3.1 *Освобождение шины пересылки данных*

Протокол арбитража определяет, как и когда шина пересылки данных предоставляется в системе различным задатчикам и обработчикам прерываний. Однако этот протокол не определяет, когда задатчики и обработчики прерываний освобождают шину.

Задатчики и обработчики прерываний при принятии решения об освобождении шины пересылки данных руководствуются несколькими критериями. Обработчики прерываний освобождают шину после выполнения своего цикла подтверждения прерывания, а задатчики — после завершения пересылок данных.

Некоторые задатчики контролируют сигналы ACFAIL* и BCLR*, которые информируют их о том, что шина пересылки данных необходима для выполнения операций более высокого приоритета. В случае с сигналом BCLR*, время, которое потребуется задатчику для освобождения шины, определяется его конструкцией. Например, задатчик модуля дискового контроллера может оказаться не в состоянии освободить шину во время пересылки сектора диска без потери данных, поэтому он может удерживать шину до завершения пересылки этого сектора. Сигнал ACFAIL* информирует задатчик о том, что обнаружено прекращение подачи питания сети переменного тока и, независимо от того, с какими проблемами столкнется задатчик в связи с необходимостью уступить шину, в этом случае все они несущественны по сравнению с общесистемными потребностями.

Рекомендация 3.1. Рекомендуется проектировать задатчики так, чтобы они освобождали шину пересылки данных не более чем через 200 мкс после установки ACFAIL* низким, за исключением случаев, предусматривающих их участие в действиях, связанных с отказом питающей сети.

Замечание 3.10. Определенный в рекомендации 3.1 интервал 200 мкс предназначен для обеспечения упорядоченной остановки системы.

Независимо от того, какой критерий используется при принятии решения о моменте освобождения шины пересылки данных, арбитраж выполняется прежде, чем какой-либо другой задатчик или обработчик прерываний начнет использовать эту шину. Этот арбитраж происходит либо во время последней пересылки данных, либо после нее, в зависимости от того, когда задатчик или обработчик прерываний уведомит свой внутримодульный запросчик.

Разрешение 3.6. Задатчики и обработчики прерываний МОГУТ освобождать шину пересылки данных либо во время своей последней пересылки, либо после нее.

Например, если задатчик уведомил свой внутримодульный запросчик о том, что шина ему больше не нужна во время своей последней пересылки, то запросчик освобождает линию BBSY* и арбитраж происходит во время последней пересылки. Однако, если задатчик ожидает завершения последней пересылки прежде, чем уведомить об этом свой внутримодульный запросчик, то шина пересылки данных будет бездействовать во время арбитража (см. 2.5). Правила, касающиеся освобождения шины пересылки данных, приведены в разделах 2 и 4.

Предложение 3.3. Предлагается проектировать задатчики, выполняющие блочные пересылки, таким образом, чтобы они уведомляли свои запросчики об освобождении линии BBSY* во время последней пересылки цикла блочной пересылки. Если линия BBSY* будет освобождена в начале цикла блочной пересылки, то высокоприоритетные запросы шины, инициируемые во время блочной пересылки, могут оказаться не принятыми во внимание арбитром до следующего цикла арбитража.

3.3.3.2 *Получение права на использование шины пересылки данных*

Чтобы гарантировать невозможность попыток установления противоположных состояний ни на одной из линий шины пересылки данных двумя задатчиками или обработчиками прерываний, эти функциональные блоки при получении права управления шиной пересылки данных подчиняются определенным правилам.

Правило 3.12. *Если задатчик или обработчик прерываний получает управление шиной пересылки данных от своего внутримодульного запросчика, он НЕ ДОЛЖЕН ВКЛЮЧАТЬ свои шинные формирователи сигналов шины пересылки данных, пока не обнаружит AS* высоким.*

Замечание 3.11. Если предшествующий задатчик или обработчик прерываний освобождает шину во время своей последней пересылки данных, то правило 3.12 гарантирует, что эта пересылка данных будет закончена прежде, чем новый задатчик или обработчик прерываний начнет использовать шину пересылки данных (если предшествующий задатчик или обработчик прерываний ожидает окончания пересылки прежде, чем освободить шину, сигнал AS* будет уже высоким).

3.3.3.3 *Дополнительная информация*

Рекомендация 3.2. Для обеспечения быстрого обслуживания запросов прерывания и оптимального использования шины пересылки данных рекомендуется проектировать задатчики, которые освобождают шину при первой возможности, как только обнаружат BCLR* низким.

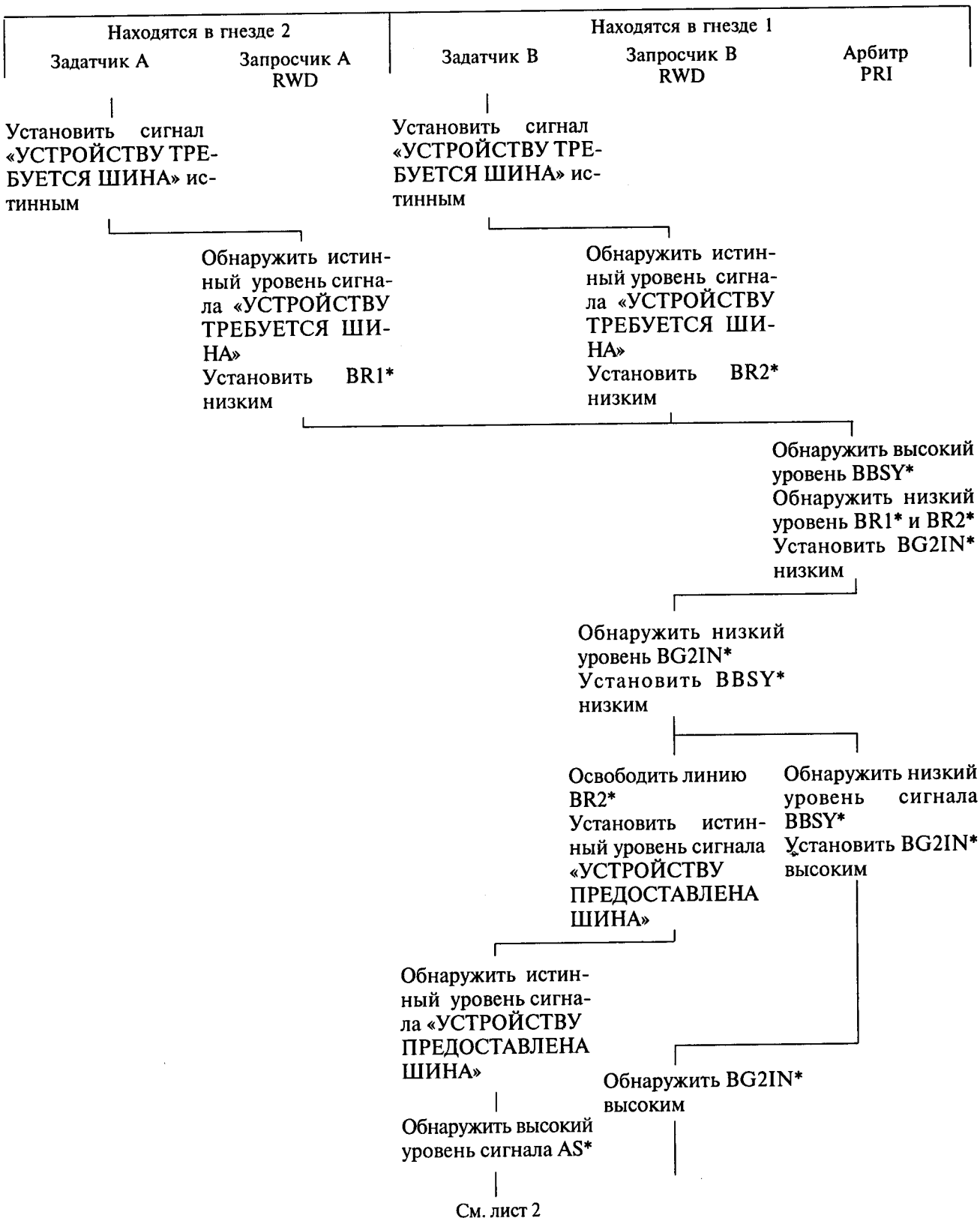
Разрешение 3.7. Задатчик или обработчик прерываний МОЖЕТ иметь более одного запросчика, причем каждый запросчик в этом случае формирует запрос шины на своей линии запроса.

Замечание 3.12. Если задатчик или обработчик прерываний имеет два или больше запросчиков, он может осуществлять высокоприоритетные пересылки данных, используя один запросчик, и низкоприоритетные пересылки, используя другой.

3.4 Типичные примеры работы

3.4.1 Арбитраж запросов шины двух различных уровней

На рисунках 3.5 и 3.6 показана последовательность событий, которые происходят, если два запросчика одновременно посылают запросы арбитру PRI по различным линиям запроса шины. В начале этой последовательности запросчик А устанавливает низким BR1*, а запросчик В устанавливает низким BR2*. Арбитр одновременно обнаруживает низкими BR1* и BR2* и устанавливает низким сигнал BG2IN* в своем гнезде (гнездо 1). Этот сигнал BG2IN* контролируется запросчиком В, также находящимся в гнезде 1. Когда запросчик В обнаруживает низкий BG2IN*, он отвечает установкой BBSY* низким. После этого зпросчик В освобождает линию BR2* и информирует свой задатчик (задатчик В), что шина пересылки данных находится в его распоряжении.



Примечание — «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА» и «УСТРОЙСТВУ ПРЕДОСТАВЛЕНА ШИНА» являются внутримодульными сигналами взаимодействия между задатчиком и его запросчиком (см. рисунок 3.4).

Рисунок 3.5, лист 1 — Алгоритм арбитража. Два запросчика, два уровня запросов

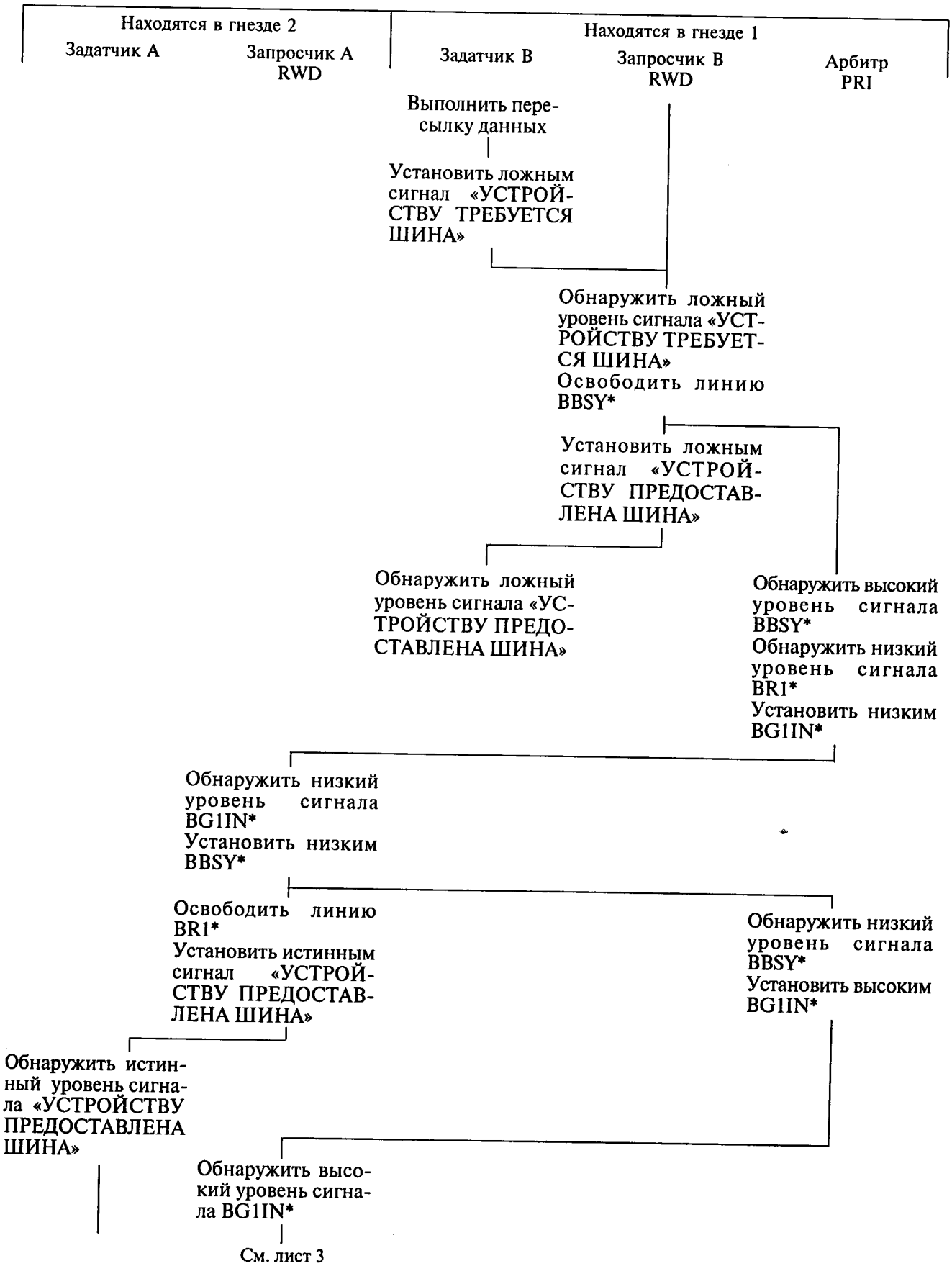


Рисунок 3.5, лист 2

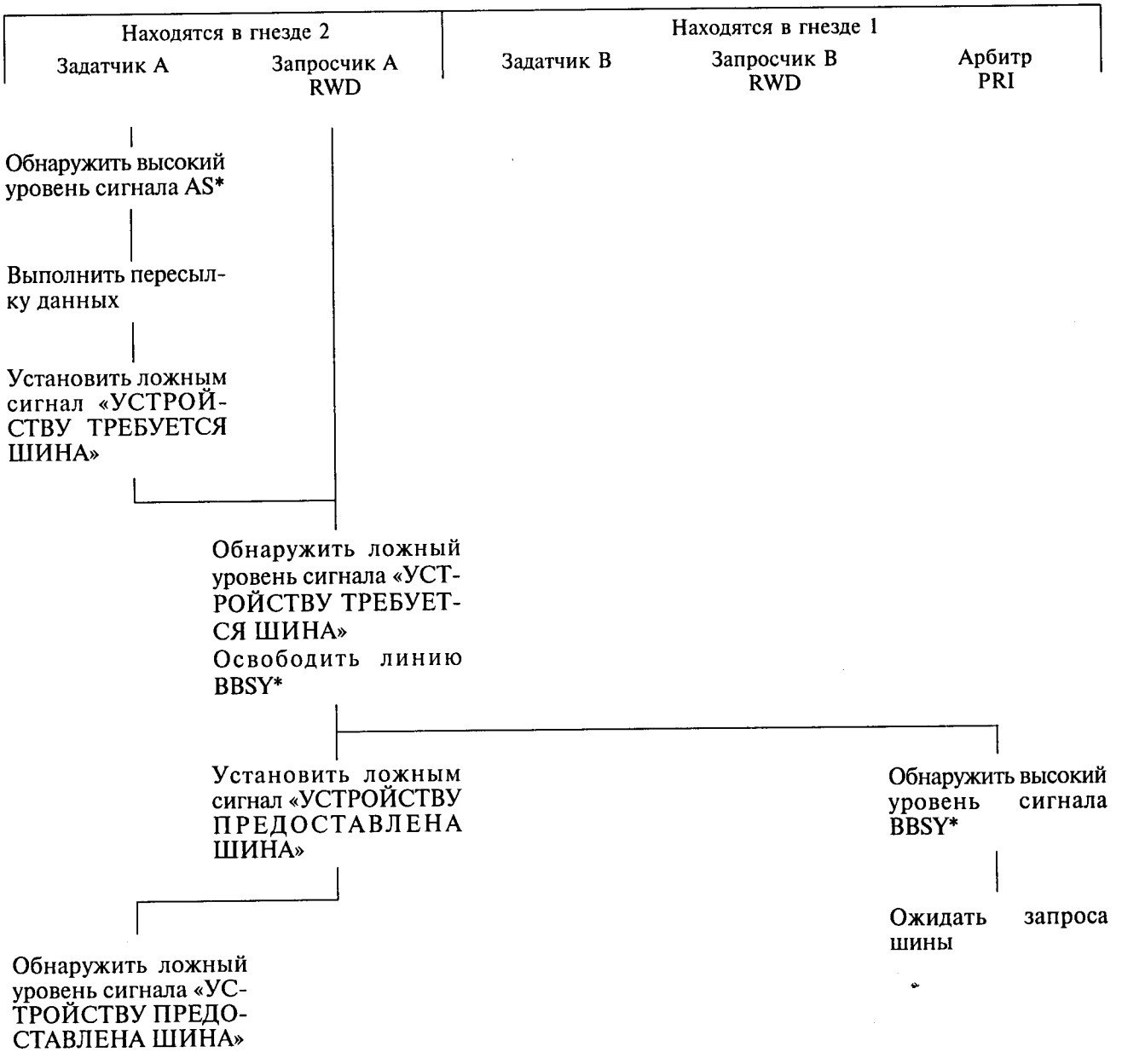


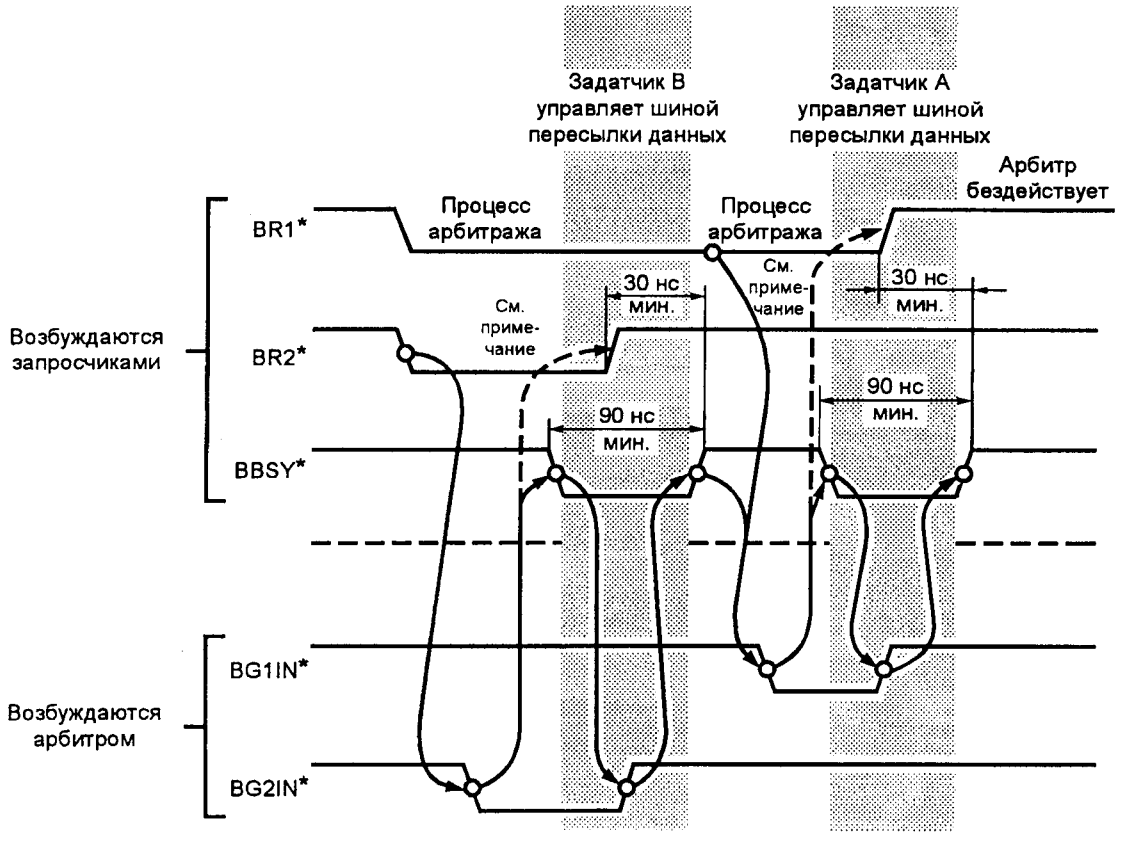
Рисунок 3.5, лист 3

Когда BBSY* становится низким, арбитр устанавливает сигнал BG2IN* гнезда 1 высоким.

Когда задатчик В заканчивает свою пересылку (пересылки) данных, он сигнализирует об этом своему запросчику В, устанавливая ложным сигнал «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА». Запросчик В после этого освобождает линию BBSY* при условии, что он принял сигнал BG2IN* высоким и прошло 30 нс после того, как он освободил BR2*.

Арбитр интерпретирует освобождение линии BBSY* как сигнал начала арбитража любых текущих запросов шины. Поскольку BR1* единственный сигнал запроса, установленный низким, арбитр предоставляет шину пересылки данных запросчику А, устанавливая BG1IN* низким. Запросчик А отвечает установкой BBSY* низким. Когда задатчик А завершает свою пересылку (пересылки) данных и сигнализирует об этом, устанавливая ложным сигнал «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА», запросчик А освобождает линию BBSY* при условии, что он принял сигнал BG1IN* высоким и прошло 30 нс после того, как он освободил BR1*.

Поскольку в данном примере к моменту освобождения линии BBSY* запросчиком А ни на одной из линий запроса шины не установлен низкий уровень, арбитр ожидает, пока не обнаружит какой-либо запрос шины.



Примечание — В данном примере каждый из запросчиков удерживает на своей линии запроса шины низкий уровень, пока ему не будет предоставлена шина пересылки данных. В некоторых случаях запросчик может освободить свою линию запроса, не получив сигнала предоставления шины (см. 3.3.2).

Рисунок 3.6 — Диаграмма последовательности арбитража. Два запросчика, два уровня запросов

Замечание 3.13. Описание работы, иллюстрируемое рисунками 3.5 и 3.6, справедливо для обоих арбитров PRI и RRS, если не рассматривать арбитр RRS для случая, когда последний активный запрос был на линии BR2*. В этом случае арбитр сначала обработает запрос BR1*, а затем перейдет к обработке запроса BR2*.

Замечание 3.14. Сигнал BBSY* и сигналы предоставления шины полностью взаимосвязаны, как показано на рисунке 3.6:

а) арбитр не устанавливает сигнал предоставления шины высоким, пока не обнаружит на линии BBSY* низкий уровень;

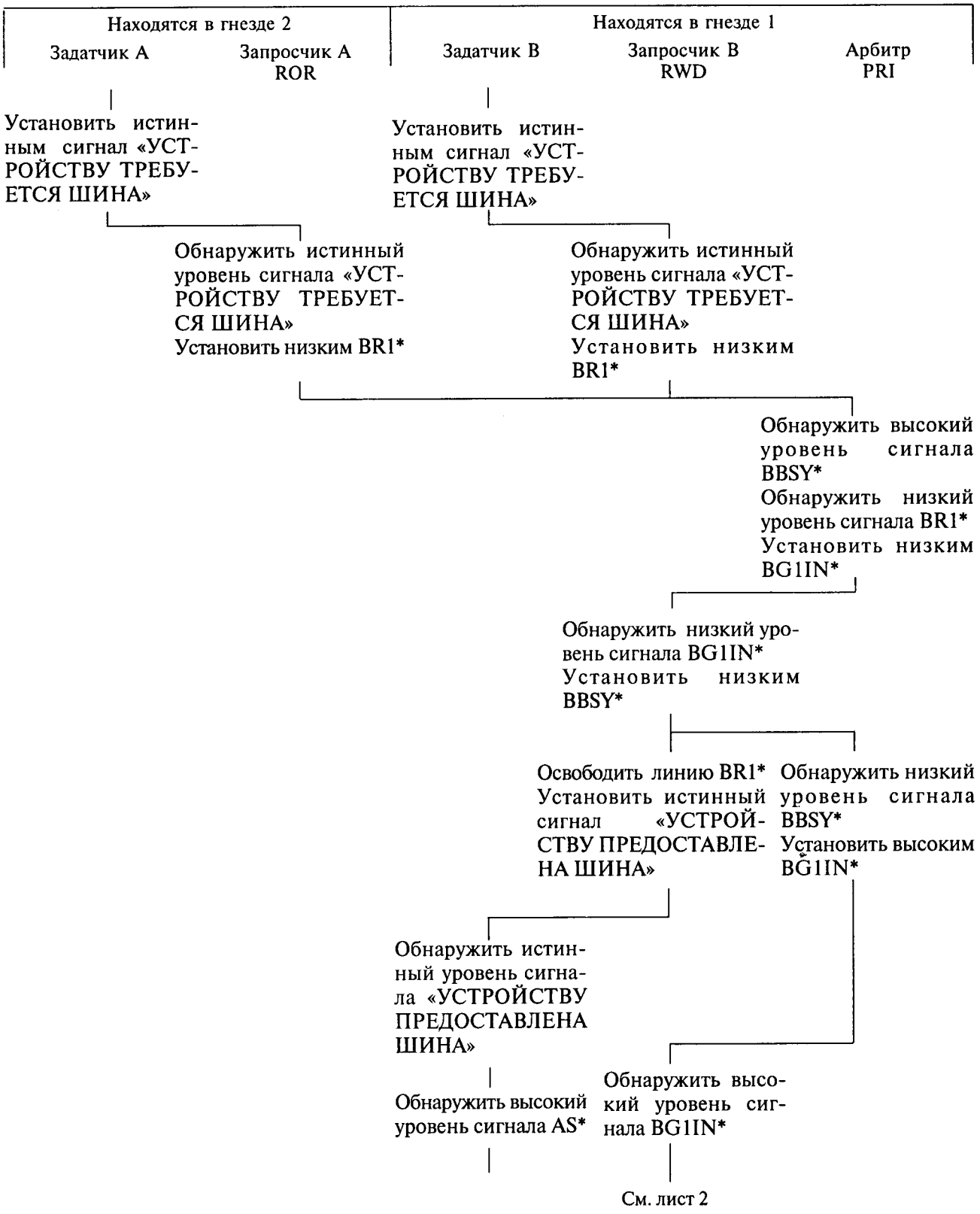
б) запросчик не освобождает линию BBSY*, пока не обнаружит сигнал предоставления шины высоким;

в) арбитр не установит низким сигнал предоставления шины, пока не обнаружит BBSY* высоким;

г) следующий запросчик не установит BBSY* низким, пока не обнаружит сигнал предоставления шины низким.

3.4.2 Арбитраж двух запросов шины на одной линии запроса

На рисунках 3.7 и 3.8 показана последовательность событий, которые происходят, если запросчик ROR и запросчик RWD одновременно посылают арбитру PRI запросы по общей линии запроса шины. В данном примере арбитр и запросчик RWD размещаются в модуле системного контроллера в гнезде 1, а запросчик ROR — в гнезде 2. В начале последовательности оба запросчика одновременно устанавливают BR1* низким. После этого арбитр устанавливает сигнал BG1IN* низким для своего собственного гнезда (гнездо 1). Этот сигнал BG1IN* контролируется запросчиком В, также находя-



Примечание — Сигналы «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА» и «УСТРОЙСТВУ ПРЕДОСТАВЛЕНА ШИНА» являются внутримодульными сигналами взаимодействия между задатчиком и его запросчиком (см. рисунок 3.4).

Рисунок 3.7, лист 1 — Алгоритм арбитража. Два запросчика, один и тот же уровень запроса

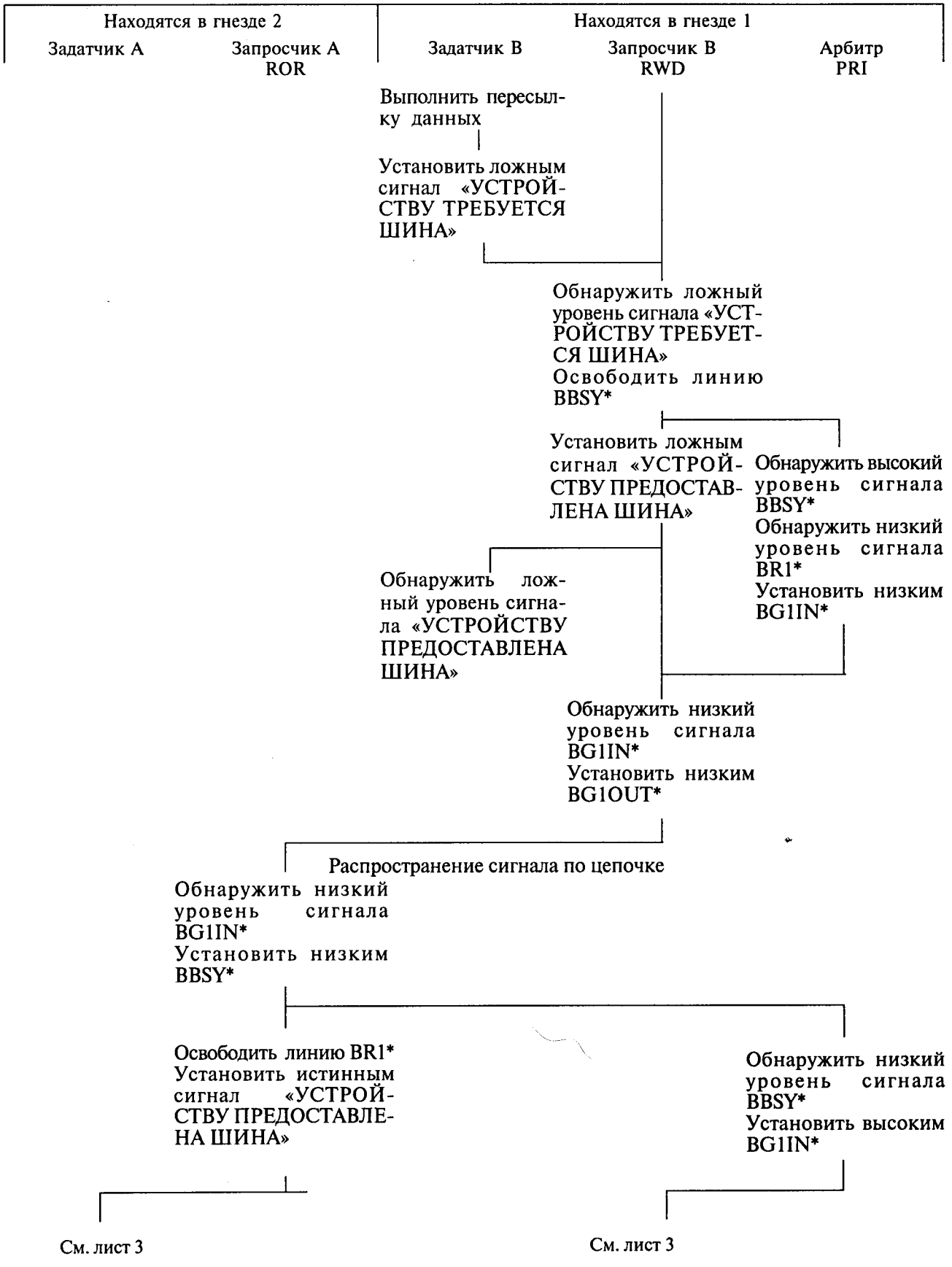


Рисунок 3.7, лист 2

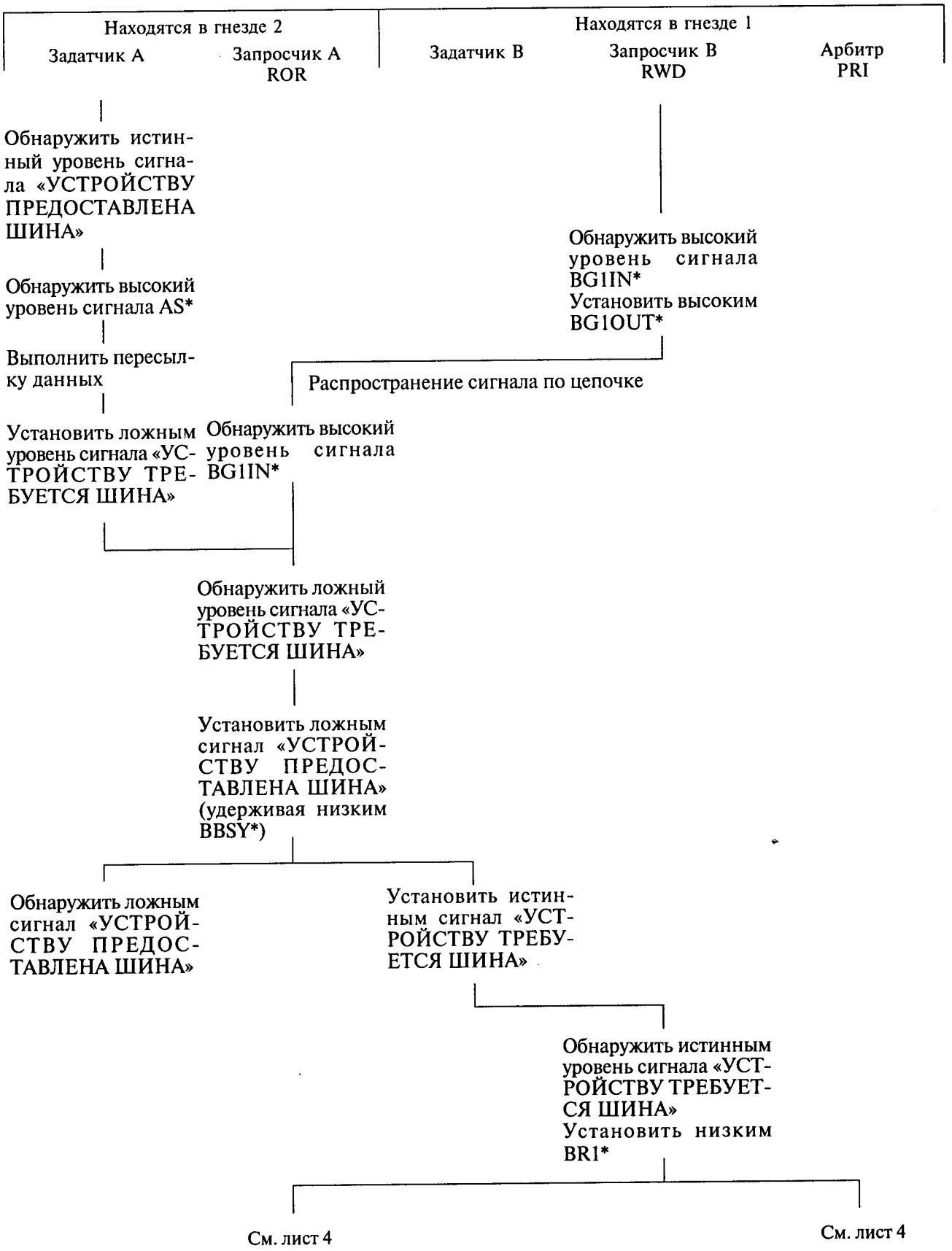


Рисунок 3.7, лист 3

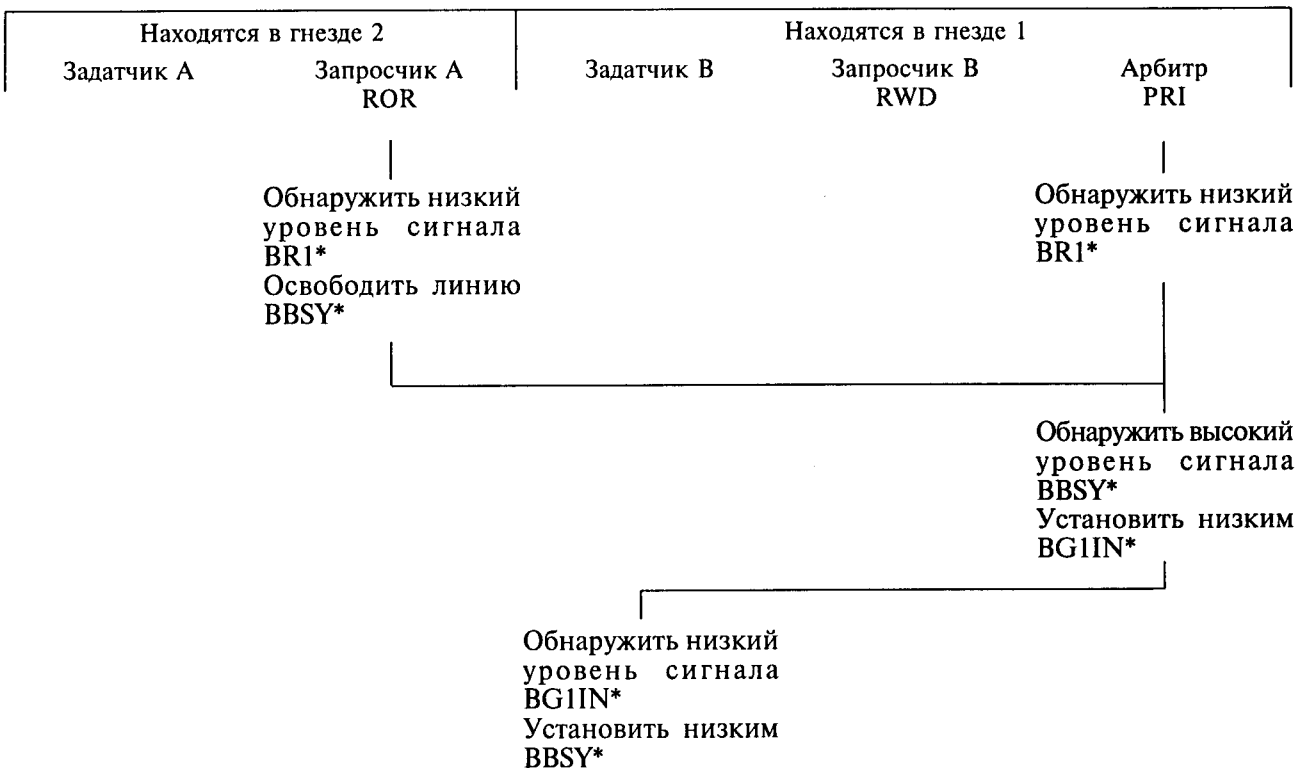


Рисунок 3.7, лист 4

щимся в гнезде 1. Когда запросчик В обнаружит сигнал BG1IN* низким, он отвечает установкой BBSY* низким. После этого запросчик В освобождает линию BR1* и информирует задатчик В о том, что шина пересылки данных находится в его распоряжении.

Замечание 3.15. Хотя запросчик В освобождает линию BR1*, запросчик А продолжает удерживать на ней низкий уровень (см. рисунки 3.7 и 3.8).

Обнаружив BBSY* низким, арбитр устанавливает высоким сигнал BG1IN*. Когда задатчик В завершит свою пересылку (пересылки) данных, он устанавливает ложным сигнал «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА». Обнаружив это, запросчик В по истечении 30 нс после освобождения им линии BR1* освободит линию BBSY*.

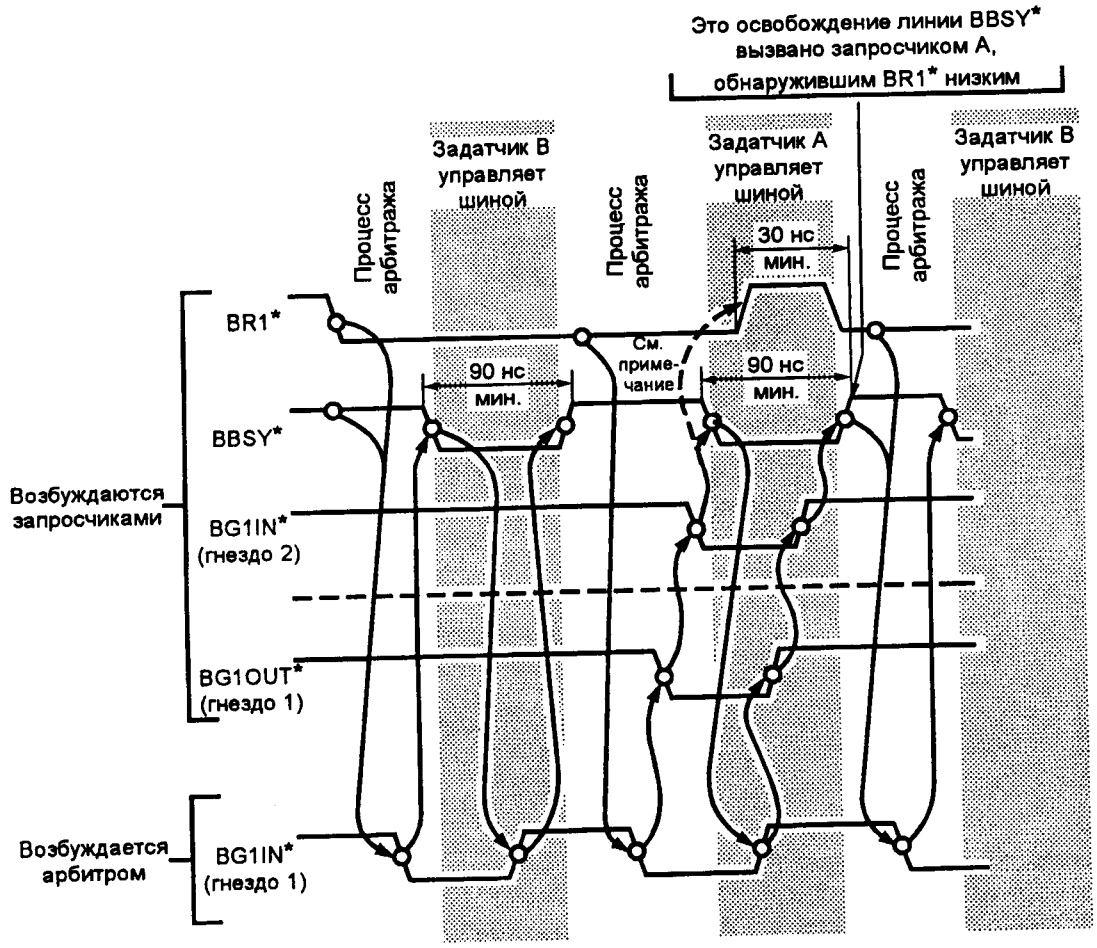
Арбитр интерпретирует освобождение BBSY* как сигнал к тому, чтобы начать арбитраж текущих запросов шины. Так как BR1* все еще установлен низким, арбитр снова устанавливает низким BG1IN*. Когда запросчик В обнаруживает BG1IN* низким, он устанавливает низким BG1OUT*, поскольку ему шина пересылки данных больше не нужна. После этого запросчик А обнаруживает низкий уровень на своей линии BG1IN* и отвечает установкой BBSY* низким, а BR1* высоким. Обнаружив на линии BBSY* низкий уровень, арбитр устанавливает высоким BG1IN*, что заставляет запросчик В установить BG1OUT* высоким.

Спустя некоторое время, когда задатчик А закончит свои пересылки данных, он устанавливает ложным сигнал «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА», показывая, что он закончил использование шины пересылки данных.

Поскольку запросчик А — это запросчик ROR, он не освобождает BBSY*, продолжая удерживать его низким. В случае, если задатчику А снова потребуется использовать шину пересылки данных, арбитраж будет не нужен. Однако в данном примере запросчик В устанавливает BR1* низким, показывая, что ему необходима шина пересылки данных, и запросчик А (который контролирует все линии запроса шины) освобождает линию BBSY*. В этом случае арбитр предоставляет шину запросчику В.

3.5 Условия возникновения гонок между сигналами запроса задатчика и предоставления шины арбитра

Предположим, что имеются два запросчика: запросчик А и запросчик В, которые совместно используют общую линию запроса шины. Запросчик В, который расположен дальше по цепочке, запрашивает шину и арбитр устанавливает на соответствующей линии предоставления шины



Примечание — В данном примере каждый из запросчиков удерживает на своей линии запроса шины низкий уровень, пока ему не будет предоставлена шина пересылки данных. В некоторых случаях запросчик может освободить свою линию запроса, не получив сигнала предоставления шины (см. 3.3.2).

Рисунок 3.8 — Диаграмма последовательности арбитража. Два запросчика, один и тот же уровень запросов

низкий уровень сигнала. Этот отрицательный перепад достигает запросчика А в тот самый момент, когда задатчик А сигнализирует, что ему требуется шина. Если запросчик А спроектирован неправильно, такая ситуация может заставить его на мгновение установить на своей линии BGxOUT* низкий уровень сигнала, а затем — снова высокой, приводя к формированию отрицательного импульса.

Правило 3.13. *Запросчики ДОЛЖНЫ БЫТЬ СПРОЕКТИРОВАНЫ таким образом, чтобы исключить возможность появления на их линии BGxOUT* кратковременных отрицательных импульсов.*

Замечание 3.16. Если запросчик спроектирован таким образом, что он запоминает в защелке состояние внутримодульного сигнала «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА» по отрицательному перепаду сигнала на его линии BGxIN* и если этот внутримодульный сигнал находится в состоянии перехода, когда возникает отрицательный перепад, то выходы защелки в течение короткого времени будут иногда либо генерировать, либо находиться в пороговой области между высоким и низким уровнями. По этой причине на запросчик не накладывается никаких временных ограничений по передаче сигнала предоставления шины. Ему запрещено только генерировать на своей линии BGxOUT короткий отрицательный импульс, который может быть интерпретирован как сигнал предоставления шины запросчикам, следующим дальше по цепочке.

Разрешение 3.8. Если запросчик обнаруживает, что его внутримодульному задатчику требуется шина между моментом получения сигнала предоставления шины для другого запросчика и моментом, когда этот сигнал передается дальше, то он МОЖЕТ рассматривать этот сигнал предоставления шины как предназначенный ему. В этом случае другой запросчик будет продолжать удерживать низким свой сигнал запроса до тех пор, пока не поступит другой сигнал предоставления шины.

4 Шина приоритетных прерываний

4.1 Введение

Магистраль VME включает в себя шину приоритетных прерываний, которая содержит сигнальные линии, необходимые для формирования и обслуживания прерываний. На рисунке 4.1 приведена схема взаимодействия шины приоритетных прерываний с функциональными блоками в составе типовой системы магистрали VME. Линии этой шины используются прерывателями для передачи запросов прерывания обработчикам прерываний, которые отвечают на эти запросы.

Любая система, которая обладает функцией прерывания, имеет соответствующие подпрограммы, которые называются подпрограммами обслуживания прерываний и вызываются для исполнения прерываниями. Системы прерываний можно подразделить на две группы:

- системы с одним обработчиком прерываний, который принимает и обслуживает все прерывания шины;
- распределенные системы, которые содержат два или больше обработчиков прерываний, принимающих и обслуживающих прерывания шины.

4.1.1 Системы прерываний с одним обработчиком

В системе прерываний с одним обработчиком все прерывания поступают на вход одного обработчика прерываний и все подпрограммы обслуживания прерываний исполняются одним процессором. На рисунке 4.2 показана структурная схема системы прерываний с одним обработчиком. Такой тип архитектуры наиболее удобен для систем управления станками и технологическими процессами, когда супервизорный процессор координирует действия специализированных процессоров, непосредственно связанных с управляемыми объектами.

Супервизорный процессор является местом назначения всех прерываний шины и обслуживает их в соответствии с приоритетами. На специализированные процессоры не возлагаются задачи обслуживания прерываний шины, и они могут основное внимание уделить управлению станками или технологическими процессами.

4.1.2 Распределенные системы прерываний

На рисунке 4.3 показана структурная схема распределенной системы прерываний. Эта система содержит два или больше обработчиков прерываний, каждый из которых обслуживает только подмножество прерываний шины. В типовых реализациях каждый из обработчиков прерываний располагается на своем процессорном модуле. Такая архитектура наиболее удобна для распределенных вычислительных систем, когда несколько равноправных процессоров выполняют прикладные программы. Поскольку каждый из этих равноправных процессоров выполняет часть системного программного обеспечения, ему может потребоваться взаимодействие с другими процессорами. В распределенной системе каждый процессор обслуживает только те прерывания, которые предназначены непосредственно ему, устанавливая специальные каналы связи со всеми другими процессорами.

4.2 Линии шины приоритетных прерываний

В процессе формирования и обработки прерываний шины используются шины пересылки данных, арбитража и приоритетных прерываний.

Последующее описание шины приоритетных прерываний предполагает понимание принципов работы шины пересылки данных, описанной в разделе 2, и шины арбитража, описанной в разделе 3.

Шина приоритетных прерываний включает в себя семь сигнальных линий запроса прерываний, одну линию подтверждения прерывания и одну цепочку подтверждения прерывания, которые получили следующие обозначения:

IRQ1* (Interrupt ReQuest 1) — линия запроса прерывания 1;
IRQ2* (Interrupt ReQuest 2) — линия запроса прерывания 2;
IRQ3* (Interrupt ReQuest 3) — линия запроса прерывания 3;
IRQ4* (Interrupt ReQuest 4) — линия запроса прерывания 4;
IRQ5* (Interrupt ReQuest 5) — линия запроса прерывания 5;
IRQ6* (Interrupt ReQuest 6) — линия запроса прерывания 6;
IRQ7* (Interrupt ReQuest 7) — линия запроса прерывания 7;
IACK* (Interrupt ACKnowledge) — линия подтверждения прерывания;
IACKIN*/IACKOUT* (Interrupt ACKnowledge INput/Interrupt ACKnowledge OUTput) — цепочка подтверждения прерывания.

4.2.1 Линии запроса прерывания

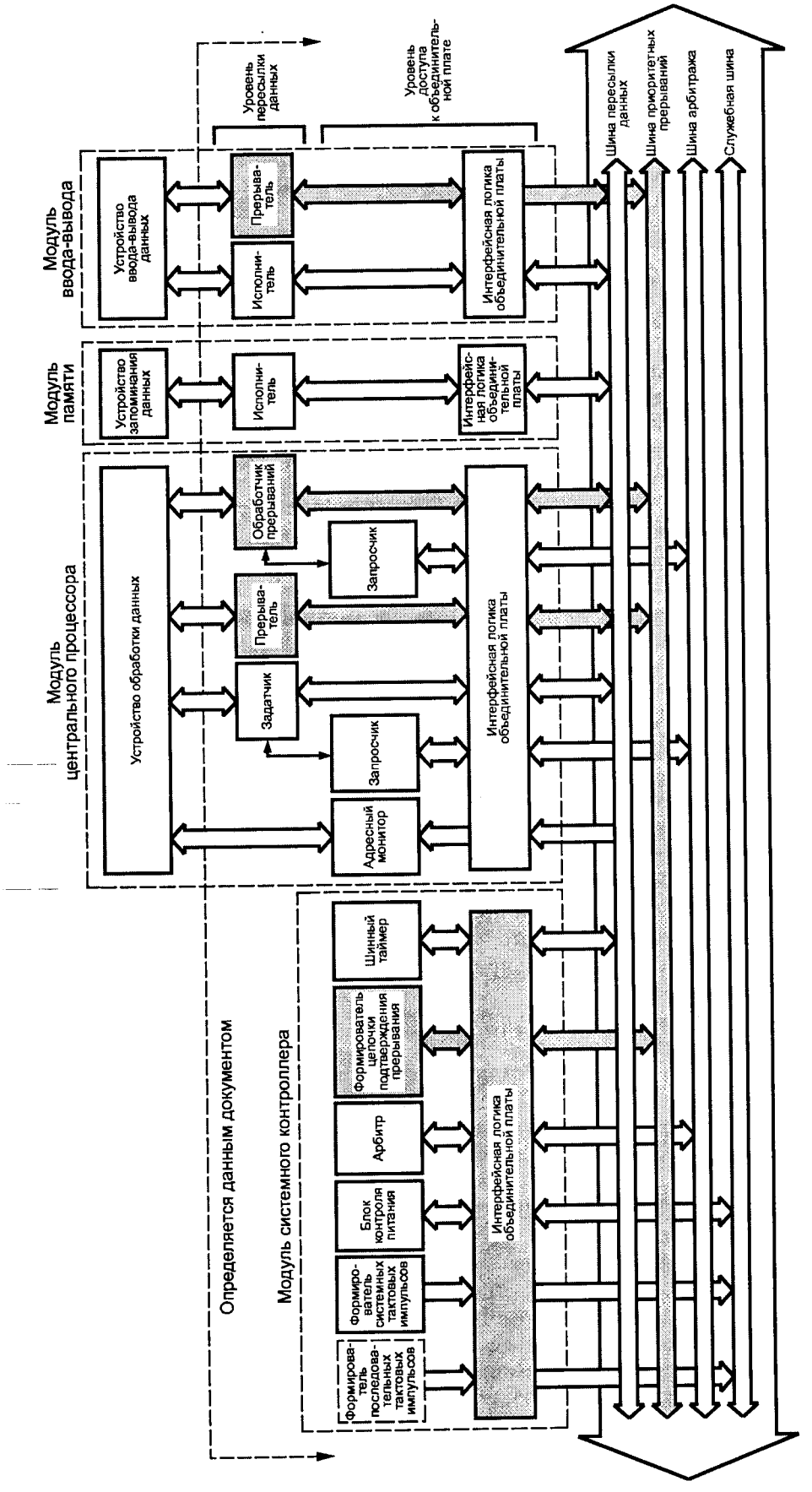


Рисунок 4.1 — Схема взаимодействия шины приоритетных прерываний с функциональными блоками

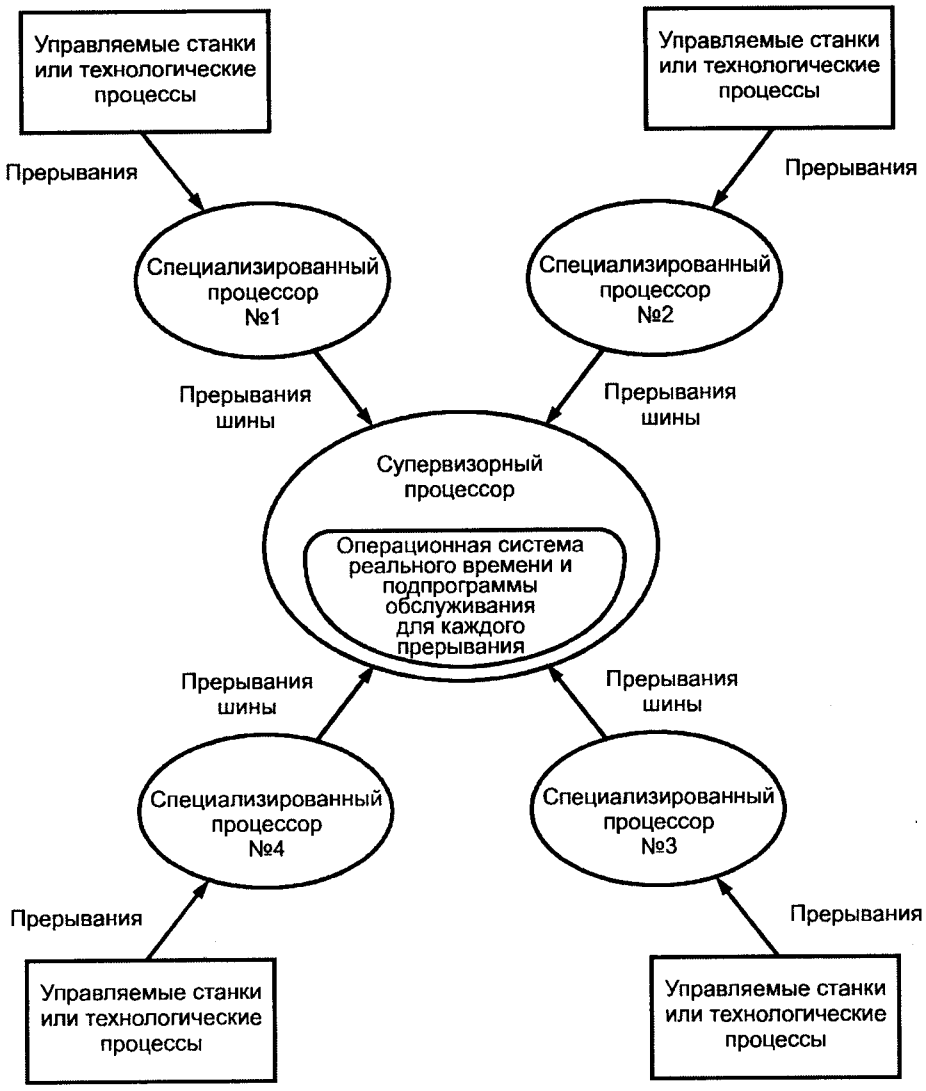


Рисунок 4.2 — Структурная схема системы прерывания с одним обработчиком

Прерыватели запрашивают прерывания, устанавливая низкий уровень сигнала на линии запроса прерывания. В системе с одним обработчиком эти линии запроса прерывания распределены по приоритетам, причем наивысший приоритет имеет линия IRQ7*.

4.2.2 Линия подтверждения прерывания

Линия IACK* проходит по всей длине объединительной платы и подсоединена к контакту IACKIN* гнезда 1 (см. рисунок 4.4). Когда на контакте IACKIN* устанавливается низкий уровень сигнала, формирователь цепочки подтверждения прерывания, расположенный в гнезде 1, распространяет по ней этот отрицательный перепад.

4.2.3 Цепочка подтверждения прерывания (IACKIN*/IACKOUT*)

Каждую из семи линий запроса прерывания могут совместно использовать два или более прерывателей. Цепочка подтверждения прерывания гарантирует, что только один прерыватель отвечает на цикл подтверждения прерывания. Эта цепочка проходит через все модули магистрали VME. Каждый прерыватель, который установил низкий уровень на линии запроса прерывания, ждет, пока на его вход IACKIN* цепочки поступит отрицательный перепад. Только после приема этого отрицательного перепада прерыватель отвечает на цикл подтверждения прерывания. При этом он не передает этот отрицательный перепад дальше по цепочке, блокируя таким образом участие других прерывателей в данном цикле подтверждения прерывания.

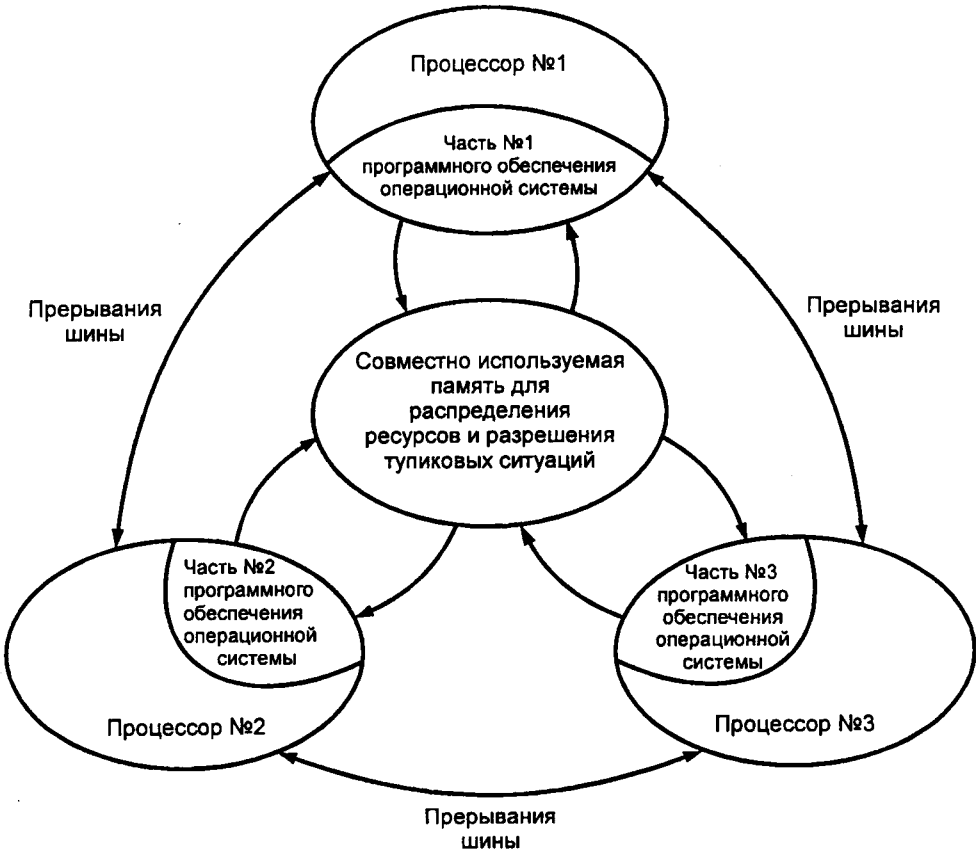


Рисунок 4.3 — Структурная схема распределенной системы прерывания

Правило 4.1. *Если какое-либо гнездо объединительной платы не занято модулем, а дальше по цепочке подтверждения прерывания модули установлены, то для пропускания сигнала по цепочке в пустом гнезде ДОЛЖНЫ БЫТЬ УСТАНОВЛЕНЫ перемычки.*

4.3 Функциональные блоки шины приоритетных прерываний. Общее описание

С шиной приоритетных прерываний связаны три типа функциональных блоков: прерыватели, обработчики прерываний и формирователи цепочки подтверждения прерывания. Функциональные возможности обработчиков прерываний и прерывателей определяются в перечне мнемонических обозначений, показывающих, какие типы циклов подтверждения прерывания они могут формировать и принимать, соответственно.

В 4.3.1 — 4.3.3 представлены схемы взаимодействия трех типов функциональных блоков с магистралью VME: обработчика прерываний, прерывателя и формирователя цепочки подтверждения прерывания.

Правило 4.2. *Линии выходных сигналов, показанные на рисунках 4.5—4.7 сплошными линиями, ДОЛЖНЫ ВОЗБУЖДАТЬСЯ соответствующим функциональным блоком, если он не устанавливает на них высокий уровень постоянно.*

Замечание 4.1. Если линия выходного сигнала не возбуждается, то оконечные нагрузки на объединительной плате гарантируют, что на ней удерживается высокий уровень.

Правило 4.3. *Линии входных сигналов, показанные на рисунках 4.5—4.7 сплошными линиями, ДОЛЖНЫ КОНТРОЛИРОВАТЬСЯ и на их сигналы ДОЛЖНЫ ДАВАТЬСЯ соответствующие ответы.*

Замечание 4.2. Правила и разрешения для возбуждения и контроля сигнальных линий, изображенных на рисунках 4.5, 4.7 пунктиром, приведены в таблицах 4.1, 4.2.

4.3.1 Обработчики прерываний

Обработчик прерываний выполняет следующие функции:

- распределяет поступающие запросы прерываний в соответствии с их приоритетами в пределах назначенной ему группы линий запросов прерывания (максимально IRQ1*—IRQ7*);

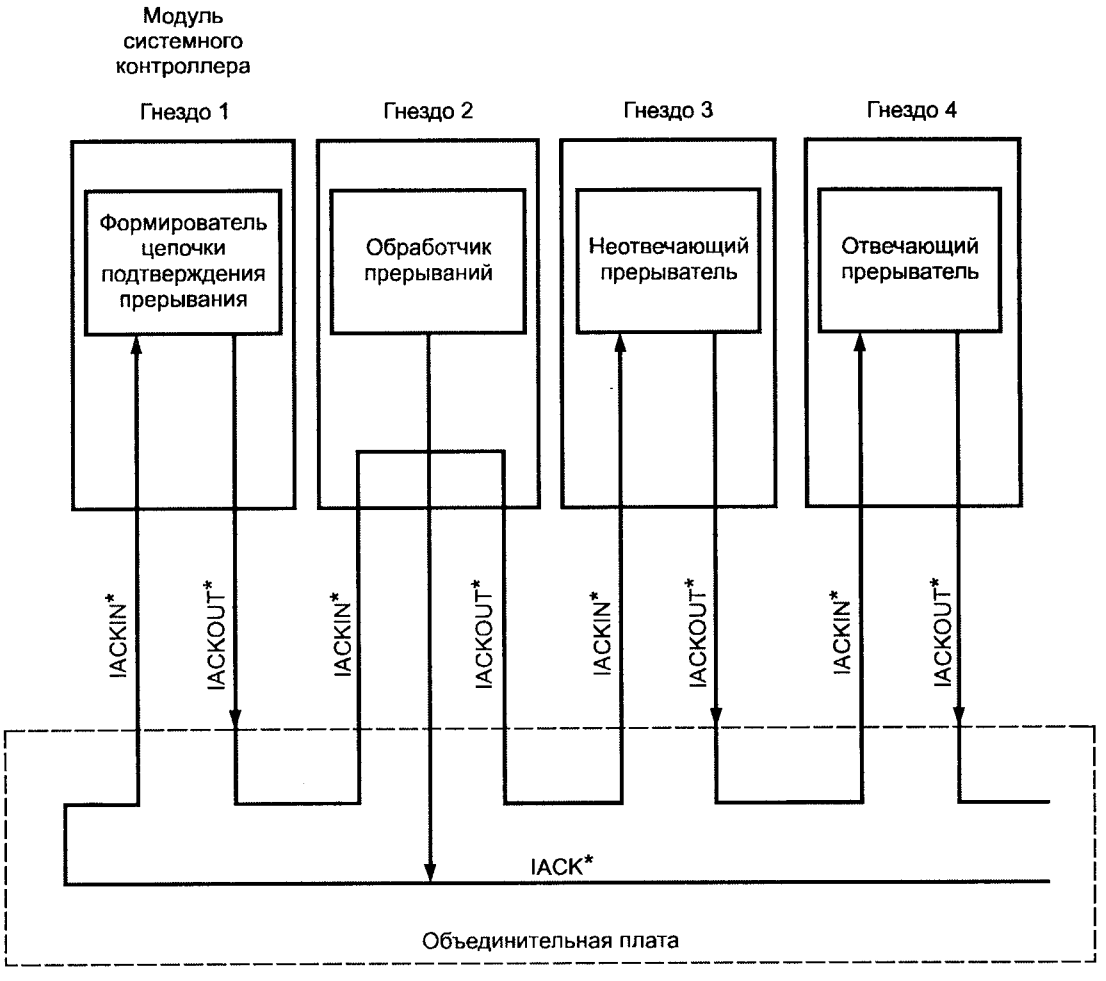


Рисунок 4.4 — Последовательная цепочка IACKIN*/IACKOUT*

- использует свой внутримодульный запросчик для запроса шины пересылки данных и, когда эта шина будет ему предоставлена, инициирует цикл подтверждения прерывания, считывая информацию статуса/идентификации из прерывателя, чей запрос подтверждается;
- инициирует соответствующую последовательность обслуживания прерывания на основе полученной информации статуса/идентификации.

Замечание 4.3. Действия, предпринимаемые во время последовательности обслуживания прерывания, настоящим стандартом не определены. Обслуживание прерываний может включать, а может и не включать использование магистрали VME.

Обработчик прерываний использует шину пересылки данных для считывания информации статуса/идентификации из прерывателя. В этом смысле обработчик прерываний действует как задатчик, а прерыватель — как исполнитель. Однако имеются следующие четыре важных отличия:

- обработчик прерываний всегда устанавливает низким сигнал IACK*;
- обработчик прерываний не возбуждает линии модификатора адреса;
- обработчик прерываний использует только три младшие адресные линии (A01—A03);
- обработчик прерываний никогда не возбуждает линии данных.

Обработчик прерываний при обращении к шине всегда устанавливает сигнал на линии IACK* низким. Задатчик же либо устанавливает сигнал на ней высоким, либо не возбуждает ее вообще.

Обработчику прерываний не нужно устанавливать достоверный код на линиях модификатора адреса, он только устанавливает достоверную информацию на трех младших адресных линиях (A01—A03). Уровни сигналов на этих трех адресных линиях определяют, для какой из семи линий запроса прерывания дается подтверждение, как показано в таблице 4.7. Задатчик же устанавливает на адресных линиях (количество которых может быть 15, 23 или 31, в зависимости от режима адресации)

адрес исполнителя, к которому обращается, а на линиях модификатора адреса — соответствующий код.

Обработчик прерываний не возбуждает линии данных (т.е. он не выполняет операцию записи в прерыватель) и, поскольку никогда не устанавливает сигнал на линии WRITE* низким, то и не возбуждает ее. Задатчик же использует линии данных для пересылки данных между собой и исполнителем в обоих направлениях и во время обычного использования устанавливает WRITE* низким или высоким, по мере необходимости.

Схема взаимодействия обработчика прерываний с магистралью VME приведена на рисунке 4.5. Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных пунктиром, приведены в таблице 4.1.

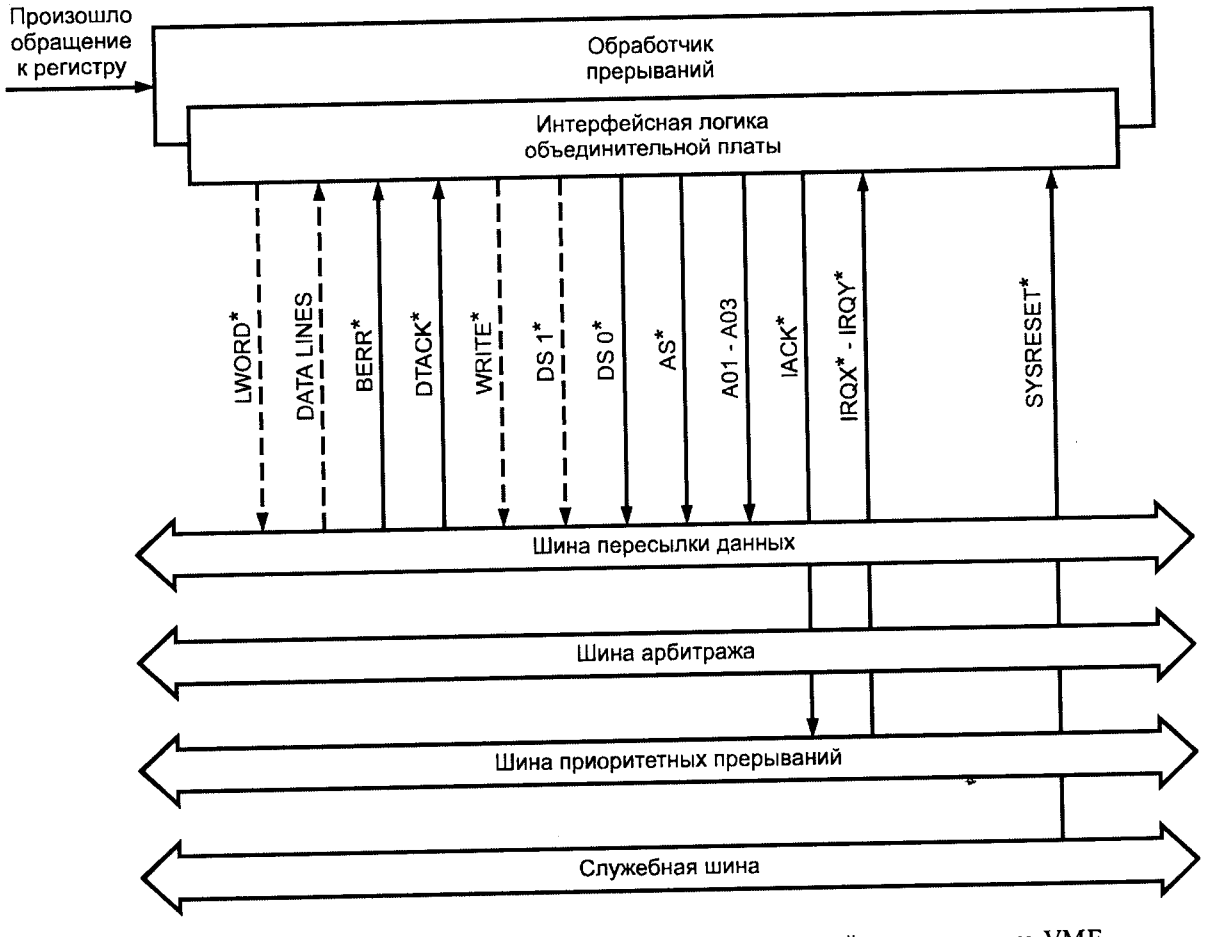


Рисунок 4.5 — Схема взаимодействия обработчика прерываний с магистралью VME

Таблица 4.1 — Обработчик прерываний. Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных на рисунке 4.5 пунктиром

Тип обработчика	Правила и разрешения прерываний
D08(O)	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линии D00—D07* МОЖЕТ возбуждать или МОЖЕТ не возбуждать линии LWORD* и DS1* МОЖЕТ контролировать или МОЖЕТ не контролировать линии D08—D31
D16	*ДОЛЖЕН ВОЗБУЖДАТЬ линию DS1** *ДОЛЖЕН КОНТРОЛИРОВАТЬ линии D00—D15* МОЖЕТ возбуждать или МОЖЕТ не возбуждать линию LWORD* МОЖЕТ контролировать или МОЖЕТ не контролировать линии D16—D31

Окончание таблицы 4.1

Тип обработчика	Правила и разрешения прерываний
D32	*ДОЛЖЕН ВОЗБУЖДАТЬ линии DS1* и LWORD** *ДОЛЖЕН КОНТРОЛИРОВАТЬ линии D00—D31*
ВСЕ	*НЕ ДОЛЖНЫ УСТАНАВЛИВАТЬ сигнал на линии WRITE* низким*
Примечание — Мнемонические обозначения D08(O), D16 и D32 применительно к обработчику прерываний определены в таблице 4.5.	

4.3.2 Прерыватель и его функции

Прерыватель выполняет следующие функции:

- запрашивает прерывание у обработчика прерываний, который контролирует его линию запроса прерывания;

- если прерыватель принимает отрицательный перепад на входе цепочки подтверждения прерывания и если он запрашивает прерывание, и если уровни сигналов на трех младших адресных линиях соответствуют используемой им линии запроса прерывания, и если разрядность запрашиваемой информации статуса/идентификации либо равна, либо больше, чем разрядность, которую он может обеспечить, то он устанавливает на линиях данных информацию статуса/идентификации. В противном случае, он пропускает отрицательный перепад дальше по цепочке подтверждения прерывания.

Каждый прерыватель возбуждает только одну линию запроса прерывания. Модуль, который формирует запросы прерывания по нескольким линиям, рассматривается данным стандартом как имеющий несколько прерывателей.

Разрешение 4.1. Поскольку прерыватель — это всего лишь концептуальная модель, внутримодульная логика МОЖЕТ совместно использоваться несколькими прерывателями.

Прерыватель использует для запроса прерывания одну из семи линий. После этого он контролирует три младшие линии адресной шины (A01—A03), линию IACKIN* и дополнительно линию IACK*, чтобы определить, когда его прерывание подтверждается. При подтверждении он помещает свою информацию статуса/идентификации на шину данных и сообщает обработчику прерываний о том, что эта информация достоверна, устанавливая DTACK* низким.

Имеется пять основных отличий в использовании шины пересылки данных прерывателем и исполнителем. Прерыватель использует шину пересылки данных следующим образом:

- отвечает только тогда, когда его входной сигнал IACKIN* низкий;
- не контролирует линии модификатора адреса ;
- контролирует только три младшие адресные линии;
- не контролирует линию WRITE*;
- ему разрешено отвечать данными, разрядность которых отличается от разрядности запрашиваемых данных.

Исполнитель контролирует линию AS* и интерпретирует отрицательный перепад на ней как сигнал достоверности текущего цикла шины. После этого исполнитель приступает к декодированию соответствующего числа адресных линий (15, 23 или 31) и линий модификатора адреса и на основе этой информации определяет, произошло ли к нему обращение. При этом исполнитель отвечает только тогда, когда IACK* установлен высоким.

Что касается прерывателя, то он интерпретирует отрицательный перепад на своей линии IACKIN* как разрешающий сигнал, чтобы ответить на текущий цикл подтверждения прерывания. Он декодирует только три младшие адресные линии (A01—A03), игнорируя линии модификатора адреса.

Прерывателю не требуется контролировать линию WRITE*, поскольку в него никогда не записываются никакие данные. Исполнителям требуется контролировать линию WRITE*, чтобы они могли отличать циклы считывания от циклов записи.

Прерыватель помещает на шину информацию статуса/идентификации и отвечает сигналом DTACK*, даже если линии LWORD*, DS1* и DS0* требуют от прерывателя информацию статуса/идентификации, разрядность которой больше, чем он может обеспечить. Например, обработчик прерываний может установить низким LWORD* и оба сигнала DS0* и DS1*, показывая, что он намерен считать 32 разряда информации статуса/идентификации с линий D00—D31; однако прерыва-

тель D08(O) ответит восьмиразрядной информацией статуса/идентификации по линиям D00—D07. В противоположность этому, если исполнитель не может обеспечить запрашиваемую разрядность данных, он либо отвечает сигналом BERR*, либо не отвечает совсем, приводя в результате к формированию тайм-аута шины.

Замечание 4.4. Когда прерыватель помещает информацию статуса/ идентификации на шину данных, любые невозбуждаемые линии данных считываются обработчиком прерываний как линии, на которых удерживается высокий уровень из-за оконечных нагрузок шины. Например, если обработчик прерываний D16 инициирует двухбайтовый цикл подтверждения прерывания, прерыватель D08(O) поместит восьмиразрядную информацию статуса/идентификации на линии D00—D07. Старшие восемь разрядов, считанные обработчиком прерываний с линий D08—D15, будут представлять собой единицы (высокий уровень), поскольку эти линии не возбуждаются прерывателем D08(O).

Правило 4.4. *Прежде чем ответить на цикл подтверждения прерывания, прерыватель ДОЛЖЕН ПРОВЕРИТЬ выполнение следующих обязательных условий:*

- условие 1— имеется необработанный запрос прерывания;

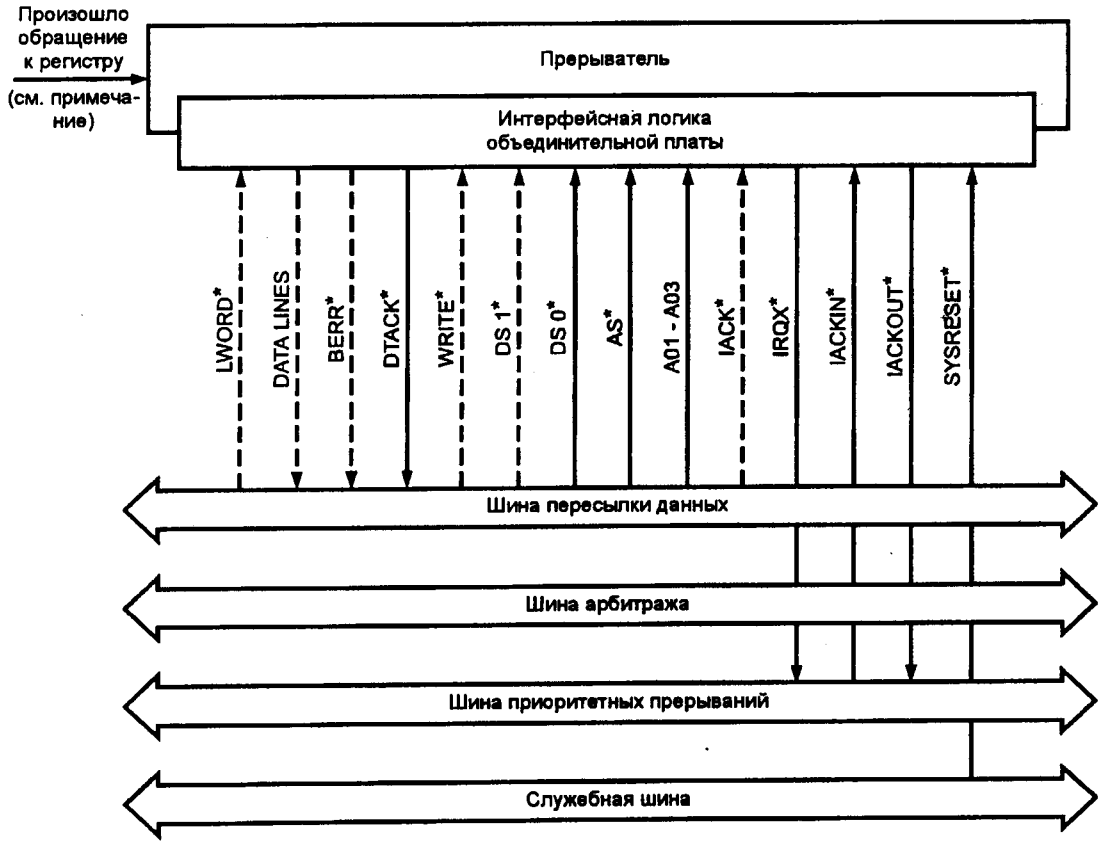
- условие 2— уровень этого запроса соответствует уровню,указываемому на линиях A01—A03;

- условие 3— разрядность запрашиваемой информации статуса/идентификации равна или больше разрядности данных, которыми он может ответить;

- условие 4 — он принял отрицательный перепад на свой вход цепочки IACKIN.*

Если какое-либо из этих четырех условий не соблюдается, прерыватель НЕ ДОЛЖЕН ОТВЕЧАТЬ на данный цикл подтверждения прерывания. Если условие 4 соблюдено, а одно из условий 1, 2 или 3 — нет, то прерыватель ДОЛЖЕН ПРОПУСКАТЬ отрицательный перепад сигнала IACKIN к следующему прерывателю по цепочке.*

Схема взаимодействия прерывателя с магистралью VME показана на рисунке 4.6. Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных на рисунке пунктиром, приведены в таблице 4.2.



Примечание — Этот входной сигнал присутствует только у прерывателей RORA.

Рисунок 4.6 — Схема взаимодействия прерывателя с магистралью VME

Таблица 4.2 — Прерыватели. Правила и разрешения для возбуждения и контроля сигнальных линий, отмеченных на рисунке 4.6 пунктиром

Тип прерывателя	Правила и разрешения
D08(O)	*ДОЛЖЕН ВОЗБУЖДАТЬ линии D00—D07* *НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ низкие уровни сигналов на линиях D08—D31* МОЖЕТ контролировать или МОЖЕТ НЕ контролировать линии LWORD* или DS1*
D16	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линию DS1** *ДОЛЖЕН ВОЗБУЖДАТЬ линии D00—D15* *НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ низкие уровни сигналов на линиях D16—D31* МОЖЕТ контролировать или МОЖЕТ НЕ контролировать линию LWORD*
D32	*ДОЛЖЕН КОНТРОЛИРОВАТЬ линии DS1* и LWORD** *ДОЛЖЕН ВОЗБУЖДАТЬ линии D00—D31*
ВСЕ	МОГУТ контролировать или МОГУТ НЕ контролировать линии WRITE* и IACK* МОГУТ возбуждать или МОГУТ НЕ возбуждать линию BERR*
Примечание — Мнемонические обозначения D08(O), D16 и D32 применительно к прерывателю определены в таблице 4.5.	

4.3.3 Формирователь цепочки подтверждения прерывания

Формирователь цепочки подтверждения прерывания является еще одним функциональным блоком, который взаимодействует с обработчиками прерываний и прерывателями для координации обслуживания прерываний. Он формирует отрицательный перепад сигнала в цепочке подтверждения прерывания всякий раз, когда обработчик прерываний инициирует цикл подтверждения прерывания.

Схема взаимодействия формирователя цепочки прерывания с магистралью VME приведена на рисунке 4.7.

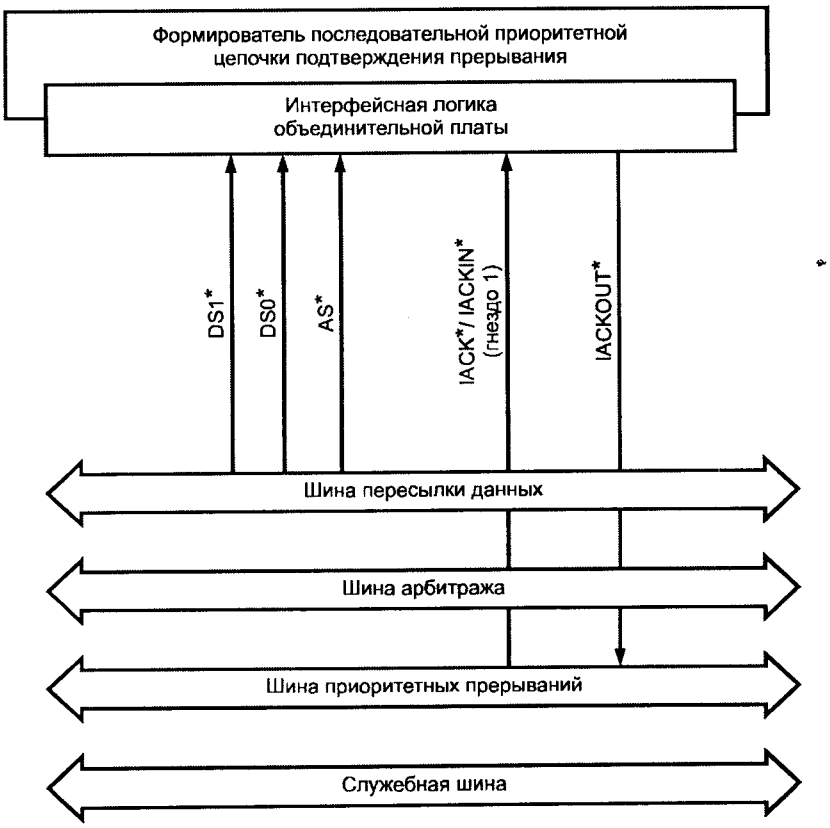


Рисунок 4.7 — Схема взаимодействия формирователя цепочки подтверждения прерывания с магистралью VME

4.3.4 Возможности обработки прерываний

Обработчики прерываний могут быть спроектированы для обработки запросов прерывания, принимаемых ими по линиям запросов от одной до семи. В таблице 4.3 показано, как используется мнемоническое обозначение IH( ) (Interrupt Handler — обработчик прерываний) для описания возможностей обработки прерываний обработчиками прерываний.

Таблица 4.3 — Использование мнемонического обозначения IH ( ) для описания возможностей обработки прерываний

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
IH(x—y)	Обработчик прерываний	Может формировать циклы подтверждения прерывания в ответ на запросы прерывания по линиям IRQx*—IRQy*
IH(x)	Обработчик прерываний	Может формировать циклы подтверждения прерывания в ответ на запросы прерывания по линиям IRQx*

4.3.5 Возможности запроса прерывания

Прерыватели могут быть спроектированы так, чтобы формировать запросы прерывания по любой из семи линий запроса прерывания. В таблице 4.4 показано, как используется мнемоническое обозначение I( ) (Interrupter — прерыватель) для описания возможностей прерывателей по формированию запросов прерывания.

Таблица 4.4 — Использование мнемонического обозначения I( ) для описания возможностей формирования запроса прерывания

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
I(x)	Обработчик прерываний	Может формировать запрос прерывания по линии IRQx*

4.3.6 Возможности пересылки информации статуса/идентификации

Имеются три варианта функциональных возможностей пересылки информации статуса/идентификации: D08(O), D16 и D32. В таблице 4.5 показано, как эти мнемонические обозначения используются для описания возможностей обработчиков прерываний и прерывателей по пересылке информации статуса/идентификации.

Таблица 4.5 — Мнемонические обозначения для описания возможностей по пересылке информации статуса/идентификации

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
D08(O)	Прерыватель	Отвечает на 8-, 16- и 32-разрядные циклы подтверждения прерывания, предоставляя 8-разрядную информацию статуса/идентификации на линии D00—D07
	Обработчик прерываний	Формирует 8-разрядные циклы подтверждения прерывания в ответ на запросы по линии (линиям) запроса прерывания и считывает информацию статуса/идентификации с линий D00—D07
D16	Прерыватель	Отвечает на 16- и 32-разрядные циклы подтверждения прерывания, предоставляя 16-разрядную информацию статуса/идентификации на линии D00—D15

Окончание таблицы 4.5

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
D32	Обработчик прерываний	Формирует 16-разрядные циклы подтверждения прерывания в ответ на запросы по линии (линиям) запроса прерывания и считывает 16-разрядную информацию статуса/идентификации с линий D00—D15
	Прерыватель	Отвечает на 32-разрядные циклы подтверждения прерывания, предоставляя 32-разрядную информацию статуса/идентификации на линии D00—D31
	Обработчик прерываний	Формирует 32-разрядные циклы подтверждения прерывания в ответ на запросы по линии (линиям) запроса прерывания и считывает 32-разрядную информацию статуса/идентификации с линий D00—D31

4.3.7 Возможность снятия запросов прерываний

Многие широко используемые периферийные интегральные схемы формируют запросы прерывания. К сожалению, не существует стандартного метода, определяющего, когда этим интегральным схемам необходимо снять свои запросы прерывания с шины. Обычно используются три следующих метода:

-метод 1, когда соответствующий процессор обнаруживает запрос прерывания от периферийного устройства, он вызывает подпрограмму обслуживания прерывания и считывает содержимое статусного регистра этого устройства; периферийное устройство интерпретирует этот цикл считывания своего статусного регистра как указание снять запрос прерывания;

-метод 2, когда соответствующий процессор обнаруживает запрос прерывания от периферийного устройства, он вызывает подпрограмму обслуживания прерывания и осуществляет запись в управляющий регистр этого устройства; периферийное устройство интерпретирует этот цикл записи как указание снять запрос прерывания;

-метод 3, когда соответствующий процессор обнаруживает запрос прерывания от периферийного устройства, он считывает информацию статуса/идентификации из этого устройства; периферийное устройство интерпретирует этот цикл считывания как указание снять запрос прерывания.

Прерыватели, которые используют методы 1 и 2, называются в настоящем стандарте прерывателями типа RORA (Release On Register Access — снимающими запрос прерывания при обращении к регистру), а прерыватели, использующие метод 3, — прерывателями типа ROAK (Release On AcKnowledge — снимающими запрос при подтверждении прерывания). На рисунке 4.8 показано, как прерыватель ROAK освобождает свою линию запроса прерывания, когда обработчик прерываний считывает его информацию статуса/идентификации, и как прерыватель RORA снимает свой запрос прерывания при обращении к управляющему или статусному регистру.

Замечание 4.5. Исполнитель, обеспечивающий обращение к управляющему или статусному регистру прерывателя, обычно находится на том же модуле, что и прерыватель, и формирует внутримодульный сигнал прерывателю, когда закончит обращение к регистру.

Правило 4.5. *Прерыватель RORA НЕ ДОЛЖЕН ОСВОБОЖДАТЬ свою линию запроса прерывания до тех пор, пока не обнаружит отрицательный перепад DSA* во время цикла обращения к регистру, и ДОЛЖЕН ОСВОБОЖДАТЬ эту линию запроса прерывания в течение 2 мкс после установки последнего строба данных высоким в конце цикла обращения к регистру.*

Правило 4.6. *Прерыватель ROAK НЕ ДОЛЖЕН ОСВОБОЖДАТЬ свою линию запроса прерывания до тех пор, пока не обнаружит отрицательный перепад на линии DSA* во время цикла подтверждения прерывания, подтверждающего его прерывание, и ДОЛЖЕН ОСВОБОЖДАТЬ эту линию запроса прерывания в течение 500 нс после установки высоким последнего строба данных в конце цикла считывания информации статуса/идентификации.*

Правило 4.7. *Прерыватели RORA и ROAK ДОЛЖНЫ ПРЕДОСТАВЛЯТЬ информацию статуса/идентификации во время цикла подтверждения прерывания, который был инициирован в ответ на их запрос прерывания.*

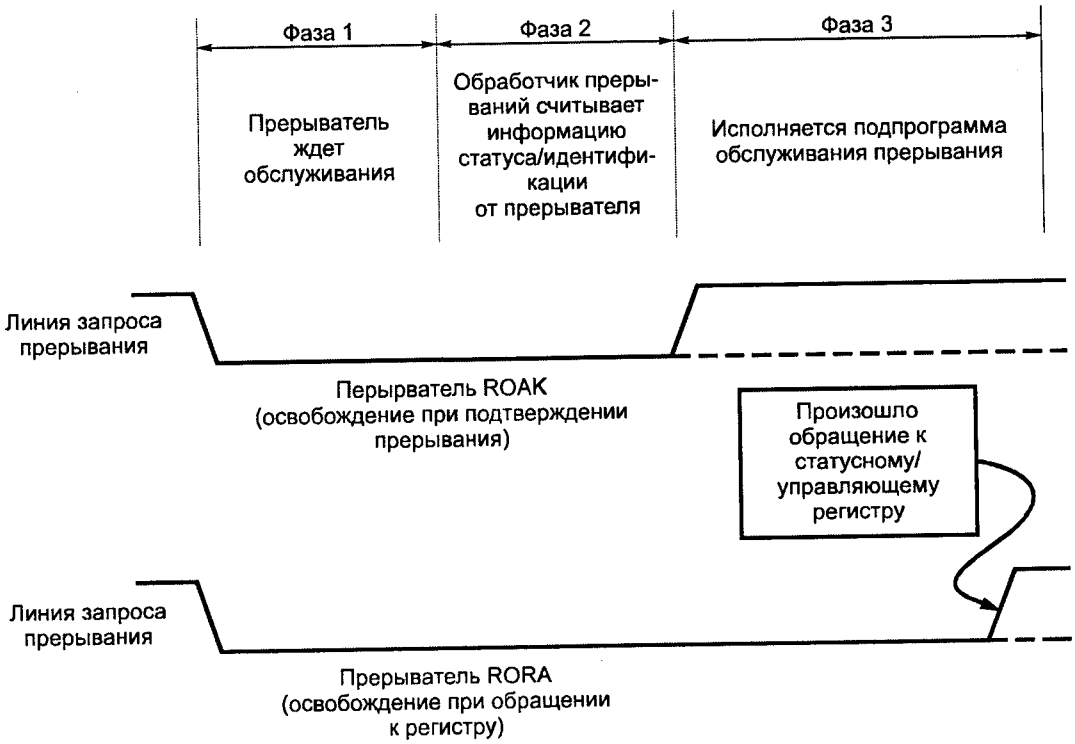


Рисунок 4.8 — Освобождение линий запроса прерывания прерывателями типов ROAK и RORA

Правило 4.8. *После того как обработчик прерываний инициировал цикл подтверждения прерывания и считал информацию статуса/идентификации из прерывателя RORA, он ДОЛЖЕН ИГНОРИРОВАТЬ низкий уровень сигнала на линии запроса прерывания в течение 2 мкс после установки истинным его внутримодульного сигнала «ПРОИЗОШЛО ОБРАЩЕНИЕ К РЕГИСТРУ».*

Замечание 4.6. Правило 4.8 предотвращает ложную интерпретацию обработчиком прерываний сигнала низкого уровня на линии запроса прерывания как нового запроса на этой линии.

Замечание 4.7. Задатчик, осуществляющий обращение к управляющему или статусному регистру прерывателя, обычно находится на том же модуле, что и обработчик прерываний, и формирует внутримодульный сигнал обработчику прерываний, когда закончит обращение к регистру.

Разрешение 4.2. Если предусмотрена процедура, позволяющая задатчику сигнализировать обработчику прерываний об обращении к управляющему или статусному регистрам прерывателя, то задатчик и обработчик прерываний МОГУТ быть расположены в разных модулях.

В таблице 4.6 показано, как мнемонические обозначения RORA и ROAK используются для описания прерывателей.

Таблица 4.6 — Мнемонические обозначения для описания возможностей освобождения линии запроса прерывания

Мнемоническое обозначение	Функциональный блок	Смысловое содержание
RORA	Прерыватель	Освобождает свою линию запроса прерывания, когда какой-либо задатчик обращается к его внутримодульному статусному или управляющему регистру
ROAK	Прерыватель	Освобождает свою линию запроса прерывания, когда считывается его информация статуса/идентификации во время цикла подтверждения прерывания

4.3.8 Взаимодействие между функциональными блоками шины приоритетных прерываний

Ниже рассмотрены несколько внутримодульных сигналов, описывающих взаимодействие между функциональными блоками прерывателей, обработчиков прерываний и другой внутримодульной логикой. Эти сигналы приведены исключительно для того, чтобы проиллюстрировать, какая информация передается между функциональными блоками, а не для описания их конструкции.

Разрешение 4.3. Модули МОГУТ быть спроектированы с внутримодульными сигналами, отличающимися от рассматриваемых в последующем описании.

На рисунке 4.4 показано, как последовательная приоритетная цепочка IACKIN*/IACKOUT* проходит через типичную конфигурацию модулей на магистрали VME.

Линия IACK* проходит по всей длине объединительной платы и может возбуждаться любым обработчиком прерываний, который управляет шиной пересылки данных. Объединительная плата соединяет линию IACK* с контактом IACKIN* гнезда 1. Формирователь цепочки подтверждения прерывания располагается в гнезде 1 и контролирует уровень сигнала на линии IACKIN* гнезда 1.

Когда обработчик прерываний устанавливает IACK* (и IACKIN* гнезда 1) низким, а затем устанавливает низким DSA*, формирователь цепочки подтверждения прерывания формирует отрицательный перепад сигнала на своем контакте IACKOUT*. Этот контакт соединен с контактом IACKIN* гнезда 2. Перемычка на модуле в гнезде 2 направляет отрицательный перепад с контакта IACKIN* на контакт IACKOUT* и через объединительную плату — на контакт IACKIN* модуля в гнезде 3. Прерыватель в гнезде 3 не имеет ждущего обработки запроса прерывания, поэтому он передает отрицательный перепад дальше на свой контакт IACKOUT*. После этого прерыватель в гнезде 4 обнаруживает этот отрицательный перепад на своей линии IACKIN* и отвечает размещением своей информации статуса/идентификации на шину данных с последующей установкой DTACK* низким.

Разрешение 4.4. Прерыватель МОЖЕТ быть расположен на модуле системного контроллера, установленном в гнезде 1, вместе с формирователем цепочки подтверждения прерывания. На рисунке 4.9 показано, как могут быть соединены эти два функциональных блока.

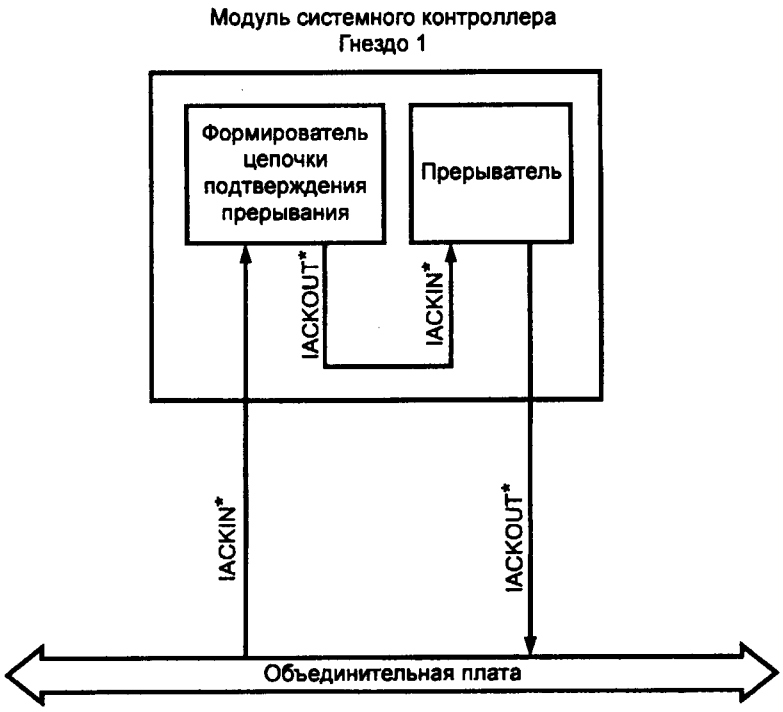


Рисунок 4.9 — Формирователь цепочки подтверждения прерывания и прерыватель в одном и том же модуле

Разрешение 4.5. В модуле МОЖЕТ находиться более чем один прерыватель. На рисунке 4.10 показано, как это можно сделать.

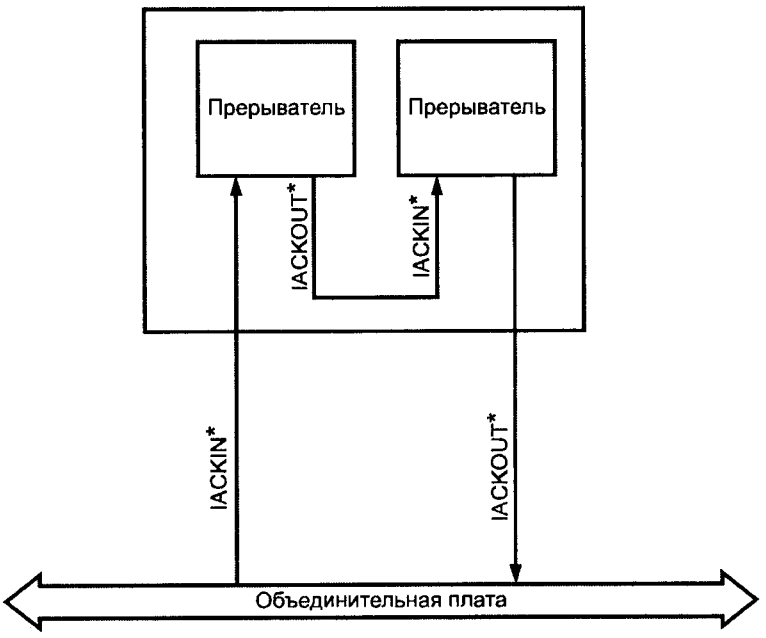


Рисунок 4.10 — Два прерывателя в одном и том же модуле

Замечание 4.8. В некоторых случаях разработчики модулей могут не знать, будет ли модуль, который они проектируют, установлен в гнезде 1 или в каком-либо другом гнезде системы магистрали VME.

Рекомендация 4.1. Если какой-либо модуль содержит как формирователь цепочки подтверждения прерывания, так и прерыватель, и этот модуль может быть установлен или не установлен в гнезде 1, рекомендуется проектировать его, как показано на рисунке 4.9.

Разрешение 4.6. В системе магистрали VME МОЖЕТ быть установлено несколько модулей, содержащих формирователи цепочки подтверждения прерывания.

4.4 Типичные примеры работы

Типичная последовательность прерывания подразделяется на следующие три фазы:

- фаза 1 — фаза запроса прерывания;
- фаза 2 — фаза подтверждения прерывания;
- фаза 3 — фаза обслуживания прерывания.

На рисунке 4.11 показана временна́я последовательность этих трех фаз.

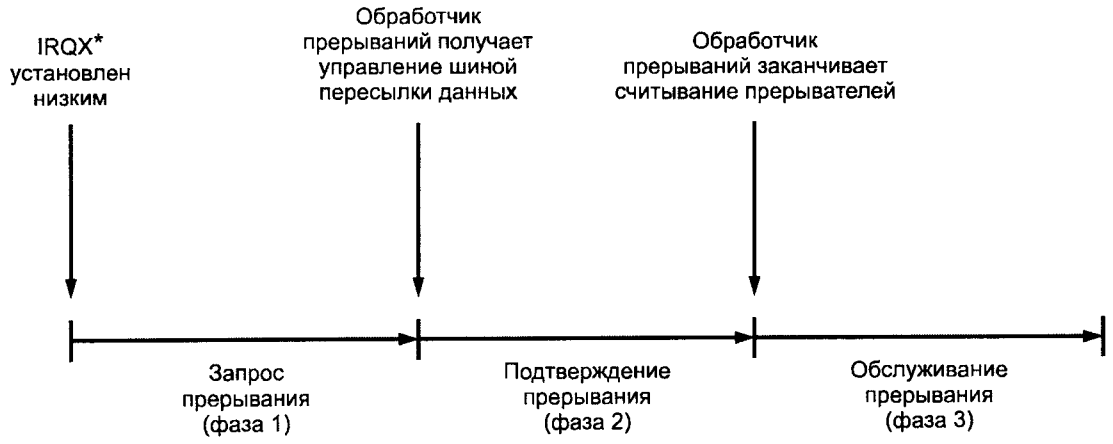


Рисунок 4.11 — Три фазы последовательности обработки прерывания

Фаза 1 начинается, когда прерыватель устанавливает на линии запроса прерывания низкий уровень сигнала, и заканчивается, когда обработчик прерываний получает управление шиной пересылки данных. Во время фазы 2 обработчик прерываний использует шину пересылки данных для считывания информации статуса/идентификации прерывателя. Во время фазы 3 исполняется программа обслуживания прерывания, которая может включать или не включать пересылки данных по магистрали.

Протокол подсистемы прерывания описывает требуемое взаимодействие между модулями во время фаз 1 и 2. Любые пересылки данных, которые происходят во время фазы 3, подчиняются протоколу шины пересылки данных, описанному в разделе 2.

4.4.1 Работа с одним обработчиком прерываний

В системах прерывания с одним обработчиком все семь линий запроса прерывания контролируются одним обработчиком прерываний. Эти линии распределены по приоритету таким образом, что IRQ7* имеет наивысший приоритет, а IRQ1* — низший. Когда обработчик прерываний обнаруживает запросы одновременно на двух линиях, он подтверждает сначала запрос прерывания с более высоким приоритетом.

4.4.2 Работа распределенных систем прерывания

Распределенные системы прерывания содержат от двух до семи обработчиков прерываний. Для ясности дальнейшее описание распределенных систем прерываний подразделено на две части:

- распределенные системы прерывания с семью обработчиками прерываний;
- распределенные системы прерывания с двумя-шестью обработчиками прерываний.

4.4.2.1 *Распределенные системы прерывания с семью обработчиками прерываний*

В распределенных системах прерывания с семью обработчиками прерываний каждая из линий запроса прерывания контролируется отдельным обработчиком прерываний. Каждый обработчик прерываний получает управление шиной пересылки данных прежде, чем считает информацию статуса/идентификации от прерывателя, который устанавливает запрос на его линии запроса прерывания.

Замечание 4.9. Не существует определенной взаимосвязи между линией запроса прерывания, обслуживаемой обработчиком прерываний, и линией запроса шины, используемой его внутримодульным запросчиком. Например, обработчик прерываний, который обслуживает линию IRQ7*, может иметь запросчик, использующий линию BR0*, а обработчик прерываний, обслуживающий линию IRQ1*, может иметь запросчик, использующий линию BR3*. Из сказанного ясно, что между линиями, обслуживаемыми различными обработчиками прерываний, никаких приоритетов прерываний не подразумевается.

На рисунке 4.12 показана распределенная система прерывания, в которой обработчик прерываний А контролирует линию IRQ2* и связан с внутримодульным запросчиком, который запрашивает шину пересылки данных по линии BR2*. Обработчик прерываний В контролирует линию IRQ5* и связан с внутримодульным запросчиком, который запрашивает шину пересылки данных по линии BR3*. Два прерывателя одновременно устанавливают низкими IRQ2* и IRQ5*, й два обработчика прерываний заставляют свои внутримодульные запросчики одновременно установить низкими BR2* и BR3*. В рассматриваемом примере используется приоритетный арбитраж, и, поскольку оба сигнала запроса шины устанавливаются низкими одновременно, арбитр сначала предоставляет управление шиной пересылки данных запросчику обработчика прерываний В, а обработчик прерываний А ждет, пока обработчик прерываний В не закончит использование шины пересылки данных.

Замечание 4.10. Если используется круговой арбитраж, шина пересылки данных может быть предоставлена в первую очередь любому из обработчиков прерываний, показанных на рисунке 4.12.

4.4.2.2 *Распределенные системы прерывания с двумя—шестью обработчиками*

Можно сконфигурировать также распределенную систему прерывания, в которой две (или больше) линии запроса прерывания контролируются одним обработчиком прерываний. На рисунке 4.13 показана система, сконфигурированная с использованием двух обработчиков прерываний. Обработчик прерываний А контролирует линии IRQ1*—IRQ4*, а обработчик прерываний В — линии IRQ5*—IRQ7*. В рассматриваемом случае линии IRQ1*—IRQ4* распределены по приоритетам, причем IRQ4* обладает наивысшим приоритетом для обработчика прерываний А. Линии IRQ5*—IRQ7* также распределены по приоритетам, и IRQ7* обладает наивысшим приоритетом для обработчика прерываний В. Однако вопрос о первоочередности предоставления шины пересылки данных какому-либо из обработчиков прерываний решается в процессе арбитража.

4.4.3 Типичный пример работы системы прерывания с одним обработчиком

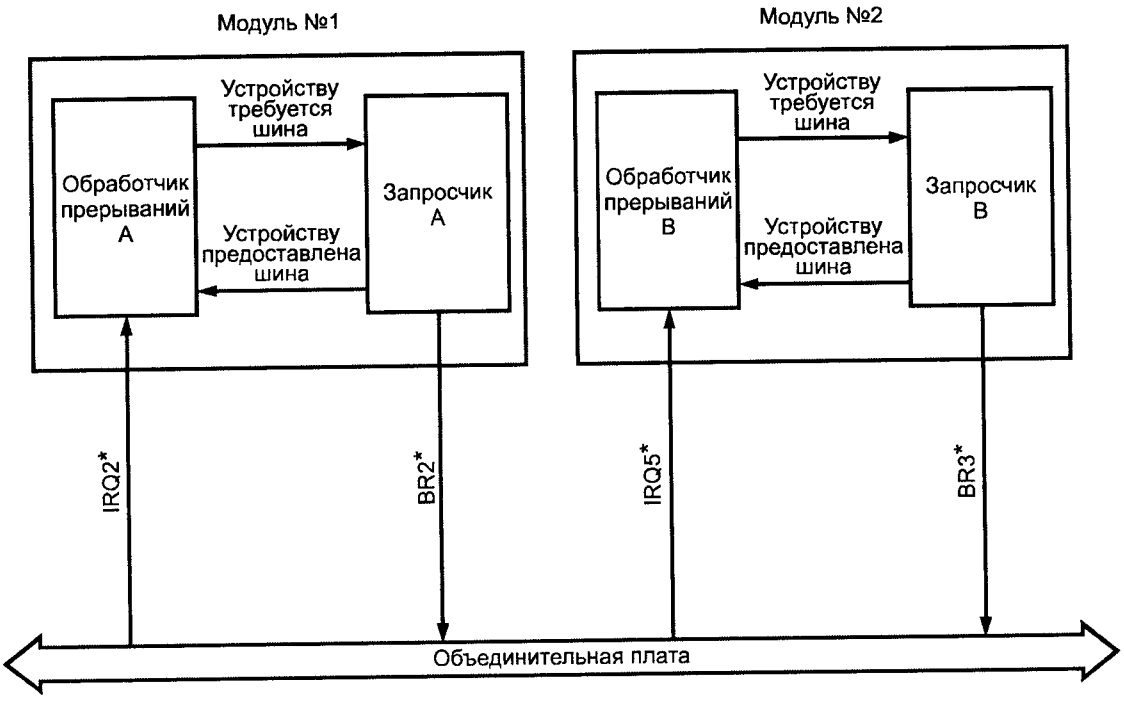


Рисунок 4.12 — Два обработчика прерываний, каждый из которых контролирует одну линию запроса прерывания

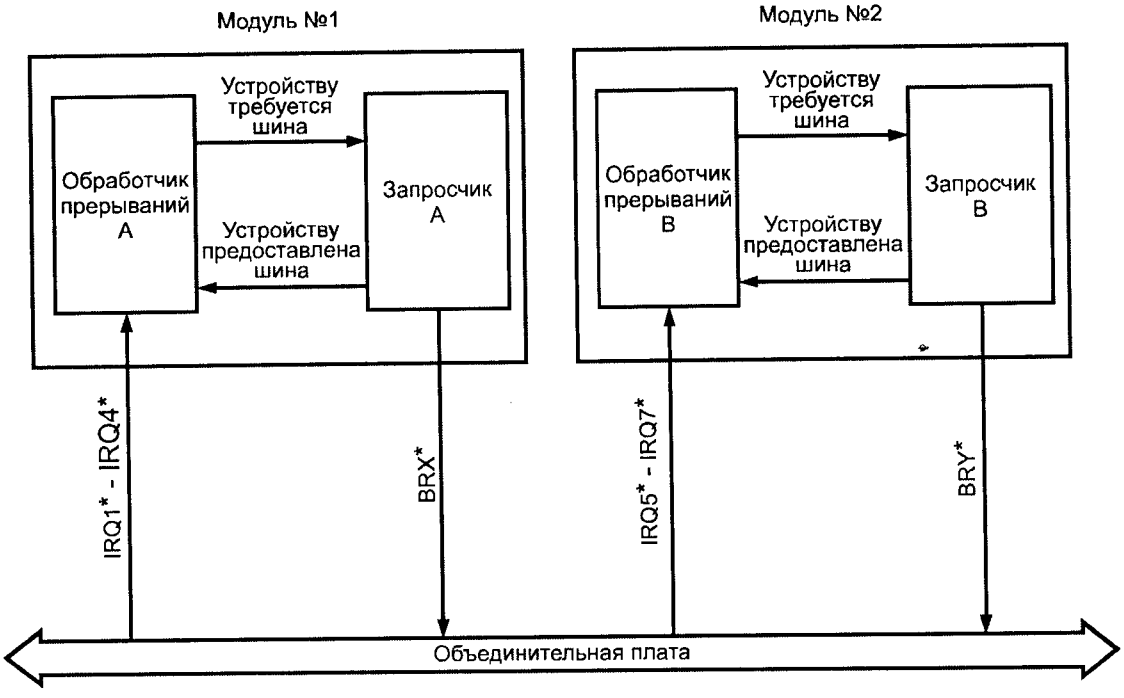


Рисунок 4.13 — Два обработчика прерываний, каждый из которых контролирует несколько линий запроса прерывания

На рисунке 4.14 показан алгоритм работы системы прерывания с одним обработчиком прерываний, который контролирует и распределяет по приоритетам все семь линий запроса прерывания. В начале алгоритма задатчик, запросчику которого была предоставлена шина пересылки данных по запрету линии BR2*, использует шину для пересылки данных. Прерыватель в гнезде 4 запрашивает прерывание установкой низким IRQ4*. Когда обработчик прерываний обнаруживает низкий уровень IRQ4*, он устанавливает истинным сигнал «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА» для своего внутри-

модульного запросчика, показывая, что ему требуется шина. Этот запросчик устанавливает затем низким BR3*. После обнаружения запроса шины арбитр устанавливает низким BCLR*, показывая, что шина пересылки данных требуется запросчику более высокого приоритета (предполагается, что в рассматриваемом примере используется арбитр PRI). Когда задатчик А обнаруживает низкий уровень на линии BCLR*, он останавливает пересылку данных и позволяет своему запросчику уступить управление шиной пересылки данных, освободив линию BBSY* переводом сигнала на ней в высокое состояние.

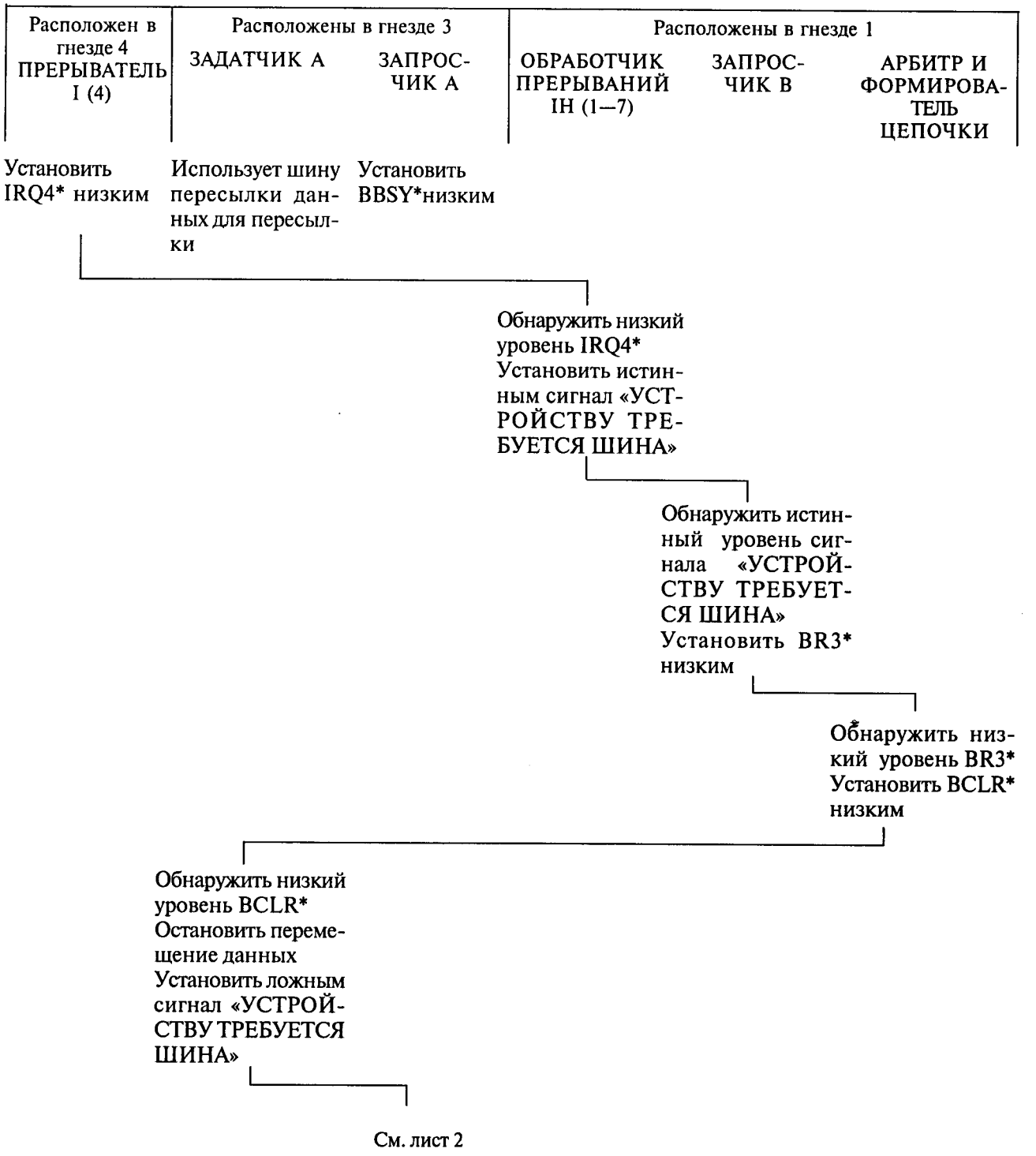


Рисунок 4.14, лист 1 — Алгоритм работы системы прерывания с одним обработчиком

Расположен в гнезде 4	Расположены в гнезде 3		Расположены в гнезде 1		
ПРЕРЫВАТЕЛЬ I (4)	ЗАДАТЧИК А	ЗАПРОСЧИК А	ОБРАБОТЧИК ПРЕРЫВАНИЙ IH (1—7)	ЗАПРОСЧИК В	АРБИТР И ФОРМИРОВАТЕЛЬ ЦЕПОЧКИ

Обнаружить ложным сигнал «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА»
Освободить BBSY*

Обнаружить высокий уровень BBSY*
Установить BG3IN* низким

Обнаружить низкий уровень BG3IN*
Установить низким BBSY*
Установить истинным сигнал «УСТРОЙСТВУ ПРЕДОСТАВЛЕНА ШИНА»

Обнаружить истинный уровень сигнала «УСТРОЙСТВУ ПРЕДОСТАВЛЕНА ШИНА»
Поместить трехразрядный код на линии A01—A03
Установить IACK* низким
Установить AS* низким
Установить низкими строб (стробы) данных

См. лист 3

Рисунок 4.14, лист 2

Расположен в гнезде 4	Расположены в гнезде 3		Расположены в гнезде 1		
ПРЕРЫВАТЕЛЬ I (4)	ЗАДАТЧИК А	ЗАПРОСЧИК А	ОБРАБОТЧИК ПРЕРЫВАНИЙ IH (1—7)	ЗАПРОСЧИК В	АРБИТР И ФОРМИРОВАТЕЛЬ ЦЕПОЧКИ

Обнаружить низкими IACK* и DSA*
Установить IACKOUT* низким

Распространение сигнала по цепочке IACKIN*/IACKOUT*

Обнаружить низкий уровень IACKIN*
Проверить трехразрядный код на A01—A03
Обнаружить низкими стробы данных
Поместить информацию статуса/идентификации на шину данных
Установить DTACK* низким

Обнаружить низкий уровень DTACK*
Считать информацию статуса/идентификации
Инициировать исполнение последовательности обслуживания прерывания

Рисунок 4.14, лист 3

Замечание 4.11. Активному задатчику не требуется уступать шину пересылки данных в течение какого-то определенного времени, однако быстрый ответ на сигнал BCLR* позволяет быстрее обслужить прерывание.

Когда арбитр обнаруживает BBSY* высоким, он предоставляет шину пересылки данных запросчику В, который информирует свой обработчик прерываний, что шина находится в его распоряжении (см. рисунок 2.26). После этого обработчик прерываний помещает трехразрядный код на адресных линиях A01—A03, чтобы показать, что он подтверждает запрос прерывания по линии IRQ4* (см. таблицу 4.7), и устанавливает IACK* низким, чтобы показать, что он подтверждает прерывание, а

также устанавливает низким AS*. Низкий уровень IACK* поступает по сигнальному проводнику на объединительной плате на контакт IACKIN* гнезда 1 и заставляет формирователь цепочки подтверждения прерывания сформировать отрицательный перепад, распространяющийся по цепочке IACKIN*/IACKOUT*.

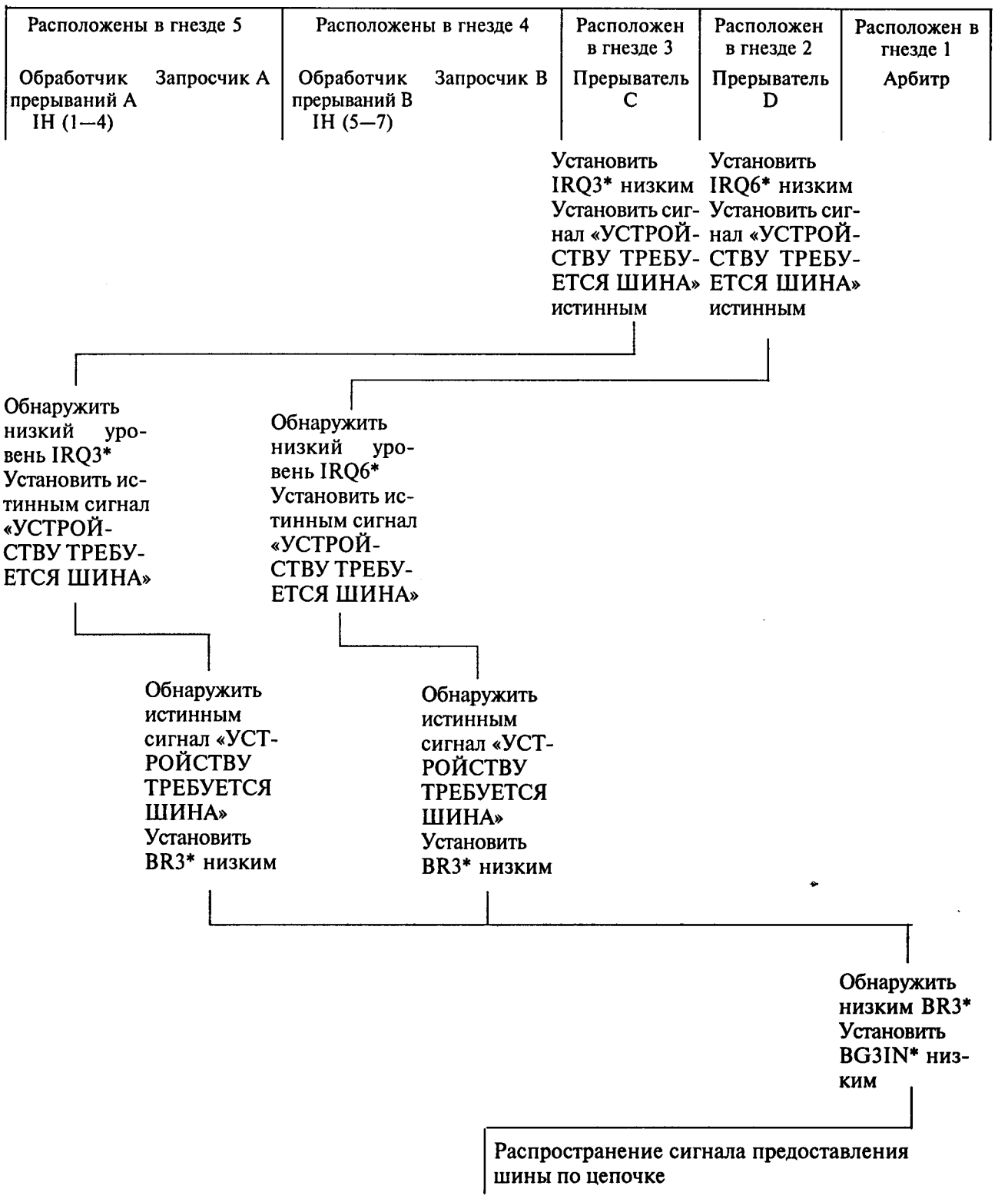
Когда прерыватель обнаруживает отрицательный перепад на своем входе IACKIN*, он проверяет уровни сигналов на линиях A01—A03 на соответствие линии запроса прерывания, которую он установил низкой. Поскольку трехразрядный код соответствует номеру линии, на которой он установил свой запрос прерывания, прерыватель помещает свою информацию статуса/идентификации на шину данных и устанавливает DTACK* низким после обнаружения строба (стробов) данных низкими. Когда обработчик прерываний обнаружит DTACK* низким, он считывает информацию статуса/идентификации и инициирует исполнение соответствующей программы обработки прерывания.

Таблица 4.7 — Трехразрядный код подтверждения прерывания

Подтверждаемая линия запроса прерывателя	Использование адресных линий для широковещательной передачи трехразрядного кода подтверждения прерывания		
	A03	A02	A01
IRQ1*	L	L	H
IRQ2*	L	H	L
IRQ3*	L	H	H
IRQ4*	H	L	L
IRQ5*	H	L	H
IRQ6*	H	H	L
IRQ7*	H	H	H
H — высокий уровень, L — низкий уровень			

4.4.4 Пример назначения приоритетов двум прерываниям в распределенной системе прерывания

На рисунке 4.15 показан алгоритм работы распределенной системы прерывания с двумя обработчиками прерываний. Обработчик прерываний A контролирует линии IRQ1* — IRQ4*, обработчик прерываний B контролирует линии IRQ5* — IRQ7*. В качестве источника прерывания наивысшего приоритета обработчик прерываний A рассматривает IRQ4*, а обработчик прерываний B — IRQ7*. В начале алгоритма прерыватель C устанавливает низким IRQ3*, а прерыватель D устанавливает низким IRQ6*. Оба обработчика прерываний обнаруживают низкие уровни сигналов на своих соответствующих линиях запроса прерывания и оба одновременно сигнализируют своим внутримодульным запросчикам, что им требуется шина пересылки данных. Оба запросчика устанавливают низким BR3*. После обнаружения BR3* низким арбитр шины устанавливает низким BG3IN* в гнезде 1. Этот отрицательный перепад распространяется по цепочке BG3IN*/BG3OUT*, пока не будет обнаружен запросчиком B в гнезде 4. Этот запросчик сигнализирует затем своему внутримодульному обработчику прерываний B, что шина пересылки данных находится в его распоряжении. После этого обработчик прерываний B считывает информацию статуса/идентификации из прерывателя D.



См. лист 2

Рисунок 4.15, лист 1 — Алгоритм работы распределенной системы прерывания с двумя обработчиками прерываний

Расположены в гнезде 5		Расположены в гнезде 4		Расположен в гнезде 3	Расположен в гнезде 2	Расположен в гнезде 1
Обработчик прерываний А IH (1—4)	Запросчик А	Обработчик прерываний В IH (5—7)	Запросчик В	Прерыватель С	Прерыватель D	Арбитр

Обнаружить низким BG3IN*
Установить BBSY* низким
Установить истинным сигнал «УСТРОЙСТВУ ПРЕДОСТАВЛЕНА ШИНА»

Обнаружить истинным сигнал «УСТРОЙСТВУ ПРЕДОСТАВЛЕНА ШИНА»

и т. д.

Рисунок 4.15, лист 2

4.5 Условия возникновения гонок

Предположим, что имеются два прерывателя: А и В. Прерыватель В, расположенный дальше по цепочке подтверждения прерывания, запрашивает прерывание. После того, как соответствующему обработчику прерываний будет предоставлена шина, он подтверждает этот запрос прерывания, устанавливая низким IACK*. В результате отрицательный перепад по цепочке подтверждения прерывания достигает прерывателя А как раз к моменту, когда он близок к установлению низкого уровня на своей собственной линии запроса прерывания. Если прерыватель А спроектирован неправильно, он может в этой ситуации установить на короткое время низким свой IACKOUT*, а затем снова высоким, приводя к возникновению отрицательного импульса в цепочке подтверждения прерывания.

Правило 4.49. *Прерыватели ДОЛЖНЫ БЫТЬ СПРОЕКТИРОВАНЫ таким образом, чтобы исключить возможность появления кратковременных отрицательных импульсов на их линии IACKOUT*.*

Замечание 4.50. Если прерыватель спроектирован таким образом, что он фиксирует состояние внутримодульного сигнала запроса прерывания по отрицательному перепаду сигнала на своей линии IACKIN*, и если внутримодульный сигнал находится в состоянии перехода, когда возникает указанный отрицательный перепад, то сигналы на выходах фиксатора в течение короткого времени будут в некоторых случаях генерировать либо оставаться в пороговой области между высоким и низким уровнями. Поэтому на прерыватель не накладывается никаких временных ограничений по передаче дальше сигнала подтверждения прерывания. Прерывателю только запрещено генерировать отрицательные импульсы на своей линии IACKOUT*, которые могли бы быть интерпретированы как подтверждение прерывателем, расположенным дальше по цепочке. В приложении D приведена примерная схема логики последовательной цепочки подтверждения прерывания.

Разрешение 4.11. Если прерыватель близок к тому, чтобы установить низким уровень сигнала на линии запроса прерывания между моментом времени, когда он принимает подтверждение прерывания, предназначенное для другого прерывателя, и моментом, когда он пропустит это подтверждение прерывания дальше, он МОЖЕТ рассматривать это подтверждение прерывания как свое собственное. В этом случае другой прерыватель продолжает удерживать свой запрос прерывания до тех пор, пока не будет сформировано еще одно подтверждение прерывания.

4.6 Правила и замечания по временны́м соотношениям сигналов шины приоритетных прерываний

В данном подразделе приведены правила и замечания по временным соотношениям, которые определяют поведение обработчиков прерываний, прерывателей и формирователей цепочки подтверждения прерывания во время выбора отвечающего прерывателя, то есть прерывателя, который должен представить свою информацию статуса/идентификации в ответ на цикл подтверждения прерывания. Эта информация по временны́м соотношениям приведена в виде рисунков и таблиц.

Цикл подтверждения прерывания начинается с выбора отвечающего прерывателя. Эта часть цикла называется фазой выбора прерывателя. Как только прерыватель ответит, обработчик прерываний считывает его информацию статуса/идентификации. Эта часть цикла называется фазой пересылки информации статуса/идентификации.

Когда обработчик прерываний инициирует цикл подтверждения прерывания, между ним и прерывателем, для которого формируется подтверждение, могут быть прерыватели, которые либо не имеют ожидающего обработки запроса прерывания, либо имеют ожидающий обработки запрос прерывания, но на линии запроса прерывания, отличной от той, для которой формируется данное подтверждение.

Хотя эти прерыватели не отвечают информацией статуса/идентификации, они участвуют в цикле подтверждения прерывания, пропуская отрицательный перепад сигнала подтверждения со своей линии IACKIN* на свою линию IACKOUT*. Поэтому такие прерыватели называются участвующими прерывателями.

Первый прерыватель в цепочке, который имеет ожидающий обработки запрос на той линии запроса прерывания, которая подтверждается, отвечает информацией статуса/идентификации. Поэтому он называется отвечающим прерывателем.

Все другие прерыватели называются неучаствующими прерывателями.

Таблица 4.8 содержит перечень временных диаграмм, которые определяют работу обработчика прерываний и прерывателя.

Таблица 4.9 содержит перечень временны́х диаграмм, которые определяют работу формирователя цепочки подтверждения прерывания.

Таблица 4.10 содержит перечень временны́х диаграмм, которые определяют работу участвующего прерывателя.

Таблица 4.11 содержит перечень временны́х диаграмм, которые определяют работу отвечающего прерывателя.

Таблица 4.12 содержит определения мнемонических обозначений, используемых в таблицах 4.13—4.15.

Таблицы 4.13—4.15 определяют использование сигнальных линий магистрали функциональными блоками шины приоритетных прерываний.

Таблицы 4.16—4.19 определяют значения параметров временны́х соотношений сигналов для функциональных блоков шины приоритетных прерываний (ссылочные номера, используемые в таблицах 4.17—4.19, соответствуют номерам временны́х параметров в таблице 4.16).

Таблица 4.8 — Перечень временны́х диаграмм, определяющих работу обработчика прерываний и прерывателя

Мнемоническое обозначение	Тип цикла	Номер рисунка временно́й диаграммы	
		выбора прерывателя	пересылки информации статуса/ идентификации
D08(O)	Однобайтовое считывание информации статуса/идентификации	2.12, 4.16	4.20
D16	Двухбайтовое считывание информации статуса/идентификации	2.12, 4.16	4.21
D32	Четырехбайтовое считывание информации статуса/идентификации	2.12, 4.16	4.21

Таблица 4.9 — Перечень временны́х диаграмм, определяющих работу формирователя цепочки подтверждения прерывания

Тип цикла	Номер рисунка временно́й диаграммы выбора прерывателя
Однобайтовое считывание информации статуса/идентификации	4.17
Двухбайтовое считывание информации статуса/идентификации	4.17
Четырехбайтовое считывание информации статуса/идентификации	4.17

Таблица 4.10 — Перечень временны́х диаграмм, определяющих работу участвующего прерывателя

Тип цикла	Номер рисунка временно́й диаграммы выбора прерывателя
Однобайтовое считывание информации статуса/идентификации	4.18
Двухбайтовое считывание информации статуса/идентификации	4.18
Четырехбайтовое считывание информации статуса/идентификации	4.18

Таблица 4.11 — Перечень временны́х диаграмм, определяющих работу отвечающего прерывателя

Мнемоническое обозначение	Тип цикла	Номер рисунка временно́й диаграммы	
		выбора прерывателя	пересылки информации статуса/ идентификации
D08(O)	Однобайтовое считывание информации статуса/идентификации	4.19	4.22
D16	Двухбайтовое считывание информации статуса/идентификации	4.19	4.23
D32	Четырехбайтовое считывание информации статуса/идентификации	4.19	4.23

Таблица 4.12 — Определение мнемонических обозначений, используемых в таблицах 4.13—4.15

Мнемоническое обозначение	Описание	Комментарий
DLBIH	Driven Low By Interrupt Handler — устанавливаются обработчиком прерываний низкими	***Правило 4.10.*** *Обработчик прерываний ДОЛЖЕН УСТАНАВЛИВАТЬ на линиях, обозначаемых DLBIH, низкие уровни сигналов*
DHBIH	Driven High By Interrupt Handler— устанавливаются обработчиком прерываний высокими	***Правило 4.11.*** *Обработчик прерываний ДОЛЖЕН УСТАНАВЛИВАТЬ на линиях, обозначаемых DHBIH, высокие уровни сигналов*

Окончание таблицы 4.12

Мнемоническое обозначение	Описание	Комментарий
dhbih?	Driven High By Interrupt Handler?— устанавливаются обработчиком прерываний высокими?	**Разрешение 4.7.** Обработчик прерываний МОЖЕТ установить на линиях, обозначаемых dhbih?, высокие уровни сигналов ***Правило 4.12.*** *Обработчик прерываний во время цикла НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ низкие уровни сигналов на линиях, обозначаемых dhbih?*
DVBI	Driven Valid By Interrupter — устанавливаются прерывателем достоверными	***Правило 4.13.*** *Прерыватель ДОЛЖЕН УСТАНАВЛИВАТЬ на линиях, обозначаемых DVBI, достоверные уровни сигналов*
dhbi?	Driven High By Interrupter ?— устанавливаются прерывателем высокими?	**Разрешение 4.9.** Прерыватель МОЖЕТ установить на линиях, обозначаемых dhbi?, высокие уровни сигналов ***Правило 4.14.*** *Прерыватель во время цикла НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ низкие уровни на линиях, обозначаемых dhbi?*

Таблица 4.13 — Использование линий A01—A03 и IACK* во время циклов подтверждения прерывания

Подтверждаемая линия запроса прерывания	A03	A02	A01	IACK*
IRQ1*	DLBIH	DLBIH	DHBIH	DLBIH
IRQ2*	DLBIH	DHBIH	DLBIH	DLBIH
IRQ3*	DLBIH	DHBIH	DHBIH	DLBIH
IRQ4*	DHBIH	DLBIH	DLBIH	DLBIH
IRQ5*	DHBIH	DLBIH	DHBIH	DLBIH
IRQ6*	DHBIH	DHBIH	DLBIH	DLBIH
IRQ7*	DHBIH	DHBIH	DHBIH	DLBIH

Таблица 4.14 — Использование линий DS1*, DS0*, LWORD* и WRITE* во время циклов подтверждения прерывания

Мнемоническое обозначение	Тип цикла	DS1*	DS0*	LWORD*	WRITE*
D08(O)	Однобайтовый цикл подтверждения прерывания	dhbih?	DLBIH	dhbih?	dhbih?
D16	Двухбайтовый цикл подтверждения прерывания	DLBIH	DLBIH	dhbih?	dhbih?
D32	Четырехбайтовый цикл подтверждения прерывания	DLBIH	DLBIH	DLBIH	dhbih?

Таблица 4.15 — Использование линий данных D00—D31 для пересылки информации статуса/идентификации

Мнемоническое обозначение	Тип цикла	D24—D31	D16—D23	D08—D15	D00—D07
D08(O)	Одно-, двух- и четырехбайтовые циклы подтверждения прерывания	dhbi?	dhbi?	dhbi?	DVBI
D16	Двух- и четырехбайтовые циклы подтверждения прерывания	dhbi?	dhbi?	DVBI	DVBI
D32	Четырехбайтовые циклы подтверждения прерывания	DVBI	DVBI	DVBI	DVBI

Таблица 4.16 — Значения параметров временных соотношений для обработчиков прерываний, прерывателей и формирователей цепочки подтверждения прерывания

Номер параметра	Значение параметра, нс					
	Обработчик прерываний (см. таблицу 4.17)		Прерыватель (см. таблицу 4.18)		Формирователь цепочки подтверждения прерывания (см. таблицу 4.19)	
	мин.	макс.	мин.	макс.	мин.	макс.
1	0	—	—	—	—	—
2	0	—	—	—	—	—
3	60	—	—	—	—	—
4	35	—	10	—	—	—
5	40	—	30	—	30	—
6	—	—	0	—	—	—
7	—	—	0	—	—	—
9	0	—	0	—	—	—
10	0	—	—10	—	—	—
11	40	—	30	—	—	—
12	35	—	10	—	—	—
13	—	10	—	20	—	—
14	0	—	0	—	—	—
16	0	—	0	—	—	—
18	0	—	0	—	—	—
19	40	—	30	—	30	—
20	0	—	0	—	—	—
21	0	—	0	—	—	—
23	10	—	0	—	—	—
24A	0	—	—	—	—	—
24B	0	—	—	—	—	—
25	—	25	—	—	—	—
26	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2T	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
31	0	—	0	—	—	—
32	—	—	10	—	10	—
34	—	—	30	—	40	—
35	—	—	0	30	0	30
36	—	—	0	—	—	—
37	—	—	0	—	—	—

Окончание таблицы 4.16

Номер параметра	Значение параметра, нс					
	Обработчик прерываний (см. таблицу 4.17)		Прерыватель (см. таблицу 4.18)		Формирователь цепочки подтверждения прерывания (см. таблицу 4.19)	
	мин.	макс.	мин.	макс.	мин.	макс.
38A	—	—	0	—	—	—
38B	—	—	0	—	—	—
39	—	—	—	40	—	—
40	—	—	30	—	30	—
41	—	—	0	—	—	—
42	—	—	—	—	30	—
43	—	—	0	—	—	—
Примечания — Т — значение тайм-аута в микросекундах.						

Таблица 4.17 — Обработчик прерываний. Правила и замечания по временны́м соотношениям

Номер параметра (см. таблицу 4.16)	Правила и замечания
1	*Правило 4.15. При получении управления магистралью VME обработчик прерываний НЕ ДОЛЖЕН ВОЗБУЖДАТЬ ни одну из линий IACK*, A01—A03, LWORD*, WRITE* DS0*, DS1* или AS* до тех пор, пока предыдущий задатчик или обработчик прерываний не позволит сигналу на линии AS* превысить напряжение низкого уровня (см.раздел 2, рисунок 2.2б)* **Замечание 4.12.** Предоставление магистрали запросчику обработчика прерываний описано в разделе 3
2	***Правило 4.16.*** *При получении управления магистралью VME обработчик прерываний НЕ ДОЛЖЕН ВОЗБУЖДАТЬ ни одну из линий IACK*, A01—A03, LWORD*, WRITE*, DS0*, DS1* или AS* до тех пор, пока не убедится, что внутримодульный сигнал «УСТРОЙСТВУ ПРЕДОСТАВЛЕНА ШИНА» установился истинным* **Замечание 4.13.** Предоставление магистрали запросчику обработчика прерываний описано в разделе 3
3	***Правило 4.17.*** *При получении управления магистралью VME обработчик прерываний НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ AS* низким, пока не истечет указанное время, после того как предыдущий задатчик или обработчик прерываний позволит сигналу AS* превысить напряжение низкого уровня* **Замечание 4.14.** Правило 4.17 гарантирует прерывателям и исполнителям соблюдение временного параметра 5, когда происходит смена управления шиной пересылки данных
4	***Правило 4.18.*** *Обработчик прерываний НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ AS* низким, пока IACK* не будет выдержан низким, а LWORD* и A01—A03 — достоверными в течение указанного минимального времени*
5	***Правило 4.19.*** *При использовании шины пересылки данных в течение двух последовательных циклов обработчик прерываний НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ AS* низким, пока этот сигнал не будет выдержан высоким в течение указанного минимального времени*
9	***Правило 4.20.*** *Обработчик прерываний НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DSA* низким, пока оба сигнала DTACK* и BERR* не установятся высокими.*

Продолжение таблицы 4.17

Номер параметра (см. таблицу 4.16)	Правила и замечания
10	***Правило 4.21.*** *Обработчик прерываний НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DSA* низким, пока не установит низким AS**
11	***Правило 4.22.*** *Обработчик прерываний НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DSA* низким, пока оба сигнала DS0* и DS1* не будут выдержаны одновременно высокими в течение указанного минимального времени*
12	***Правило 4.23.*** *Обработчик прерываний НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DSA* низким, пока WRITE* не будет выдержан высоким в течение указанного минимального времени*
13	***Правило 4.24.*** *Во время двух- и четырехбайтовых циклов подтверждения прерывания обработчик прерываний ДОЛЖЕН УСТАНАВЛИВАТЬ DSB* низким в пределах указанного максимального времени с момента, когда он установит низким DSA** **Замечание 4.15.** Временно́й параметр 13 не относится к однобайтовым считываниям информации статуса/идентификации
14	***Правило 4.25.*** *Во время всех циклов подтверждения прерывания обработчик прерываний ДОЛЖЕН СОХРАНЯТЬ достоверное значение трехразрядного кода подтверждения прерывания на линиях A01—A03 и ДОЛЖЕН УДЕРЖИВАТЬ соответствующий уровень LWORD*, пока не обнаружит отрицательный перепад DTACK* или BERR**
16	***Правило 4.26.*** *Во время всех циклов подтверждения прерывания обработчик прерываний ДОЛЖЕН УДЕРЖИВАТЬ IACK* низким, пока не обнаружит отрицательный перепад DTACK* или BERR**
18	***Правило 4.27.*** *Обработчик прерываний ДОЛЖЕН УДЕРЖИВАТЬ AS* низким, пока не обнаружит DTACK* или BERR* низким*
19	***Правило 4.28.*** *Обработчик прерываний ДОЛЖЕН УДЕРЖИВАТЬ AS* низким в течение указанного минимального времени*
20	***Правило 4.29.*** *Установив DSA* низким, обработчик прерываний ДОЛЖЕН УДЕРЖИВАТЬ его в этом состоянии, пока не обнаружит DTACK* или BERR* низким*
21	***Правило 4.30.*** *Установив DSB* низким, обработчик прерываний ДОЛЖЕН УДЕРЖИВАТЬ его в этом состоянии, пока не обнаружит DTACK* или BERR* низким*
23	***Правило 4.31.*** *Установив DSA* низким, обработчик прерываний ДОЛЖЕН УДЕРЖИВАТЬ WRITE* высоким в течение указанного минимального времени после того, как он установит DSB* высоким*
24A	***Правило 4.32.*** *Если обработчик прерываний устанавливает AS* высоким, либо освобождает линию этого сигнала после того, как его запросчик освободит BBSY*, он ДОЛЖЕН ОСВОБОДИТЬ линии IACK*, A01—A03, LWORD*, WRITE*, DS0* и DS1* прежде, чем позволить сигналу на линии AS* превысить напряжение низкого уровня*
24B	***Правило 4.33.*** *Если обработчик прерываний устанавливает AS* высоким, либо освобождает линию этого сигнала, прежде чем его запросчик освободит BBSY*, он ДОЛЖЕН ОСВОБОДИТЬ линии IACK*, A01—A03, LWORD*, WRITE*, DS0* и DS1* прежде, чем изменить значение своего сигнала «УСТРОЙСТВУ ТРЕБУЕТСЯ ШИНА» с истинного на ложное*
25	***Правило 4.34.*** *Если обработчик прерываний устанавливает AS* высоким, либо освобождает линию этого сигнала после освобождения его запросчиком линии BBSY*, он ДОЛЖЕН ОСВОБОДИТЬ линию AS* в пределах указанного времени с момента, когда он позволит сигналу на этой линии превысить напряжение низкого уровня*

Окончание таблицы 4.17

Номер параметра (см. таблицу 4.16)	Правила и замечания
26	**Замечание 4.16.** Временнóй параметр 26 гарантирует, что шина данных не будет возбуждена, пока обработчик прерываний не установит DSA* низким
27	**Замечание 4.17.** Обработчику прерывания гарантируется, что данные на шине данных будут удерживаться достоверными в пределах указанного времени с момента установки DTACK* низким. Это время не относится к циклам, в которых прерыватель устанавливает низким BERR*, а не DTACK*
28	**Замечание 4.18.** Обработчику прерываний гарантируется, что ни DTACK*, ни BERR* не будут установлены низкими до истечения указанного минимального времени с момента, когда он установит низким DSA*. Шинный таймер гарантирует обработчику прерываний, что он установит BERR* низким, если DTACK* не установится низким по истечении тайм-аута и в пределах удвоенного периода тайм-аута этого обработчика
29	**Замечание 4.19.** Обработчику прерываний гарантируется, что данные на шине данных будут удерживаться достоверными, пока он не установит DSA* высоким
30	**Замечание 4.20.** Данный временной параметр гарантирует, что ни DTACK*, ни BERR* не установятся высокими, пока обработчик прерываний не установит высокими оба сигнала DS0* и DS1*
31	**Замечание 4.21.** Обработчику прерываний гарантируется, что шина данных будет освобождена к моменту, когда DTACK* и BERR* установятся высокими

Таблица 4.18 — Прерыватель. Правила и замечания по временным соотношениям

Номер параметра (см. таблицу 4.16)	Правила и замечания
4	**Замечание 4.22.** Прерывателям гарантируется, что на линиях IACK*, LWORD* и A01—A03 будут удерживаться достоверные значения в течение указанного минимального времени до момента, когда они обнаружат отрицательный перепад сигнала на линии AS*
5	**Замечание 4.23.** Всем прерывателям гарантируется указанное минимальное время удержания высокого AS* между циклами шины пересылки данных
6	**Замечание 4.24.** Отвечающему прерывателю гарантируется, что ни одна из линий D00—D31 не будет возбуждаться никаким другим функциональным блоком, пока этот отвечающий прерыватель не освободит DTACK* и BERR*
7	**Замечание 4.25.** Отвечающему прерывателю гарантируется, что линии D00—D31 будут освобождены всеми другими функциональными блоками к моменту установки низким DSA*
9	**Замечание 4.26.** Отвечающему прерывателю гарантируется, что ни DS0*, ни DS1* не установятся низкими до установки высокими обоих сигналов DTACK* и BERR* из предыдущего цикла
11	**Замечание 4.27.** Прерывателям гарантируется указанное минимальное время одновременного удержания высокими обоих сигналов DS0* и DS1* между циклами

Продолжение таблицы 4.18

Номер параметра (см. таблицу 4.16)	Правила и замечания
12	**Замечание 4.28.** Прерывателям гарантируется, что WRITE* будет выдержан высоким в течение указанного минимального времени до момента, когда они обнаружат отрицательный перепад DSA*
13	**Замечание 4.29.** Если обработчик прерываний устанавливает низкими оба сигнала DS0* и DS1*, отвечающему прерывателю гарантируется, что DSB* установится низким в пределах указанного максимального времени, после того как это сделает DSA*. Поэтому, если DSB* не установится низким в пределах этого максимального времени, отвечающий прерыватель предполагает, что он должен ответить однобайтовой информацией статуса/идентификации
14	**Замечание 4.30.** Отвечающему прерывателю гарантируется, что на линиях LWORD* и A01—A03 будут удерживаться достоверные уровни сигналов до тех пор, пока он не установит низким DTACK* или BERR* в пределах тайм-аута шины
16	**Замечание 4.31.** Отвечающему прерывателю гарантируется, что IACK* будет удерживаться низким до тех пор, пока он не установит низким DTACK* или BERR* в пределах тайм-аута шины
18	**Замечание 4.32.** Отвечающему прерывателю гарантируется, что AS* будет удерживаться низким до тех пор, пока он не установит низким DTACK* или BERR* в пределах тайм-аута шины
19	**Замечание 4.33.** Прерывателям гарантируется, что AS* будет удерживаться низким в течение указанного минимального времени
20	**Замечание 4.34.** Отвечающему прерывателю гарантируется, что как только DSA* установится низким, он будет удерживаться в этом состоянии до установки этим прерывателем низким DTACK* или BERR* в пределах тайм-аута шины
21	**Замечание 4.35.** Отвечающему прерывателю гарантируется, что как только DSB* установится низким, он будет удерживаться в этом состоянии до установки этим прерывателем низким DTACK* или BERR* в пределах тайм-аута шины
23	**Замечание 4.36.** Прерывателям гарантируется, что WRITE* будет удерживаться высоким, пока оба сигнала DS0* и DS1* не установятся высокими
26	***Правило 4.35.*** *Отвечающий прерыватель НЕ ДОЛЖЕН ВОЗБУЖДАТЬ ни одну из линий D00—D31, пока DSA* не установится низким*
27	***Правило 4.36.*** *Отвечающий прерыватель НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ DTACK* низким до установления им достоверной информации статуса/идентификации на линиях данных* **Замечание 4.37.** Это время не относится к циклам, в которых отвечающий прерыватель устанавливает низким BERR*, а не DTACK*
28	***Правило 4.37.*** *Прежде чем установить низким DTACK* или BERR*, отвечающий прерыватель ДОЛЖЕН ОЖИДАТЬ в течение указанного минимального времени после установки DSA* низким*
29	***Правило 4.38.*** *Установив DTACK* низким, отвечающий прерыватель НЕ ДОЛЖЕН ИЗМЕНЯТЬ состояние линий D00—D31, пока DSA* не установится высоким*

Окончание таблицы 4.18

Номер параметра (см. таблицу 4.16)	Правила и замечания
30	***Правило 4.39.*** *Установив DTACK* или BERR* низким, отвечающий прерыватель НЕ ДОЛЖЕН ОСВОБОЖДАТЬ линию этого сигнала, пока не обнаружит высокими оба сигнала DS0* и DS1**
31	***Правило 4.40.*** *Отвечающий прерыватель ДОЛЖЕН ОСВОБОДИТЬ все линии D00—D31 до освобождения DTACK* и BERR**
32	**Замечание 4.38.** Отвечающему прерывателю гарантируется, что на линиях IACK*, LWORD* и A01—A03 будут удерживаться достоверные уровни сигналов в течение указанного минимального времени к тому моменту, когда он обнаружит отрицательный перепад DSA*. Это время получается из значений временных параметров 4 и 10
34	**Замечание 4.39.** Прерывателю гарантируется, что DSA* будет удерживаться низким в течение указанного минимального времени к тому моменту, когда он обнаружит отрицательный перепада IACKIN*
35	***Правило 4.41.*** *Участвующий прерыватель ДОЛЖЕН УСТАНОВИТЬ свой IACKOUT* высоким в течение указанного максимального времени после возникновения положительного перепада AS**
36	***Правило 4.42.*** *Отвечающий прерыватель НЕ ДОЛЖЕН ВОЗБУЖДАТЬ ни одну из линий D00—D31, пока IACKIN* не установится низким*
37	***Правило 4.43.*** *Если участвующий прерыватель возбуждает любую из линий D00—D31, он ДОЛЖЕН ОСВОБОДИТЬ их до установки своего IACKOUT* низким*
38A	***Правило 4.44.*** *Участвующий прерыватель НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ свой IACKOUT* низким до обнаружения им низкого уровня сигнала на линии IACKIN**
38B	***Правило 4.45.*** *Отвечающий прерыватель НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ низким DTACK* до обнаружения им низкого IACKIN**
39	**Замечание 4.40.** Данный параметр гарантирует, что IACKIN* каждого из прерывателей установится высоким в течение указанного времени после возникновения положительного перепада AS*. Это время получается из значения временного параметра 35, который предписывает формирователю цепочки подтверждения прерывания и участвующим прерывателям устанавливать IACKOUT* высоким в пределах указанного максимального времени
40	**Замечание 4.41.** Всем прерывателям гарантируется, что на их линиях IACKIN* будут удерживаться высокие уровни сигналов в течение указанного минимального времени между двумя последовательными циклами шины пересылки данных
41	**Замечание 4.42.** Данный параметр гарантирует, что на линиях A01—A03 и LWORD* будут удерживаться достоверные уровни сигналов в течение указанного времени после установки участвующим прерывателем низким своего IACKOUT* в пределах тайм-аута шины
43	**Замечание 4.43.** Данный параметр гарантирует, что AS* будет оставаться низким в течение указанного минимального времени после установки участвующим прерывателем низким своего IACKOUT* в пределах тайм-аута шины

Таблица 4.19 — Формирователь цепочки подтверждения прерывания. Правила и замечания по временным соотношениям

Номер параметра (см. таблицу 4.16)	Правила и замечания
	Замечание 4.44. Поскольку объединительная плата соединяет линию IACK* с контактом IACKIN* гнезда 1, эти два сигнала эквивалентны. Поэтому все правила и замечания, относящиеся к одному из них, относятся и к другому
5	**Замечание 4.45.** Формирователю цепочки подтверждения прерывания гарантируется указанное минимальное время удержания высокого AS* между циклами шины пересылки данных
19	**Замечание 4.46.** Формирователю цепочки подтверждения прерывания гарантируется удержание AS* низким в течение указанного минимального времени. Это время получается из значений временных параметров 8, 16 и 27 прерывателя
32	**Замечание 4.47.** Формирователю цепочки подтверждения прерывания гарантируется, что IACK* (и IACKIN* гнезда 1) будет удерживаться достоверным в течение указанного минимального времени к моменту обнаружения им отрицательного перепада на линии DSA*
34	***Правило 4.46.*** *Если IACKIN* установлен низким и формирователь цепочки подтверждения прерывания обнаруживает отрицательный перепад DSA*, то этот формирователь ДОЛЖЕН УСТАНОВИТЬ IACKOUT* низким, но только по истечении указанного времени после появления отрицательного перепада DSA**
	Замечание 4.48. Формирователь цепочки подтверждения прерывания не устанавливает IACKOUT* низким каждый раз, когда DSA* устанавливается низким. Он делает это, только если IACK* также установлен низким, показывая, что исполняется цикл подтверждения прерывания
35	***Правило 4.47.*** *Если формирователь цепочки подтверждения прерывания установил IACKOUT* низким, он ДОЛЖЕН УСТАНОВИТЬ IACKOUT* высоким в течение указанного времени после возникновения положительного перепада AS**
40	***Правило 4.48.*** *Формирователь цепочки подтверждение прерывания НЕ ДОЛЖЕН УСТАНАВЛИВАТЬ IACKOUT* низким до тех пор, пока этот сигнал не будет выдержан высоким в течение указанного минимального времени*
42	**Замечание 4.49.** Если формирователь цепочки подтверждения прерывания устанавливает IACKOUT* низким в течение тайм-аута шины, этот временной параметр гарантирует, что IACK* (и IACKIN* гнезда 1) удерживается достоверным в течение указанного минимального времени

Рисунки 4.16—4.23 содержат временны́е диаграммы, которые определяют временны́е соотношения во время циклов подтверждения прерывания.

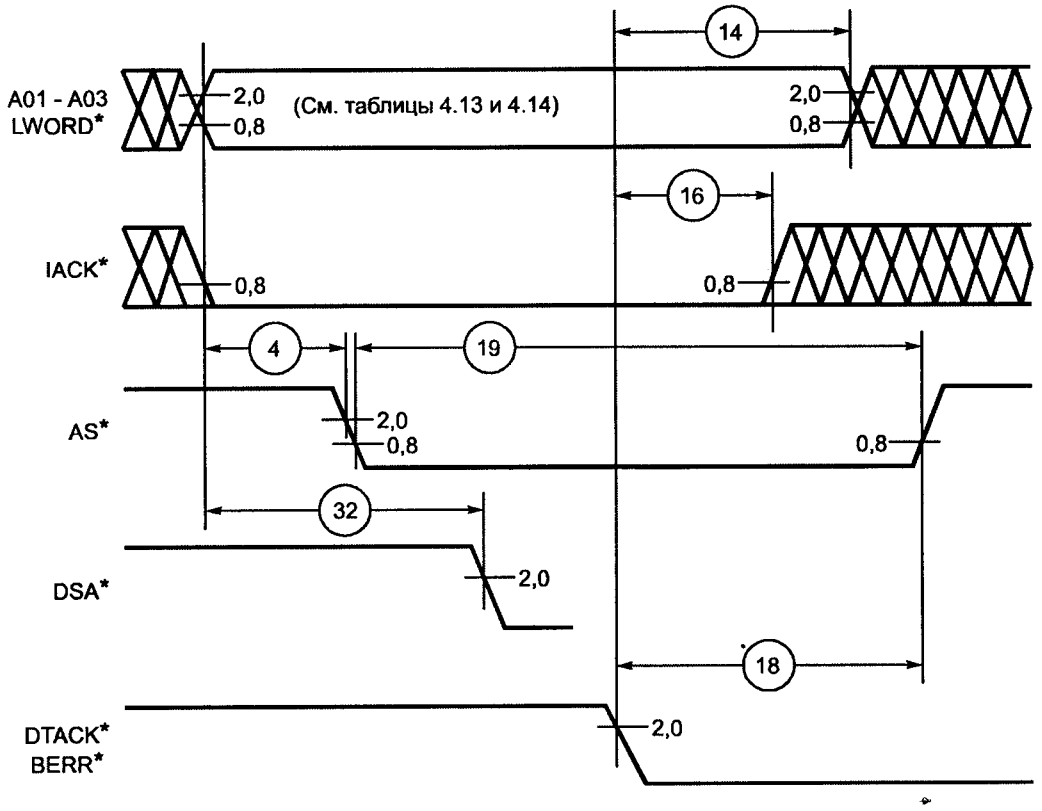
Рисунок 4.24 определяет дополнительные временны́е соотношения для цепочки IACKIN*/IACKOUT* между циклами.

Все правила и замечания, связанные с рисунками, перечисленными ниже, также относятся к обработчикам прерываний, прерывателям и формирователям цепочки потверждения прерывания.

Рисунки 2.22—2.24 в разделе 2 определяют временны́е соотношения для стробов адреса и данных между циклами пересылки данных.

Рисунок 2.25 определяет временны́е соотношения цикла с тайм-аутом.

Рисунок 2.26 определяет временны́е соотношения во время процесса передачи управления шиной.

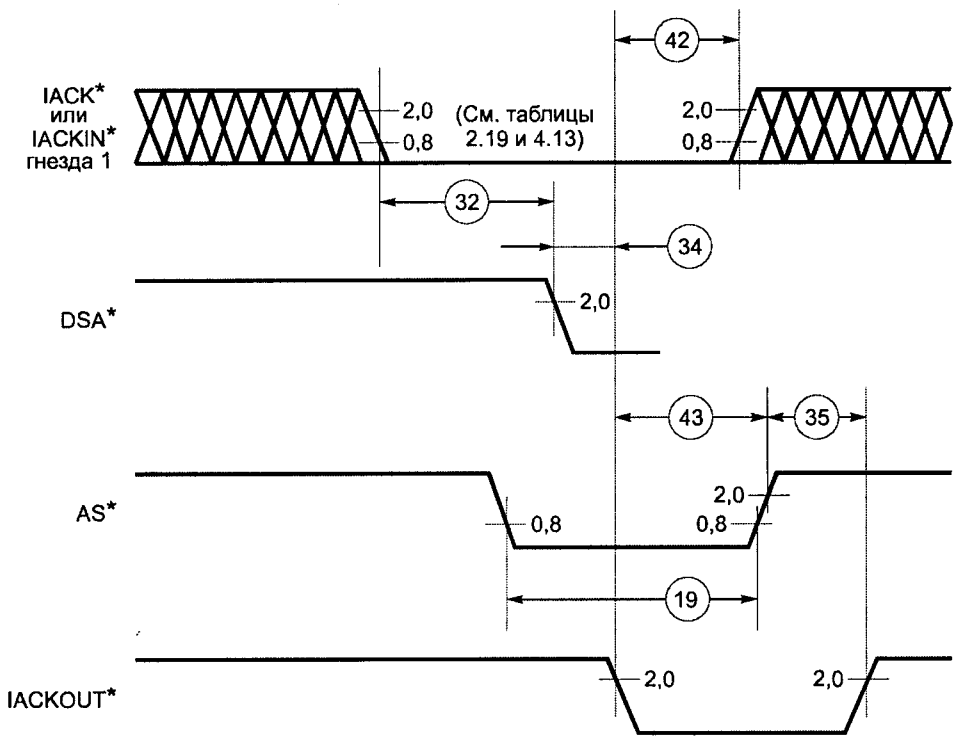


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Обработчик прерываний		Прерыватель		Формирователь цепочки подтверждения прерывания	
	мин.	макс.	мин.	макс.	мин.	макс.
4	35	—	10	—	—	—
14	0	—	0	—	—	—
16	0	—	0	—	—	—
18	0	—	0	—	—	—
19	40	—	30	—	30	—
32	—	—	10	—	10	—

Примечание — Все значения параметров в наносекундах.

Рисунок 4.16 — Временна́я диаграмма выбора прерывателя.
ОБРАБОТЧИК ПРЕРЫВАНИЙ и ПРЕРЫВАТЕЛЬ. (Одно-, двух- и четырехбайтовые циклы подтверждения прерывания)

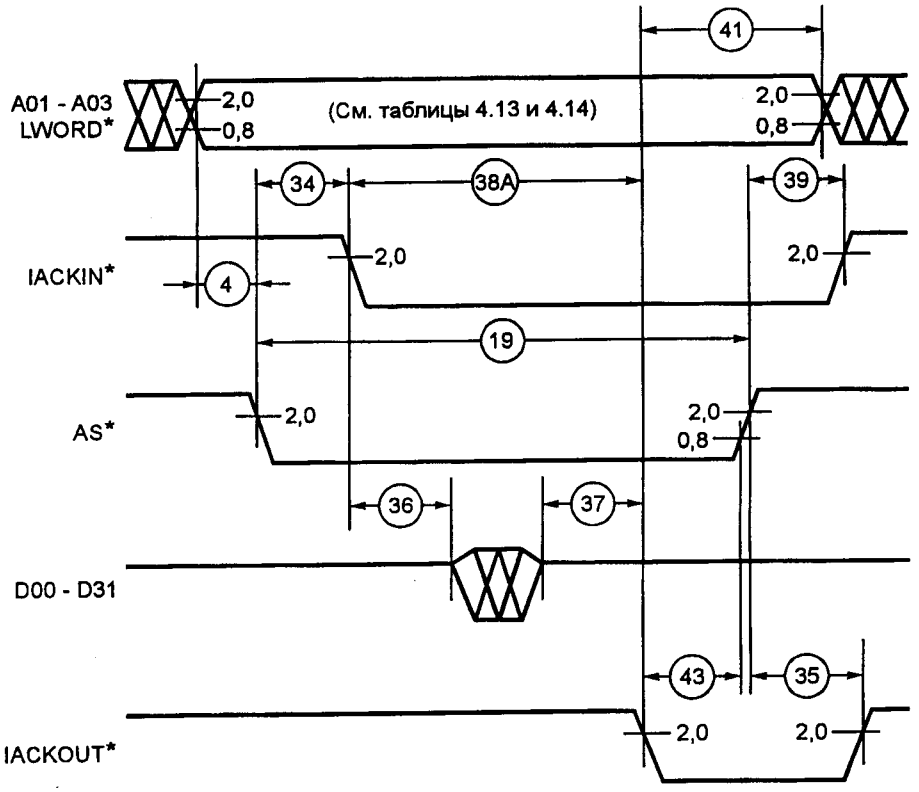


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Обработчик прерываний		Прерыватель		Формирователь цепочки подтверждения прерывания	
	мин.	макс.	мин.	макс.	мин.	макс.
19	40	—	30	—	30	—
32	—	—	10	—	10	—
34	—	—	30	—	40	—
35	—	—	0	30	0	30
42	—	—	—	—	30	—
43	—	—	0	—	—	—

Примечание — Все значения параметров в наносекундах.

Рисунок 4.17 — Временна́я диаграмма выбора прерывателя.
ФОРМИРОВАТЕЛЬ ЦЕПОЧКИ ПОДТВЕРЖДЕНИЯ ПРЕРЫВАНИЯ. (Одно-, двух- и четырехбайтовые циклы подтверждения прерывания)

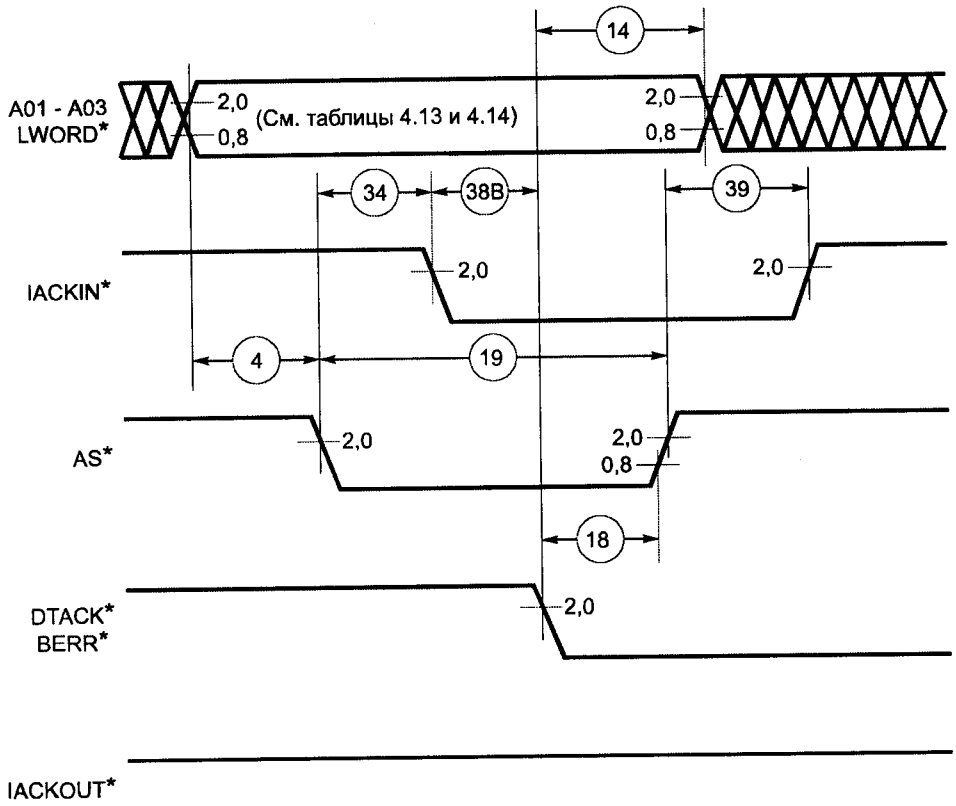


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Обработчик прерываний		Прерыватель		Формирователь цепочки подтверждения прерывания	
	мин.	макс.	мин.	макс.	мин.	макс.
4	35	—	10	—	—	—
19	40	—	30	—	30	—
34	—	—	30	—	40	—
35	—	—	0	30	0	30
36	—	—	0	—	—	—
37	—	—	0	—	—	—
38A	—	—	0	—	—	—
39	—	—	—	40	—	—
41	—	—	0	—	—	—
43	—	—	0	—	—	—

Примечание — Все значения параметров в наносекундах.

Рисунок 4.18 — Временна́я диаграмма выбора прерывателя.
Участвующий ПРЕРЫВАТЕЛЬ. (Одно-, двух- и четырехбайтовый циклы подтверждения прерывания)

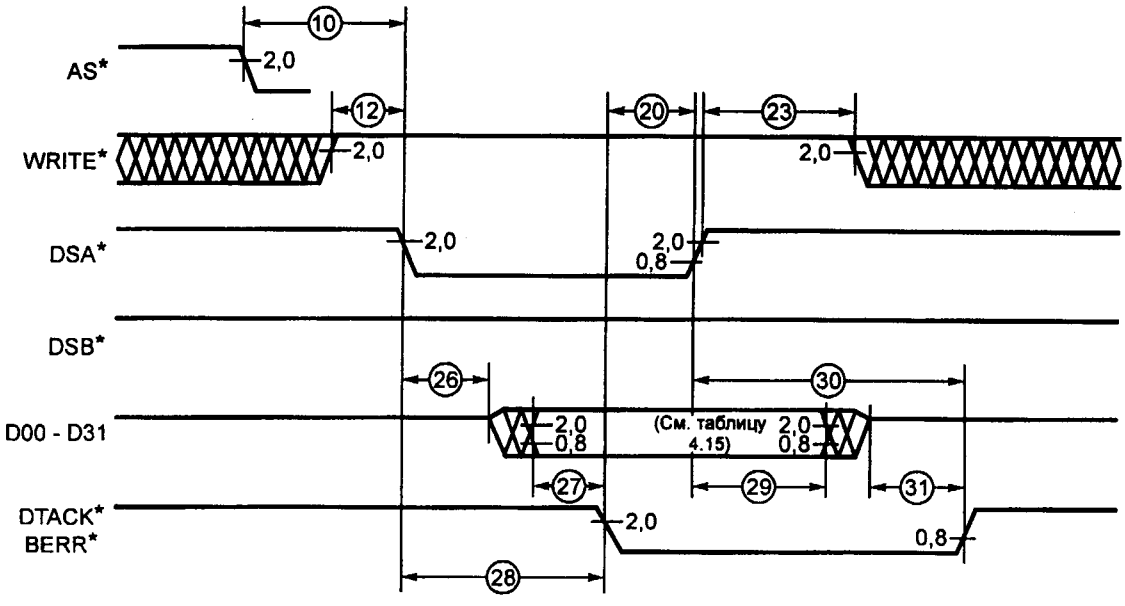


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Обработчик прерываний		Прерыватель		Формирователь цепочки подтверждения прерывания	
	мин.	макс.	мин.	макс.	мин.	макс.
4	35	—	10	—	—	—
14	0	—	0	—	—	—
18	0	—	0	—	—	—
19	40	—	30	—	30	—
34	—	—	30	—	40	—
38B	—	—	0	—	—	—
39	—	—	—	40	—	—

Примечание — Все значения параметров в наносекундах.

Рисунок 4.19 — Временна́я диаграмма выбора прерывателя.
Отвечающий ПРЕРЫВАТЕЛЬ. (Одно-, двух- и четырехбайтовый циклы подтверждения прерывания)

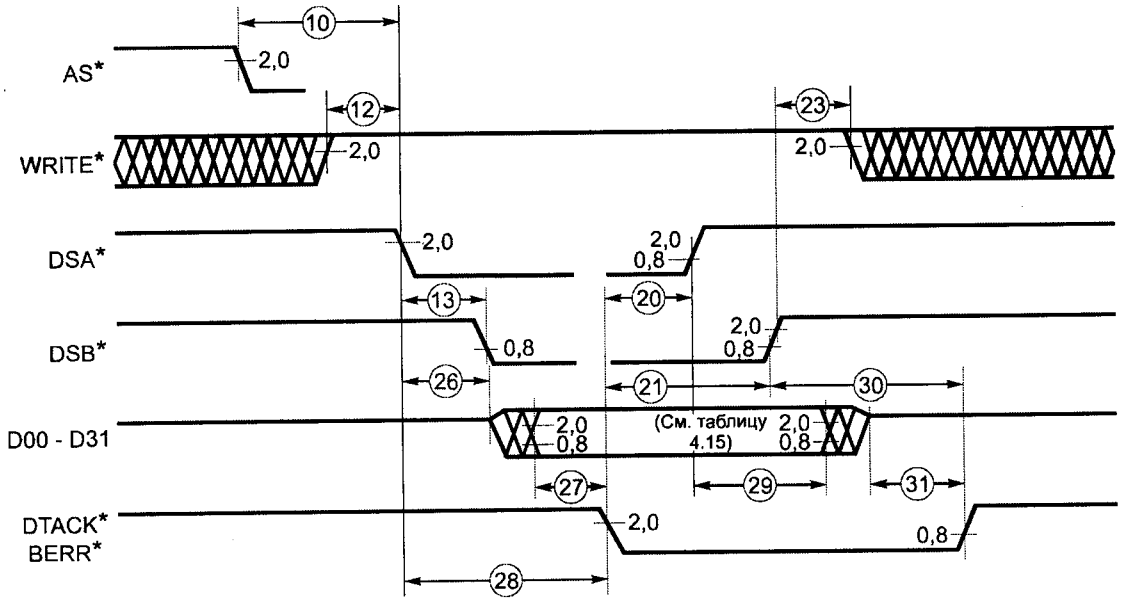


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Обработчик прерываний		Прерыватель		Формирователь цепочки подтверждения прерывания	
	мин.	макс.	мин.	макс.	мин.	макс.
10	0	—	—10	—	—	—
12	35	—	10	—	—	—
20	0	—	0	—	—	—
23	10	—	0	—	—	—
26	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
31	0	—	0	—	—	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 4.20 — Временна́я диаграмма пересылки информации статуса/идентификации. ОБРАБОТЧИК ПРЕРЫВАНИЙ. (Однобайтовый цикл подтверждения прерывания)

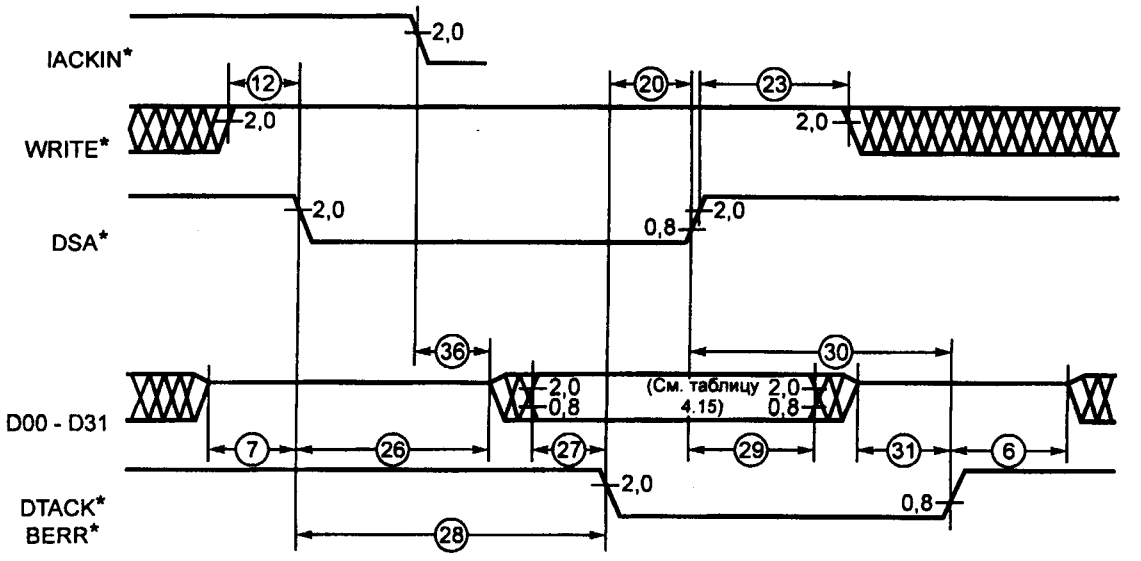


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Обработчик прерываний		Прерыватель		Формирователь цепочки подтверждения прерывания	
	мин.	макс.	мин.	макс.	мин.	макс.
10	0	—	—10	—	—	—
12	35	—	10	—	—	—
13	—	10	—	20	—	—
20	0	—	0	—	—	—
21	0	—	0	—	—	—
23	10	—	0	—	—	—
26	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
31	0	—	0	—	—	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 4.21 — Временна́я диаграмма пересылки информации статуса/идентификации. ОБРАБОТЧИК ПРЕРЫВАНИЙ. (Двух- и четырехбайтовый циклы подтверждения прерывания)

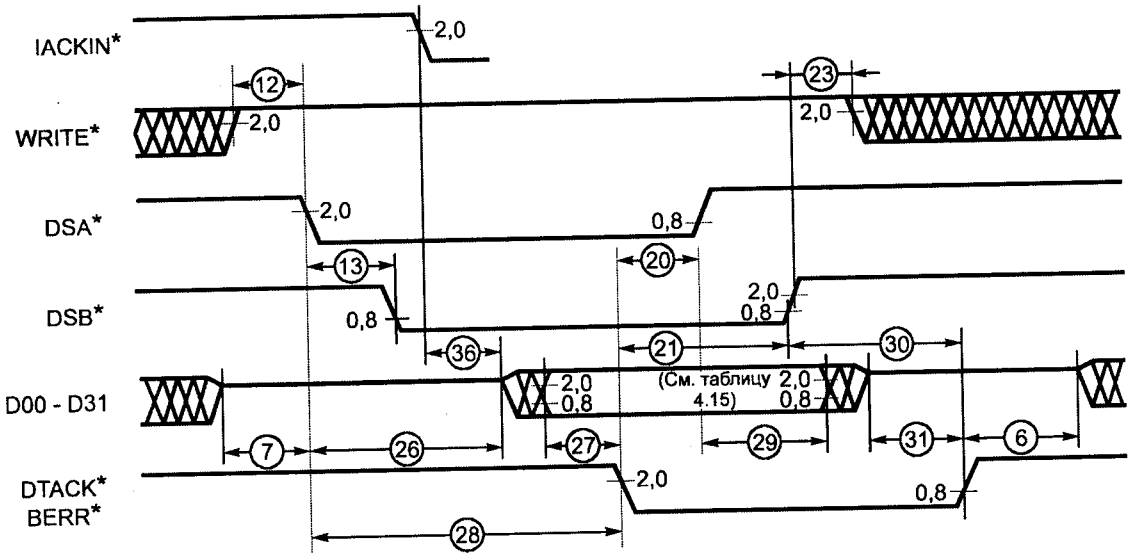


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Обработчик прерываний		Прерыватель		Формирователь цепочки подтверждения прерывания	
	мин.	макс.	мин.	макс.	мин.	макс.
6	—	—	0	—	—	—
7	—	—	0	—	—	—
12	35	—	10	—	—	—
20	0	—	0	—	—	—
23	10	—	0	—	—	—
26	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
31	0	—	0	—	—	—
36	—	—	0	—	—	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 4.22 — Временна́я диаграмма пересылки информации статуса/идентификации. Отвечающий ПРЕРЫВАТЕЛЬ. (Однобайтовый цикл подтверждения прерывания)

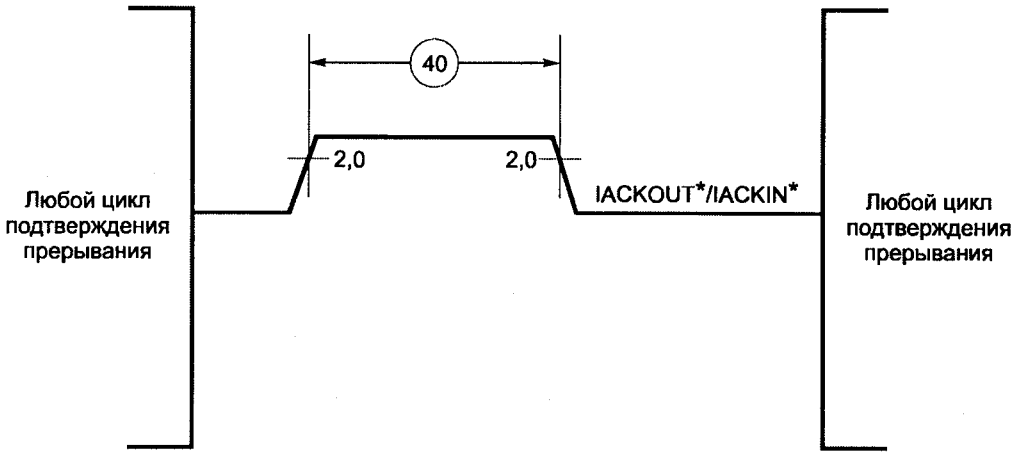


Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Обработчик прерываний		Прерыватель		Формирователь цепочки подтверждения прерывания	
	мин.	макс.	мин.	макс.	мин.	макс.
6	—	—	0	—	—	—
7	—	—	0	—	—	—
12	35	—	10	—	—	—
13	—	10	—	20	—	—
20	0	—	0	—	—	—
21	0	—	0	—	—	—
23	10	—	0	—	—	—
26	0	—	0	—	—	—
27	—25	—	0	—	—	—
28	30	2*T*	30	—	—	—
29	0	—	0	—	—	—
30	0	—	0	—	—	—
31	0	—	0	—	—	—
36	—	—	0	—	—	—

Примечания
1 Все значения параметров в наносекундах.
2 *T* — значение тайм-аута в микросекундах.

Рисунок 4.23 — Временна́я диаграмма пересылки информации статуса/идентификации. Отвечающий ПРЕРЫВАТЕЛЬ (Двух- и четырехбайтовый циклы подтверждения прерывания)



Значения напряжения пороговых уровней указаны в вольтах

Номер параметра	Обработчик прерываний		Прерыватель		Формирователь цепочки подтверждения прерывания	
	мин.	макс.	мин.	макс.	мин.	макс.
40	—	—	30	—	30	—
Примечание — T — значение времени тайм-аута в микросекундах.						

Рисунок 4.24 — Временна́я диаграмма работы цепочки подтверждения прерывания между циклами ФОРМИРОВАТЕЛЬ ЦЕПОЧКИ ПОДТВЕРЖДЕНИЯ ПРЕРЫВАНИЯ, отвечающий ПРЕРЫВАТЕЛЬ и участвующий ПРЕРЫВАТЕЛЬ.

С целью соблюдения указанных требований по временны́м соотношениям, проектировщики модулей должны учитывать задержки распространения в используемых ими шинных формирователях и приемниках для наихудшего случая. Задержки распространения формирователей зависят от их выходных нагрузок. Однако в техдокументации изготовителей не всегда содержится достаточная информация для расчета задержек распространения при различных нагрузках. Чтобы помочь проектировщикам модулей, в разделе 6 предложены некоторые рекомендации.

Замечание определяет временны́е соотношения перепадов сигналов входных линий. Этим соотношениям можно доверять до тех пор, пока не нарушаются правила нагрузки объединительной платы, приведенные в разделе 6. Правила для оконечных нагрузок, содержащиеся в разделе 6, гарантируют, что для сигнальных линий, которые освобождаются после того, как были возбуждены, временны́е параметры будут соблюдены.

Обычно для каждого правила, определяющего временны́е соотношения, имеется соответствующее замечание. Однако значение временно́го параметра, гарантируемое замечанием, может отличаться от значения, определенного правилом. Например, внимательный анализ временны́х диаграмм показывает, что от обработчика прерываний требуется обеспечить время установления адреса не меньше 35 нс, а прерывателю гарантируется только 10 нс. Это связано с тем, что формирователи адреса не всегда способны полностью перевести сигнальные линии объединительной платы из низкого состояния в высокое через пороговую область, пока этот перепад не распространится до конца объединительной платы и не отразится обратно. Правда, отрицательный перепад адресного строба обычно пересекает пороговое значение 0,8 В без ожидания отражения. Поэтому результирующее время установления у прерывателя равно времени установления обработчика прерывания, уменьшенному на удвоенное время распространения по магистрали.

Для описания временны́х соотношений стробов данных используется специальная система обозначений, подробно рассмотренная в 2.6.

5 Служебная шина

5.1 Введение

В данном разделе определяются сигнальные линии и функциональные блоки, которые обеспечивают выполнение таких служебных функций, как генерация тактовых сигналов, инициализация и возможность диагностики магистрали (см. рисунок 5.1).

5.2 Сигнальные линии служебной шины

Ниже перечислены сигнальные линии служебной шины:

SYSCLK (system clock) — линия сигнала системного тактирования;

SERCLK (serial clock) — линия сигнала тактирования последовательной магистрали;

SERDAT* (serial data) — линия данных последовательной магистрали;

ACFAIL* (AC failure) — линия сигнала отказа сети переменного тока;

SYSRESET* (system reset) — линия сигнала системного сброса;

SYSFAIL* (system failure) — линия сигнала системного отказа;

5.3 Функциональные блоки служебной шины

5.3.1 Формирователь системного тактового сигнала

Формирователь системного тактового сигнала генерирует независимый нестробируемый сигнал (SYSCLK) фиксированной частоты 16 МГц с номинальной скважностью 2. Формирователь помещается в модуле системного контроллера, вставляемого в гнездо 1 (см.раздел 1). Временны́е параметры сигнала SYSCLK используются при определении временны́х задержек. Временна́я диаграмма работы формирователя приведена на рисунке 5.2.

Замечание 5.1. Сигнал SYSCLK не имеет фиксированных фазовых соотношений с другими сигналами синхронизации магистрали.

5.3.2 Формирователь тактового сигнала последовательной магистрали

Формирователь тактового сигнала последовательной магистрали формирует сигнал фиксированной частоты и специальной формы, используемый блоками магистрали МЭК 823, которые находятся в модулях магистрали VME. Форма этого сигнала определяется стандартом МЭК 823. Для удобства разработчиков временны́е параметры, действующие в настоящее время, опубликованы в настоящем стандарте и содержатся в приложении С.

5.3.3 Блок контроля питания

На рисунке 5.3 приведена схема взаимодействия блока контроля питания с устройствами магистрали VME. Этот блок обнаруживает отказы питания и своевременно сигнализирует об этом системе, чтобы она выполнила нормальное завершение своей работы. Когда подача питания в систему восстанавливается, блок контроля питания обеспечивает инициализацию функциональных блоков магистрали.

Блок контроля питания может также инициализировать систему магистрали VME при нажатии оператором кнопки ручного управления.

Между моментами выхода значений напряжения питания постоянного тока за пределы, установленные техническими требованиями на источник питания, и моментами изменения уровней сигналов на линиях ACFAIL* и SYSRESET* имеются определенные временны́е соотношения. Эти соотношения показаны на рисунках 5.4, 5.5.

Разрешение 5.1. Системы на основе магистрали VME МОГУТ строиться с блоком контроля питания или без него.

Правило 5.1. *Блоки контроля питания ДОЛЖНЫ СОБЛЮДАТЬ временны́е соотношения, показанные на рисунках 5.4, 5.5.*

Разрешение 5.2. Для инициализации системы линия SYSRESET* МОЖЕТ устанавливаться кнопкой ручного управления с любого модуля магистрали VME. Соблюдение временны́х соотношений рисунков 5.4, 5.5 не распространяется на случаи, когда какой-либо модуль возбуждает линию SYSRESET*, но не возбуждает линию ACFAIL*.

Правило 5.2. *Если какой-либо модуль устанавливает SYSRESET* низким, он ДОЛЖЕН УДЕРЖИВАТЬ его в этом состоянии не менее 200 мс.*

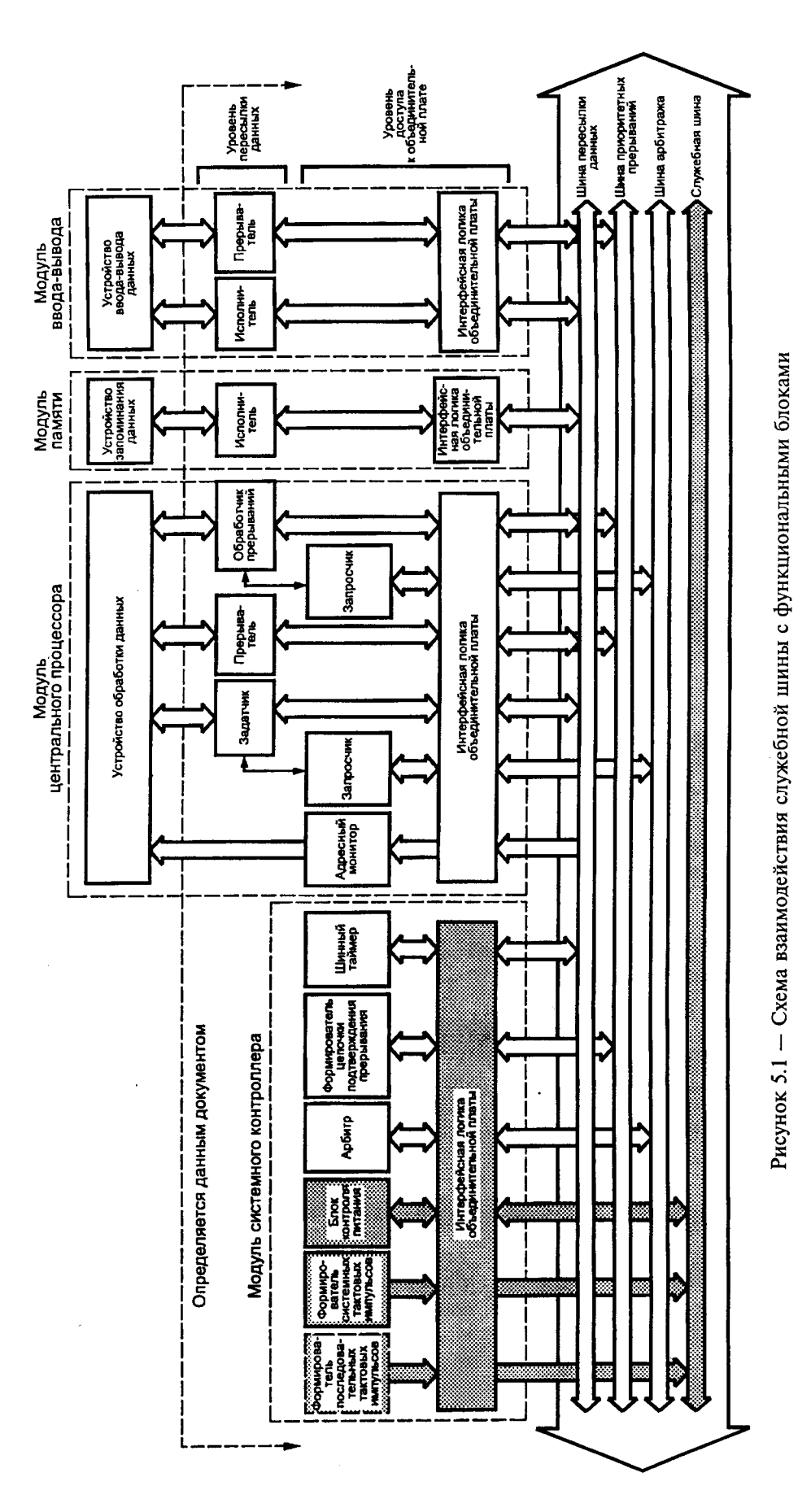


Рисунок 5.1 — Схема взаимодействия служебной шины с функциональными блоками

Рисунок 5.2 — Временна́я диаграмма работы формирователя системного тактового сигнала

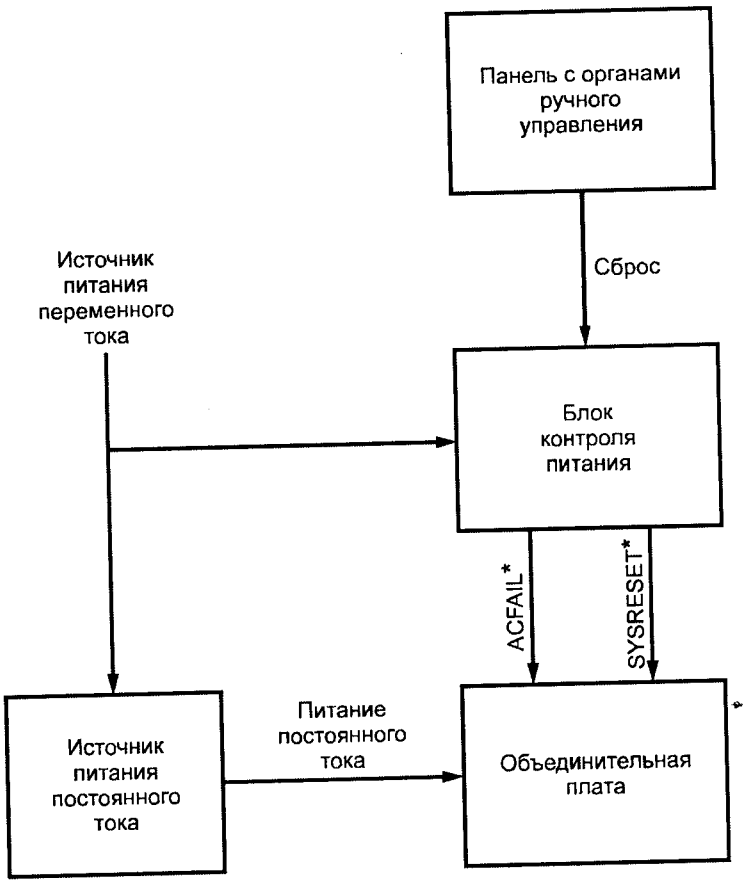


Рисунок 5.3 — Схема взаимодействия блока контроля питания с устройствами магистрали VME

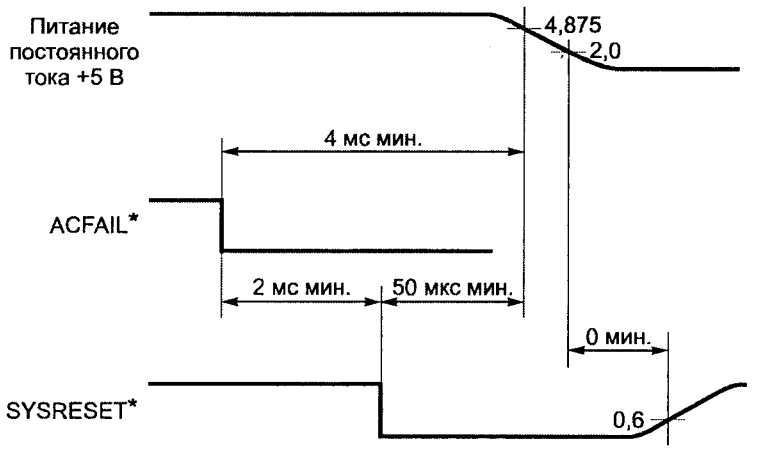


Рисунок 5.4 — Временна́я диаграмма работы блока контроля питания в случае отказа питания

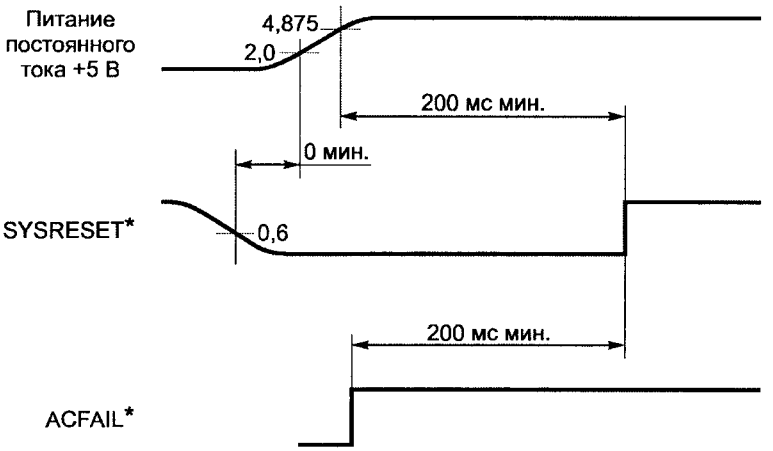


Рисунок 5.5 — Временна́я диаграмма работы блока контроля питания при включении системы

5.4 Инициализация и диагностика системы

Ниже дано описание протоколов, позволяющих выполнять операции завершения работы и включения питания упорядоченным способом. Для формирования последовательности операций включения и выключения питания используются две сигнальные линии: ACFAIL* и SYSRESET*. Сигнальная линия SYSFAIL* используется при формировании последовательности операций включения питания.

Ниже определено поведение различных функциональных блоков при выполнении последовательности операций выключения питания.

Рекомендация 5.1. Рекомендуется проектировать задатчики так, чтобы по истечении 200 мкс после установки низкого уровня ACFAIL* они запрашивали шину только для действий, связанных с отказом питания.

Рекомендация 5.2. Если задатчики или обработчики прерываний выставляют запрос шины раньше, чем обнаружен низкий уровень ACFAIL*, то рекомендуется, чтобы они завершили все свои последующие действия, не относящиеся к процедуре отключения питания, в течение 200 мкс.

Замечание 5.2. Обращения к магистрали, необходимые для сохранения и восстановления системных данных в памяти магистрали, зависят от прикладной задачи и в настоящем стандарте не определяются. Восстановление данных, сохраненных в процессе останова системы, должна обеспечивать операционная система. Процесс восстановления должен завершаться перед возобновлением работы системы. В случае мультипроцессорной системы это может потребовать некоторых межпроцессорных взаимодействий.

Сигнал на линии SYSRESET* возбуждается формирователем с открытым коллектором и управляется блоком контроля питания или любым модулем при нажатии кнопки ручного сброса.

Замечание 5.3. При использовании кнопок сброса следует предусматривать специальные антидребезговые схемы, гарантирующие поддержание низкого уровня SYSRESET* в течение интервала не менее 200 мс.

Правило 5.3. *Формирователь системного тактового сигнала ДОЛЖЕН ФОРМИРОВАТЬ заданный сигнал SYSCLK независимо от уровня SYSRESET*.*

Разрешение 5.4. При установке SYSRESET* низким любой модуль, которому требуется для завершения инициализации более 200 мс, МОЖЕТ включить в низкое состояние свой формирователь линии SYSRESET* для удержания на линии сигнала низкого уровня в течение требуемого времени.

Правило 5.4. *Если в момент установки SYSRESET* низким выходное напряжение источника постоянного тока + 5 В находится в заданных для него пределах, то, начиная с этого момента, ДОЛЖНЫ СОБЛЮДАТЬСЯ требования, приведенные в таблице 5.1.*

Таблица 5.1 — Правила возбуждения функциональными блоками сигналов в процессе включения и выключения питания

Функциональный блок	Правило	
	НЕ ДОЛЖЕН	по истечении
Задатчик и обработчик прерываний	*УСТАНАВЛИВАТЬ из высокого состояния в низкое сигналы AS*, DS0* или DS1**	5 мкс
Задатчик и обработчик прерываний	*ВОЗБУЖДАТЬ линии IACK*, LWORD*, AS*, DS0*, DS1*, AM0—AM5, A01—A31, WRITE* и D00—D31*	20 мкс
Исполнитель и прерыватель	*ВОЗБУЖДАТЬ линии D00—D31, DTACK* и BERR**	30 мкс
Прерыватель	*ВОЗБУЖДАТЬ линии IRQ1*—IRQ7**	30 мкс
Шинный таймер	*ВОЗБУЖДАТЬ линию BERR**	30 мкс
Арбитр	*УСТАНАВЛИВАТЬ из высокого состояния в низкое сигналы BG0IN*—BG3IN**	5 мкс
Арбитр	*УДЕРЖИВАТЬ в низком состоянии сигналы BG0IN*—BG3IN**	30 мкс
Запросчик	*ВОЗБУЖДАТЬ линию BBSY**	30 мкс

Правило 5.5. *Если к моменту установления выходного напряжения источника постоянного тока + 5 В в заданные для него пределы SYSRESET* является низким, то, начиная с этого момента, ДОЛЖНЫ СОБЛЮДАТЬСЯ требования, приведенные в таблице 5.1.*

Правило 5.6. *После истечения соответствующих временных интервалов, указанных в таблице 5.1, функциональные блоки НЕ ДОЛЖНЫ ИЗМЕНЯТЬ состояния своих формирователей до установки SYSRESET* высоким, если выходное напряжение источника постоянного тока + 5 В не выходит за заданные для него пределы.*

Правило 5.7. *Если в момент установки SYSRESET* низким выходное напряжение источника постоянного тока + 5 В находится в заданных для него пределах, а задатчик или обработчик прерываний устанавливает AS*, DS0* или DS1* низкими, то они ДОЛЖНЫ УДЕРЖИВАТЬ эти сигналы низкими в течение времени, достаточного для выполнения требований по минимальным временным интервалам удержания низкого уровня, указанным в разделах 2 и 4.*

Сигнал на линии SYSFAIL*, возбуждаемой формирователем с открытым коллектором, при включении питания удерживается низким, и это состояние сохраняется до завершения процесса самотестирования системы (см. рисунок 5.6). К этому имеет отношение следующая информация:

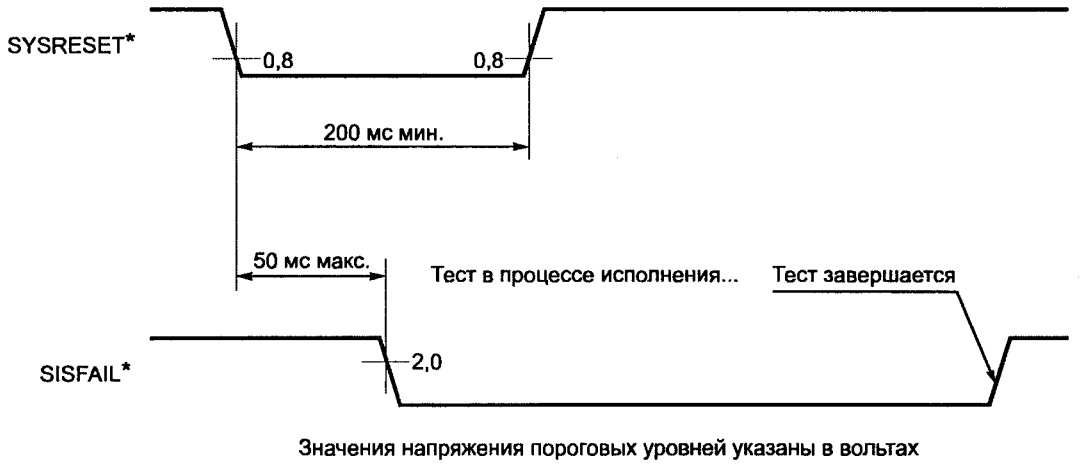


Рисунок 5.6 — Временна́я диаграмма сигналов SYSRESET* и SYSFAIL*

Предложение 5.1. В любом модуле с функцией задатчика, имеющем встроенный микропроцессор, предлагается предусматривать в управляющем регистре локально доступный разряд (доступный только собственному микропроцессору), который устанавливает на контакте SYSFAIL* модуля низкий уровень сигнала при первоначальной подаче на него напряжения питания. Это позволяет собственному микропроцессору выполнять локальное самотестирование и освобождать линию SYSFAIL* только в случае его успешного завершения.

Предложение 5.2. Модули без встроенного микропроцессора предлагается проектировать с глобально доступным разрядом (разрядом, доступным любому микропроцессору системы VME) в его управляющем регистре. Этот разряд инициализируется для установки на линии SYSFAIL* низкого уровня, что позволяет любому модулю со встроенным микропроцессором сначала выполнить тестирование безмикропроцессорного модуля, а затем осуществить запись в глобально доступный разряд его управляющего регистра для освобождения его линии SYSFAIL*.

Предложение 5.3. В любом модуле магистрали VME, имеющем в управляющем регистре разряд управления линией SYSFAIL*, предлагается устанавливать на его передней панели светодиодный индикатор состояния этого разряда. В таком случае при поступлении сигнала о системном отказе по линии SYSFAIL* отказавший модуль можно найти визуальной проверкой.

Правило 5.8. *Если модуль магистрали VME спроектирован с возможностью управления SYSFAIL*, то он ДОЛЖЕН УСТАНОВИТЬ SYSFAIL* низким не более чем через 50 мс после установления низким SYSRESET*, как показано на рисунке 5.6.*

Разрешение 5.4. Модуль МОЖЕТ также устанавливать SYSFAIL* низким в любое время в ходе нормальной работы при обнаружении какого-то отказа.

5.5 Контакты питания

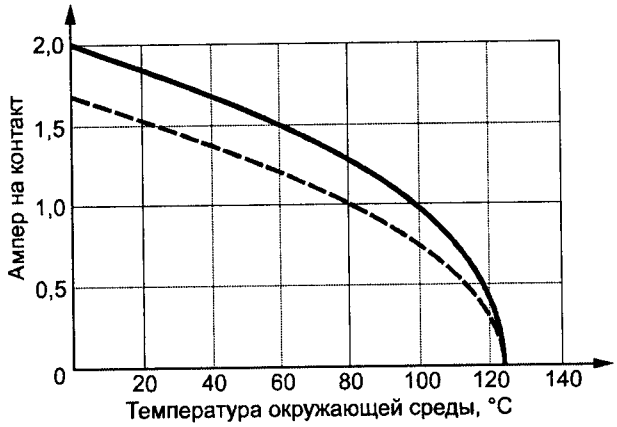
На рисунке 5.7 приведена зависимость от температуры максимального значения тока для контактов питания соединителей магистрали VME.

Замечание 5.4. Некоторые контакты соединителей, подключенных к объединительной плате, имеют незначительные отличия в значениях контактного сопротивления, что создает несбалансированные токи в контактах, соединенных параллельно. Предположим, что два контакта запараллелены и несут суммарный ток 2 А. Если контактное сопротивление одного равно 1 мОм, а другого — 2 мОм, то через один контакт будет проходить ток 1,33 А, а через другой — только 0,67 А.

Правило 5.9. *Контакты соединителей магистрали VME ДОЛЖНЫ ОБЛАДАТЬ способностью пропускать, по меньшей мере, токи, показанные на рисунке 5.7 сплошной линией.*

Замечание 5.5. Если один или несколько контактов питания полностью выходят из строя, вся токовая нагрузка достается оставшимся контактам. Например, если выходит из строя половина контактов, через остальные контакты потечет ток, вдвое превышающий нормальный. При больших токах нагрузки это может вызвать повреждение оставшихся исправных контактов.

Предложение 5.4. При проектировании модулей магистрали VME с высокой токовой нагрузкой предлагается разделить площадь платы модуля на зоны, запитываемые от отдельных шин питания. Эти шины не следует соединять между собой на плате, а каждую из них присоединять к своему контакту питания.



Примечания

1 Пунктирной линией показано изменение максимального значения тока, протекающего по каждому контакту питания соединителя магистрали от источника постоянного тока +5 В, для случаев, когда два или более контакта подсоединены к одной общей шине питания на плате модуля.

2 Сплошной линией показано изменение максимального значения тока, протекающего по каждому контакту питания соединителя магистрали от источника постоянного тока +5 В, для случаев, когда каждый контакт питания подсоединен к отдельной шине питания на плате модуля.

Рисунок 5.7 — Оценка значения тока для контактов питания

Замечание 5.6. Если в каркас, имеющий только объединительную плату J1, вставляется модуль двойной высоты, потребляющий бóльшую мощность, чем может обеспечить его соединитель P1, то контакты питания P1 могут выйти из строя в результате перегрева.

5.6 Резервная линия

Замечание 5.7. Как определено в таблице 6.3 раздела 6, резервная линия имеет оконечную нагрузку и подсоединена к соответствующим контактам всех соединителей объединительной платы.

Правило 5.10. *Резервная линия предназначена для использования в будущем и при проектировании каких-либо модулей магистрали VME НЕ ДОЛЖНА ПРИМЕНЯТЬСЯ.*

6 Электрические характеристики магистрали VME

6.1 Введение

Передача данных между такими модулями магистрали VME, как процессоры, запоминающие устройства и устройства ввода/вывода, осуществляется с использованием одной или двух объединительных плат, в зависимости от конструкции базового блока. Соблюдение правил данного раздела гарантирует надлежащую синхронизацию, минимальный уровень шума и перекрестных помех на сигнальных линиях объединительных плат. Изложенные ниже правила являются руководством по проектированию объединительных плат магистрали VME.

Правило 6.1. *Длина сигнальных проводников объединительной платы НЕ ДОЛЖНА ПРЕВЫШАТЬ 500 мм (19,68 ").*

Правило 6.2. *Объединительная плата магистрали VME НЕ ДОЛЖНА СОДЕРЖАТЬ более 21 гнезда.*

Правило 6.3. *Для линий, требующих установки оконечных нагрузок (см.подраздел 6.7), на объединительной плате ДОЛЖНЫ ПРЕДУСМАТРИВАТЬСЯ средства, обеспечивающие их установку на обоих концах сигнальной линии.*

Правило 6.4. *Объединительная плата ДОЛЖНА СОДЕРЖАТЬ силовые проводники для распределения питания от источников + 5 В, + 5 В (резервный), +12 В и минус 12 В, подсоединенные ко всем контактам питания, указанным в 7.6.*

Правило 6.5. *Объединительная плата ДОЛЖНА ОБЕСПЕЧИВАТЬ присоединение к "Земле" всех контактов GND ("Земля"), указанных в 7.6.*

Разрешение 6.1. Сигнальные линии магистрали обычно возбуждаются выполненными по биполярной технологии формирователями, но МОГУТ использоваться формирователи, выполненные по любой технологии и отвечающие требованиям настоящего стандарта.

6.2 Распределение питания

Питание в системе магистрали VME осуществляется путем распределения на объединительной плате (платах) стабилизированных напряжений постоянного тока следующих номиналов:

+5 В постоянного тока — основной источник питания; требуется для большинства схем системы, включая схемы транзисторно-транзисторной логики (ТТЛ), микропроцессоры, выполненные по технологии металл—окисел—полупроводник (МОП) и память;

плюс/минус 12 В постоянного тока — часто используются для питания формирователей интерфейса RS232C. Они также иногда используются для питания цифровых МОП и аналоговых схем. В некоторых случаях напряжение смещения минус 5 В постоянного тока и напряжение минус 5,2 В постоянного тока для питания схем эмиторно-связанной логики (ЭСЛ) также получается от источника минус 12 В с помощью внутримодульных стабилизаторов. Обычно мощность этих источников много меньше мощности источника +5 В;

+5 В постоянного тока (резервный) — используется для сохранения содержимого памяти, часов текущего времени и т.п. при отключении основного источника +5 В постоянного тока.

6.2.1 Технические требования к источникам питания постоянного тока

Технические требования к источникам постоянного тока приведены в таблице 6.1. Эти требования определяют максимально допустимые отклонения напряжений, измеренных на контактах соединителей любого модуля, вставленного в объединительную плату.

Таблица 6.1 — Технические характеристики питающих напряжений магистрали

Обозначение	Характеристики		
	Описание	Допустимые отклонения, В (см. замечания)	Пульсация/шум с частотой менее 10 МГц (от пика до пика), мВ
+5 V	+5 В пост. тока	+0,25 /—0,125	50
+12 V	Питание +12 В пост. тока	+0,60 /—0,36	50
—12 V	Питание —12 В пост. тока	—0,60 /+0,36	50
+5 V STDBY	Резервное питание +5 В пост. тока	+0,25 /—0,125	50
GND	"Земля"	Опорный потенциал	—

Рекомендация 6.1. Объединительные платы рекомендуется проектировать, а соединения выполнять на них таким образом, чтобы точка подводки линий обратной связи источника питания к объединительной плате располагалась вблизи центра объединительной платы и как можно ближе к точке, где на объединительную плату подводится питание.

Замечание 6.1. Размещение точки подводки линий обратной связи источника питания вблизи точки подводки силовой линии исключает опасность поступления слишком высокого напряжения на модули, находящиеся вблизи точки подводки силовой линии.

Замечание 6.2. Несимметричные отклонения напряжений, указанные в таблице 6.1, гарантируют сохранение параметров питания постоянного тока в пределах, допустимых для большинства интегральных схем (ИС), несмотря на падения напряжения, которые обычно имеют место в цепях подводки питания.

Замечание 6.3. Ток потребления некоторых систем при нормальной работе изменяется во времени в широких пределах. Например, значительные колебания могут возникнуть при регенерации динамической памяти, если обновляется сразу большое количество модулей памяти. В таком случае приобретает значение время отклика системы распределения напряжения.

Рекомендация 6.2. Для сведения к минимуму импульсных помех на платах модулей магистрали VME рекомендуется использовать конденсаторы развязки по питанию.

6.2.2 Электрические параметры гнездовых и штыревых соединителей

Правило 6.6. *Используемый в магистрали 96-контактный соединитель ДОЛЖЕН ОБЛАДАТЬ следующими электрическими параметрами:*

- номинальное значение рабочего напряжения между контактами— не менее 100 В постоянного тока;

- контактное сопротивление при номинальном токе— не более 50 мОм;

- сопротивление изоляции между контактами— не менее 100 МОм.

Замечание 6.17. Соединители, имеющие различную диэлектрическую проницаемость материала корпуса, обладают различной восприимчивостью к перекрестным помехам.

Предложение 6.9. Используйте соединители, материал корпуса которых сводит к минимуму перекрестные помехи.

6.3 Электрические характеристики сигналов

Правило 6.7. *Модули магистрали НЕ ДОЛЖНЫ ВОЗБУЖДАТЬ ни одной сигнальной линии объединительной платы до более высокого установившегося значения напряжения, чем наибольшее значение напряжения на любом из ее контактов питания + 5 В, или до более низкого установившегося значения напряжения, чем самое низкое значение напряжения на любом из ее контактов GND ("Земля").*

Правило 6.8. *В модулях магистрали ДОЛЖНЫ ИСПОЛЬЗОВАТЬСЯ формирователи и приемники логических сигналов со следующими статическими характеристиками:*

низкий выходной уровень формирователя— не более 0,6 В;

низкий входной уровень приемника— не более 0,8 В;

высокий выходной уровень формирователя— не менее 2,4 В;

высокий входной уровень приемника— не менее 2,0 В.

На рисунке 6.1 эти уровни представлены графически.

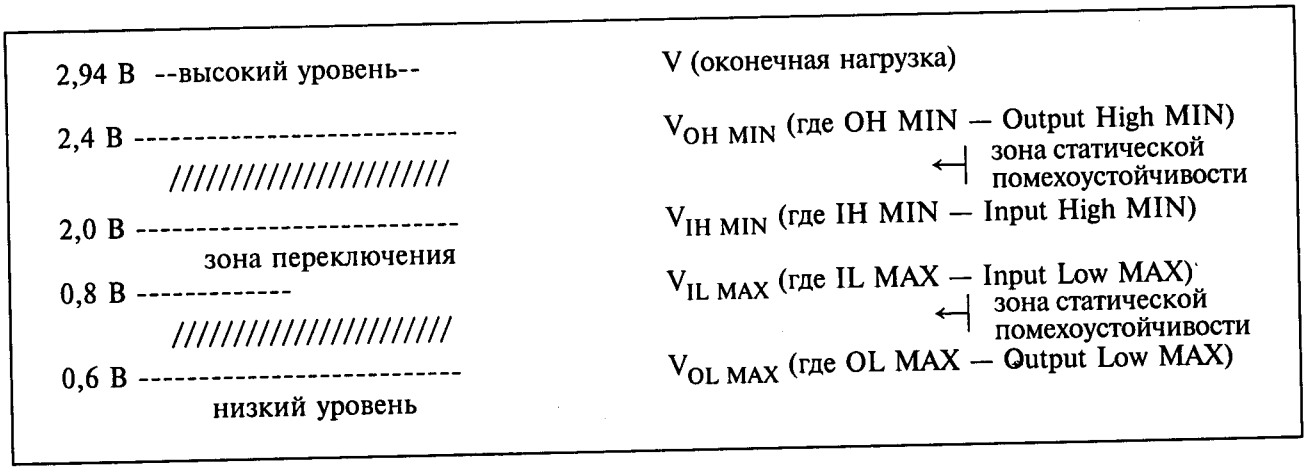


Рисунок 6.1 — Уровни логических сигналов

Модули магистрали VME возбуждают линии объединительной платы формирователями с тремя и двумя состояниями, а также с открытым коллектором на выходе. Требования по возбуждению и нагрузке различных сигнальных линий определены в подразделе 6.4. В подразделе 6.7 приведена обобщающая информация о типах формирователей, которые используются для возбуждения каждой сигнальной линии.

Правило 6.9. *При измерении пороговых напряжений на модуле для проверки соответствия техническим требованиям по временным соотношениям опорный потенциал "Земля" ДОЛЖЕН БЫТЬ ВЗЯТ на контакте GND ("Земля") модуля, ближайшем к контакту измеряемого сигнала, а напряжение сигнала ДОЛЖНО ИЗМЕРЯТЬСЯ на контакте соединителя этого модуля.*

6.4 Требования к характеристикам формирователей и приемников сигналов магистрали

Данный подраздел определяет технические требования к характеристикам формирователей и приемников всех сигнальных линий магистрали. В таблице 6.2 перечисляются все сигналы и указывается, в каких подразделах они рассматриваются.

Таблица 6.2 — Требования к характеристикам формирователей и приемников сигналов магистрали

Наименование сигнала	Требование, номер подпункта	Наименование сигнала	Требование, номер подпункта
A01—A31	6.4.2.2	DS1*	6.4.2.1
ACFAIL*	6.4.2.5	DTACK*	6.4.2.5
AM0—AM5	6.4.2.2	IACK*	6.4.2.2; 6.4.2.5
AS*	6.4.2.1	IACKOUT*	6.4.2.4
BBSY*	6.4.2.5	IRQ1*—IRQ7*	6.4.2.5
BCLR*	6.4.2.3	LWORD*	6.4.2.2
BERR*	6.4.2.5	SERCLK	6.4.2.3
BG0OUT*—BG3OUT*	6.4.2.4	SYSCLK	6.4.2.3
BR0*—BR3*	6.4.2.5	SYSFAIL*	6.4.2.5
D00—D31	6.4.2.2	SYSRESET *	6.4.2.5
DS0*	6.4.2.1	WRITE*	6.4.2.2

6.4.1 Определение типов формирователей магистрали

Ниже приведены определения формирователей с двумя и тремя состояниями, а также открытым коллектором на выходе.

Формирователь с двумя состояниями на выходе — активный формирователь в обоих состояниях. В состоянии низкого уровня ток втекает в формирователь, в состоянии высокого уровня — вытекает из него. Этот тип формирователя используется для сигналов, имеющих только один формирователь на каждой линии (например в цепочках).

Формирователь с тремя состояниями на выходе аналогичен предыдущему формирователю, но кроме высокого и низкого состояний он может перейти в высокоимпедансное состояние (формирователь выключен). Для линии, которая может возбуждаться несколькими устройствами в разных ее точках (например для адресных линий или линий данных), используется несколько таких формирователей. На такой линии в любой момент времени активным (включенным) может быть только один формирователь.

Формирователь с открытым коллектором на выходе. В состоянии низкого уровня ток втекает в формирователь, а в состоянии высокого уровня — вытекает из него, но очень незначительный. Если на этой линии не устанавливается низкий уровень, установление высокого уровня на сигнальной линии с такими формирователями обеспечивается оконечными нагрузочными резисторами на объединительной плате. Такие формирователи используются для сигнальных линий, которые могут возбуждаться несколькими устройствами одновременно (например для линий запроса прерывания и запроса шины пересылки данных).

6.4.2 Требования к характеристикам формирователей и нагрузок для всех линий

Правило 6.10. *Все модули магистрали VME ДОЛЖНЫ ИМЕТЬ на каждой контролируемой ими сигнальной линии средства ограничения напряжения, которые исключают его отклонение ниже минус 1,5 В.*

Замечание 6.4. Стандартные микросхемы серий 555 (74LSXXX) и 1531 (74FXXX) имеют на своих входах внутренние ограничительные диоды, обеспечивающие выполнение правила 6.10.

Правило 6.11. *Приемники ДОЛЖНЫ ГАРАНТИРОВАТЬ обнаружение высокого логического уровня свыше порога 2,0 В, указанного на рисунке 6.1.*

Правило 6.12. *Приемники ДОЛЖНЫ ГАРАНТИРОВАТЬ обнаружение низкого логического уровня ниже порога 0,8 В, указанного на рисунке 6.1.*

Разрешение 6.2. Формирователь с тремя состояниями на выходе МОЖЕТ использоваться в качестве формирователя с двумя состояниями, если он постоянно находится во включенном состоянии.

6.4.2.1 *Требования к характеристикам формирователей с тремя состояниями и нагрузок для сильноточных линий AS*, DS0*, DS1**

Правило 6.13. *Если модуль магистрали VME возбуждает линии AS*, DS0* или DS1*, то его формирователи для этих линий ДОЛЖНЫ УДОВЛЕТВОРЯТЬ следующим техническим требованиям:*

- втекающий ток в состоянии низкого уровня I_{OL} — не менее 64 мА;

- напряжение низкого уровня V_{OL} — не более 0,6 В при токе I_{OL}= 64 мА;

- вытекающий ток в состоянии высокого уровня I_{OH} — не менее 3 мА;

- напряжение высокого уровня V_{OH} — не менее 2,4 В при токе I_{OH} = 3 мА;

- минимальный вытекающий ток при коротком замыкании I_{OS} — не менее 50 мА при напряжении 0В;

- максимальный вытекающий ток при коротком замыкании I_{OS} — не более 225 мА при напряжении 0В.

Правило 6.14. *При выключении формирователей модули магистрали VME ДОЛЖНЫ ОГРАНИЧИВАТЬ нагрузку каждой из линий AS*, DS0*, DS1* до следующих значений:*

- вытекающий ток модуля при напряжении 0,6 В, включая ток утечки, $I_{OZL}+I_{IL}$ — не более 450 мкА;

- втекающий ток модуля при 2,4 В, включая ток утечки, $I_{OZH}+I_{IH}$ — не более 100 мкА;

- общая емкостная нагрузка на сигнал, включая емкость сигнального проводника, C_T — не более 20 пФ.

Замечание 6.5. Втекающий и вытекающий токи, определяемые правилами 6.13 и 6.14, учитывают токи как формирователя, так и приемника, втекающие в модуль и вытекающие из него.

Предложение 6.1. Для возбуждения линий AS*, DS0* и DS1* предлагается использовать микросхемы типов 531АП4 (74S241) или 1531АП4/АП5 (74F241/244). В качестве приемников на линиях AS*, DS0* и DS1* предлагается использовать микросхемы типов 555АП3 (74LS240), 555АП4 (74LS241) или 555АП5 (74LS244).

6.4.2.2 *Требования к характеристикам формирователей с тремя состояниями и нагрузок для стандартных линий A01—A31, D00—D31, AM0—AM5, IACK*, LWORD*, WRITE**

Правило 6.15. *Если модуль магистрали VME возбуждает линии A01—A31, D00—D31, AM0—AM5, IACK*, LWORD* или WRITE*, то формирователи для этих линий ДОЛЖНЫ УДОВЛЕТВОРЯТЬ следующим требованиям:*

- втекающий ток в состоянии низкого уровня I_{OL} — не менее 48 мА;

- напряжение низкого уровня V_{OL} — не более 0,6 В при токе I_{OL} = 48 мА;

- вытекающий ток в состоянии высокого уровня I_{OH} — не менее 3 мА;

- напряжение высокого уровня V_{OH} — не менее 2,4 В при токе I_{OH} = 3 мА;

- минимальный вытекающий ток при коротком замыкании I_{OS} — не менее 50 мА при напряжении 0 В;

- максимальный вытекающий ток при коротком замыкании I_{OS} — не более 225 мА при напряжении 0 В.

Правило 6.16. *При выключении формирователей модули магистрали VME ДОЛЖНЫ ОГРАНИЧИВАТЬ нагрузку каждой из линий A01—A31, D00—D31, AM0—AM5, IACK*, LWORD* и WRITE* до следующих значений:*

- вытекающий ток модуля при напряжении 0,6 В, включая ток утечки, $I_{OZL}+I_{IL}$ — не более 700 мкА;

- втекающий ток модуля при напряжении 2,4 В, включая ток утечки, $I_{OZH}+I_{IH}$ — не более 150 мкА;

- общая емкостная нагрузка на сигнал, включая емкость сигнального проводника, C_T — не более 20 пФ.

Замечание 6.6. Втекающий и вытекающий токи, определяемые правилами 6.15 и 6.16, учитывают токи как формирователей, так и приемников, втекающие в модуль и вытекающие из него.

Предложение 6.2. Для возбуждения линий A01—A31, D00—D31, AM0—AM5, IACK*, LWORD* и WRITE* предлагается использовать микросхемы типов 1533АП8 (74ALS645-1), 1531АП5 (74F244), 1530ИР33 (74AS573) или 74AS580.

В качестве приемников на линиях A01—A31, D00—D31, AM0—AM5, IACK*, LWORD* и WRITE* предлагается использовать микросхемы типов 555АП3 (74LS240), 555АП4 (74LS241) или 555АП5 (74LS244).

В качестве приемопередатчиков на линиях A01—A31, D00—D31, AM0—AM5, IACK*, LWORD* и WRITE* предлагается использовать микросхемы типов 1533АП8 (74ALS645-1), 1533АП6 (74ALS245A-1), 1533АП10 (74ALS646-1) или 74ALS648-1.

6.4.2.3 *Требования к характеристикам формирователей с двумя состояниями и нагрузок для сильноточных линий SERCLK, SYSCLK, BCLR**

Правило 6.17. *Системы магистрали VME ДОЛЖНЫ ИМЕТЬ только один модуль, возбуждающий каждую из линий SERCLK, SYSCLK или BCLR*. Его формирователи для этих линий ДОЛЖНЫ УДОВЛЕТВОРЯТЬ следующим требованиям:*

- втекающий ток в состоянии низкого уровня I_{OL} — не менее 64 мА;

- напряжение низкого уровня V_{OL} — не более 0,6 В при токе I_{OL} = 64 мА;

- вытекающий ток в состоянии высокого уровня I_{OH} — не менее 3 мА;

- напряжение высокого уровня V_{OH} — не менее 2,4 В при токе I_{OH} = 3 мА;

- минимальный вытекающий ток при коротком замыкании I_{OS} — не менее 50 мА при напряжении 0 В;

- максимальный вытекающий ток при коротком замыкании I_{OS} — не более 255 мА при напряжении 0 В.

Правило 6.18. Все модули ДОЛЖНЫ ОГРАНИЧИВАТЬ нагрузку каждой из линий SERCLK, SYSCLK и BCLR* до следующих значений:

- вытекающий ток модуля при напряжении 0,6 В, включая ток утечки, I_{OZL} +I_{IL} — не более 600 мкА;

- втекающий ток модуля при напряжении 2,4 В, включая ток утечки, I_{OZH} +I_{IH} — не более 50 мкА;

- общая емкостная нагрузка на сигнал, включая емкость сигнального проводника, для системных контроллеров, имеющих формирователи, C_T — не более 20 пФ;

- общая емкостная нагрузка на сигнал, включая емкость сигнального проводника, для прочих модулей, не имеющих формирователей, C_T — не более 12 пФ.

Замечание 6.7. Втекающий и вытекающий токи, определяемые в правилах 6.17 и 6.18, учитывают токи как формирователей, так и приемников, втекающие в модуль и вытекающие из него.

Предложение 6.3. Для возбуждения линий SERCLK, SYSCLK и BCLR* ПРЕДЛАГАЕТСЯ использовать микросхемы типов 531АП4 (74S241) или 1531АП4/АП5 (74F241/244). В качестве приемников для линий SERCLK, SYSCLK и BCLR* предлагается использовать микросхемы типов 555АП3 (74LS240), 555АП4 (74LS241) или 555АП5 (74LS244).

6.4.2.4 *Требования к характеристикам формирователей с двумя состояниями и нагрузок для стандартных линий BG0OUT*— BG3OUT*/ BG0IN*— BG3IN*, IACKOUT*/IACKIN**

Правило 6.19. *Если модуль магистрали VME возбуждает линии BG0OUT*—BG3OUT*/BG0IN*—BG3IN* или IACKOUT*/IACKIN*, то его формирователи этих линий ДОЛЖНЫ УДОВЛЕТВОРЯТЬ следующим требованиям:*

- втекающий ток в состоянии низкого уровня I_{OL} — не менее 8 мА;

- напряжение низкого уровня V_{OL} — не более 0,6 В при I_{OL} = 8 мА;

- вытекающий ток в состоянии высокого уровня I_{OH} — не менее 400 мкА;

- напряжение высокого уровня V_{OH} — не менее 2,7 В при I_{OH} = 400 мкА.

Правило 6.20. *Все модули магистрали VME ДОЛЖНЫ ОГРАНИЧИВАТЬ нагрузку каждой из линий BG0OUT*BG3OUT*/BG0IN*—BG3IN* и IACKOUT*/IACKIN* до следующих значений:*

- вытекающий ток модуля при напряжении 0,6 В, включая ток утечки, I_{OZL} + I_{IL} — не более 600 мкА;

- втекающий ток модуля при напряжении 2,4 В, включая ток утечки, I_{OZH} + I_{IH} — не более 50 мкА;

- общая емкостная нагрузка на сигнал, включая емкость сигнального проводника, C_T — не более 20 пФ.

Замечание 6.8. Втекающий и вытекающий токи, определяемые в правилах 6.19 и 6.20, учитывают токи как формирователей, так и приемников, втекающие в модуль и вытекающие из него.

Предложение 6.4. Для возбуждения линий BG0OUT*—BG3OUT*/ BG0IN*—BG3IN* и IACKOUT*/IACKIN* предлагается использовать любые стандартные формирователи, удовлетворяющие указанным выше требованиям. В качестве приемников для линий BG0OUT*—BG3OUT*/ BG0IN*—BG3IN* и IACKOUT*/IACKIN* предлагается использовать микросхемы типов 555АП3 (74LS240), 555АП4 (74LS241) или 555АП5 (74LS244).

6.4.2.5 *Требования к характеристикам формирователей с открытым коллектором и нагрузок для линий BR0*—BR3*, BBSY*, IRQ1*—IRQ7*, DTACK*, BERR*, SYSFAIL*, SYSRESET*, ACFAIL* и IACK**

Правило 6.21. *Если модуль магистрали VME возбуждает линии BR0*—BR3*, BBSY*, IRQ1*—IRQ7*, DTACK*, BERR*, SYSRESET*, SYSFAIL*, ACFAIL* или IACK*, то его формирователи этих линий ДОЛЖНЫ УДОВЛЕТВОРЯТЬ следующим требованиям:*

- втекающий ток в состоянии низкого уровня I_{OL} — не менее 48 мА;

- напряжение низкого уровня V_{OL} — не более 0,6 В при I_{OL} = 48 мА.

Правило 6.22. *Все модули магистрали VME ДОЛЖНЫ ОГРАНИЧИВАТЬ нагрузку каждой из линий BR0*—BR3*, BBSY*, IRQ1*—IRQ7*, DTACK*, BERR*, SYSFAIL*, SYSRESET*, ACFAIL* и IACK* до следующих значений:*

- вытекающий ток модуля при напряжении 0,6 В, включая ток утечки, $I_{OZL} + I_{IL}$ — не более 400 мкА (DTACK и BERR*), не более 600 мкА (все прочие);*

- втекающий ток модуля при напряжении 2,4 В, включая ток утечки, $I_{OZH} + I_{IH}$ — не более 50 мкА;

- общая емкостная нагрузка на сигнал, включая емкость сигнального проводника, C_T — не более 20 пФ.

Замечание 6.9. Втекающий ток модуля, определяемый в правилах 6.21 и 6.22, учитывает токи как формирователей, так и приемников, втекающие в модуль и вытекающие из него.

Предложение 6.5. Для возбуждения линий BR0*—BR3*, BBSY*, IRQ1*—IRQ7*, DTACK*, BERR*, SYSFAIL*, ACFAIL* и IACK* предлагается использовать микросхемы типа 530ЛА13 (74S38). В качестве приемников для линий BR0*—BR3*, BBSY*, IRQ1*—IRQ7*, DTACK*, BERR*, SYSFAIL*, SYSRESET*, ACFAIL* и IACK* предлагается использовать микросхемы типов 555АП3 (74LS240), 555АП4 (74LS241) или 555АП5 (74LS244).

Предложение 6.6. Поскольку большинство ТТЛ-формирователей работает ненадежно, когда параметры источника питания +5 В находятся вне пределов, заданных для них, то в блоке контроля питания для возбуждения линии SYSRESET* предлагается использовать формирователь, построенный на основе транзистора с высоким коэффициентом усиления, работающего в режиме малого сигнала.

6.5 Межсоединения сигнальных линий объединительной платы

Магистраль VME является интерфейсной системой с высокими техническими характеристиками. При ее проектировании необходимо учитывать, что на объединительной плате проявляются эффекты, присущие линиям передачи. При определении требований разделов 2 и 4 к значениям времени установления адресов и данных учитывается то обстоятельство, что большая часть формирователей ненадежно переключает сигнальные линии объединительной платы с низкого на высокий уровень до тех пор, пока не придет отраженный сигнал от конца магистрали. Хотя эти отражения играют определенную полезную роль, они не могут быть избыточными, так как из-за этого возникнет "звон". В следующих ниже пунктах определяются требования к характеристикам объединительной платы, соблюдение которых обеспечивает нужный результат.

6.5.1 Оконечные нагрузки

Правило 6.23. *На каждом из концов всех сигнальных линий, кроме линий последовательных приоритетных цепочек, ДОЛЖНЫ ИСПОЛЬЗОВАТЬСЯ оконечные нагрузки.*

Замечание 6.10. Оконечные нагрузки выполняют следующие 4 функции:

- уменьшают отражения от концов сигнальных линий объединительных плат;
- поднимают напряжение до значения соответствующего высокому уровню для формирователей с открытым коллектором;
- восстанавливают на сигнальных линиях состояние высокого уровня при выключении формирователей с тремя состояниями;
- обеспечивают непрерывный ток для выключения принимающего втекающий ток транзистора формирователя, что вызывает более быстрые положительные перепады на сигнальной линии.

На рисунке 6.2 показана эквивалентная схема оконечной нагрузки. Также показан делитель напряжения, который обеспечивает это значение оконечной нагрузки.

Замечание 6.11. Если максимальные допуски для сопротивлений и напряжения источника в оконечной нагрузке, приведенной на рисунке 6.2, выдерживаются в пределах ± 5 %, то эта схема будет удовлетворять допускам, указанным для эквивалентной схемы.

Замечание 6.12. Оконечная нагрузка, приведенная на рисунке 6.2, будет обеспечивать импеданс эквивалентной схемы только в том случае, если ее источник + 5 В будет надлежащим образом развязан относительно «Земли» с помощью конденсатора развязки по питанию.

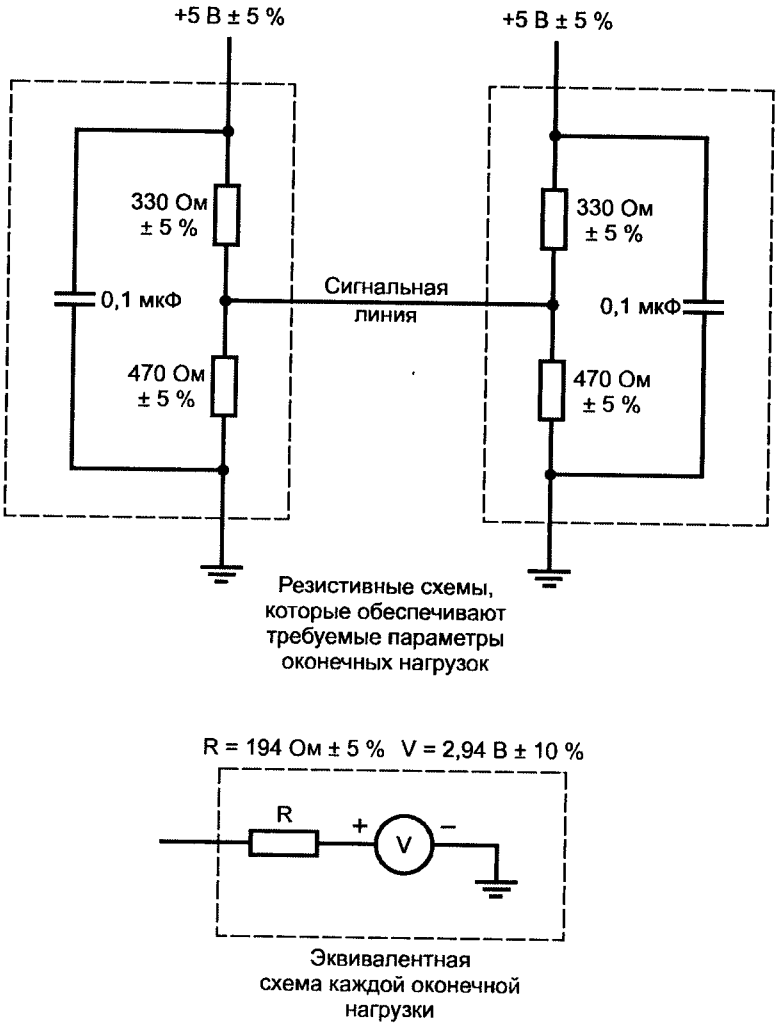


Рисунок 6.2 — Стандартные оконечные нагрузки магистрали

Рекомендация 6.3. Рекомендуется устанавливать конденсатор развязки от 0,01 мкФ до 0,1 мкФ как можно ближе к контакту V_{cc} каждой резисторной сборки оконечных нагрузок.

Разрешение 6.3. Для обеспечения оконечной нагрузки МОГУТ использоваться любые резисторные схемы и источники напряжений, если их эквивалентная схема будет соответствовать показанной на рисунке 6.2.

6.5.2 Волновое сопротивление

Каждая сигнальная линия на объединительной плате имеет соответствующее волновое сопротивление Z_0. Это волновое сопротивление важно потому, что неоднородности линии, связанные с емкостными эффектами и нагрузками в магистрали, и рассогласование между Z_0 и оконечными нагрузками могут вызвать искажения формы сигналов. На рисунке 6.3 показано поперечное сечение микрополосковой сигнальной линии многослойной печатной объединительной платы. Z_0 является функцией ширины и толщины линии, толщины диэлектрика и его относительной диэлектрической проницаемости. На рисунке 6.4 представлена зависимость волнового сопротивления от ширины микрополосковой линии для стеклотекстолитовой платы общепринятой толщины. Оконечные нагрузки на сигнальных линиях магистрали VME уменьшают искажение формы проходящих по ним сигналов. Хотя идеального согласования волновых сопротивлений (которое бы полностью исключало искажения, вызванные отражениями) оконечных нагрузок и сигнальных линий не достигается, важно не допустить слишком большого рассогласования, например в случае, когда величина Z_0 сигнальной линии слишком мала.

Рекомендация 6.4. При проектировании объединительной платы рекомендуется выбирать такую ширину сигнальной линии и толщину платы, которые дают значение Z_0, определяемое с помощью рисунка 6.4, максимально приближающееся к 100 Ом.

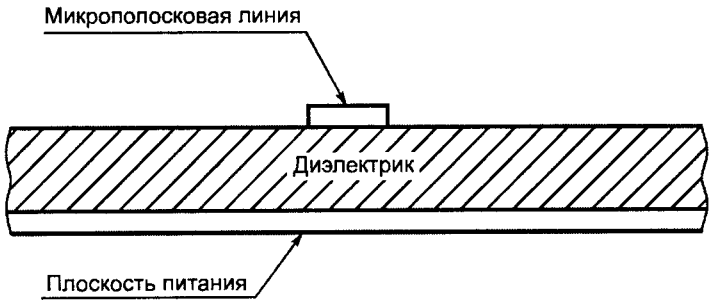
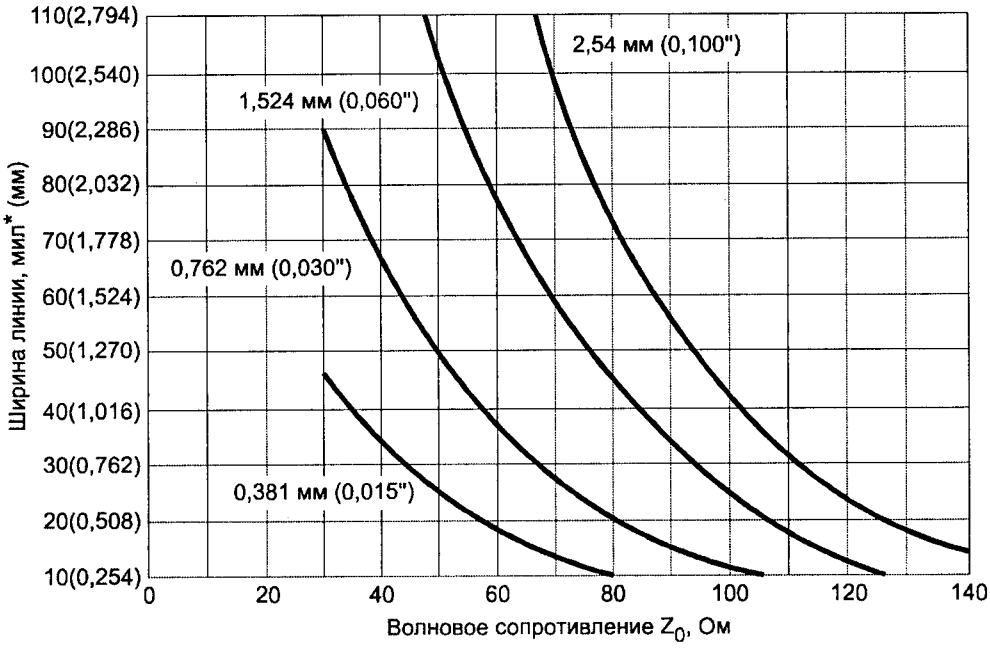


Рисунок 6.3 — Поперечное сечение микрополосковой сигнальной линии объединительной платы



* 1 мил = 0,001"

Толщина микрополосковых линий составляет 0,038мм (0,0015"). Материал G-10 (СФ-2-35-1,5) с диэлектрической проницаемостью 4,7

Рисунок 6.4 — Зависимость волнового сопротивления Z_0 от ширины сигнальной линии

Фактическое волновое сопротивление сигнальной линии объединительной платы называется эффективным волновым сопротивлением (Z_0'). Оно будет ниже 100 Ом из-за емкости сквозных металлизированных отверстий и контактов соединителей. Хотя металлизированные отверстия необходимы для установки соединителей, количество других отверстий должно быть сведено к минимуму.

Эффективное волновое сопротивление сигнальной линии объединительной платы (без вставленных в нее модулей) может быть рассчитано по формуле

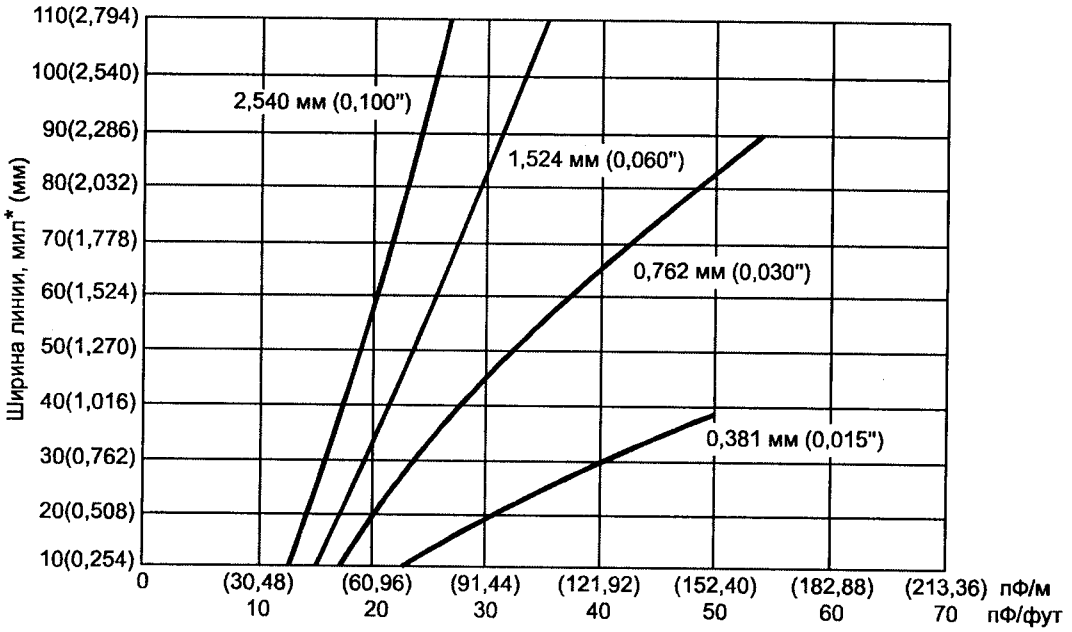
$$Z_0' = \frac{Z_0}{\sqrt{1 + C_d/C_0}},$$

где Z_0 — волновое сопротивление микрополосковой линии без учета нагрузочных эффектов вставляемых модулей, соединителей и сквозных металлизированных отверстий (см. рисунок 6.4);

C_d — распределенная емкость металлизированных сквозных отверстий и соединителей объединительной платы на единицу длины линии;

C_0 — собственная емкость микрополосковой линии на единицу длины без учета нагрузочных эффектов вставляемых модулей, соединителей и сквозных металлизированных отверстий (см. рисунок 6.5);

Z_0' — волновое сопротивление сигнальной линии объединительной платы, включая нагрузочное влияние соединителей и сквозных металлизированных отверстий, но без учета нагрузочного влияния вставляемых модулей.



* 1 мил = 0,001"

Толщина микрополосковых линий составляет 0,038 мм (0,0015"). Материал G-10 (СФ-2-35-1,5) с диэлектрической проницаемостью 4,7

Рисунок 6.5 — Зависимость погонной емкости C_0 от ширины сигнальной линии

Замечание 6.13. Типичные значения Z_0' для объединительной платы магистрали VME без вставленных модулей составляют 50—60 Ом. Если это волновое сопротивление составляет 50 Ом или выше, то будет обеспечена удовлетворительная работа.

6.5.3 Дополнительные сведения

Правило 6.24. *Длина печатных проводников от 96-контактных соединителей до внутримодульных схем НЕ ДОЛЖНА ПРЕВЫШАТЬ 50,8 мм (2").*

Замечание 6.14. Если печатный проводник от 96-контактного соединителя до внутримодульных схем имеет ветвление, то длина ответвлений складывается, а полученная суммарная длина удовлетворяет правилу 6.24.

Правило 6.25. *Каждая из линий SYSCLK и SERCLK ДОЛЖНА ВОЗБУЖДАТЬСЯ только одним формирователем.*

Правило 6.26. *Если в системе имеется функциональный блок формирователя тактового сигнала SYSCLK или функциональный блок формирователя SERCLK, то они ДОЛЖНЫ БЫТЬ УСТАНОВЛЕНЫ в гнездо 1 объединительной платы.*

Замечание 6.15. Размещение формирователей тактовых сигналов на модуле в гнезде 1 сводит к минимуму искажение формы их сигналов, вызываемое отражениями от концов линий объединительной платы.

Предложение 6.7. Если фактические величины емкостной нагрузки в технических данных изготовителя отсутствуют, то для оценки суммарной емкостной нагрузки модуля магистрали VME можно использовать следующие значения:

- типичная емкость приемника 3 — 5 пФ

- типичная емкость формирователя 10 — 12 пФ
- типичная емкость приемопередатчика 15 — 18 пФ
- типичная емкость печатного проводника длиной 50,8 мм (2") 2 — 3 пФ.

Замечание 6.16. Схемные проводники, проходящие параллельно друг другу, например, на объединительной плате, иногда индуцируют взаимные изменения уровней сигналов. Это явление широко известно под названием перекрестной помехи. При проектировании объединительных плат следует помнить, что расстояние между линиями и их расположение относительно плоскостей питания и "Земли" оказывают существенное влияние на величину перекрестных помех.

Предложение 6.8. Задержки распространения сигналов через формирователи обычно зависят от их нагрузки, а сигнальные линии магистрали VME обычно представляют значительные нагрузки. Это следует учитывать при расчете временных соотношений между сигналами для наихудшего случая. Если в технических данных изготовителя для формирователя указана задержка распространения сигнала для нагрузки 300 пФ, предлагается использовать это значение при расчете для наихудшего случая. Если же значения задержки распространения даны только для нагрузки 30 пФ, к значению задержки распространения следует прибавить 10 нс, к задержке включения — 15 нс.

6.6 Сигналы, определяемые пользователем

Рекомендация 6.5. Если модуль имеет 96-контактный соединитель Р2, рекомендуется не допускать, чтобы на любых контактах Р2 возбуждалось напряжение, превышающее ± 15 В. Это снизит вероятность серьезного повреждения системы в случае короткого замыкания сигнального проводника, подводимого к одному из этих контактов, на любую другую сигнальную линию.

6.7 Формирователи и оконечные нагрузки сигнальных линий

В данном подразделе приводится сводная таблица типов формирователей, которые должны использоваться для каждой из сигнальных линий магистрали.

В целях упрощения таблицы 6.3 используются следующие сокращения для описания различных типов формирователей:

с двумя состояниями (сильноточный) — TP HC—totem—pole (high current)
с двумя состояниями (стандартный) — TP STD — totem—pole (standard)
с тремя состояниями (сильноточный) — 3 HC — three—state (high current)
с тремя состояниями (стандартный) — 3 STD — three—state (standard)
с открытым коллектором — OC — open—collector

Подробные технические требования к характеристикам приведены в подразделе 6.4.

Таблица 6.3 — Сводная таблица типов формирователей магистрали

Обозначение сигналов или сигнальных линий	Наименование сигналов или сигнальных линий	Тип формирователя	Наличие оконечной нагрузки и выхода на магистральные линии
A01—A31 (31 линия)	Линия адреса	3 STD	Да
ACFAIL*	Отказ питания переменного тока	OC	Да
AM0—AM5 (6 линий)	Модификатор адреса	3 STD	Да
AS*	Адресный строб	3 HC	Да
BBSY*	Шина занята	OC	Да
BCLR*	Очистить шину	TP HC	Да
BERR*	Ошибка в магистрали	OC	Да
BG0IN*—BG3IN*/ BG0OUT*—BG3OUT* (цепочка)	Цепочка предоставления шины	TP STD	Нет

Окончание таблицы 6.3

Обозначение сигналов или сигнальных линий	Наименование сигналов или сигнальных линий	Тип формирователя	Наличие оконечной нагрузки и выхода на магистральные линии
BR0*—BR3* (4 линии)	Запрос на использование шины	ОС	Да
D00—D31 (32 линии)	Шина пересылки данных	3 STD	Да
DS0*—DS1* (2 линии)	Стробы данных	3 НС	Да
DTACK*	Подтверждение пересылки данных	ОС	Да
IACK*	Подтверждение прерывания	3 STD или ОС	Да
IACKIN*/IACKOUT* (цепочка)	Цепочка подтверждения прерывания	ТР STD	Нет
IRQ1*—IRQ7* (7 линий)	Запрос прерываний	ОС	Да
LWORD*	Длинное слово	3 STD	Да
RESERVED	Зарезервировано		Да
SERCLK	Тактовый сигнал последовательной магистрали	ТР НС	Да
SERDAT*	Последовательные данные	ОС	Да
SYSCLK	Системный тактовый сигнал	ТР НС	Да
SYSFAIL*	Системный отказ	ОС	Да
SYSRESET*	Системный сброс	ОС	Да
WRITE*	Запись	3 STD	Да

7 Требования к механическим конструкциям

7.1 Введение

Соблюдение требований, содержащихся в данном разделе, гарантирует механическую совместимость модулей, объединительных плат, каркасов и всех других сопутствующих механических принадлежностей магистрали VME.

Размеры механических конструкций, приведенные в этом разделе, соответствуют требованиям стандартов МЭК [1], [2], [3]. Электрические характеристики соединителей в том виде, как они изложены в разделах 5 и 6, заменяют соответствующие им характеристики публикации МЭК [3], где они отличаются.

Стандарт МЭК [3] описывает семейство соединителей, имеющих обозначение 603-2-IEC-C096 xx-xxx. Все соединители P1/J1 и P2/J2, которые используются в модулях и объединительных платах магистрали VME, относятся к этому семейству. В этом разделе обозначение 603-2-IEC-xxxxxx-xxx используется при упоминании всех типов соединителей, относящихся к этому семейству, как группы. Обозначение 603-2-IEC-C096Mx-xxx (вилка) используется при упоминании 96-контактных соединителей штыревого типа, которые устанавливаются на печатных узлах модулей. Обозначение 603-2-IEC-

С096Fx-ххх (розетка) используется при упоминании 96-контактных соединителей гнездового типа, которые устанавливаются на объединительной плате.

На рисунке 7.1 показан вид спереди каркаса шириной 482,6 мм (19"), на котором видно, как в нем располагаются модули одинарной и двойной высоты. Модули устанавливаются в каркас спереди вертикально, печатный узел ориентирован стороной элементов вправо.

Разрешение 7.1. Система магистрали VME может быть скомпонована из модулей одинарной высоты, двойной высоты или их сочетания.

***Правило** 7.1. Каркасы с модулями одинарной высоты ДОЛЖНЫ ИМЕТЬ объединительную плату с одним рядом соединителей J1.*

***Правило** 7.2. Каркасы для модулей двойной высоты ДОЛЖНЫ ИМЕТЬ три варианта исполнения:*

а) объединительная плата с соединителями J1 в верхней части каркаса;

б) объединительная плата с соединителями J1 в верхней части каркаса и объединительная плата с соединителями J2 в нижней части каркаса;

в) объединительная плата двойной высоты с соединителями J1 и J2.

***Правило** 7.3. Объединительные платы магистрали VME НЕ ДОЛЖНЫ ИМЕТЬ более 21 гнезда для модулей.*

Разрешение 7.2. В случае, если объединительная плата имеет менее 21 гнезда, каркас МОЖЕТ быть меньше по ширине, чем стандартный размер 482,6 мм.

***Правило** 7.4. Ширина каркаса изменяется в зависимости от необходимого количества гнезд. Все его остальные размеры ДОЛЖНЫ СООТВЕТСТВОВАТЬ указанным в настоящем разделе, чтобы обеспечить взаимную механическую совместимость модулей и каркасов.*

7.2 Печатные платы модулей магистрали VME

Рекомендация 7.1. Рекомендуется изготавливать печатные платы модулей толщиной (1,6 ± 0,2) мм.

Замечание 7.1. Толщина печатной платы важна потому, что конструкция направляющих рассчитана именно на этот размер. Более толстые платы могут не войти в направляющие, а платы с меньшей толщиной могут плохо попадать в соединители J1 и J2 на объединительной плате.

Замечание 7.2. Конструкция соединителя 603-2-IEC-хххххх-ххх имеет определенный размер между направляющими кромками соединителя и осевой линией каждого контакта. Это обеспечивает точную центровку контактов соединителей P1, P2 на модулях и J1, J2 на объединительной плате.

Разрешение 7.3. Толщина печатной платы модуля магистрали VME МОЖЕТ быть больше 1,6 мм, если:

а) толщина верхней и нижней кромок печатной платы, которые входят в направляющие, уменьшена до 1,6 мм на размере 2,5 мм вдоль кромки (см. рисунки 7.2 и 7.3);

б) опорная поверхность печатной платы, предназначенная для установки соединителя по стандарту МЭК [3], находится на расстоянии 4,07 мм от межмодульной разделительной плоскости (см. рисунок 7.5).

Установлены два стандартных размера печатных плат модулей магистрали VME: одинарной высоты и двойной высоты (см. рисунки 7.2, 7.3).

7.2.1 Платы модулей одинарной высоты

Замечание 7.3. Размеры платы одинарной высоты: высота 100 мм, глубина 160 мм, площадь приблизительно 16000 мм^2.

***Правило** 7.5. Все платы одинарной высоты ДОЛЖНЫ БЫТЬ СПРОЕКТИРОВАНЫ в соответствии с размерами, приведенными на рисунке 7.2.*

***Правило** 7.6. Расположение отверстий для контактов 96-контактного соединителя 603-2-IEC-CO96Mx-ххх P1 ДОЛЖНО СООТВЕТСТВОВАТЬ рисунку 7.2.*

Предложение 7.1. Предлагается использовать расположение координатной сетки печатной платы в соответствии с рисунком 7.2.

Замечание 7.4. Многие изготовители поставляют модульное оборудование, которое при монтаже по сетке, показанной на рисунке 7.2, надлежащим образом совмещается с сеткой передней панели, приведенной на рисунке 7.7.

Разрешение 7.4. Координатная сетка для других радиоэлементов (кроме соединителей 603-2-IEC-CO96Mx-ххх) МОЖЕТ не совпадать с координатной сеткой отверстий под контакты этих соединителей.

7.2.2 Платы модулей двойной высоты

Замечание 7.5. Размеры платы двойной высоты: высота 233,35 мм, глубина 160 мм, площадь приблизительно 37340 мм².

Правило 7.7. *Все платы двойной высоты ДОЛЖНЫ БЫТЬ СПРОЕКТИРОВАНЫ в соответствии с размерами, приведенными на рисунке 7.3.*

Правило 7.8. *Расположение отверстий под контакты 96-контактного соединителя 603-2-IEC-CO96Mx-xxx P1 ДОЛЖНО СООТВЕТСТВОВАТЬ рисунку 7.3.*

Правило 7.9. *Если в качестве соединителя P2 используется тип 603-2-IEC-CO96Mx-xxx, то расположение отверстий для контактов ДОЛЖНО СООТВЕТСТВОВАТЬ рисунку 7.3.*

Замечание 7.6. Как и в случае с печатной платой одинарной высоты (7.2.1), взаимное расположение координатной сетки на плате по рисунку 7.3 и модульных элементов передней панели модуля соответствуют рисунку 7.8.

Разрешение 7.5. Координатная сетка для других радиоэлементов (кроме соединителей 603-2-IEC-xxxxxx-xxx) МОЖЕТ не совпадать с координатной сеткой отверстий под контакты этих соединителей.

Замечание 7.7. Между координатными сетками (шаг 2,54 мм) верхней и нижней половины платы имеется разрыв 1,27 мм.

7.2.3 Соединители печатных плат модулей

Платы одинарной высоты имеет только один соединитель на задней кромке, который обозначается P1. Платы двойной высоты имеют один или два соединителя на задней кромке. В случае одного соединителя он обозначается P1 и расположен в верхней части задней кромки. В случае двух соединителей верхний обозначается P1, а нижний — P2.

Правило 7.10. *Соединители P1 и P2 печатных плат модулей магистрали VME ДОЛЖНЫ СООТВЕТСТВОВАТЬ требованиям по механике для соединителей 603-2-IEC-CO96Mx-xxx класса 2 или УДОВЛЕТВОРЯТЬ более жестким требованиям и ДОЛЖНЫ УСТАНАВЛИВАТЬСЯ согласно рисунку 7.4.*

Замечание 7.8. Соединители по стандарту МЭК [3] класса 2 обеспечивают минимум 400 циклов сочленения/расчленения.

Замечание 7.9. Символ симметричности в рамке под каждой платой на рисунке 7.4 устанавливает верхний предел того, на сколько центральная линия соединителя может быть наклонена по отношению к нижней кромке платы. Это значение определено в правиле 7.11.

Правило 7.11. *Расстояние в перпендикулярном направлении (d1) от нижней кромки платы до точки A, как указано на рисунке 7.4, НЕ ДОЛЖНО ОТЛИЧАТЬСЯ от расстояния (d2) до точки B более чем на 0,3 мм.*

Правило 7.12. *Если при проектировании модуля магистрали VME центральный ряд контактов соединителя P2 предполагается использовать для расширения шины адреса или шины пересылки данных или если модуль магистрали VME требует увеличения потребляемой мощности больше, чем может обеспечить соединитель P1, то ДОЛЖЕН ИСПОЛЬЗОВАТЬСЯ 96-контактный соединитель P2 типа 603-2-C096Mx-xxx, который ДОЛЖЕН УСТАНАВЛИВАТЬСЯ согласно рисунку 7.4.*

Разрешение 7.6. Если не требуется расширение шины адреса и шины пересылки данных и если плата не требует увеличения мощности питания, превышающей возможности соединителя P1, то на платах двойной высоты в качестве соединителя P2 МОЖНО использовать любой соединитель группы 603-2-IEC-xxxxxx-xxx или спроектировать ее вообще без соединителя P2.

Разрешение 7.7. В платах двойной высоты два внешних ряда контактов соединителя P2 МОЖНО использовать по усмотрению потребителя (см. 7.6.2).

Разрешение 7.8. Кабели ввода/вывода МОГУТ подсоединяться к передней кромке печатной платы. При этом тип соединителей не регламентируется.

Предложение 7.2. По возможности предлагается избегать кабельного подключения к передней кромке печатной платы модуля. Это упрощает ее установку и извлечение из каркаса во время обслуживания.

7.2.4 Компоновка модуля

В общем случае печатный узел состоит из печатной платы с одним или двумя соединителями типа 603-2-IEC-xxxxxx-xxx на ее задней кромке, электрорадиоэлементов и передней панели с рукоятками (при необходимости). Более подробную информацию о передних панелях содержит 7.3.

Правило 7.13. *Пайки, проводники и элементы на печатной плате НЕ ДОЛЖНЫ РАСПОЛАГАТЬСЯ ближе 2,5 мм от ее верхней и нижней кромок, чтобы обеспечить гарантированный зазор между ними и направляющими. Эти размеры указаны на рисунках 7.2 и 7.3.*

На рисунке 7.5 изображен поперечный разрез печатной платы модуля, его передняя панель, задний соединитель и объединительная плата. Номинальные размеры на рисунке находятся в соответствии с размерами, указанными на других рисунках этого раздела.

7.2.5 Ширина модуля

Разрешение 7.9. Модули по ширине могут занимать несколько гнезд. Модули, предназначенные для размещения в одном гнезде каркаса, называются модулями одинарной ширины.

7.2.6 Коробление печатной платы модуля, длина выводов и высота элементов

При изготовлении печатных плат иногда возникает их коробление.

Правило 7.14. *Суммарный размер, учитывающий коробление печатной платы и длину выводов элементов, НЕ ДОЛЖЕН ПРЕВЫШАТЬ 2,47 мм от стороны паек идеальной (неискривленной) печатной платы, а суммарный размер, учитывающий коробление печатной платы и высоту элементов (в противоположном направлении), НЕ ДОЛЖЕН ПРЕВЫШАТЬ [13,71 + 20,32 (N-1)] мм от стороны элементов идеальной печатной платы,где N— число гнезд,занимаемых модулями.*

Замечание 7.10. В процессе установки модуля в каркас выводы элементов могут касаться правой кромки передней панели соседнего модуля. Поэтому следует соблюдать осторожность, чтобы избежать повреждения выводов.

Предложение 7.3. Где это возможно, предлагается уменьшить длину выводов элементов до 1,52 мм. Это облегчит установку и извлечение модуля. Кроме того, за счет этого допустимо увеличение искривления платы модуля.

Замечание 7.11. Размеры, приведенные в предложении 7.3, гарантируют зазор, по меньшей мере, 2,54 мм между элементами платы модуля и выводами элементов соседней платы справа. Это пространство обеспечивает прохождение надлежащего воздушного потока и уменьшает опасность нежелательных электрических контактов при вибрации.

Правило 7.15. *После сборки модуля магистрали VME ДОЛЖНЫ БЫТЬ ИЗМЕРЕНЫ размеры коробления платы, высоты выводов и элементов, чтобы убедиться в том, что их суммарный размер не выходит за установленные пределы. Для надлежащего выполнения этих измерений модуль ДОЛЖЕН БЫТЬ ВСТАВЛЕН в каркас (или подобное ему контрольное приспособление).*

На рисунке 7.6 показан модуль (одинарной или двойной высоты), установленный в каркас, и указаны размеры, которые надлежит контролировать. Межмодульные разделительные плоскости являются базой, от которой ведется отсчет при выполнении измерений.

Замечание 7.12. Для ускорения проверки размеров, указанных на рисунке 7.6, полезно иметь специальное контрольное приспособление, имитирующее каркас.

7.3 Передние панели

В этом подразделе приводятся требования к механическим конструкциям передних панелей для модулей одинарной и двойной высоты и сопутствующим конструктивным элементам.

Разрешение 7.10. Модули магистрали VME могут изготавливаться с передними панелями или без них.

Рекомендация 7.2. Передние панели и сопутствующие конструктивные элементы рекомендуется использовать для повышения устойчивости модулей магистрали VME к вибрациям и для формирования охлаждающего воздушного потока через каркас.

Правило 7.16. *Если передние панели используются, то они ДОЛЖНЫ ИМЕТЬ крепежные винты M2,5 × 0,45, расположенные в верхней и нижней частях для крепления к каркасу (см. рисунок 7.7).*

На рисунке 7.7 изображена передняя панель модуля одинарной высоты и ширины. На рисунке 7.8 изображена передняя панель модуля двойной высоты и одинарной ширины. Размер и расположение координатной сетки на задней плоскости передних панелей совпадают с координатной сеткой печатных плат, изображенных на рисунках 7.2 и 7.3 соответственно.

Предложение 7.4. Элементы на передней панели, такие как светодиоды и переключатели, предлагается устанавливать в узлах координатной сетки.

7.3.1 Рукоятки

Разрешение 7.11. Передние панели модулей магистрали VME МОГУТ быть с рукоятками или без них.

Рекомендация 7.3. Для удобства извлечения модулей магистрали VME из каркаса на передних панелях рекомендуется устанавливать рукоятки.

Замечание 7.13. Рукоятки различных изготовителей несколько отличаются по форме.

Предложение 7.5. Предлагается выбирать рукоятки, размеры которых по глубине и высоте соответствуют указанным на рисунках 7.7 и 7.8.

Рекомендация 7.4. При сборке рекомендуется устанавливать рукоятки в одном или нескольких местах передней панели модулей магистрали VME, указанных на рисунках 7.7, 7.8, 7.11 и 7.12.

Разрешение 7.12. На модулях одинарной высоты рукоятки МОГУТ быть установлены в следующих комбинациях:

а) только сверху,

б) только снизу,

в) сверху и снизу.

Разрешение 7.13. На модулях двойной высоты рукоятки МОГУТ быть установлены в следующих комбинациях:

- только сверху,
- только в середине,
- только снизу,
- сверху и в середине,
- снизу и в середине,
- сверху и снизу.

Замечание 7.14. Когда модули двойной и одинарной высоты устанавливаются в один каркас, средние рукоятки модуля двойной высоты находятся на одном уровне с рукоятками модуля одинарной высоты, не образуя ломаной линии, что улучшает внешний вид.

Замечание 7.15. Усилие извлечения модуля двойной высоты, имеющего оба разъема P1 и P2, составляет 180 Н. Установка на нем рукояток сверху и снизу максимально облегчит его извлечение из каркаса.

7.3.2 Крепление передних панелей

Рекомендация 7.5. Если используется передняя панель, рекомендуется на плате модуля оставлять площадки, показанные на рисунках 7.9 и 7.10, свободные от элементов, чтобы разместить угольники для крепления этой панели. Отверстие для крепления угольника рекомендуется располагать в соответствии с рисунками 7.2 и 7.3.

Рекомендация 7.6. Если передняя панель устанавливается на модуле двойной высоты, рекомендуется использовать для ее крепления дополнительно хотя бы один из средних угольников, изображенных на рисунке 7.10.

Правило 7.17. *Если используется передняя панель, то ДОЛЖЕН БЫТЬ ВЫДЕРЖАН контрольный размер, показанный на рисунках 7.9, 7.10 (от задней плоскости передней панели до задней плоскости соединителя).*

Замечание 7.16. Контрольный размер от задней плоскости передней панели до передней плоскости объединительной платы гарантирует полную сочленяемость соединителей P1 и P2. При этом крепежные винты плотно прижмут переднюю панель модуля к каркасу.

7.3.3 Размеры передней панели

Отсчет размеров передней панели ведется от базовой точки, расположенной на 0,15 мм левее ее верхнего левого угла, если смотреть спереди.

Правило 7.18. *Размеры передних панелей одинарной ширины ДОЛЖНЫ СООТВЕТСТВОВАТЬ указанным на рисунках 7.7 и 7.8.*

Рекомендация 7.7. Номинальная толщина передних панелей 2,5 мм.

Замечание 7.17. Ширина передней панели (20,02 мм) одногнездового модуля на 0,3 мм меньше шага расположения гнезд (20,32 мм). Это обеспечивает гарантированный зазор между соседними панелями и согласуется с допусками на печатные узлы и каркасы.

Правило 7.19. *Если модуль занимает более одного гнезда и имеет переднюю панель, то ширина его передней панели ДОЛЖНА БЫТЬ [20,02 + 20,32 (N—1)] мм, где N— число гнезд, занимаемых модулем.*

Правило 7.20. *Передняя панель одногнездового модуля ДОЛЖНА ИМЕТЬ два крепежных винта: один сверху, другой снизу, как показано на рисунках 7.7 и 7.8.*

7.3.4 Панели-заглушки

Панели-заглушки устанавливаются в том случае, если в передней панели каркаса образуются значительные зазоры, например, при наличии пустых гнезд. Панели-заглушки не требуют установки крепежных угольников, т.к. модули отсутствуют. Эти панели крепятся к каркасу винтами или поворотными защелками, расположенными сверху и снизу, как в передних панелях модулей магистрали VME.

Правило 7.21. *Панели-заглушки ДОЛЖНЫ БЫТЬ СПРОЕКТИРОВАНЫ в соответствии с размерами, указанными на рисунках 7.11 и 7.12.*

Правило 7.22. *Панель-заглушка одинарной ширины ДОЛЖНА ИМЕТЬ два винта, расположенных сверху и снизу, как показано на рисунках 7.11 и 7.12.*

Рекомендация 7.8. Рекомендуется использовать панели-заглушки, т.к. они улучшают условия охлаждения и внешний вид системы магистрали VME.

Предложение 7.6. На панелях-заглушках предлагается устанавливать рукоятки. Это улучшает внешний вид системы магистрали VME, поскольку все модули магистрали VME, вставленные в каркас, имеют рукоятки.

Предложение 7.7. В панелях-заглушках шириной более 101,6 мм для их надежного крепления к каркасу предлагается предусматривать дополнительные крепежные отверстия.

7.3.5 Эжекторы и инжекторы модулей

Замечание 7.18. Разные изготовители предлагают различные типы устройств, облегчающих установку/извлечение модулей магистрали VME (инжекторы/эжекторы).

Замечание 7.19. Усилие сочленения одного соединителя 603-2-C096Mx-xxx, может доходить до 90 Н.

Разрешение 7.14. Модули магистрали VME МОГУТ снабжаться любыми эжекторами, инжекторами или держателями при условии, что они не влияют на совместимость этих модулей с другими модулями или каркасами, выполненными в соответствии с требованиями настоящего стандарта.

7.4 Объединительные платы

Основная объединительная плата обозначается как объединительная плата J1. В некоторых случаях это единственная объединительная плата системы магистрали VME. Когда используется каркас двойной высоты, такая объединительная плата устанавливается в его верхней части. Когда используется система магистрали VME повышенной разрядности, вторая объединительная плата, обозначаемая как объединительная плата J2, устанавливается под первой в нижней части каркаса. В этой плате для образования магистральных линий используется средний ряд контактов J2 (ряд *b*), а два внешних ряда (ряды *a* и *c*) могут использоваться либо для реализации подсистемной шины VSB [4] магистрали VME, либо для других функций по усмотрению пользователя. Термин J1/J2 используется для обозначения комбинированной объединительной платы, которая имеет оба ряда соединителей J1 и J2.

Гнезда для модулей обозначаются номерами 1,2,3...21, начиная с левой стороны каркаса, если смотреть спереди. Направление распространения сигнала по цепочке от гнезда 1 к гнезду 21.

Правило 7.23. *Объединительные платы J1 ДОЛЖНЫ РАСПРОСТРАНЯТЬ все сигналы во все гнезда, по магистральным линиям, за исключением сигналов цепочек (см. 7.6.1).*

Правило 7.24. *В объединительных платах J2 или J1/J2, используемых для пересылок 32-разрядных адреса и данных, сигналы ДОЛЖНЫ РАСПРОСТРАНЯТЬСЯ по магистральным линиям на все контакты центрального ряда (ряда b) тех гнезд, для которых имеются соединители (см. 7.6.2).*

Правило 7.25. *Во всех объединительных платах J1 и J2 магистрали VME ДОЛЖНЫ ИСПОЛЬЗОВАТЬСЯ 96-контактные соединители 603-2-IEC-CO96Fx-xxx.*

Правило 7.26. *Во всех объединительных платах J1 ДОЛЖНЫ ПРЕДУСМАТРИВАТЬСЯ возможность установки каких-либо перемычек для обеспечения непрерывности цепочек подтверждения прерывания и предоставления шины в гнездах, где не вставлены модули.*

Предложение 7.8. Чтобы обеспечить замыкание цепочек на объединительной плате J1, предлагается использовать соединители 603-2-IEC-C096Fx-xxx, которые имеют выводы для монтажа накруткой.

Предложение 7.9. Если используются соединители без выводов под накрутку, то контакты перемычек предлагается располагать рядом с подлежащими замыканию контактами соединителей J1.

Рекомендация 7.15. Для улучшения качества заземления в расширенных конфигурациях систем рекомендуется использовать комбинированную объединительную плату J1/J2.

Предложение 7.10. На объединительной плате J2 предлагается использовать соединители 603-2-IEC-C096Fx-xxx с выводами под накрутку. Это позволит подсоединять к таким выводам гибкие плоские кабели и дополнительные объединительные платы.

7.4.1 Требования к размерам объединительной платы

На рисунке 7.13 изображена 21-гнездовая объединительная плата одинарной высоты, а на рисунке 7.15 — 21-гнездовая объединительная плата двойной высоты.

Разрешение 7.16. Объединительная плата МОЖЕТ иметь до 21 гнезда.

Рекомендация 7.9. Ширину объединительных плат, спроектированных менее чем на 21 гнездо, рекомендуется рассчитывать по формулам:

$$[(N \times 20{,}32) - 1{,}44] \text{ мм} + 0/-0{,}3 \text{ мм}$$

или

$$[(N \times 20{,}32) + 0/-0{,}3] \text{ мм},$$

где N — число гнезд.

Такие размеры позволяют располагать объединительные платы в каркасе рядом без потери места для гнезд.

Рекомендация 7.10. Не рекомендуется проектировать объединительные платы шириной более чем (425,28 +0/—0,3) мм. Они не устанавливаются в каркасы стандартного размера, выпускаемые многими изготовителями.

Разрешение 7.17. Объединительные платы магистрали VME МОГУТ быть с резьбовыми шпильками для подключений кабеля питания или без них.

Правило 7.27. *Объединительные платы J1 и J2 магистрали VME ДОЛЖНЫ БЫТЬ СПРОЕКТИРОВАНЫ в соответствии с размерами, указанными на рисунках 7.13, 7.14, за исключением ширины, которая зависит от количества гнезд.*

Правило 7.37. *Объединительные платы J1/J2 магистрали VME ДОЛЖНЫ БЫТЬ СПРОЕКТИРОВАНЫ в соответствии с размерами, указанными на рисунках 7.15, 7.16, за исключением ширины, которая зависит от количества гнезд.*

Замечание 7.20. Размеры, указанные на рисунке 7.14, повторяются через каждые 20,32 мм.

7.4.2 Оконечные нагрузки сигнальных линий

Правило 7.28. *Объединительные платы магистрали VME ДОЛЖНЫ ИМЕТЬ оконечные нагрузки для всех сигнальных линий, указанных в подразделе 6.7.*

Замечание 7.25. Если оконечные нагрузки встроены в объединительную плату, то обеспечивается лучшее качество сигнала, чем если бы они находились на отдельных подключаемых платах.

Рекомендация 7.15. Оконечные нагрузки рекомендуется встраивать в объединительную плату.

Правило 7.29. *Длина сигнальных проводников объединительной платы, включая любые платы подключаемых нагрузок, НЕ ДОЛЖНА ПРЕВЫШАТЬ 508 мм.*

7.5 Сборка каркасов магистрали VME

Этот подраздел содержит сведения о сборке каркасов. Все горизонтальные размеры отсчитываются от левой кромки проема каркаса.

7.5.1 Каркасы и ширина гнезд

На рисунке 7.19 показан типичный каркас двойной высоты на 21 гнездо.

Разрешение 7.19. В каркас двойной высоты МОЖНО устанавливать только модули двойной высоты. Такой каркас МОЖНО также разделить с помощью профильных планок на две секции одинарной высоты, расположенные одна над другой.

Замечание 7.22. Профильная планка, указанная в разрешении 7.19, имеет сверху и снизу направляющие для модулей, устанавливаемых в верхнюю и нижнюю секции.

Предложение 7.11. По возможности предлагается располагать крайние левые направляющие модуля (гнездо 1) так, чтобы ось их находилась на расстоянии 3,27 мм от кромки проема каркаса. Это обеспечит необходимый зазор между выводами элементов платы модуля, установленного в гнездо, и кромкой проема, а также экономит место в горизонтальном направлении. (Если это расстояние будет больше, то справа невозможно будет разместить 21 гнездо).

7.5.2 Размеры каркаса

Правило 7.30. *Все каркасы двойной высоты ДОЛЖНЫ СООТВЕТСТВОВАТЬ размерам, указанным на рисунке 7.19, за исключением ширины, которая зависит от количества гнезд.*

Правило 7.31. *Все каркасы одинарной высоты ДОЛЖНЫ СООТВЕТСТВОВАТЬ размерам, указанным на рисунке 7.19, за исключением ширины, которая зависит от количества гнезд, и вертикального размера между нижними и верхними направляющими, который составляет (100,2 + 0,4/—0) мм, вместо 233,35 мм.*

Замечание 7.23. Размер от задней опорной поверхности передней панели до передней плоскости объединительной платы особенно важен, так как он гарантирует правильное сочленение соединителей.

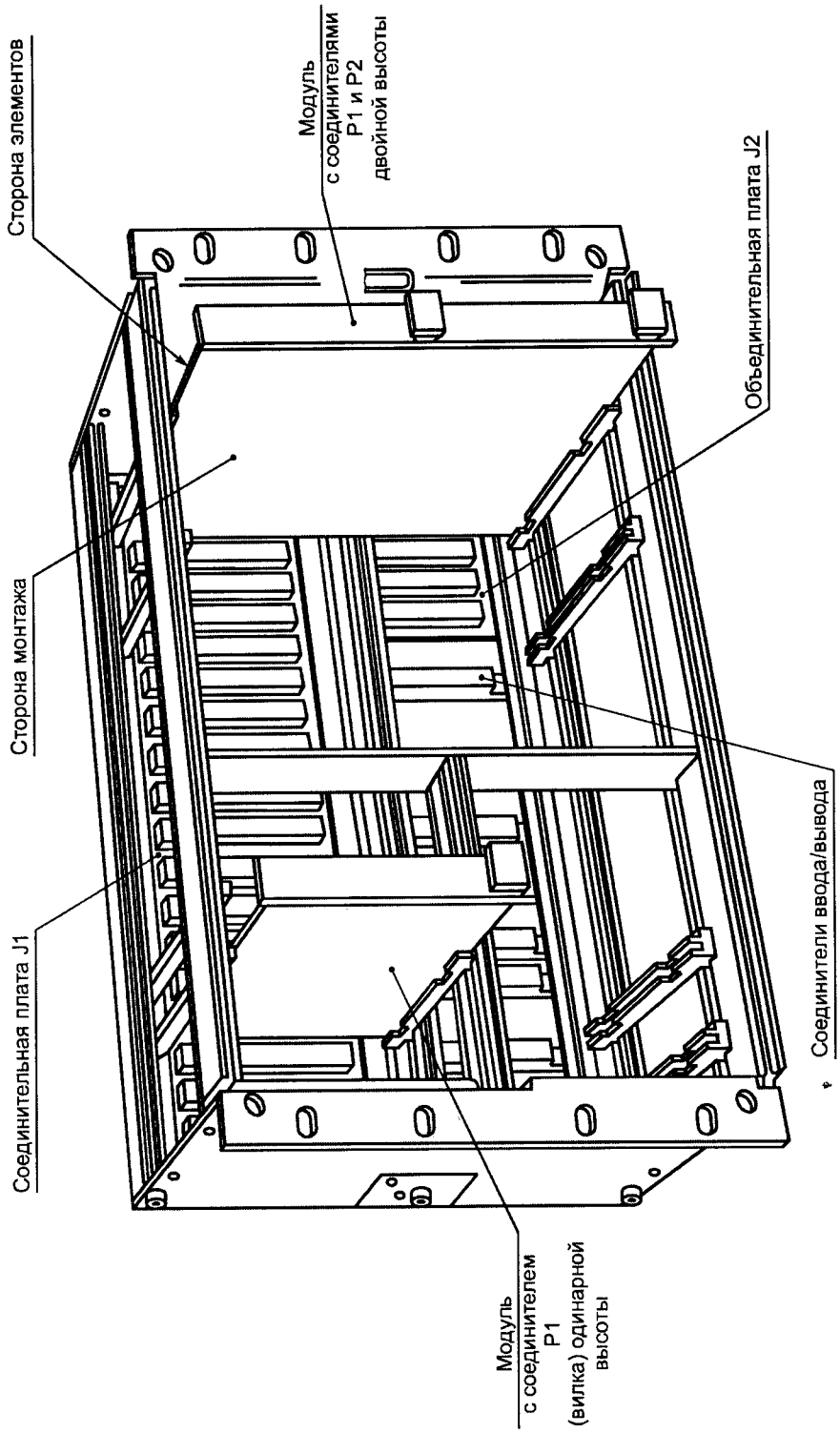
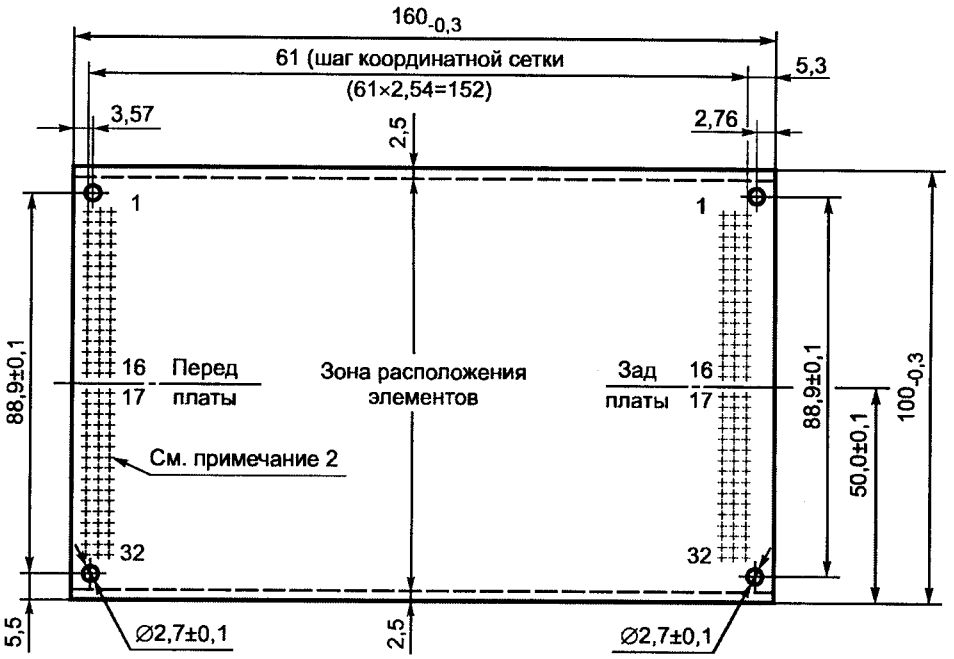


Рисунок 7.1 — Каркас с модулями разной высоты

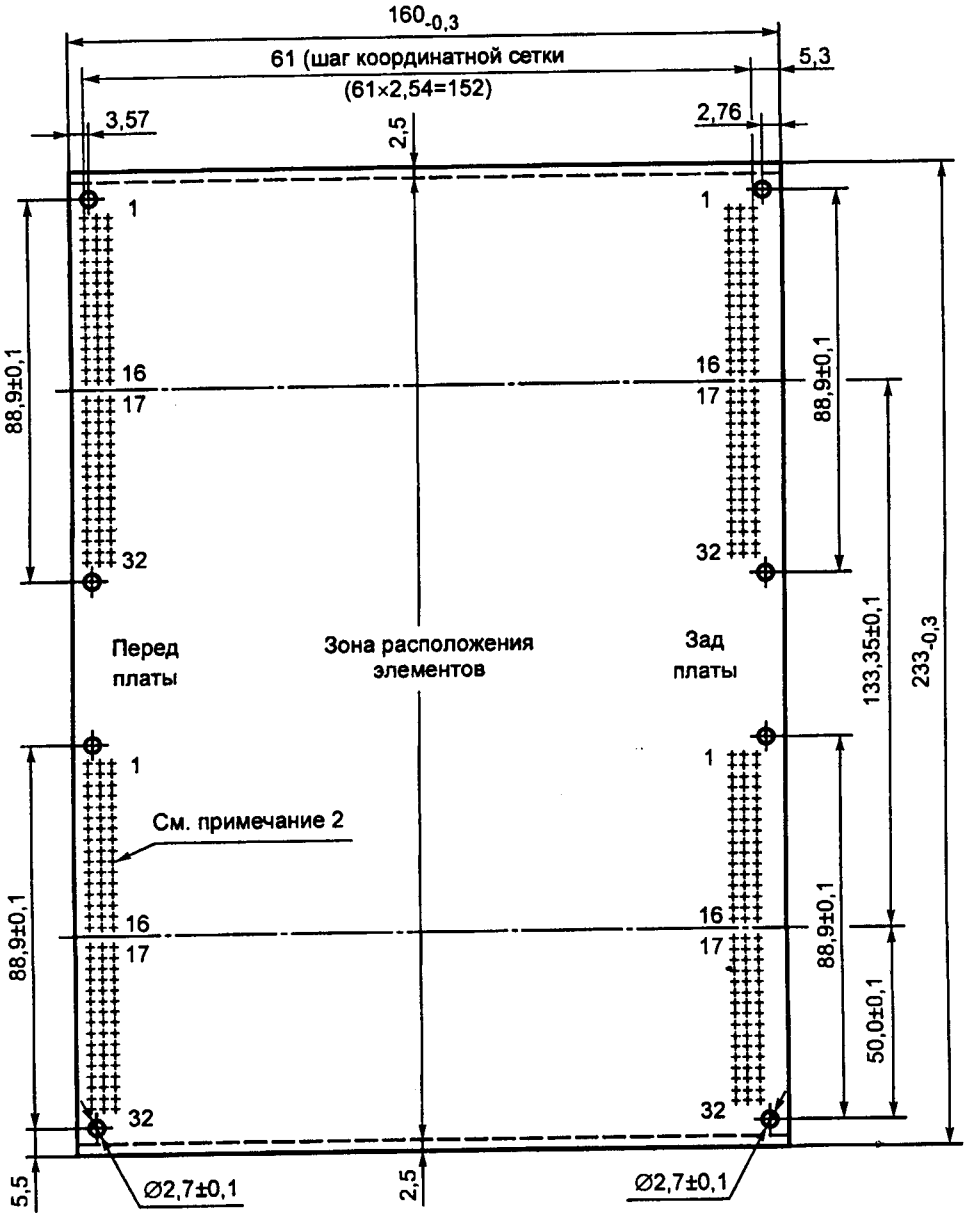


Примечания

1 ***Правило 7.32.*** *Платы модулей в зоне направляющих ДОЛЖНЫ ИМЕТЬ толщину (1,6±0,2) мм.*

2 Координатная сетка показана для того, чтобы помочь конструктору совместить ее с координатной сеткой передней панели.

Рисунок 7.2 — Основные размеры печатной платы одинарной высоты



Примечания

1 ***Правило 7.33.*** *Платы модулей в зоне направляющих ДОЛЖНЫ ИМЕТЬ толщину (1,6±0,2) мм.*

2 Координатная сетка показана для того, чтобы помочь конструктору совместить ее с координатной сеткой передней панели.

Рисунок 7.3 — Основные размеры печатной платы двойной высоты

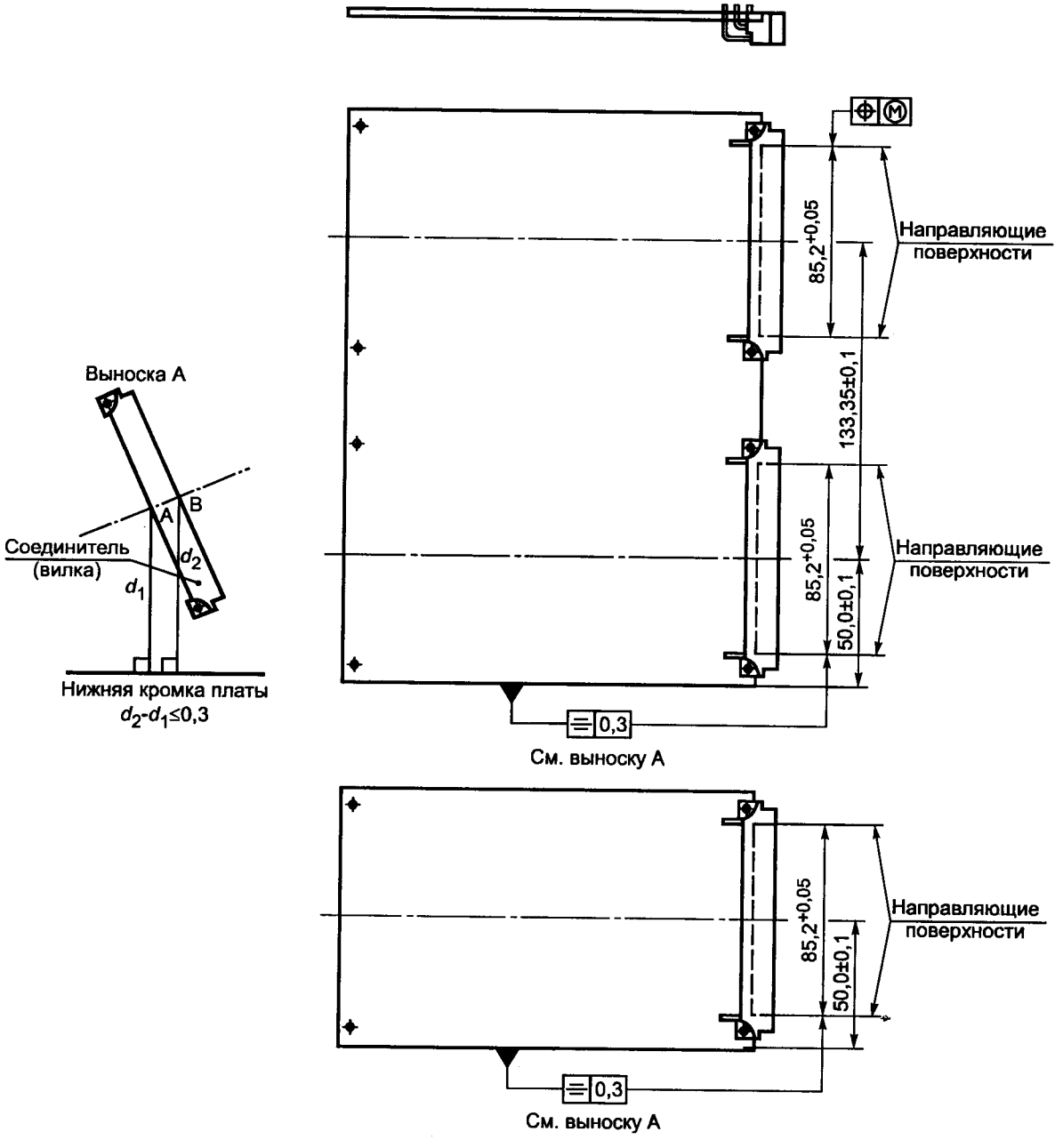
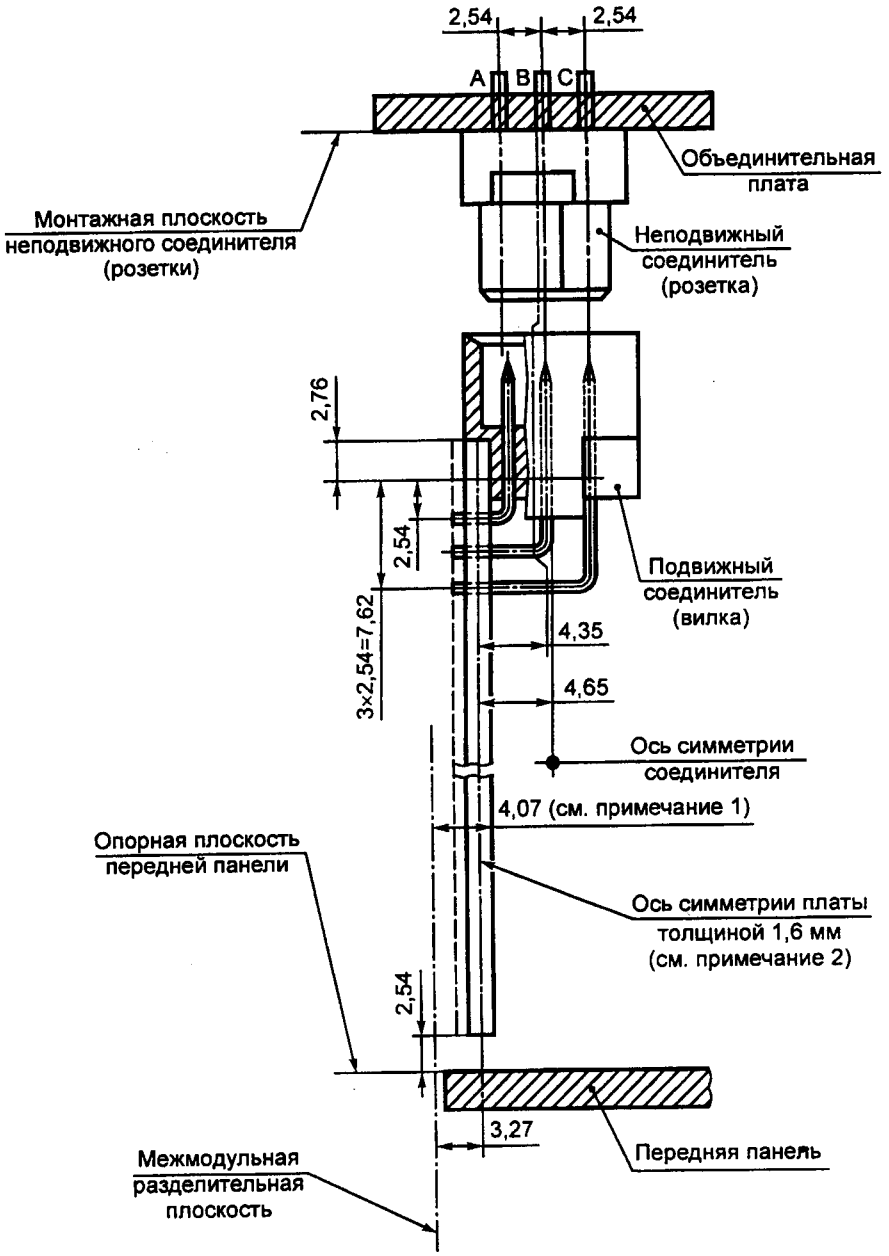


Рисунок 7.4 — Расположение соединителей на платах одинарной и двойной высоты

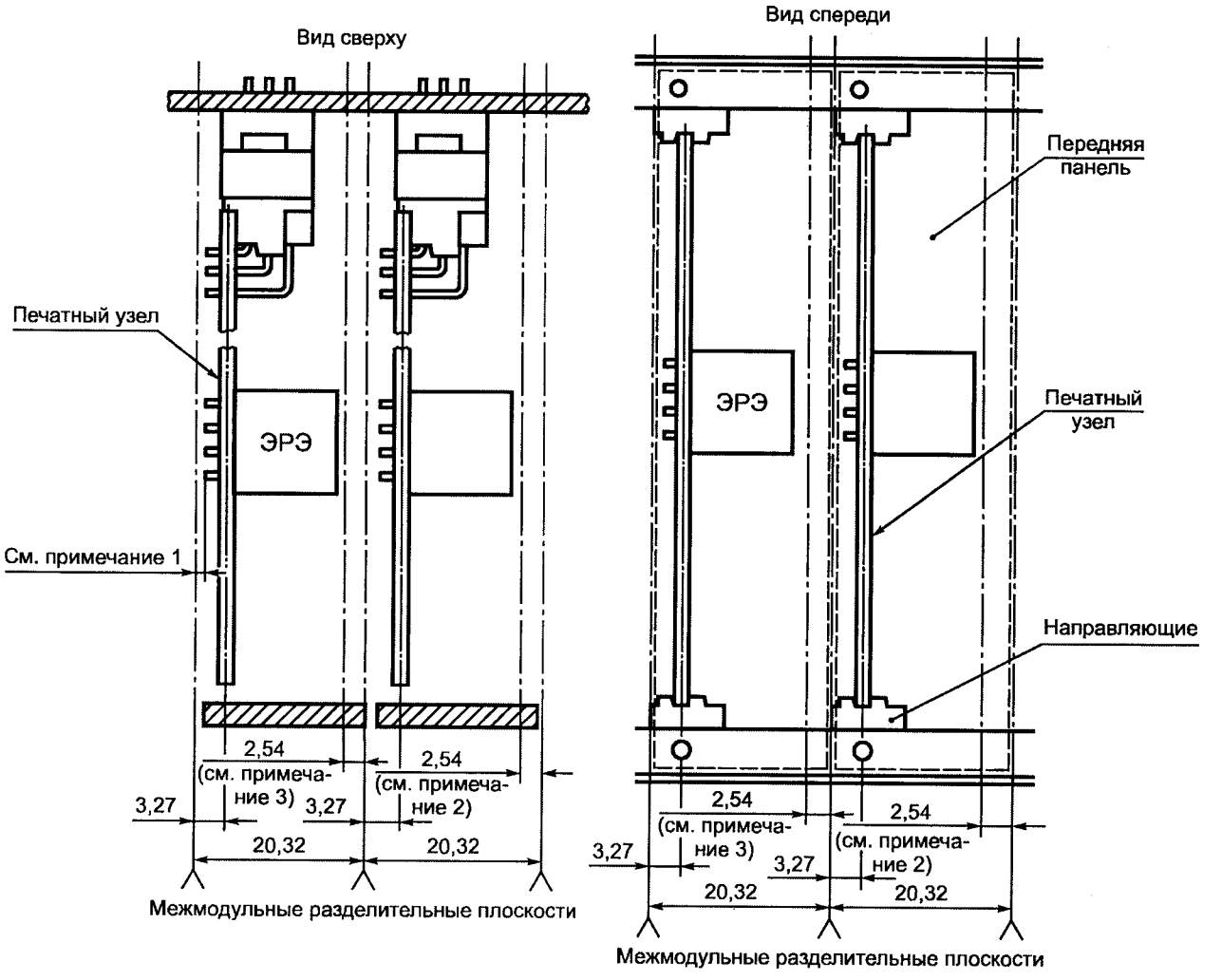


Примечания

1 Дополнительную информацию о допустимой толщине плат см. в подразделе 7.2.

2 Размер 4,07 мм не зависит от толщины печатной платы.

Рисунок 7.5 — Поперечный разрез печатной платы, соединителя, объединительной платы и передней панели



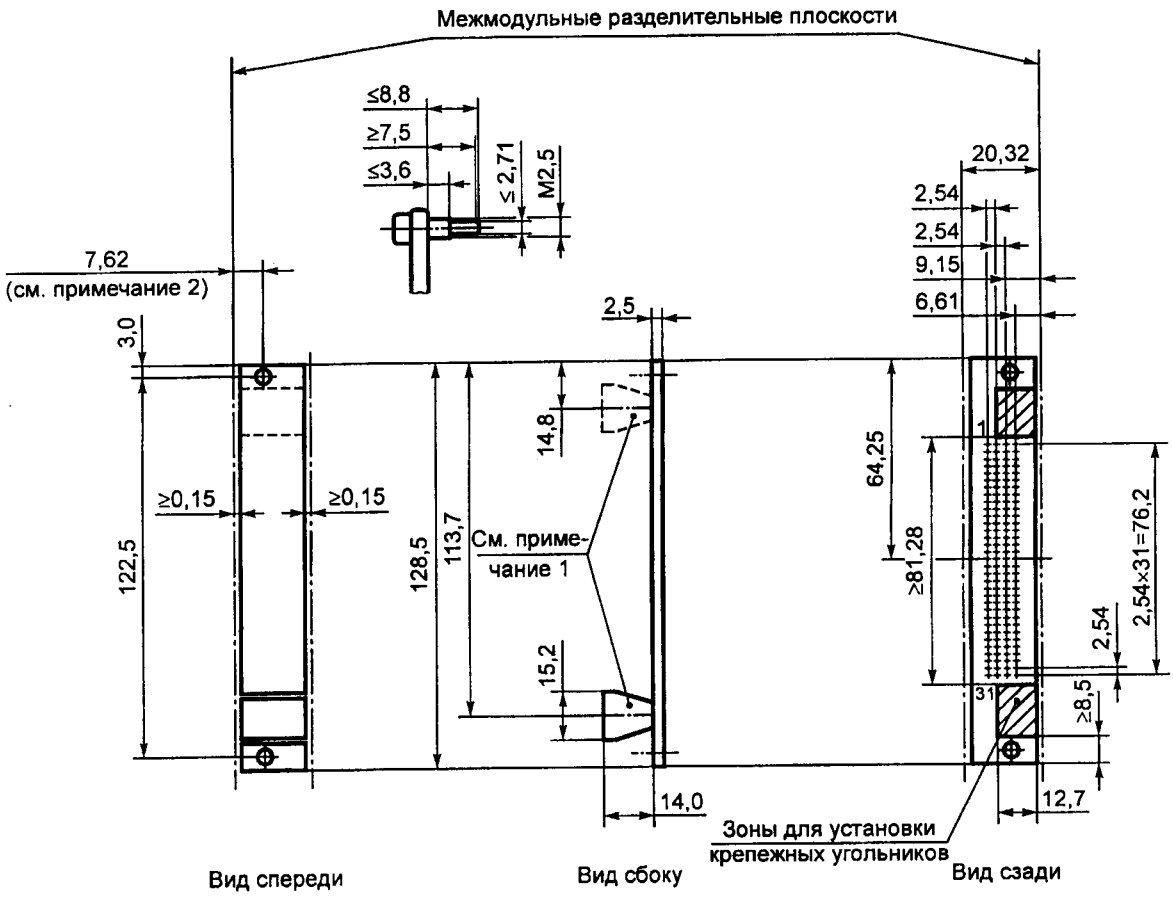
Примечания

1 **Предложение 7.12.** Длину выводов элементов предлагается ограничить максимально до 1,5 мм.

2 ***Правило 7.34.*** *После полной установки модуля в объединительную плату выводы ЭРЭ и сами ЭРЭ, смонтированные на стороне паек, НЕ ДОЛЖНЫ ВЫСТУПАТЬ за межмодульную разделительную плоскость.*

3 ***Правило 7.35.*** *После полной установки модуля в объединительную плату ЭРЭ, смонтированные со стороны элементов, НЕ ДОЛЖНЫ НАХОДИТЬСЯ ближе 2,54 мм от межмодульной разделительной плоскости.*

Рисунок 7.6 — Высота ЭРЭ, длина выводов, коробление печатных плат



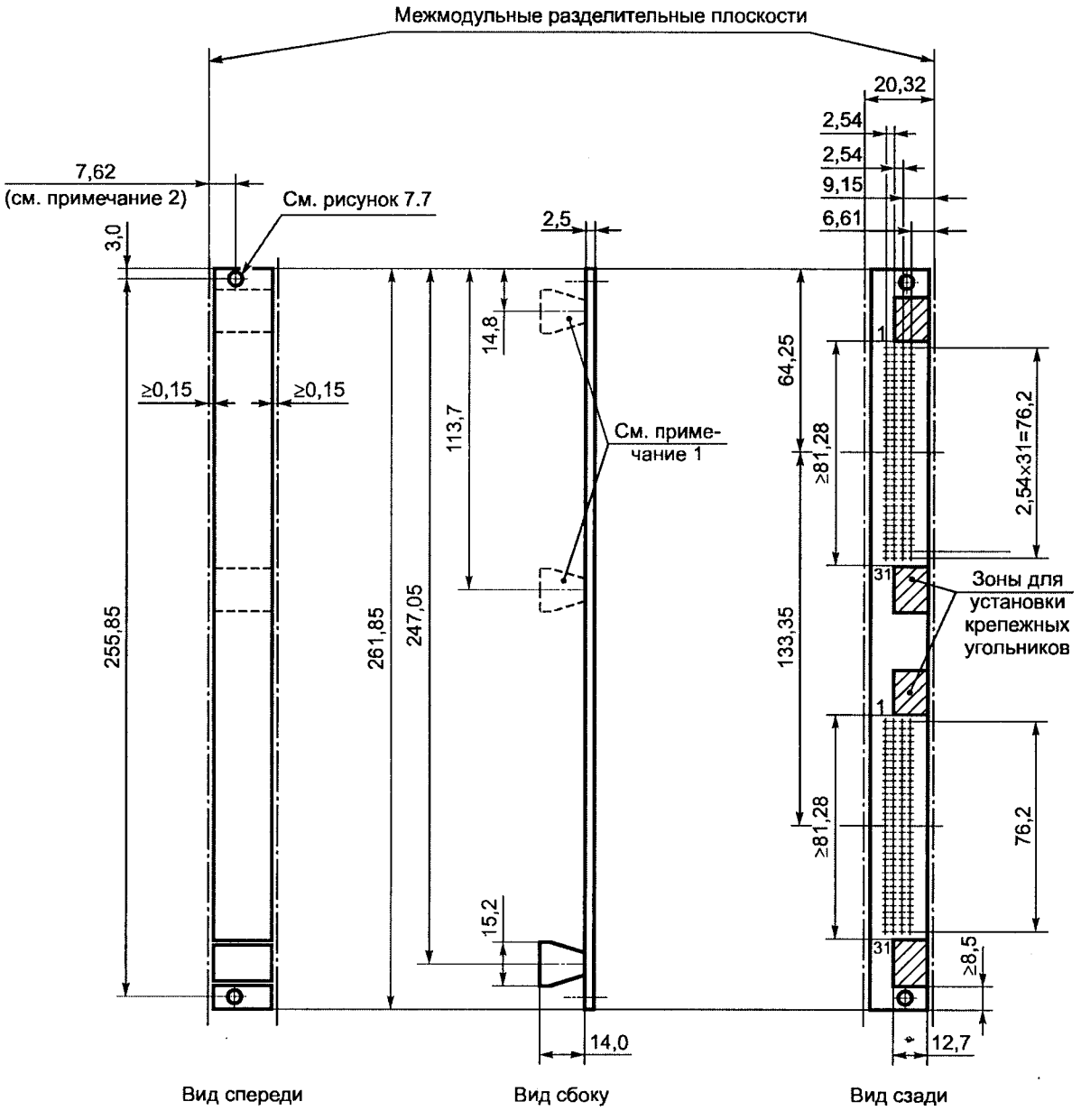
Примечания

1 Размеры рукояток приведены только в качестве рекомендаций.

2 **Рекомендация 7.11** Крепежное отверстие рекомендуется располагать на расстоянии 7,62 мм от межмодульной разделительной плоскости.

3 **Разрешение 7.20.** Крепежное отверстие МОЖНО располагать на расстоянии 12,7 мм от межмодульной разделительной плоскости.

Рисунок 7.7 — Передняя панель одинарной высоты, одинарной ширины



Примечания

1 Размеры рукояток приведены только в качестве рекомендаций.

2 **Рекомендация 7.12** Крепежное отверстие рекомендуется располагать на расстоянии 7,62 мм от межмодульной разделительной плоскости.

3 **Разрешение 7.21.** Крепежное отверстие МОЖНО располагать на расстоянии 12,7 мм от межмодульной разделительной плоскости.

Рисунок 7.8 — Передняя панель двойной высоты, одинарной ширины

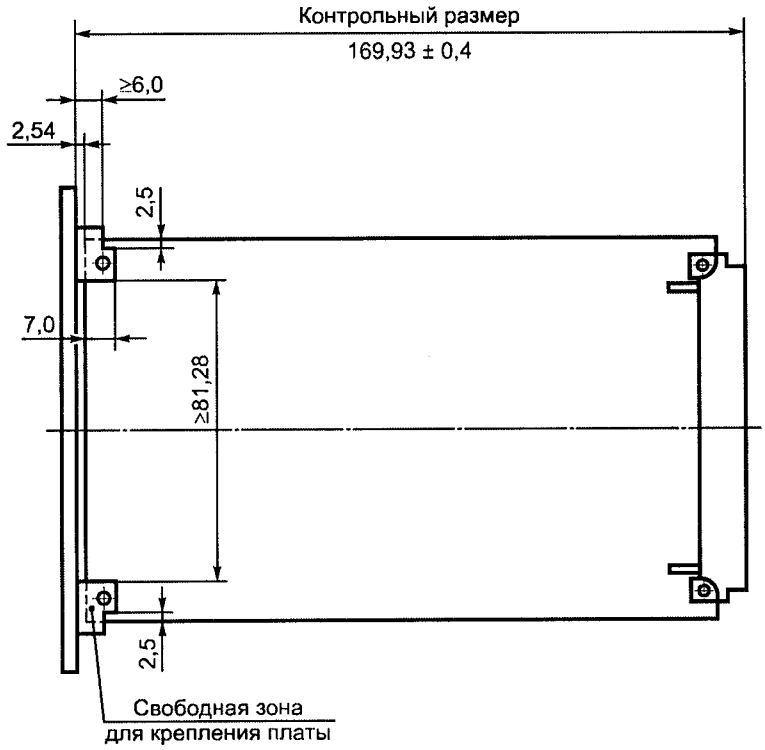


Рисунок 7.9 — Крепежные угольники передней панели и размеры модуля одинарной высоты

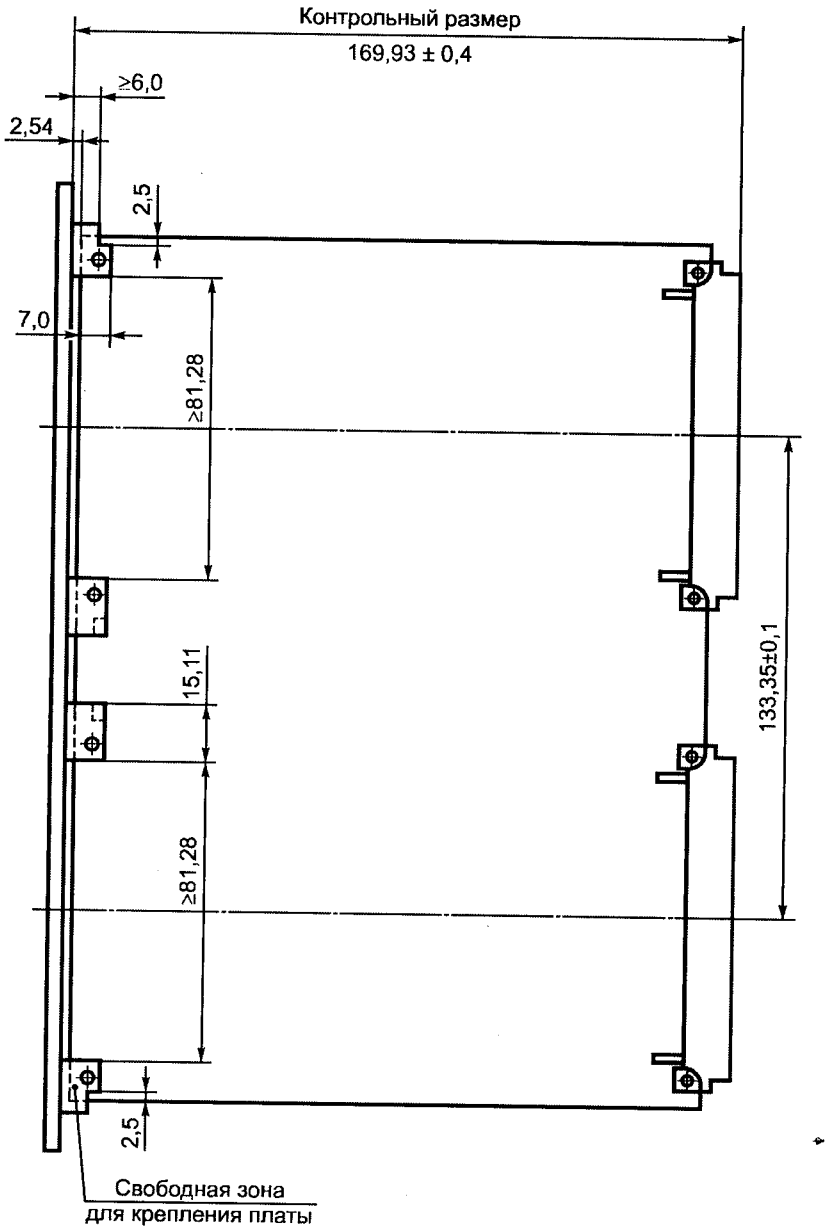
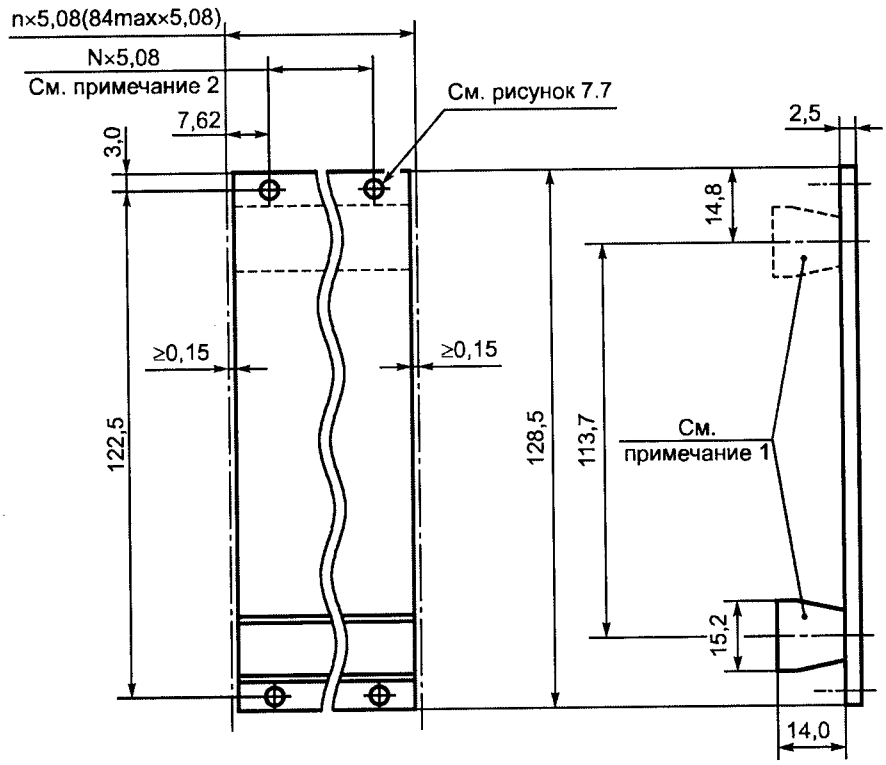


Рисунок 7.10 — Крепежные угольники передней панели и размеры модуля двойной высоты

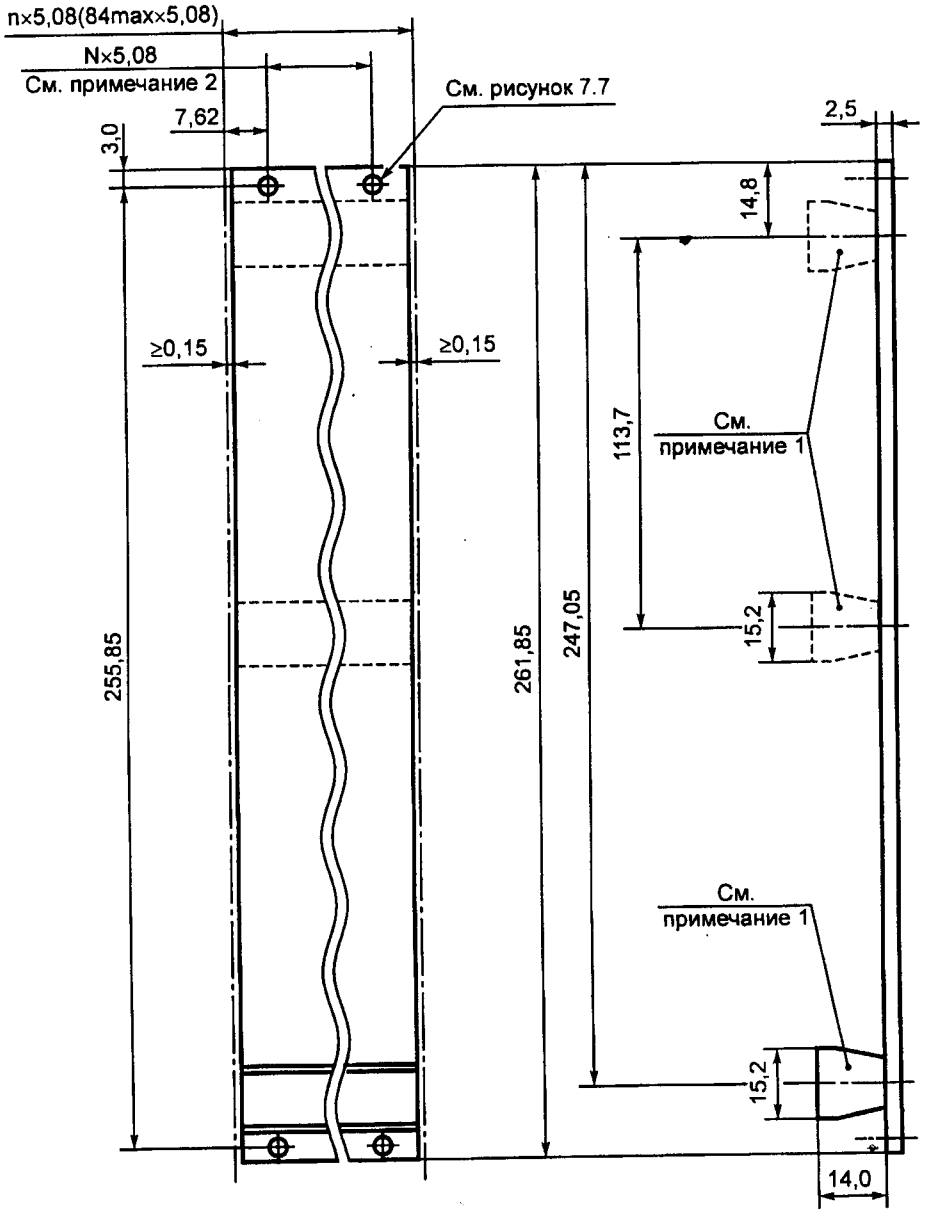


Примечания

1 Размеры рукояток по высоте и ширине приводятся только в качестве рекомендаций.

2 **Рекомендация 7.13.** При ширине панели-заглушки более 50,8 мм рекомендуется использовать, по меньшей мере, 4 крепежных отверстия: два снизу и два сверху.

Рисунок 7.11 — Панель-заглушка одинарной высоты

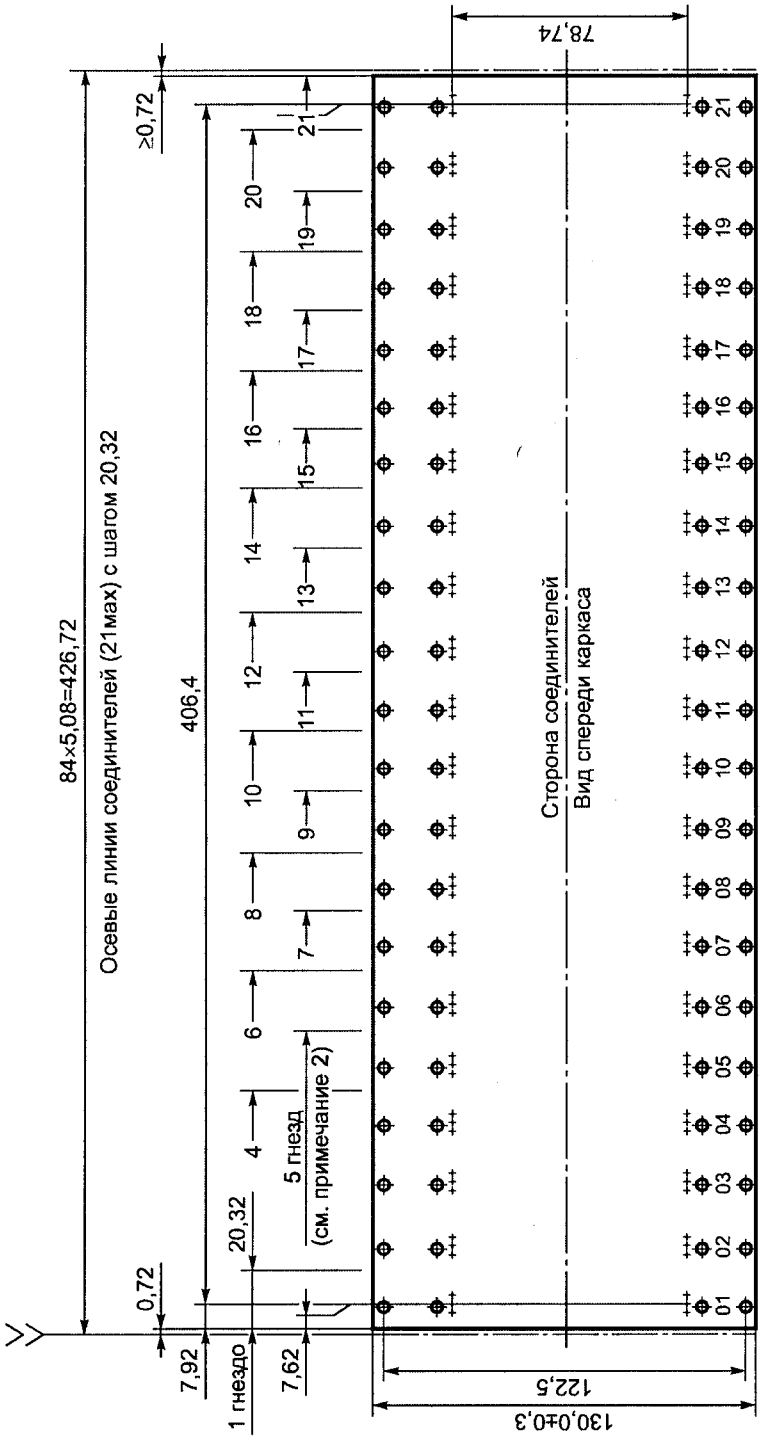


Примечания

1 Размеры рукояток по высоте и ширине приводятся только в качестве рекомендаций.

2 **Рекомендация 7.14.** При ширине панели-заглушки более 50,8 мм рекомендуется использовать, по меньшей мере, четыре крепежных отверстия: два снизу и два сверху.

Рисунок 7.12 — Панель-заглушка двойной высоты



П р и м е ч а н и я

1 **Разрешение 7.23.** Высота объединительных плат J1 или J2 МОЖЕТ составлять (128,7+0/—0,3) мм

2 Ширина объединительной платы изменяется в зависимости от количества гнезд.

Рисунок 7.13 — Габаритные и присоединительные размеры объединительных плат J1 и J2

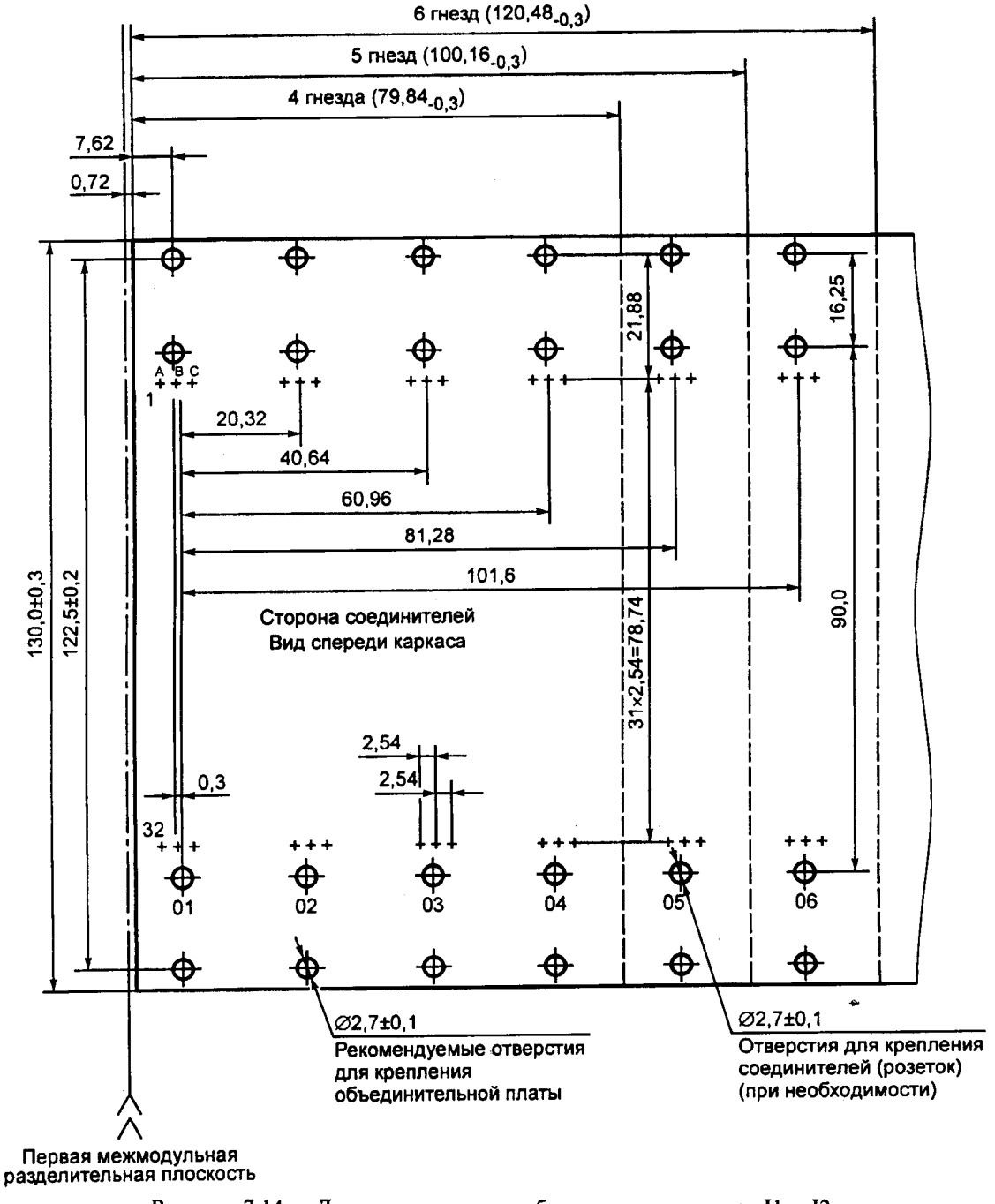
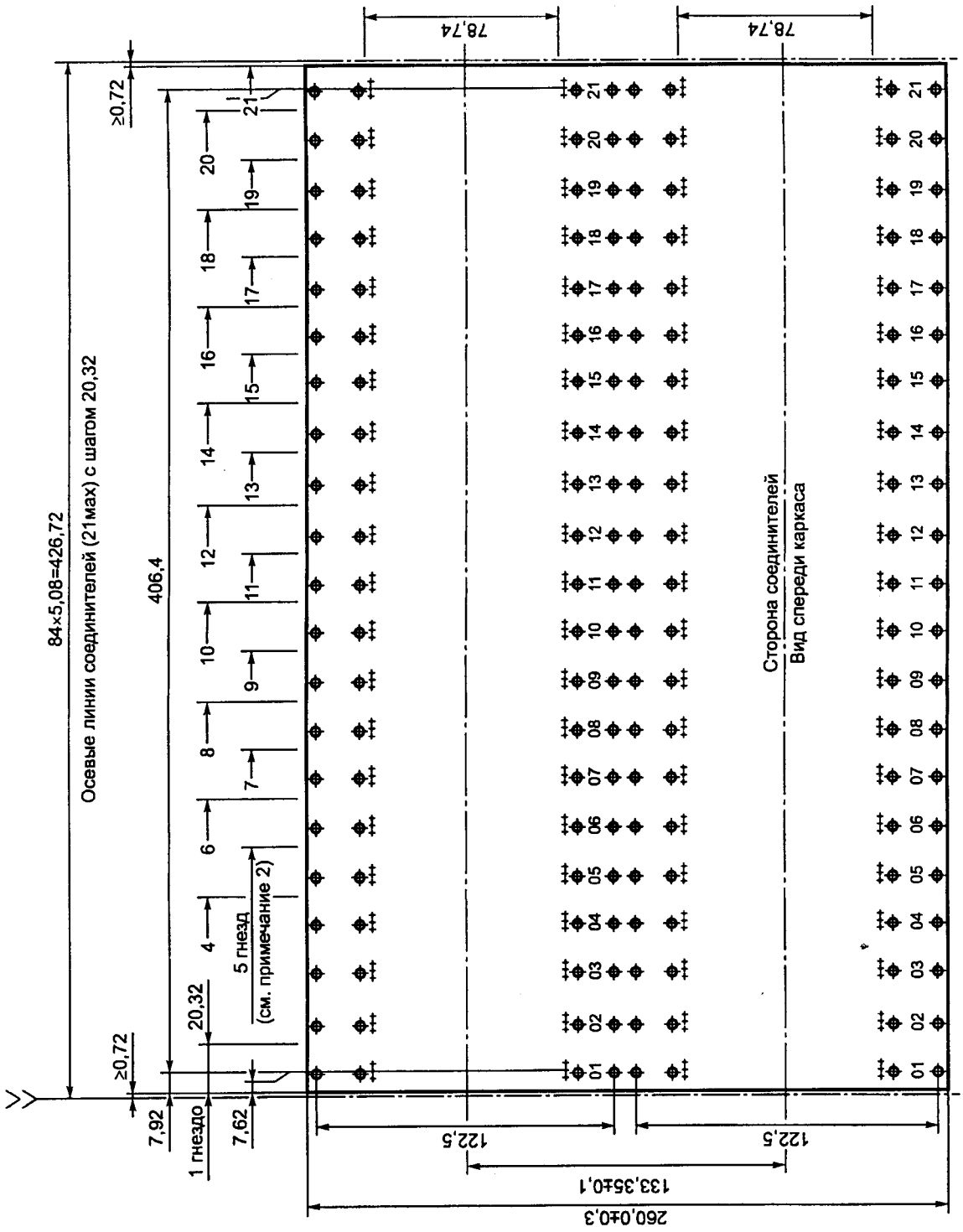


Рисунок 7.14 — Детальные размеры объединительных плат J1 и J2



Примечания

1 **Разрешение 7.24.** Высота объединительной платы J1/J2 МОЖЕТ составлять (262,05+0/—0,3) мм.

2 Ширина объединительной платы зависит от количества гнезд.

Рисунок 7.15 — Габаритные и присоединительные размеры комбинированной объединительной платы J1/J2

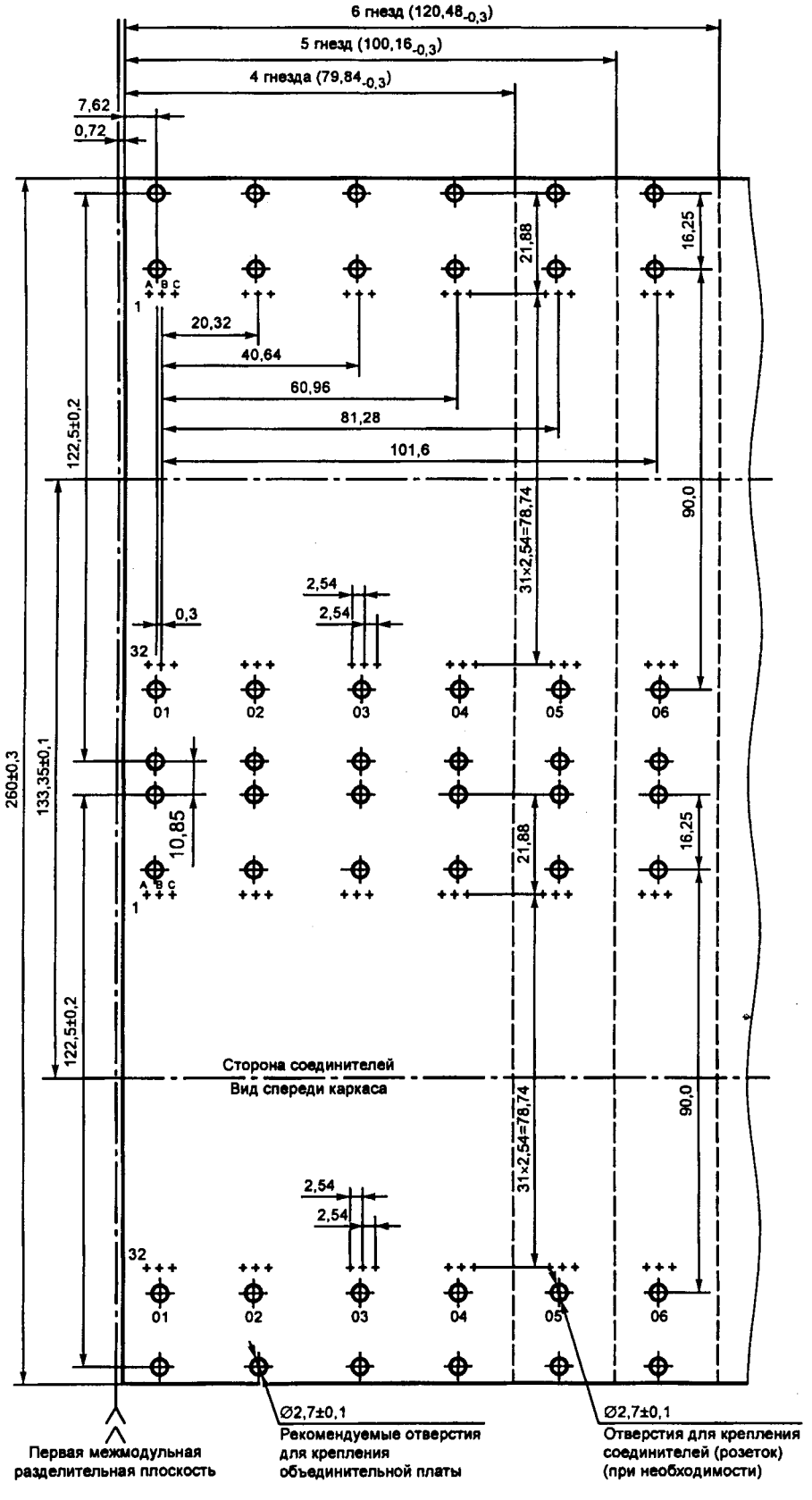


Рисунок 7.16 — Детальные размеры комбинированной объединительной платы J1/J2

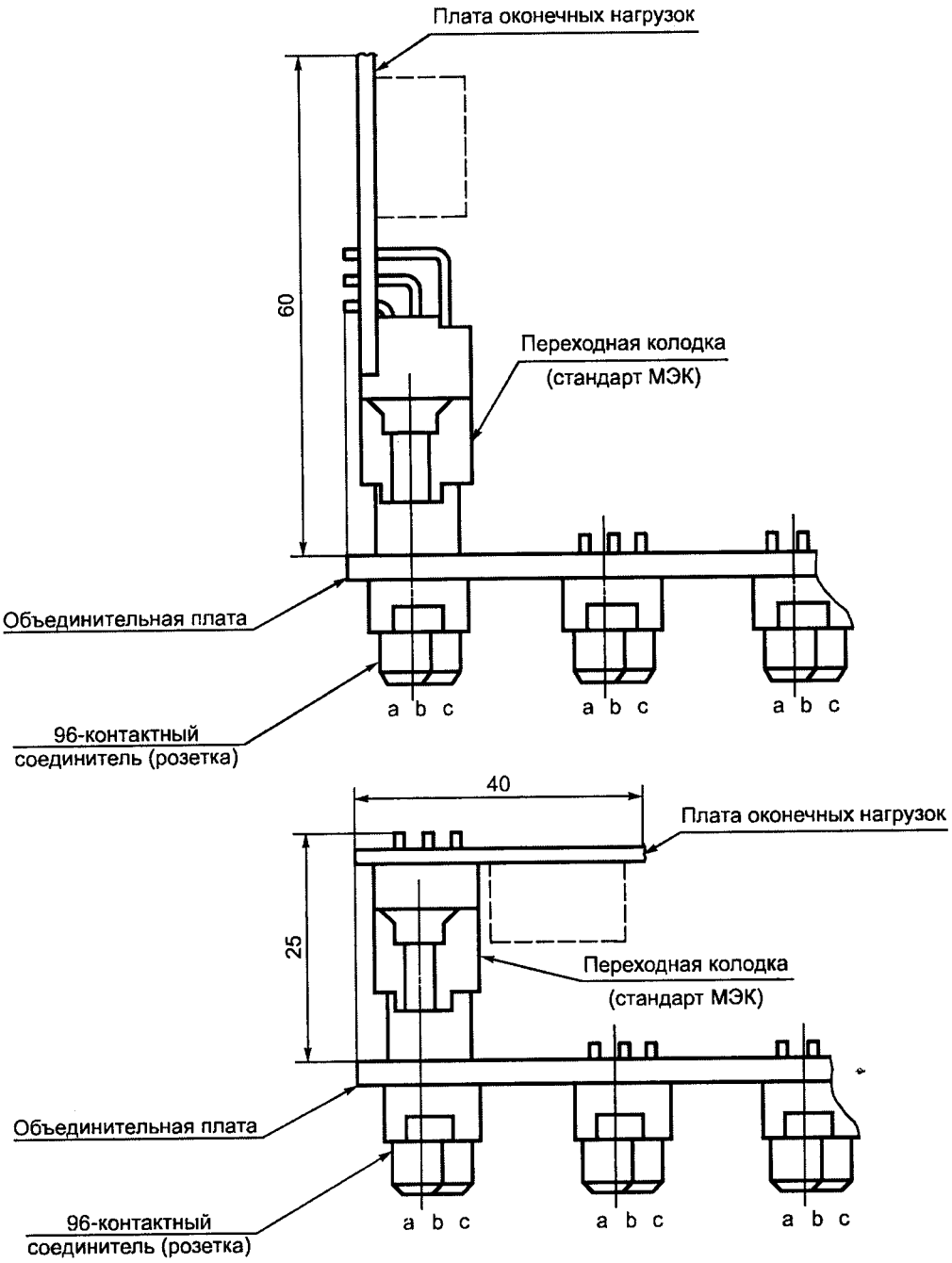


Рисунок 7.17 — Расположение оконечных нагрузок вне объединительной платы (вид на объединительную плату сверху)

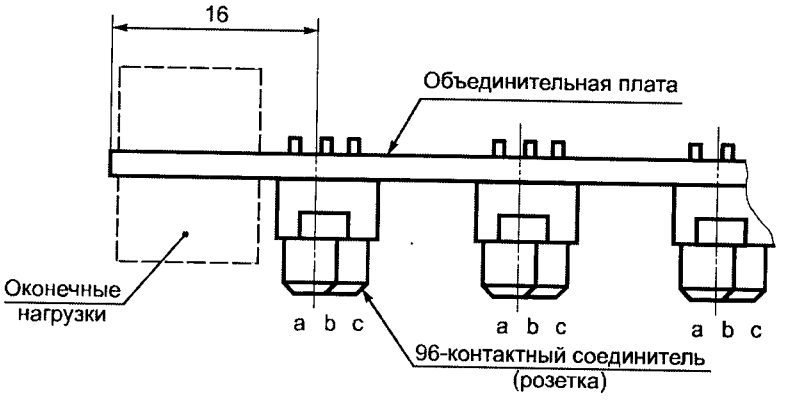
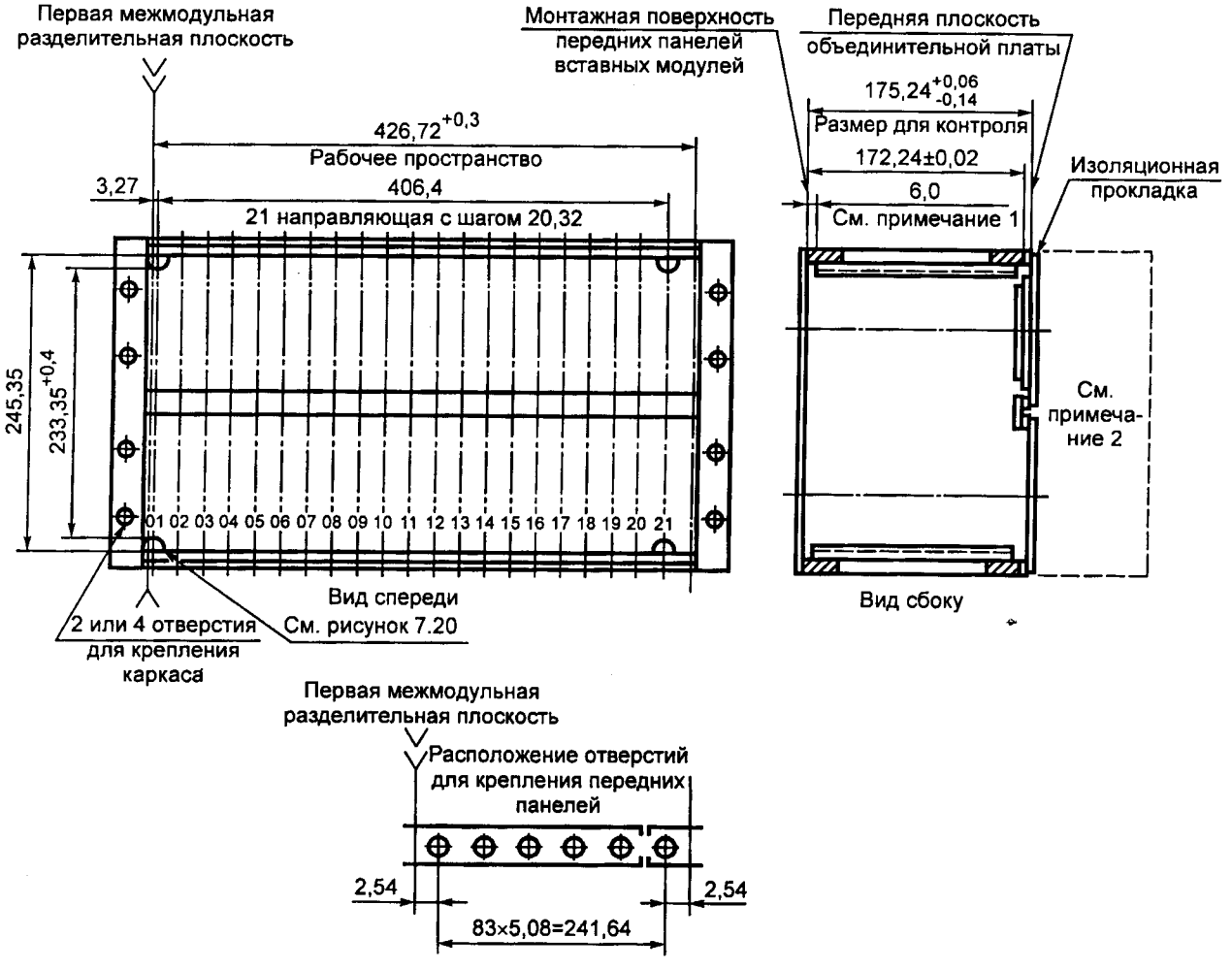


Рисунок 7.18 — Расположение оконечных нагрузок непосредственно на объединительной плате (вид на объединительную плату сверху)



Примечания

1 ***Правило 7.36.*** *Толщина изоляционной прокладки (вид сбоку показан на рисунке справа) ДОЛЖНА БЫТЬ ВЫБРАНА такой, чтобы выдерживался размер от опорной поверхности передней панели до передней плоскости объединительной платы.*

2 **Замечание 7.24.** У разных изготовителей толщина прокладки может быть разной.

3 **Разрешение 7.22.** Боковые панели каркаса МОГУТ выступать за пределы объединительной платы в глубину, если это необходимо.

4 Дополнительную информацию смотри в стандартах МЭК [1], [2].

Рисунок 7.19 — Каркас на 21 гнездо

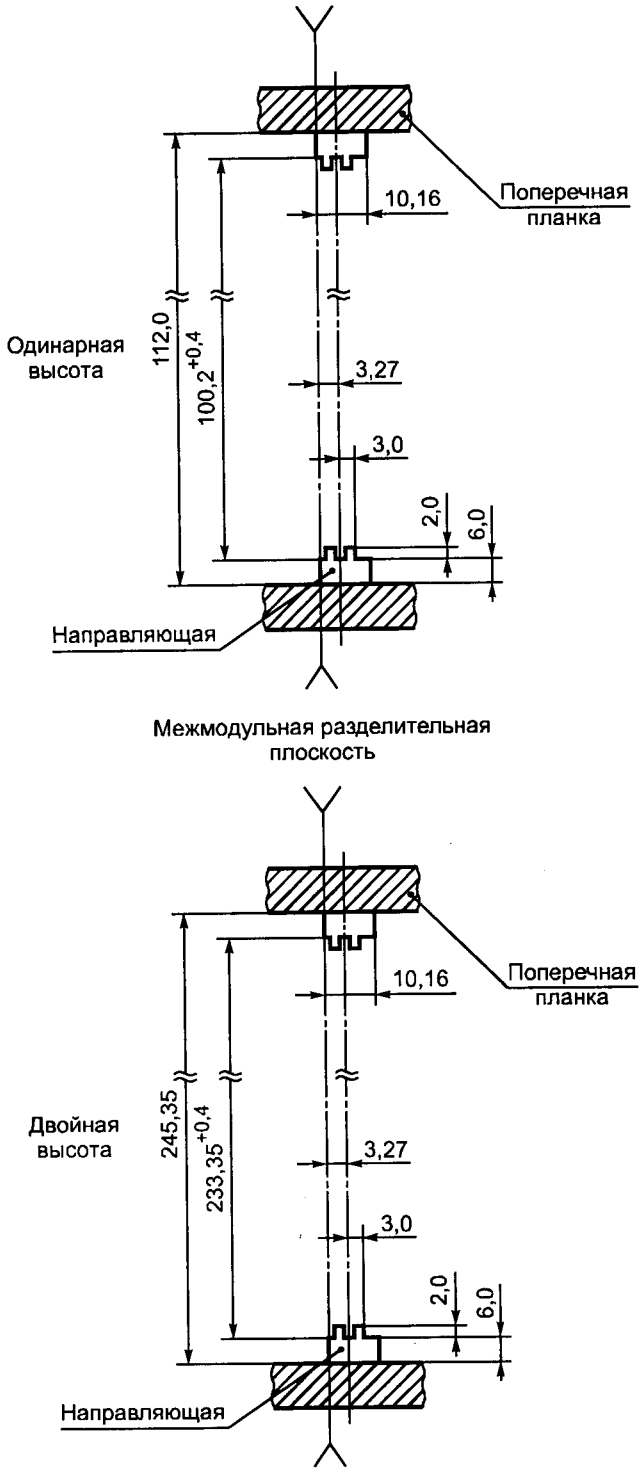


Рисунок 7.20 — Направляющие модулей

7.6 Соединители объединительных плат и модулей магистрали VME

7.6.1 Распределение сигналов по контактам соединителей J1/P1

В таблице 7.1 приводятся мнемонические обозначения сигналов для контактов соединителей J1/P1. Соединитель состоит из трех рядов контактов, обозначаемых как ряды *a, b, c*.

Таблица 7.1 — Распределение сигналов по контактам соединителей J1/P1

Номер контакта	Мнемоническое обозначение ряда		
	a	*b*	*c*
1	D00	BBSY*	D08
2	D01	BCLR*	D09
3	D02	ACFAIL*	D10
4	D03	BG0IN*	D11
5	D04	BG0OUT*	D12
6	D05	BG1IN*	D13
7	D06	BG1OUT*	D14
8	D07	BG2IN*	D15
9	GND	BG2OUT*	GND
10	SYSCLK	BG3IN*	SYSFAIL
11	GND	BG3OUT*	BERR*
12	DS1*	BR0*	SYSRESET*
13	DS0*	BR1*	LWORD*
14	WRITE*	BR2*	AM5
15	GND	BR3*	A23
16	DTACK*	AM0	A22
17	GND	AM1	A21
18	AS*	AM2	A20
19	GND	AM3	A19
20	IACK*	GND	A18
21	IACKIN*	SERCLK (1)	A17
22	IACKOUT*	SERDAT* (1)	A16
23	AM4	GND	A15
24	A07	IRQ7*	A14
25	A06	IRQ6*	A13
26	A05	IRQ5*	A12
27	A04	IRQ4*	A11
28	A03	RQ3*	A10
29	A02	IRQ2*	A09
30	A01	IRQ1*	A08
31	—12 V	+5 V STDBY	+12 V
32	+5 V	+5 V	+5 V

7.6.2 Распределение сигналов по контактам соединителей J2/P2

В таблице 7.2 приводятся мнемонические обозначения сигналов для контактов соединителей J2/P2. Соединитель состоит из трех рядов контактов, обозначаемых как ряды *a, b, c*.

Таблица 7.2 — Распределение сигналов по контактам соединителей J2/P2

Номер контакта	Мнемоническое обозначение ряда		
	a	*b*	*c*
1	Определяется пользователем	+5 V	Определяется пользователем
2	То же	GND	То же
3	»	RESERVED	»
4	»	A24	»
5	»	A25	»
6	»	A26	»
7	»	A27	»
8	»	A28	»
9	»	A29	»
10	»	A30	»
11	»	A31	»
12	»	GND	»
13	»	+5 V	»
14	»	D16	»
15	»	D17	»
16	»	D18	»
17	»	D19	»
18	»	D20	»
19	»	D21	»
20	»	D22	»
21	»	D23	»
22	»	GND	»
23	»	D24	»
24	»	D25	»
25	»	D26	»
26	»	D27	»
27	»	D28	»
28	»	D29	»
29	»	D30	»
30	»	D31	»
31	»	GND	»
32	»	+5 V	»

ПРИЛОЖЕНИЕ А
(справочное)

Словарь специальных терминов, используемых для описания магистрали VME

A.16 (Address 16): Мнемоническое обозначение типа функционального блока, формирующего или декодирующего адрес на адресных линиях A01—A15.

A.24 (Address 24): Мнемоническое обозначение типа функционального блока, формирующего или декодирующего адрес на адресных линиях A01—A23.

A.32 (Address 32): Мнемоническое обозначение типа функционального блока, формирующего или декодирующего адрес на адресных линиях A01—A31.

ADO (ADdress—Only): Мнемоническое обозначение типа задатчика, способного исполнять циклы Только Адрес.

адресный монитор (location monitor): Функциональный блок, контролирующий пересылки по шине пересылки данных с целью обнаружения обращений к ячейкам, которые ему предписано контролировать. Если происходит обращение к одной из этих ячеек, адресный монитор вырабатывает внутримодульный сигнал.

арбитр (arbiter): Функциональный блок, который принимает запросы на использование шины от запросчиков и предоставляет управление шиной пересылки данных одновременно только одному запросчику.

арбитраж (arbitration): Процесс назначения управления шиной пересылки данных по запросу.

блок контроля питания (power monitor module): Функциональный блок, контролирующий состояние первичного источника питания системы магистрали VME и сигнализирующий о выходе параметров питания за пределы, гарантирующие надежную работу системы. Поскольку большинство систем запитывается от источника переменного тока, блок контроля питания обычно проектируется с возможностью обнаружения состояния снижения напряжения или обесточивания сети переменного тока.

BLT (BLock Transfer): Мнемоническое обозначение типа задатчика, который способен инициировать циклы блочной пересылки данных, а также исполнителя, который способен отвечать на эти циклы.

гнездо (slot): Место перед объединительной платой (платами), куда помещается модуль для сочленения с соединителем (соединителями) этой платы. Если система имеет обе объединительные платы J1 и J2 или комбинированную плату J1/J2, то каждое гнездо имеет два 96-контактных соединителя. Если система имеет только одну плату J1, то каждое гнездо имеет один 96-контактный соединитель.

D08(O) (Data 08 (Odd)): Мнемоническое обозначение:

- исполнителя, который посылает или получает восьмибитные данные одновременно по линиям D00—D07;
- обработчика прерываний, который получает восьмибитную информацию статуса/идентификации одновременно по линиям D00—D07;
- прерывателя, который посылает восьмибитную информацию статуса/идентификации одновременно по линиям D00—D07.

D08(EO) (Data 08 (Even and Odd)): Мнемоническое обозначение:

- задатчика, который посылает или получает 8-битные данные одновременно по линиям D00—D07 или D08—D15;
- исполнителя, который посылает или получает 8-битные данные одновременно по линиям D00—D07 или D08—D15.

D16 (Data 16): Мнемоническое обозначение:

- задатчика, который посылает и получает 16-битные данные одновременно по линиям D00—D15;
- исполнителя, который посылает и получает 16-битные данные одновременно по линиям D00—D15;
- обработчика прерываний, который получает 16-битную информацию статуса/идентификации одновременно по линиям D00—D15;
- прерывателя, который посылает 16-битную информацию статуса/идентификации одновременно по линиям D00—D15.

D32 (Data 32): Мнемоническое обозначение:

- задатчика, который посылает и получает 32-битные данные одновременно по линиям D00—D31;
- исполнителя, который посылает и получает 32-битные данные одновременно по линиям D00—D31;
- обработчика прерываний, который получает 32-битную информацию статуса/идентификации одновременно по линиям D00—D31;
- прерывателя, который посылает 32-битную информацию статуса/идентификации одновременно по линиям D00—D31.

задатчик (master): Функциональный блок, который инициирует циклы шины пересылки данных для пересылки данных между ним и исполнителем.

запросчик (requester): Функциональный блок, находящийся на одной плате с задатчиком или обработчиком прерываний и запрашивающий право на использование шины пересылки данных всякий раз, когда это потребуется его задатчику или обработчику прерываний.

интерфейсная логика объединительной платы (backplane interface logic): Определенная логика, учитывающая характеристики объединительной платы: полное сопротивление ее сигнальных линий, время распространения сигналов, значения оконечных нагрузок и т.д. Настоящий стандарт устанавливает правила проектирования такой логики с учетом максимальной длины объединительной платы и максимального количества гнезд для модулей.

исполнитель (slave): Функциональный блок, который обнаруживает циклы шины пересылки данных, инициируемые задатчиком, и, когда эти циклы предусматривают его (исполнителя) участие, осуществляет пересылки данных между ним и задатчиком.

каркас (subrack): Жесткая рамочная конструкция, обеспечивающая механическую поддержку модулей, вставляемых в объединительную плату, надлежащее сочленение соединителей, отсутствие контакта между соседними модулями, распределение охлаждающих воздушных потоков в системе и невозможность нарушения контакта между вставленными модулями и объединительной платой вследствие вибрации или ударов.

модуль (board): Печатная плата с набором электрорадиоэлементов и одним—двумя 96-контактными соединителями, которые могут сочленяться с соединителями объединительной платы.

модуль системного контроллера (system controller board): Модуль, размещаемый в гнезде 1 объединительной платы магистрали VME и имеющий в своем составе формирователь системного тактового сигнала, арбитр, формирователь цепочки подтверждения прерывания и шинный таймер. Некоторые контроллеры содержат также формирователь тактового сигнала последовательной пересылки либо блок контроля питания, либо то и другое.

обработчик прерываний (interrupt handler): Функциональный блок, обнаруживающий запросы прерывания, которые формируются прерывателями, и отвечающий на эти запросы требованием о предоставлении ему информации статуса/идентификации.

объединительная плата (backplane): Печатная плата с 96-контактными соединителями и печатными сигнальными проводниками, которые соединяют соответствующие контакты этих соединителей. Некоторые системы магистрали VME имеют единственную печатную плату — объединительную плату J1. Она содержит печатные сигнальные проводники, требуемые для выполнения основных операций. Другие системы магистрали VME имеют также дополнительную печатную плату — объединительную плату J2. Она содержит дополнительные 96-контактные соединители и печатные сигнальные проводники, требуемые для пересылок данных, и адреса повышенной разрядности. Наконец, третьи имеют комбинированную печатную плату — объединительную плату J1/J2, которая содержит печатные сигнальные проводники и соединители объединительных плат J1 и J2.

последовательная цепочка (daisy-chain): Определенный тип сигнальной линии магистрали VME, используемой для распространения уровня сигнала от модуля к модулю, начиная с первого гнезда и кончая последним гнездом. Магистраль VME имеет четыре цепочки предоставления шины и одну цепочку подтверждения прерывания.

прерыватель (interrupter): Функциональный блок, формирующий запрос прерывания на шине приоритетных прерываний, а затем предоставляющий информацию статуса/идентификации по требованию обработчика прерываний.

служебная шина (utility bus): Одна из четырех шин объединительной платы магистрали VME, по которой передаются периодические системные синхросигналы и сигналы, координирующие последовательность действий системы при включении и выключении питания.

UAT (UnAligned Transfer): Мнемоническое обозначение задатчика, который посылает или получает данные невыровненным (неупорядоченным) способом.

формирователь последовательной цепочки подтверждения прерывания (IACK daisy—chain driver): Функциональный блок, устанавливающий в активное состояние цепочку подтверждения прерывания всякий раз, когда обработчик прерываний подтверждает запрос прерывания. Такая цепочка гарантирует пересылку информации о статусе/идентификации по шине пересылки данных только от одного прерывателя, даже если запрос прерывания сформирован несколькими прерывателями.

формирователь системного тактового сигнала (system clock driver): Функциональный блок, подающий на служебную шину синхронизирующий сигнал частотой 16 МГц.

формирователь тактового сигнала последовательной магистрали (serial clock driver): Функциональный блок, подающий периодический тактовый сигнал, который синхронизирует работу магистрали МЭК 823. Хотя стандарт магистрали VME определяет формирователь тактового сигнала последовательной пересылки и в нем предусматриваются две сигнальные линии на объединительной плате, используемые магистралью МЭК 823, протокол последней абсолютно не зависит от магистрали VME. Технические требования к временным параметрам для формирователя тактового сигнала последовательной магистрали приведены в приложении С.

функциональный блок (functional module): Совокупность электронных схем, расположенных на одном модуле магистрали VME и совместно выполняющих какую-то определенную задачу.

цикл арбитража (arbitration cycle): Цикл арбитража начинается, когда арбитр обнаруживает запрос на использование шины пересылки данных. Арбитр предоставляет шину запросчику, который сигнализирует, что шина занята. Запросчик заканчивает цикл снятием сигнала занятости шины, что побуждает арбитр снова считывать запросы шины.

цикл блочной записи (block write cycle): Цикл шины пересылки данных, используемый для пересылки блока размером от 1 до 256 байтов от задатчика исполнителю. Эта пересылка выполняется с использованием ряда последовательных одно-, двух- и четырехбайтовых пересылок данных. Задатчик не освобождает шину пересылки данных до тех пор, пока не будут переданы все байты. Цикл блочной записи отличается от ряда из нескольких обычных циклов записи тем, что задатчик только один раз (в начале цикла) выполняет широковещательную пересылку адреса и модификатора адреса. Затем исполнитель наращивает этот адрес при каждой пересылке, чтобы данные следующей пересылки записывались в следующую ячейку.

цикл блочного считывания (block read cycle): Цикл шины пересылки данных, используемый для пересылки блока размером от 1 до 256 байтов от исполнителя задатчику. Эта пересылка выполняется с использованием ряда последовательных одно-, двух- и четырехбайтовых пересылок. После начала блочной пересылки задатчик не освобождает шину пересылки данных до тех пор, пока не будут считаны им все байты. Цикл блочного считывания отличается от ряда из нескольких обычных циклов считывания тем, что задатчик только один раз (в начале цикла) выполняет широковещательную пересылку адреса и модификатора адреса. Затем, при каждой пересылке исполнитель наращивает этот адрес, чтобы данные следующей пересылки извлекались из следующей ячейки.

цикл записи (write cycle): Цикл шины пересылки данных, используемый для пересылки одного, двух, трех или четырех байтов от задатчика исполнителю. Цикл начинается, когда задатчик выполняет широковещательную пересылку адреса и модификатора адреса и помещает данные на шину пересылки данных. Каждый исполнитель принимает адрес и модификатор адреса и проверяет, не он ли должен отвечать в этом цикле. Если это так, он запоминает эти данные и затем подтверждает пересылку. После этого задатчик завершает цикл.

цикл подтверждения прерывания (interrupt acknowledge cycle): Цикл шины пересылки данных, инициируемый обработчиком прерываний, который выполняет считывание информации статуса/идентификации от прерывателя. Обработчик прерываний вырабатывает этот цикл всякий раз, когда обнаруживает запрос прерывания от прерывателя, а шина пересылки данных находится под его управлением.

цикл считывания (read cycle): Цикл шины пересылки данных, используемый для пересылки одного, двух, трех или четырех байтов от исполнителя задатчику. Цикл начинается, когда задатчик выполняет широковещательную пересылку адреса и модификатора адреса. Каждый исполнитель принимает адрес и модификатор адреса и проверяет, не он ли должен отвечать в этом цикле считывания. Если это так, он извлекает данные из своей внутренней памяти, помещает их на шину пересылки данных и подтверждает пересылку. После этого задатчик завершает цикл.

цикл Только Адрес (address—only cycle): Цикл шины пересылки данных, состоящий только из широковещательной пересылки адреса без пересылки данных. Исполнители не подтверждают такие циклы, а задатчики завершают такой цикл, не ожидая подтверждения.

цикл Чтение—Модификация—Запись (read—modify—write cycle): Цикл шины пересылки данных, используемый задатчиком для обращения к ячейке исполнителя как в режиме считывания, так и в режиме записи при запрещении доступа к этой ячейке со стороны других задатчиков в течение этого цикла. Этот цикл очень полезен для мультипроцессорных систем, в которых определенные ячейки памяти используются для управления доступом к определенным системным ресурсам (например семафорные ячейки).

цикл шины пересылки данных (data transfer bus cycle): Последовательность перепадов напряжения логических уровней на сигнальных линиях шины пересылки данных, которая реализует пересылку адреса или адреса и данных между задатчиком и исполнителем. Существуют 34 типа циклов шины пересылки данных.

шина арбитража (arbitration bus): Одна из четырех шин объединительной платы магистрали VME. Позволяет одному арбитру и нескольким запросчикам координировать порядок использования шины пересылки данных.

шина пересылки данных (data transfer bus): Одна из четырех шин объединительной платы. Позволяет задатчикам устанавливать направление пересылок двоичных данных между ними и исполнителями.

шина приоритетных прерываний (priority interrupt bus): Одна из четырех шин объединительной платы магистрали VME, которая позволяет прерывателям посылать запросы прерывания в обработчики прерываний, а обработчикам прерываний — подтверждать эти прерывания.

шинный таймер (bus timer): Функциональный блок, который выполняет отсчет времени каждой пересылки по шине пересылки данных и завершает цикл шины, если это время превышает разумные пределы. Без этого блока может возникнуть ситуация, при которой задатчик попытается выполнить операцию пересылки данных в/из какой-то несуществующей ячейки исполнителя и неопределенно долго ждать результаты. Шинный таймер предотвратит эту задержку завершением цикла.

ПРИЛОЖЕНИЕ В
(справочное)

Описание сигналов на выводах соединителей магистрали VME

Данное приложение содержит описание сигнальных линий магистрали VME. В таблице В.1 определены мнемонические обозначения сигналов магистрали, их описание и характеристики.

Таблица В.1 — Мнемонические обозначения и описание сигналов магистрали VME.

Обозначение	Название и описание сигнала
A01—A15	Линии **адреса (address)** (разряды 1—15). Сигналы на линиях адреса, возбуждаемые формирователями с тремя состояниями на выходе, используются для широковещательной передачи короткого, стандартного или расширенного адреса
A16—A23	Линии **адреса (address)** (разряды 16—23). Сигналы на линиях адреса, возбуждаемые формирователями с тремя состояниями на выходе, используются совместно с сигналами линий A01—A15 для широковещательной передачи стандартного или расширенного адреса
A24—A31	Линии **адреса (address)** (разряды 24—31). Сигналы на линиях адреса, возбуждаемые формирователями с тремя состояниями на выходе, используются совместно с сигналами линий A01—A23 для широковещательной передачи расширенного адреса
ACFAIL*	**Отказ питания переменного тока (a.c.failure).** Сигнал, возбуждаемый формирователем с открытым коллектором на выходе, показывает, что питание переменного тока больше не поступает или не выдерживаются требуемые уровни входного напряжения переменного тока
AM0—AM5	**Модификатор адреса (address modifier)** (разряды 0—5). Сигналы, возбуждаемые формирователями с тремя состояниями на выходе, используются для широковещательной передачи информации, такой как размер адреса, тип цикла и/или идентификации задатчика
AS*	**Адресный строб (address strobe).** Сигнал, возбуждаемый формирователем с тремя состояниями на выходе, показывает, что на линии данных помещен истинный адрес
BBSY*	**Шина занята (bus busy).** Сигнал, возбуждаемый формирователем с открытым коллектором на выходе, устанавливается в низкое состояние запросчиком, чтобы показать, что его задатчик использует шину пересылки данных. Когда задатчик освобождает эту линию, появляющийся в результате положительный перепад сигнала побуждает арбитр считать сигналы запроса шины и предоставить ее запросчику с наивысшим приоритетом
BCLR*	**Очистить шину (bus clear).** Сигнал, возбуждаемый формирователем с двумя состояниями на выходе, формируется арбитром для указания того, что имеется запрос шины более высокого приоритета. Этот сигнал требует, чтобы текущий задатчик освободил шину пересылки данных
BERR*	**Ошибка в магистрали (bus error).** Сигнал, возбуждаемый формирователем с открытым коллектором на выходе, вырабатывается исполнителем или шинным таймером. Этот сигнал показывает, что пересылка данных не завершена
BG0IN*—BG3IN*	**Вход (0—3) предоставления шины (bus grant (0—3)IN).** Сигналы, возбуждаемые формирователями с двумя состояниями на выходе, вырабатываются арбитром. Сигналы «вход предоставления шины» и «выход предоставления шины» образуют последовательные цепочки предоставления шины. Сигнал «вход предоставления шины» показывает модулю, получающему его, что он может использовать шину пересылки данных

Продолжение таблицы В.1

Обозначение	Название и описание сигнала
BG0OUT*—BG3OUT*	**Выход (0—3) предоставления шины (bus grant (0—3)OUT).** Сигналы, возбуждаемые формирователями с двумя состояниями на выходе, вырабатываются запросчиками. Выходной сигнал предоставления шины показывает следующему модулю в цепочке, что он может использовать шину пересылки данных
BR0*—BR3*	**Запрос шины (0—3) (bus request (0—3)).** Сигналы, возбуждаемые формирователями с открытым коллектором на выходе, вырабатываются запросчиками. Низкий уровень сигнала на одной из этих линий показывает, что какому-то задатчику необходимо использовать шину пересылки данных
D00—D31	Линии **данных (data)**. Сигналы двунаправленных линий данных, возбуждаемые формирователями с тремя состояниями на выходе, используются для пересылки данных между задатчиками и исполнителями и информации статуса/идентификации от прерывателей к обработчикам прерываний
DS0*, DS1*	**Строб данных 0 (data strobe 0), строб данных 1 (data strobe 1).** Сигналы, возбуждаемые формирователями с тремя состояниями на выходе, используются совместно с сигналами LWORD* и A01 для указания количества пересылаемых байтов данных (1, 2, 3 или 4). Во время цикла записи отрицательный перепад первого из стробов указывает, что на шине пересылки данных данные истинны. В цикле считывания положительный перепад первого из стробов указывает, что данные с шины пересылки данных приняты
DTACK*	**Подтверждение пересылки данных (data transfer acknowledge).** Сигнал, возбуждаемый формирователем с открытым коллектором на выходе, вырабатывается исполнителем. Отрицательный перепад этого сигнала указывает, что данные, помещенные на линии данных во время цикла считывания, — истинны или же, что данные приняты с линий данных во время цикла записи. Положительный перепад сигнала указывает, что исполнитель освободил линии данных в конце цикла считывания
GND	**«Земля» (ground).** Опорный потенциал источников питания постоянного тока для магистрали VME
IACK*	**Подтверждение прерывания (interrupt acknowledge).** Сигнал, возбуждаемый формирователем с открытым коллектором или тремя состояниями на выходе, используется обработчиком прерываний, подтверждающим запрос прерывания. Он проходит по магистральной линии объединительной платы к контакту IACKIN* гнезда 1, где отслеживается формирователем последовательной цепочки подтверждения прерывания
IACKIN*	**Вход подтверждения прерывания (interrupt acknowledge in).** Сигнал, возбуждаемый формирователем с двумя состояниями на выходе, показывает модулю магистрали VME, принимающему этот сигнал, что ему разрешено отвечать в текущем цикле подтверждения прерывания. Сигналы IACKIN* и IACKOUT* образуют последовательную цепочку
IACKOUT*	**Выход подтверждения прерывания (interrupt acknowledge out).** Сигнал, возбуждаемый формирователем с двумя состояниями на выходе, посылается модулем для указания следующему в последовательной цепочке модулю, что ему разрешено отвечать в текущем цикле подтверждения прерывания. Сигналы IACKIN* и IACKOUT* образуют последовательную цепочку
IRQ1*—IRQ7*	**Запрос прерывания (1—7) (interrupt request (1—7)).** Сигналы, возбуждаемые формирователями с открытым коллектором на выходе, вырабатываются прерывателями, которые выдают запросы прерывания. Когда несколько линий прерывания контролируются одним обработчиком прерываний, линии с наибольшим номером присваивается наивысший приоритет

Окончание таблицы В.1

Обозначение	Название и описание сигнала
LWORD*	**Длинное слово (long word).** Сигнал, возбуждаемый формирователем с тремя состояниями на выходе, используется совместно с сигналами DS0*,DS1* и A01 для выбора байтовой (байтовых) ячейки (ячеек) в четырехбайтовой группе, доступной (доступных) во время пересылки данных
RESERVED	**Зарезервировано (reserved).** Сигнальная линия, зарезервированная для будущего усовершенствования магистрали VME
SERCLK	**Тактовый сигнал последовательной магистрали (serial clock).** Сигнал, возбуждаемый формирователем с двумя состояниями на выходе, используется для синхронизации передачи данных по магистрали МЭК 823
SERDAT*	**Данные последовательной магистрали (serial data).** Сигнал, возбуждаемый формирователем с открытым коллектором на выходе, используется для передачи данных по магистрали МЭК 823
SYSCLK	**Системный тактовый сигнал (system clock).** Сигнал, возбуждаемый формирователем с двумя состояниями на выходе, обеспечивает магистраль тактовым сигналом постоянной частоты 16 МГц, который не зависит от каких-либо других временных соотношений в магистрали
SYSFAIL*	**Системный отказ (system fail).** Сигнал, возбуждаемый формирователем с открытым коллектором на выходе, показывает, что в работе системы произошел отказ. Этот сигнал может быть сформирован любым модулем магистрали VME
SYSRESET*	**Системный сброс (sistem reset).** Сигнал, возбуждаемый формирователем с открытым коллектором на выходе, низкий уровень которого вызывает сброс системы в исходное состояние
WRITE*	**Запись (write).** Сигнал, возбуждаемый формирователем с тремя состояниями на выходе, формируется задатчиком и указывает тип цикла пересылки данных: считывание или запись. Высокий уровень сигнала указывает, что выполняется операция считывания, низкий — операция записи
+5V STDBY	**Резервное напряжение +5 В постоянного тока (+5V d.c. standby).** Обеспечивает подачу напряжения +5В постоянного тока устройствам, требующим батарейное питание
+5V	**Питание +5В постоянного тока (+5V d.c. power).** Используется логическими схемами системы
+12V	**Питание +12В постоянного тока (+12V d.c. power).** Используется логическими схемами системы
—12V	**Питание —12В постоянного тока (—12V d.c. power).** Используется логическими схемами системы

ПРИЛОЖЕНИЕ С
(справочное)

Использование линий SERCLK и SERDAT*

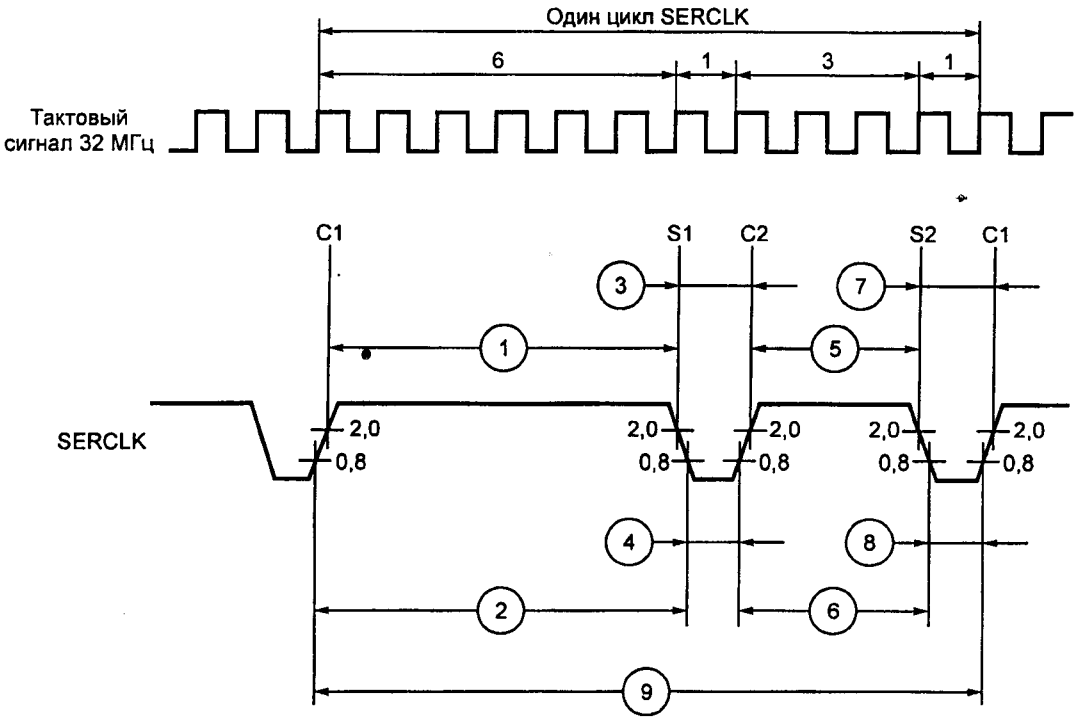
Две сигнальные линии на объединительной плате магистрали VME (SERCLK и SERDAT*) предназначены для использования магистрали МЭК 823 и обеспечивают последовательный канал связи между модулями. Протокол, используемый в магистрали МЭК 823, выходит за рамки настоящего стандарта. Поскольку разработчики модуля системного контроллера могут пожелать включить в свой модуль формирователь сигнала SERCLK, в данном приложении дается необходимая информация. Сигнал SERCLK, подобно сигналу SYSCLK, не имеет фиксированной временно́й связи с каким-либо сигналом магистрали VME (кроме сигнала SERDAT*, который несет биты данных, синхронизированные сигналом SERCLK).

Формирователи и приемники для сигналов SERCLK и SERDAT* определены в разделе 6 настоящего стандарта.

На рисунке С.1 и в таблице С.1 приведены необходимые временны́е параметры сигнала SERCLK.

Сигнал с указанными значениями временны́х соотношений может быть получен из тактового сигнала частотой 32 МГц. Значения, указанные в таблице С.1, предназначены для случаев, когда линии SERCLK и SERDAT* не выходят за пределы объединительной платы магистрали VME. Если же эти линии распространяются за пределы объединительной платы, то для получения дополнительной информации необходимо обратиться к стандарту МЭК 823.

Рекомендация С.1 При проектировании формирователя тактового сигнала последовательной магистрали рекомендуется учитывать, что задержки распространения сигнала на линии SERCLK для положительного и отрицательного перепадов будут, вероятно, различными. Это различие особенно существенно, когда линия SERCLK сильно нагружена. При расчете задержек распространения сигнала рекомендуется использовать справочные технические данные, указываемые изготовителем ИС формирователя для емкостной нагрузки 300 пкФ. Если значения задержки даются только для нагрузки 30 пкФ, рекомендуется к каждому значению задержки прибавлять 10 нс.



Значения временны́х параметров см. в таблице С.1

Рисунок С.1 — Временна́я диаграмма сигнала SERCLK

Таблица С.1 — Значения временных параметров сигнала SERCLK

Номер параметра	Значение параметра, нс	
	не менее	не более
1	167	—
2	—	194
3	—	51
4	25	—
5	74	—
6	—	100
7	—	51
8	25	—
9	340	347

Предложение С.1 Предлагается проектировать формирователь тактового сигнала последовательной магистрали так, чтобы с помощью перемычек можно было организовывать его работу от различных разрядов двоичного счетчика, который запускается от тактового генератора с частотой 32 МГц. Это позволяет использовать в качестве базовой частоты для формирователя частоты 32 МГц, 16 МГц, 8 МГц и т.д., что упрощает выбор частоты, соответствующей длине линий SERCLK и SERDAT*.

Предложение С.2 Для того, чтобы несколько модулей, содержащих формирователь тактового сигнала последовательной магистрали, могли быть установлены в одну и ту же объединительную плату, предлагается проектировать их с перемычкой, отключающей формирователь от линии SERCLK.

Замечание С.1 Если для формирования сигнала SERCLK используется генератор тактового сигнала 32 МГц, то он может быть использован также для формирования 16-мегагерцового сигнала SYSCLK магистрали VME.

ПРИЛОЖЕНИЕ D
(справочное)

Метастабильность и ресинхронизация

D.1 Введение

Правильная работа логических устройств базируется на определенных временных соотношениях между входным и тактовым сигналами (например, требованиях ко времени установления и удержания для D-триггеров). Во многих случаях знание таких соотношений не может быть гарантировано. Так сигналы подсистем, вырабатываемые по независимым синхросигналам, и сигналы, поступающие извне, служат примером асинхронного взаимодействия, где могут возникнуть всевозможные временные соотношения.

Когда не соблюдаются ограничения временных соотношений входных сигналов последовательной схемы (в простейшем случае триггер), то ее поведение может стать непредсказуемым или предсказуемым только статистически, вследствие явления, называемого **метастабильностью**, которая может стать причиной трудно выявляемых или случайных ошибок. Метастабильность неизбежна, поскольку она связана с основной неопределенностью времени принятия решения: для двух фронтов отстоящих друг от друга на 1 мкс легко решить, который будет первым, немного труднее это сделать, если их отделяет 1 нс, и, практически, невозможно, если время сдвига составляет 1 нс (любая физическая система имеет конечное временное разрешение). Однако правильная методика конструирования и выбор надлежащей технологии могут наложить известные ограничения на вероятность опасности метастабильности.

С возрастанием быстродействия цифровых систем обработка метастабильности потребует от разработчика аппаратной части определенных знаний. В данном приложении представлена основная информация и специфические советы о том, как обращаться с некоторыми наиболее важными элементами интерфейса магистрали VME. В последующих разделах приложения описана возможность появления метастабильности внутри устройств. Затем представляется модель количественного анализа и, наконец, предлагаются разработчикам пути контроля проблемы. Для знакомства с метастабильностью первым используется RS-триггер. Подробный анализ и модели приводятся для D-триггеров, которые обычно используются для ресинхронизации. Последняя часть посвящена магистрали VME и описывает возникновение метастабильности в подсистемах арбитража и прерывания и методы контроля ее влияния.

D.2 Основы метастабильности

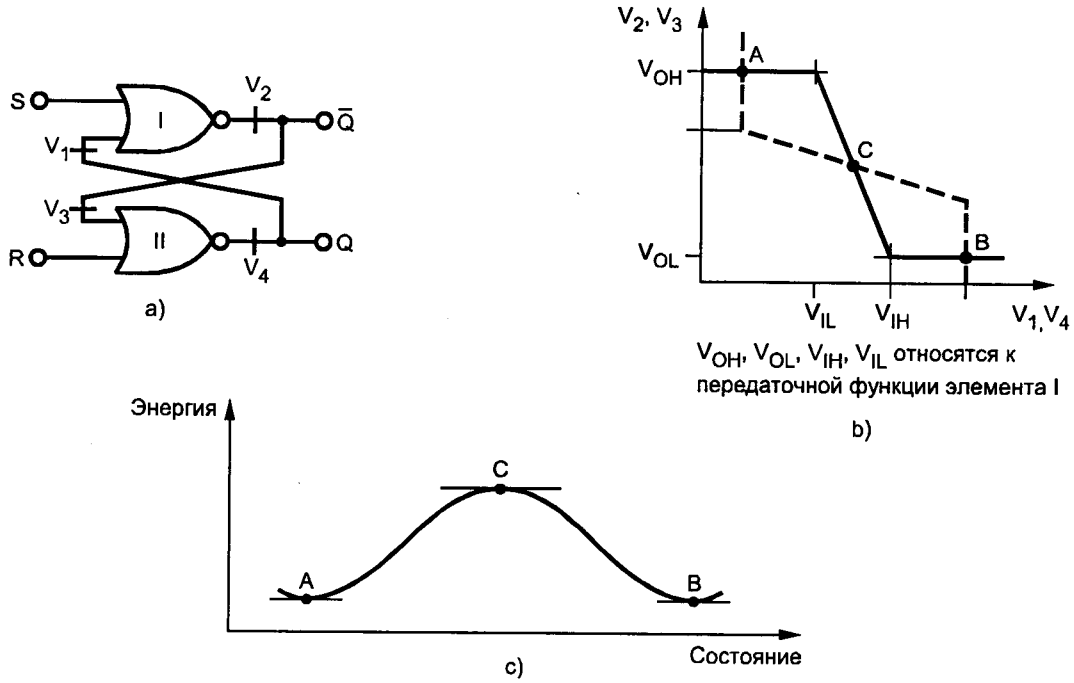
Передаточные функции двух инвертирующих логических элементов (ИЛИ—НЕ или И—НЕ), которые образуют RS-триггер (см. рисунок D.1a), изображены графически на рисунке D.1b. На этом рисунке показано, как наличие двух стабильных состояний (A,B) подразумевает также наличие третьего состояния нестабильного равновесия, называемого **метастабильным состоянием** (C). Другая точка зрения представлена на рисунке D.1c — состояниям установки и установки и сброса триггера соответствуют состояния минимума энергии. Между этими двумя минимумами всегда существует максимум, плоская вершина которого представляет точку метастабильного равновесия.

Если система устанавливается в метастабильное состояние (т.е. точно на максимум) и не подвергается возмущениям, то она будет оставаться в этом состоянии неограниченное время. Если система устанавливается вблизи максимума, то она будет переходить в направлении одного из стабильных состояний в течение времени, называемого **разрешающим временем**. Разрешающее время зависит от характеристик триггера (насколько острым является максимум) и может быть предсказано статистически.

В RS-триггере метастабильность может возникнуть под влиянием входного импульсного сигнала критической длительности или критической амплитуды (в области между V_{IL} и V_{IH}) или при принудительной установке входов S и R в «запрещенное» состояние (оба активны), а затем при установке обоих входов в неактивное состояние с критическим временным сдвигом (см. рисунок D.2a). В D-триггерах метастабильность происходит в тот момент, когда изменение сигнала на входе D и тактирующий перепад тактового сигнала слишком близки (время установления или удержания не соблюдается) или когда импульс тактового сигнала слишком узок (эти входные условия вызывают импульсы критической длительности для внутренних бистабильных схем), или если импульсы тактового сигнала находятся в диапазоне критической амплитуды (см. рисунок D.2b). Все эти условия, которые могут вызвать метастабильность, называются **критическими соотношениями входных сигналов**.

При критических условиях входных сигналов могут произойти два явления:

- **аналоговая метастабильность**, где оба выходных сигнала имеют уровни напряжения, являющиеся промежуточными между точно высоким и точно низким уровнями $V_{OH/VOL}$ (триггер работает в точке C рисунка D.1b);
- **колебательная метастабильность**, которая заставляет оба выходных сигнала колебаться в фазе (при $Q = \overline{Q}$) в течение некоторого времени. Эти два нарушения работоспособности показаны на рисунке D.3. Аналоговое или колебательное поведение зависит от структуры триггера и семейства логики, а более всего от соотношения между задержкой распространения сигнала логическим элементом и временем нарастания и



a) схема на элементах ИЛИ—НЕ; b) передаточная функция элемента;
c) энергетическая функция триггера

Рисунок D.1 — Базовый RS-триггер

спада сигнала. Колебательная метастабильность происходит вследствие сочетания времен распространения (нарастания) спада, что очень редко встречается при правильном конструировании интегральных триггеров (задержки распространения намного больше времени перехода. Поэтому далее будет рассмотрена, в основном, аналоговая метастабильность.

Правильное конструирование и использование триггеров Шмитта смогут устранить критические условия входных сигналов в области амплитуды (неправильные уровни входных сигналов) или, по крайней мере, преобразовать их в критические условия во временной области. С другой стороны, критические условия временных соотношений входных сигналов неизбежны, когда цифровая система должна поддерживать связь с внешними устройствами (обычно не синхронизированными с внутренним тактовым сигналом), или для сигналов, которыми обмениваются между собой подсистемы, использующие различные тактовые сигналы.

До тех пор, пока триггер остается в метастабильном состоянии, логический элемент, подсоединенный к его выходу, может рассматривать метастабильный уровень как высокий или низкий, в зависимости от его конкретных электрических характеристик, которые могут незначительно отличаться даже в пределах одной и той же интегральной схемы (см. рисунок D.4). Более того, этот логический элемент также может перейти в свою линейную область и распространять неверный уровень на свой выход. Серьезные сбои могут возникнуть из-за этих многочисленных интерпретаций. Например, если выходной сигнал арбитра переходит в метастабильное состояние, два запросчика могут попытаться использовать ресурс одновременно.

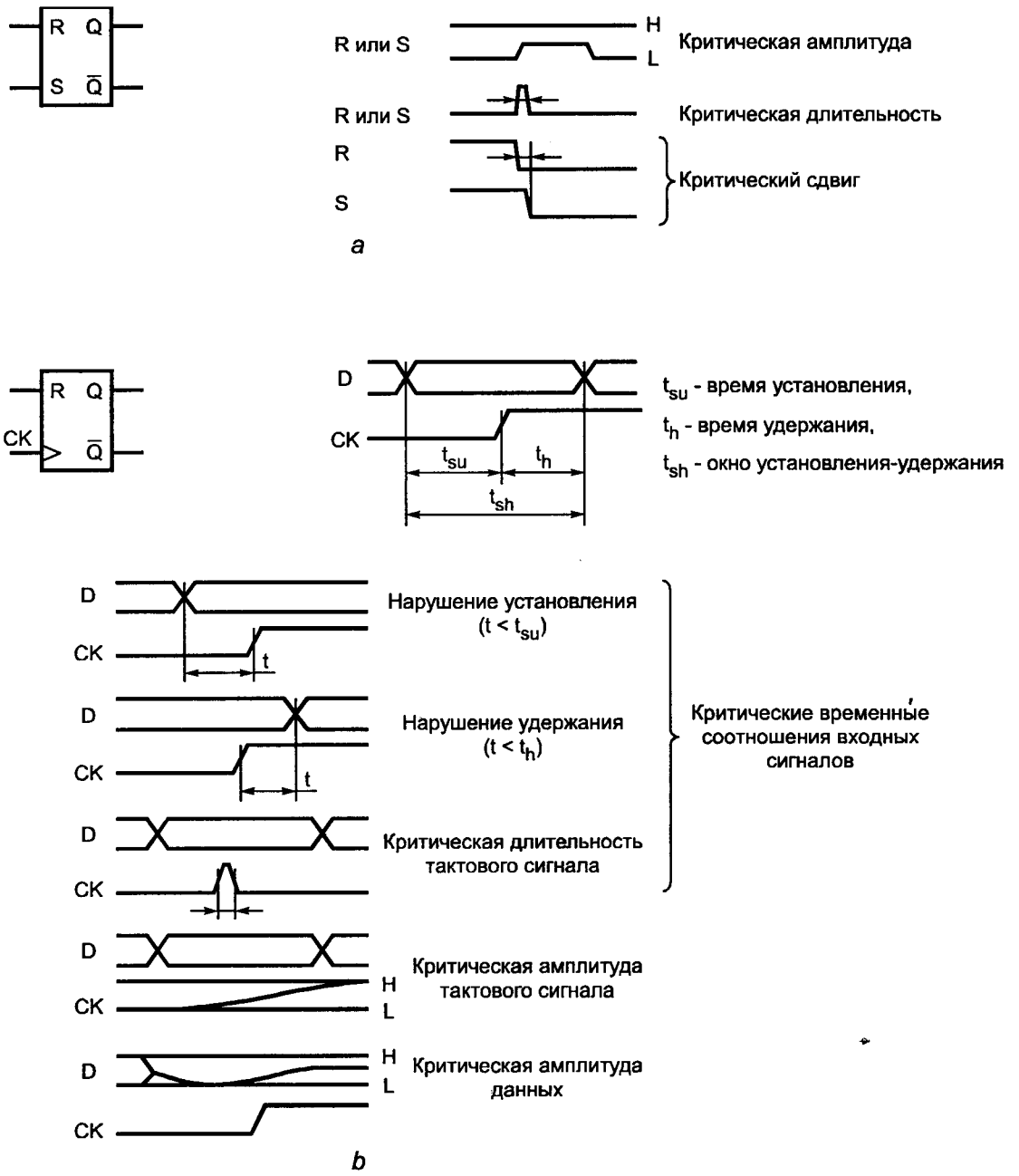
Через некоторое время триггер переходит в стабильное состояние, т.е. на его выходах устанавливаются комплементарные логические состояния со значениями напряжения уровней в допустимых пределах. Длительность метастабильного состояния может быть охарактеризована только в статистической форме. В ряде специальных работ, выходящих за рамки данного приложения, рассматривается простая, но точная модель продолжительности метастабильности и утверждается, что если бистабильная схема находится в метастабильном состоянии при $t = 0$, то вероятность оставаться в этом состоянии более, чем заранее определенное время t' равна:

$$P(\mathrm{MT}) = e^{-t'/\tau}. \qquad (D.1)$$

Параметр τ типичен для каждого семейства интегральных схем (ИС) и может быть измерен или (реже) получен из таблицы данных на ИС. Если D-триггер с частотой тактирования F_c фиксирует некоррелированный сигнал с частотой перехода F_d, то частота метастабильных состояний F_{ms}, которым требуется более t', чтобы перейти в стабильное распознаваемое состояние, равна:

$$F_{ms} = F_c \cdot F_d \cdot T_o \cdot e^{-t'/\tau}, \qquad (D.2)$$

где параметры T_o и τ зависят от технологии изготовления триггера.



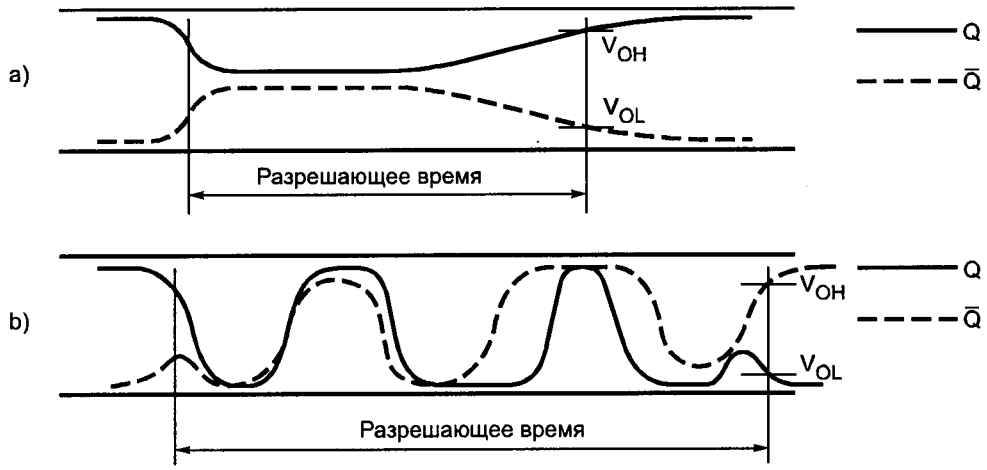
a) RS-триггер; b) D-триггер.

Рисунок D.2 — Критические условия входных сигналов

Также можно сказать, что будет сгенерировано метастабильное состояние более длительное, чем t', если переход данных происходит в пределах временного окна T_{cw}, перекрывающего тактирующий перепад тактового сигнала и имеющего длительность:

$$T_{cw} = T_o \cdot e^{-t'/\tau}. \qquad (D.3)$$

Это **критическое окно** помещается внутри более широкого временного интервала, определяемого границами времени установления и удержания. Точное положение критического окна — неизвестно. Его длительность зависит от технологии изготовления бистабильной схемы (параметры T_o и τ) и от времени t', заданного для триггера разрешением его метастабильности — чем длиннее разрешающее время, тем у́же критическое окно. Точнее, фактическое разрешающее время содержит внутреннюю задержку на прохождение — h, кото-



a) — аналоговая метастабильность; b) — колебательная метастабильность.

Рисунок D.3 — Два типа метастабильности

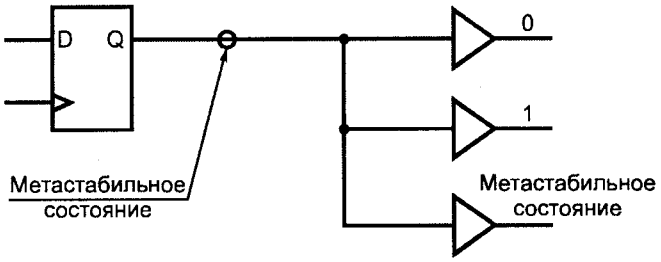


Рисунок D.4 — Различные интерпретации метастабильности выходного сигнала

рая используется для передачи сигналов с входных выводов на внутренний основной триггер-защелку и с него — на выходные выводы, и поэтому исключается для разрешающей способности метастабильности (необходимо заменить t' на $t' - h$ в вышеприведенных формулах).

Разрешающее время соразмерно задержке на прохождение: нельзя доверять выходному сигналу триггера, пока задержка, равная разрешающему времени, не истечет с момента любого изменения сигнала на входе. Прямым способом уменьшения вероятности того, что метастабильное состояние может вызвать появление ошибок, является увеличение разрешающего времени t', т.е. ожидать более длительное время. Другой возможностью является уменьшение ширины критического окна (это снижает вероятность вхождения в метастабильное состояние). Это соответствует уменьшению параметров T_0 и τ и может быть достигнуто путем выбора логического семейства с меньшими значениями T_0 и τ.

Вышеописанное поведение относится к аналоговой метастабильности. Колебательная метастабильность возникает при тех же ранее описанных критических входных условиях (временные соотношения или уровень), а продолжительность метастабильного состояния имеет то же самое статистическое распределение, что и при аналоговой метастабильности.

Следует указать, что даже избыточность аппаратной части не сможет замаскировать ошибки метастабильности. Однако можно использовать формулы (D.1—D.3) для характеризации и определения границ этих ошибок. Разработчик может решать проблему метастабильности двумя путями: на высоком уровне — путем проектирования систем, способных исправлять влияние ошибки метастабильности, и на уровне схем — путем выбора наилучшей ИС для критических устройств синхронизации.

D.3 Метастабильность в сопряжениях шины

Метастабильность возникает на границах синхронизации между асинхронными компонентами. В сопряжении шины ошибки метастабильности могут быть следствием взаимодействия сигналов, поступающих с автоматов, синхронизированных отдельными генераторами, таких как два или более процессора, или, для тактируемых шин, сигналов, генерируемых модулями, которые не синхронизованы с тактовым сигналом шины. Например, шина, синхронизируемая тактовым сигналом частотой 10 МГц, получая сигналы с процессора, имеющего тактовую частоту 16 МГц, должна осуществлять их выборку собственным тактовым сигналом (это действие преобразует скорость передачи сигналов с 16 МГц в 10 МГц) и наоборот — процессор с тактовой частотой 16 МГц получает сигналы с синхронизируемой (10 МГц) или асинхронной шины. При этой

ресинхронизации сопряжений могут встретиться любые фазовые соотношения между входными сигналами и тактовым сигналом выборки.

Сигналы с различных процессоров взаимодействуют в арбитре шины, а сигналы модуль/шина — в интерфейсе шины. Также некоторые контроллеры входа/выхода, такие как последовательные интерфейсы, используют несколько источников тактовых сигналов, но они менее чувствительны к проблеме метастабильности, поскольку пониженная скорость уменьшает вероятность ошибки и позволяет получить более длительное разрешающее время.

Метастабильность может возникнуть в любой момент, когда взаимодействуют сигналы, синхронизированные различными тактовыми сигналами и совсем не синхронизированные. Поэтому она проистекает не из протокола шин, а из наличия нескольких генераторов тактовых сигналов. Ресинхронизация сопряжений, возможно скрытая в каком-либо функциональном узле структурной схемы, должна существовать всегда. Технические требования протокола шины могут предупредить разработчика, посоветовать, где поместить ресинхронизации сопряжений, и снабдить данными для оценки вероятности ошибки.

Значительные усилия и изобретательность были проявлены при разработке триггеров, имеющих насколько возможно малые времена установления и удержания, а достижения в области технологии ИС в действительности сделали эти времена все меньше и меньше. Сейчас возможно разработать триггеры, у которых одно из этих времен равно нулю или даже отрицательное, но ни одна схема не сможет устранить оба этих времени — их сумма (называемая в дальнейшем **окном установления-удержания**) всегда больше нуля. Единственной проблемой улучшения триггера являются технологические усовершенствования, которые делают возможным достижение меньшего окна установления-удержания, дающего возможность поднять системные характеристики на более высокий уровень посредством использования более быстрых тактовых сигналов, бесконечно уменьшая частоту возможной ошибки метастабильности.

Помимо уменьшения времени установления и удержания, повышенная степень интеграции сделала последствия неудовлетворения их менее жесткими. В то время, как триггер-защелка с внешней обратной связью легко может возбудиться и вести себя как генератор в течение нескольких циклов (колебательная метастабильность), современная ИС триггера с уверенностью может обеспечивать не более одного дополнительного изменения полярности напряжения, если критическое окно будет пропущено, и/или его выходной сигнал остается в области неопределенного напряжения только в течение относительно короткого времени.

Более того, эффективное критическое окно (согласно формуле D.3), где триггер фактически работает неправильно, обычно на два-три порядка меньше, чем окно установления-удержания, которое может гарантировать изготовитель ИС. Быстродействующий триггер может иметь окно установления-удержания в несколько наносекунд. Где-то в пределах этого окна существует промежуток времени намного короче (критическое окно), во время которого триггер может вести себя непредсказуемо, если данные имеют промежуточное значение напряжения. Хотя этот факт представляет интерес, он не имеет ценности для разработчика, поскольку нельзя сказать, где действительное критическое окно будет находиться внутри окна установления-удержания.

D.4 Синхронная схема в сравнении с асинхронной

При решении проблемы устройств с запоминанием состояний возникли две школы методологии конструирования. Каждая из методологий имеет свои преимущества и недостатки и каждая из них занимает свое собственное место в области конструирования электронных систем. Фактически, большинство электронных конструкций, несмотря на преобладание той или иной, используют смесь этих двух методологий.

В **синхронных** конструкциях применяется основной тактовый сигнал, который используется для стробирования всех их триггеров. Изменения состояния системы могут произойти только в соответствии с активным перепадом тактового сигнала. Внешние сигналы синхронизируются путем подачи каждого из них на D-вход триггера. Сигналы с выходов этих триггеров поступают на внутреннюю комбинационную логическую схему, которая должна установиться до следующего активного перепада тактового сигнала. Частота тактового сигнала этих триггеров ограничена желаемой вероятностью ошибки метастабильности — период тактового сигнала должен быть больше суммы разрешающего времени и задержки на распространение сигнала в логике.

В **асинхронных** конструкциях изменения состояний синхронизируются входными сигналами, которые могут поступать непосредственно на входы синхронизации D-триггеров, или на входы установки/сброса RS-триггеров. Правильная последовательность состояний может быть гарантирована приемами конструирования, использующими один триггер на каждое состояние. Метастабильные состояния могут возникать только тогда, когда два или более входных сигнала переключаются в пределах критического окна, и могут быть замаскированы фильтрацией выходных сигналов триггеров, пропусканием их через схему и совместно с задержанным тактовым сигналом.

В обоих случаях ошибки метастабильности удается избежать за счет затраты времени — уменьшенной частоты тактового сигнала для синхронных конструкций или дополнительно задержанных входных сигналов в асинхронных конструкциях.

В то время как относительные достоинства синхронных и асинхронных конструкций могут подвергаться длительному обсуждению их сторонниками, объективное рассмотрение конкретной задачи указывает, какой тип схемы для нее является наилучшим. Когда цель разработки хорошо и четко определена, предпочтение отдается асинхронной конструкции, поскольку она исключает издержки времени по синхронизации входных

сигналов и поскольку каждая часть схемы работает настолько быстро, насколько она способна, что быстрее чем при скорости, определяемой самой медленной функцией.

По мере увеличения сложности поставленной задачи, особенно, когда она включает длинные последовательности воздействия, синхронная конструкция предпочтительна, поскольку она позволяет разработчику сосредоточиться на требуемой функции более, чем на элементах ИС, используемых для выполнения этой функции. Синхронная конструкция быстрее адаптируется к изменениям функциональных требований и отработке ситуаций, не предусмотренных на начальных этапах проектирования.

Но, в свою очередь, пригодность методологии синхронных конструкций для более сложных схем имеет свой верхний предел. В частности, с увеличением физического размера конструкции увеличивается и время, необходимое для достижения основным тактовым сигналом всех составляющих схем, и решение проблемы сдвига тактового сигнала относительно других сигналов ужесточается. За пределами определенной точки сложности конструкции лучше перейти к разбиению ее на функциональные блоки, каждый из которых реализуется синхронно или асинхронно в зависимости от того, что диктуют его характеристики.

Необходимо отметить, что обе методологии асинхронных и синхронных конструкций предусматривают знание поведения триггеров, когда нельзя удовлетворить требования времени установления и удержания. Для синхронных конструкций эта информация является составной частью определения частоты тактового сигнала и конфигурации устройств синхронизации входных сигналов, в то время как для асинхронных конструкций она обычно используется для определения характеристик линий задержки.

D.5 Определение времени разрешения триггеров

Для измерения характеристик метастабильности триггеров могут быть использованы четыре метода:

а) сигнал на входе данных переключается одновременно с тактирующим перепадом тактового сигнала;

б) оба сигнала: на входах установки и сброса — переключаются одновременно;

в) сигнал на входе данных устанавливается таким, чтобы триггер должен был изменить свое состояние, а затем в качестве тактового сигнала подается короткий импульс;

г) короткий импульс подается либо на вход установки либо на вход сброса.

Полученные результаты позволяют вычислить для ИС триггеров три константы: τ, T_0 и h. Эти значения затем могут быть использованы для расчета среднего времени между отказами (Mean Time Between Failure — MTBF) для ИС, когда она используется в качестве устройства синхронизации и ей для разрешения дается время t'. Из формулы (D.2) получаем:

$$\mathrm{MTBF} = \frac{1}{F_{ms}} = \frac{1}{F_c \cdot F_d \cdot T_0} \cdot e^{t'/\tau} \qquad (\text{для } t' > h), \tag{D.4}$$

где MTBF и T_0 — указаны в наносекундах;

частота тактового сигнала F_c и частота переключения данных F_d — в герцах;

t' и τ — выражены в одних и тех же единицах времени.

Чтобы использовать эту формулу для ситуации гонки между сигналами на входах установки и сброса, необходимо вместо частоты тактового сигнала и частоты переключения данных просто подставить среднюю частоту сигнала разблокирования каждого из входов. Для использования формулы для ситуации гонки между перепадом тактового сигнала и сигналом разблокировки входа установки или сброса следует вместо частоты переключения данных подставить среднюю частоту разблокировки.

Формула справедлива для переходов данных, равномерно распределенных по периоду тактового сигнала, и для значений t', превышающих минимальное значение h. В таблице D.1 приведены результаты лабораторных исследований для τ, T_0 и h для общедоступных ИС. В последующих колонках таблицы D.1 приведены значения MTBF для каждой ИС при t' = 20, 30 и 40 нс, частоте тактового сигнала 25 МГц и частоте переключения данных 100 кГц (эквивалентно частоте сигнала 50 кГц). В правой крайней колонке приведены значения t', которые необходимы для получения MTBF, равного 10^9с (около 32 лет) при тех же частотах переключения.

Таблица D.1 — Данные метастабильности

Тип ИС	Количество/ изготовитель	Дата изготовления	Категория значения	Экспериментальные данные			Вычисленные параметры			
				τ, нс	T_0	h, нс	MTBF, с для t'			t', нс (MTBF 10^9 с)
							20 нс	30 нс	40 нс	
7400, RS-тр.	1/T	1973		3,2	0,2 мкс	29	—	0,024	0,54	110
74S00, RS-тр.	1/T	1972		1,8	1 мкс	17	—	—	1800	64

Окончание таблицы D.1

Тип ИС	Количество/ изготовитель	Дата изготовления	Категория значения	Экспериментальные данные			Вычисленные параметры			
				τ, нс	T_0	h, нс	MTBF, с для t'			t', нс (MTBF 10^9 с)
							20 нс	30 нс	40 нс	
74LS74	3/T	1974		1,5	0,4 с	35	—	—	0,38	73
74S74	59/F	74—75	Лучшее	0,40	0,2 с	13	1×10^{10}	7×10^{20}	5×10^{31}	19
			Худшее	0,89	2 мс	13	1,1	9×10^{4}	7×10^{9}	38
	5/T	73—75	Лучшее	0,79	2 мс	15	20,0	6×10^{6}	2×10^{12}	34
			Худшее	1,02	50 мкс	15	2,6	5×10^{4}	9×10^{8}	40
	5/N	1974	Лучшее	1,14	30 мкс	15	0,6	4×10^{3}	2×10^{7}	34
			Худшее	1,36	20 мкс	15	0,05	76	1×10^{5}	52
	5/S	72—74	Лучшее	0,96	0,8 мс	16	0,6	2×10^{4}	6×10^{8}	40
			Худшее	1,70	1 мкс	16	0,05	18	7×10^{3}	60
74S374	3/T	1978	Среднее	0,91	0,4 мс	15	3,5	2×10^{5}	1×10^{10}	38
74F74	5/M	1986	Лучшее	0,50	0,3 с	8	3×10^{8}	2×10^{17}	7×10^{25}	21
			Худшее	0,56	10 мкс	8	1×10^{8}	7×10^{15}	4×10^{23}	21
	5/F	1986	Лучшее	0,33	0,2 с	8	4×10^{12}	6×10^{27}	8×10^{40}	16
			Худшее	0,47	0,5 мкс	7	2×10^{10}	4×10^{19}	7×10^{28}	18
	5/F	1986	Лучшее	0,31	0,4 с	7	1×10^{16}	1×10^{30}	1×10^{44}	15
			Худшее	0,34	8 мс	7	2×10^{15}	1×10^{28}	6×10^{40}	15
74AS74	5/T	1986	Лучшее	0,49	0,1 мс	8	2×10^{9}	2×10^{18}	1×10^{27}	20
			Худшее	0,52	40 мкс	8	5×10^{8}	1×10^{17}	3×10^{25}	20
74F175	5/M	1986	Лучшее	0,35	2 с	7	1×10^{12}	3×10^{24}	9×10^{36}	17
			Худшее	0,72	60 нс	7	8×10^{6}	8×10^{12}	9×10^{18}	24
	5/F	1986	Лучшее	0,36	2 мс	7	3×10^{14}	3×10^{26}	4×10^{38}	16
			Худшее	0,43	0,7 мс	7	9×10^{10}	1×10^{21}	1×10^{31}	18
	5/F	85—86	Лучшее	0,35	0,3 с	8	9×10^{12}	2×10^{25}	6×10^{37}	17
			Худшее	0,46	0,5 мс	8	6×10^{9}	2×10^{19}	5×10^{28}	19
74F374	2/F	1980	Среднее	0,40	0,1 мс	6	2×10^{13}	2×10^{24}	1×10^{35}	16

Из таблицы D.1 можно сделать ряд главных выводов:

(1) Базовая внутренняя схема и технологическое семейство триггера или защелки (RS-триггера) имеют достаточно большое значение при определении пригодности ИС для ее использования в качестве синхронизирующего устройства.

(2) Лучшие ИС в таблице D.1 — это 74F74 и 74F374. ИС N74SLxx может быть использована в качестве синхронизирующего устройства. Триггер-защелка, построенный из элементов «И—НЕ» с внешней обратной связью, совсем непригоден для этой цели.

Чтобы использовать информацию, приведенную в таблице D.1, разработчик может следовать нижеприведенной процедуре для каждого случая, где времена установления и удержания триггера не могут быть гарантированы.

1 Выбрать значение MTBF. Это очень часто является самым трудным шагом. Он подразумевает компромисс между рабочими характеристиками и допустимой длительностью времени разрешения; между надежностью и возможностью разрешения отказа в пределах допустимого времени и между общими требованиями надежности и областью применения. В высокоэффективных конструкциях, к которым не предъявляются исключительные требования по надежности, в качестве исходного может быть использовано значение 10^8с (3 года). Если в системе имеется много устройств синхронизации, то среднее время безотказной работы (MTBF) каждого устройства синхронизации должно быть значительно выше этого значения системы. При выборе значения MTBF следует помнить, что приблизительный возраст земли составляет около 2×10^{17}с. Поэтому желание получить среднее время безотказной работы выше 10^{10}с может означать, что вы слишком оптимистичны относительно срока службы вашего устройства.

2 Выбрать ИС, которая имеет малые значения τ и T_0, но все еще удовлетворяет специфическим требованиям конструкции. Из ИС, перечисленных в таблице D.1, предлагается выбрать серию 74F.

3 Проверить, являются ли переходы данных по отношению к тактовому сигналу независимыми (произвольными) или постоянно расположенными. Если первое, то используйте среднюю частоту переключения данных в уравнении следующего шага. Если второе, то используйте среднюю частоту переключения данных вблизи тактирующего перепада, т.е. в промежутке оценочных времени установления и удержания.

4 Используя полученные значения частот тактового сигнала и данных (и/или установки и/или сброса) для этого случая, вычислить минимальное t' с помощью обратной формулы:

$$t' = \tau \ln(T_0 \cdot \mathrm{MTBF} \cdot F_c \cdot F_d).$$

5 Если выходной(ые) сигнал(ы) триггера проходит(ят) через промежуточные вентили, чьи выходные сигналы также могут иметь аномалии, необходимо прибавить к t' максимальное значение времени распространения вентиля(ей).

6 Если указанный выше аномальный сигнал используется в качестве входного для триггера, то следует прибавить время установления этого триггера. Если аномальный сигнал используется в логической схеме, которой маскируется аномалия, необходимо прибавить максимальное время маскирования и вычесть минимальное время демаскирования этой схемой. Если аномальные сигналы используются несколькими способами, необходимо прибавить наибольший из этих показателей.

7 В типичной синхронной конструкции следует обеспечить, чтобы период основного тактового сигнала был больше или равен наибольшему значению, вычисленному в шагах 1—6 для всех триггеров конструкции, которые могут перейти в метастабильное состояние. В асинхронной конструкции следует выбрать или определить линию задержки, которая обеспечивает чистый сигнал, где он необходим.

D.6 Метастабильность в магистрали VME

Протокол пересылки данных магистрали VME является асинхронным, поэтому, если времена установления/удержания и сдвиг сигнала учитываются правильно, риск возникновения метастабильности среди данных или адресов и относящихся к ним стробов отсутствует. Функциональные блоки с тактируемой логикой имеют предрасположенность к метастабильности на их асинхронных входах (сигналы поступают как извне, так и с шины). Стандарт магистрали гарантирует временные запасы между стробами и статическими сигналами, но устройства синхронизации интерфейса шина—модуль остаются за разработчиком модуля. Благодаря асинхронному протоколу нет необходимости в синхронизации модуль—шина.

Операцией протокола, которая должна быть защищена от ошибок метастабильности, является арбитраж шины, так как запросы шины синхронизируются генератором тактовых сигналов каждого процессора, а следовательно являются взаимно асинхронными. Как показано на рисунке D.5, критические условия входного сигнала могут появиться в двух секциях: последовательной цепочке (сигнал предоставления с рядом расположенного модуля используется в качестве тактового для D-триггера локального запроса) и центролизованном арбитре (запросы с разных процессоров — асинхронны).

Управляющие сигналы протокола арбитража BRx* и BGxIN* обеспечивают асинхронное квитирование, гарантирующее, что задатчик начинает использовать шину только после получения разрешения от арбитра. С другой стороны, распространение сигнала в последовательной цепочке BGxIN*/BGxOUT* не квитируется в каждом модуле и запрос может быть выставлен в то время, как предоставление распространяется по цепочке.

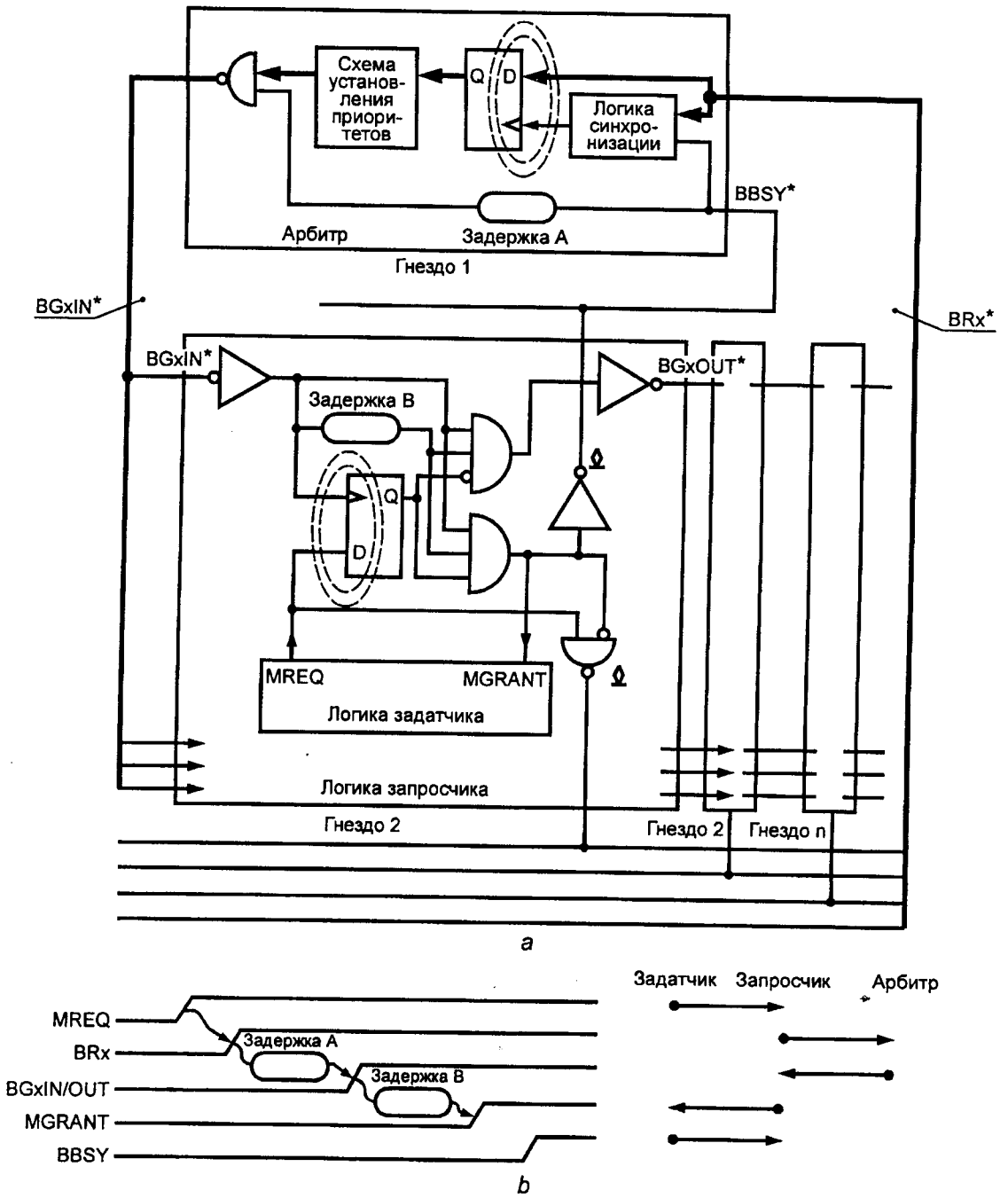
Для того, чтобы замаскировать возможное метастабильное состояние на выходе триггера синхронизации запроса, необходимо вставить задержку между фиксацией запроса и активацией локального предоставления. Эта задержка может быть получена либо с помощью линии задержки (как в подразделе D.6.1, который описывает асинхронный подход), либо с помощью тактируемого автомата (синхронный подход, см. подраздел D.6.2).

Последовательная цепочка подтверждения прерывания имеет подобные же проблемы, поскольку запросы синхронизируются локальными тактовыми сигналами прерывателя, как подробно описано в подразделе D.6.3.

Пунктирной линией окружены места, где могут возникнуть критические условия для входного сигнала.

D.6.1 Асинхронная обработка последовательной цепочки арбитража

На рисунке D.6 показан пример того, как запросчик может обрабатывать сигнал последовательной цепочки предоставления шины с помощью асинхронной логики. Логика декодирования адресов, связанная с внутримодульным задатчиком, вырабатывает истинный (высокий) сигнал «Задатчику требуется шина» (master wants bus - MWB). Когда этот сигнал находится в высоком состоянии, запросчик устанавливает сигнал на линии BRx* в низкое состояние. Сигнал «Задатчику требуется шина» не имеет определенных временных соотношений с входным сигналом BGxIN*. Когда BGxIN* переходит в низкое состояние, запросчик определяет: установить ли сигнал BBSY* в низкое состояние и принять на себя управление шиной или же передает низкий уровень со своего входа BGxIN* далее по последовательной цепочке, устанавливая сигнал на выходе BGxOUT* в низкое состояние.



a) упрощенная структурная схема; b) управляющие сигналы с указанием петли квитирования (ЗАДАТЧИК—ЗАПРОСЧИК—АРБИТР—ЗАПРОСЧИК—ЗАДАТЧИК)

Рисунок D.5 — Структура арбитража магистрали VME

Сигнал BGxIN* принимается инвентором «А». Таким образом, отрицательный перепад сигнала BGxIN* становится положительным для сигнала BGxIN, что заставляет триггер «В» (74F74) зафиксировать сигнал «Задатчику требуется шина». BGxIN также поступает в линию задержки «С», задержка которой была вычислена для выбранного MTBF, как описано в разделе D.5.

Ко времени перехода сигнала BGxDEL в высокое состояние, сигнал MWBSAMP на выходе триггера «В» должен быть достоверным (при выбранном значении MTBF). В частности, сигнал MWBSAMP будет в высоком состоянии, если сигнал MWB был в высоком состоянии. В этом случае логический элемент «D» («И—НЕ») установит сигнал MYBG* в низкое состояние, тем самым устанавливая триггер «Е» так, что сигнал BBSY* устанавливается в низкое состояние, а сигнал DRx* освобождается в результате перевода его в

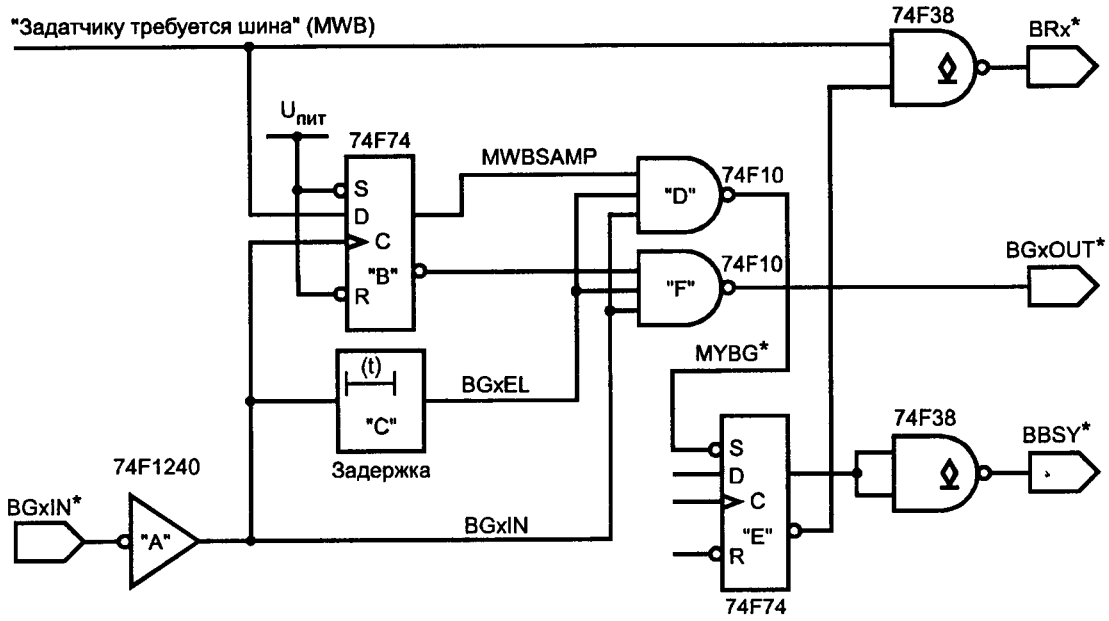


Рисунок D.6 — Асинхронная обработка последовательной цепочки арбитража

высокое состояние. Если сигнал MWBSAMP находится в низком состоянии, когда DGxDEL переходит в высокое состояние, логический элемент «F» («И-НЕ») установит сигнал BGxOUT* в низкое состояние. Отметим также, что сигнал BGxIN сам является входным для логических элементов «D», «F» («И—НЕ»), обеспечивая быструю отмену BGxOUT* и MYBG*, когда BGxIN* переходит в высокое состояние.

D.6.2 Синхронная обработка последовательной цепочки арбитража

На рисунке D.7 показан пример того, как можно спроектировать запросчик для синхронной обработки сигнала последовательной цепочки арбитража. Четыре входных сигнала фиксируются в устройстве (регистре) «А» (74F175) каждым отрицательным перепадом сигнала CLK. В то время, как BGxIN* находится в высоком состоянии, а BGxIN — в низком состоянии, логические элементы «В» и «С» разрешают высокому состоянию сигнала «Задатчику требуется шина» (MWB) распространяться и устанавливать сигнал «Запрос шины до предоставления шины» (Bus Request Before Bus Grant) BRB4BG* в низкое состояние. Как только он перейдет в низкое состояние, обратная связь через логический элемент «В» будет удерживать BRB4BG* в низком состоянии до тех пор, пока сигнал MWB снова не перейдет в низкое состояние. После того как BGxIN* перейдет в низкое состояние, а BGxIN — в высокое, сигнал MWB блокируется и BRB4BG* не может перейти в низкое состояние до тех пор, пока BGxIN остается в высоком состоянии (хотя он будет оставаться в низком состоянии, если сигнал MWB был в высоком состоянии до перехода BGxIN в низкое состояние). Последний тактирующий перепад, после которого BRB4BG* может перейти в низкое состояние, является первым, при котором BGxIN впервые фиксируется высоким, т.е. первым после которого BGDEL1 переходит в высокое состояние.

Входные сигналы MWB и BGxIN* асинхронны по отношению к сигналу CLK. Поэтому любой из сигналов BGDEL1 или BRB4BG* или их комплементарные (инверсные) сигналы могут стать метастабильными. Как было описано выше, период сигнала CLK должен быть достаточно продолжительным, чтобы перекрыть разрешающее время устройства «А» для выбранного значения MTBF плюс время на прохождение сигнала BGDEL1* через логические элементы «D» и «Е», плюс время установления для нижнего триггера в устройстве «А». Если сигнал MWB перейдет в высокое состояние перед тем как это сделает BGxIN, устанавливая BRB4BG* в высокое состояние, сигнал PREBBSY* будет зафиксирован в низком состоянии следующим тактирующим перепадом. MYBBSY перейдет в высокое состояние от этого перепада и сигнал BBSY* будет установлен в низкое состояние. В противном случае, на этом тактирующем перепаде сигнал PREBBSY* перейдет в высокое состояние. Поэтому BBSY* не будет установлен в низкое состояние. Вместо этого комбинация BGDEL2, переходящего в высокое состояние, и BRB4BG в высоком состоянии квалифицируется логическим элементом «F» (74F20), так что BGxOUT* перейдет в низкое состояние. Отметим, что сигнал BGxIN также является входным для элемента «F», обеспечивая быструю отмену BGxOUT*, когда отменен BGxIN*. Следует также отметить, что сигнал BGxIN не подключен к логическому элементу «Е», так как такое включение делает сигнал MYBBSY подверженным метастабильности, когда BGxIN* переходит в высокое состояние.

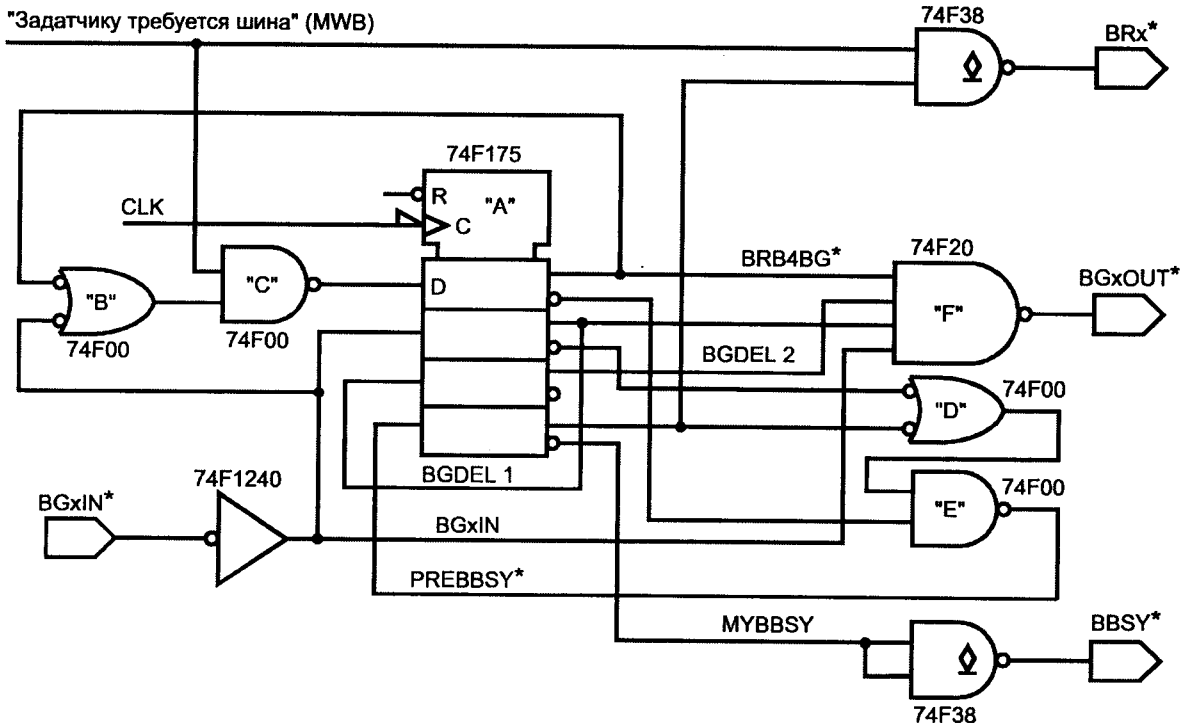


Рисунок D.7 — Синхронная обработка последовательной цепочки арбитража

MTBF этой схемы может быть вычислено из периода сигнала CLK, значений τ и T_0 для 74F175 и ожидаемых частот переключения сигналов BGxIN* и MWB.

D.6.3 Асинхронная обработка последовательной цепочки подтверждения прерывания

На рисунке D.8 приведен пример асинхронного способа обработки прерывателем сигнала последовательной цепочки подтверждения прерывания. Логическая схема, связанная с внутримодульным источником прерывания, устанавливает сигнал MYIRQ в высокое состояние, запрашивая прерывание по одной из линий IRQ1*—IRQ7*. Когда будет обнаружен цикл подтверждения прерывания, прерыватель должен определить — реагировать на цикл или нет. Если этот модуль запрашивает прерывание на подтверждаемом уровне, то он отвечает на цикл подтверждения прерывания установкой информации статуса/идентификации на линиях данных, а затем утверждает DTACK*. В противном случае он передает низкий уровень сигнала IACKIN* со своего входа далее по последовательной цепочке, устанавливая IACKOUT* в низкое состояние. Выбор между этими двумя альтернативами осуществляется устройством «А» (74F85) и отражается в сигнале MYLEVEL.

Сигнал AS* принимается и инвертируется с помощью элемента «В» (74F1240) для получения AS — входного сигнала *2*-отводной линии задержки «С». Через t_1 наносекунд после того как AS переходит в высокое состояние положительный перепад сигнала на выходе первого каскада задержки «С» фиксирует сигнал MYLEVEL в триггере «D» (74F74). Если положительный перепад фиксирует высокое состояние сигнала MYLEVEL, заставляя сигнал MYLVSAMP перейти в высокое состояние, верхний логический элемент 74F20 квалифицируется и сигнал MYIACKIN* устанавливается в низкое состояние. В противном случае, квалифицируется нижний логический элемент 74F20 и сигнал IACKOUT* устанавливается в низкое состояние. Значение t_1 вычислено таким, чтобы для сигнала MYLEVEL, формируемого из сигналов линий A01—A03, обеспечивалось время установления для триггера «D». Поскольку сигнал MYIRQ не имеет определенного временного соотношения с сигналом AS*, то нельзя полностью гарантировать время установления для триггера «D», т.е. он может перейти в состояние метастабильности, если MYIRQ переходит в высокое состояние, а AS* переходит в низкое состояние в точно правильном (неправильном) соотношении. Эта проблема решается вторым каскадом линии задержки «С». Как описано в разделе D.5, задержка t_2 между двумя выходными сигналами «С» должна быть достаточно длительной для перекрытия разрешающего времени триггера «D» при выбранном значении MTBF плюс сдвиг между элементами 74F20. Когда выходной сигнал ASD2 второго каскада линии задержки «С» переходит в высокое состояние, сигнал MYLVSAMP и его комплементарный сигнал будут достоверными. В этот момент оба элемента 74F20 будут готовы отвечать на низкий уровень сигнала IACKIN*: либо, устанавливая MYIACK* в низкое состояние и инициируя ответ этого модуля на цикл подтверждения прерывания, либо, устанавливая IACKOUT* в низкое состояние, передавая подтверждение прерывания по последовательной цепочке прерывателю следующего модуля. Отметим, что сигнал AS также является вход-

ным для элементов 74F20, обеспечивая быструю отмену IACKOUT* (или MYIACK*), когда AS* переходит в высокое состояние. Также отметим, что если в системе имеется несколько прерывателей, подобных этому, то все они будут принимать свои решения аналогично, так что прохождение сигнала по последовательной цепочке подтверждения прерывания будет продолжаться со скоростью, обеспечиваемой комбинационной логикой.

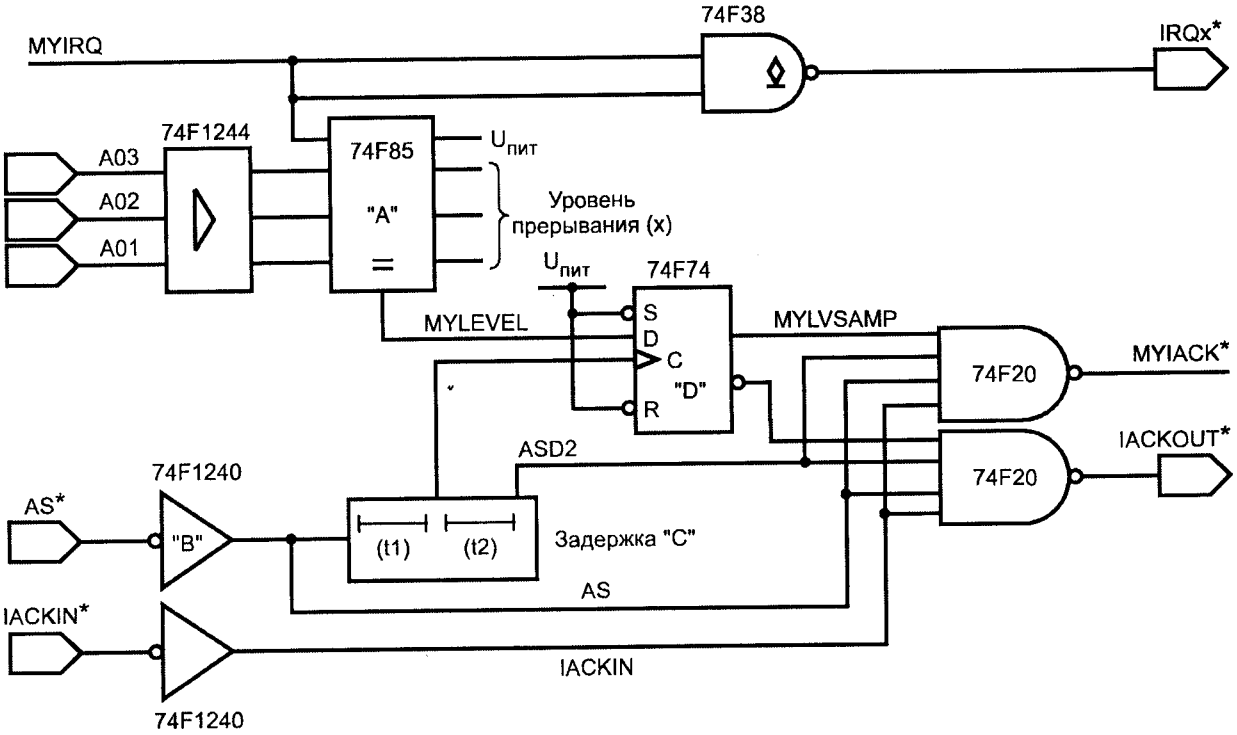


Рисунок D.8 — Асинхронная обработка сигнала последовательной цепочки прерываний

D.6.4 Асинхронная схема арбитра

На рисунке D.9 приведен пример асинхронной схемы арбитра. Сигналы четырех линий BR*, линий BBSY* и линии SYSRESET* принимаются с помощью элементов «А» и «В» (74F1244), а их выходные сигналы называются BBR0*—BBR3*, BBBSY* и BSRES* соответственно. Сигналы BBR0*—BBR3* объединяются по «ИЛИ» на элементе «С» (74F20) для получения сигнала BR, имеющего истинное значение в высоком состоянии и объединяемого с сигналами BBBSY* и BSRES* по «И» на элементе «D» (74F11) для получения на его выходе сигнала ARBGO. Отметим, что сигнал ARBGO остается в низком состоянии до тех пор, пока сигнал SYSRESET* остается в магистрали истинным. Высокое состояние ARBGO означает, что один или более из сигналов BR0*—BR3* находятся в низком состоянии и что оба сигнала BBBSY* и BSRES* находятся в высоком состоянии. Это является сигналом для арбитра начать арбитраж шины, устанавливая один из сигналов BG0IN*—BG3IN* в низкое состояние в гнезде 1 объединительной платы. Положительный перепад ARBGO фиксирует сигналы BBR0*—BBR3* в устройстве (регистре) «F» (74F175), которое может стать метастабильным, если один или более BBR3*—BBR0* переходят из высокого состояния в низкое в то время, когда ARBGO переходит в высокое состояние.

Сигнал ARBGO также является входным для линии задержки «F». Пока высокий уровень на ARBGO проходит через линию задержки, комбинаторный блок «логика предоставления» выбирает среди выходных сигналов устройства «Е» для определения следующего предоставления (логика предоставления может использовать алгоритм приоритетов, алгоритм кругового обслуживания или смесь обоих алгоритмов). Задержка «F» должна быть вычислена такой, чтобы она перекрывала время разрешения метастабильности устройства «Е» при выбранном MTBF плюс общее время на прохождение сигнала в блоке логики приоритетов, как это описано в разделе D.5. Таким образом, к тому времени, когда ARBGODEL перейдет в высокое состояние, выходные сигналы WIN0—WIN3, один из которых находится в высоком состоянии, будут стабильны. Затем выбранный логический элемент 74F00 устанавливает один из сигналов BG0IN*—BG3IN* в низкое состояние.

Отметим, что в отличие от предыдущих примеров последовательной цепочки ни BBBSY* , ни ARBGO не подаются на формирователи BGxIN* для обеспечения быстрой отмены, когда запросчик устанавливает BBSY* в низкое состояние. Схема разработана таким образом, чтобы отменять BGxIN после задержки «F» по двум причинам:

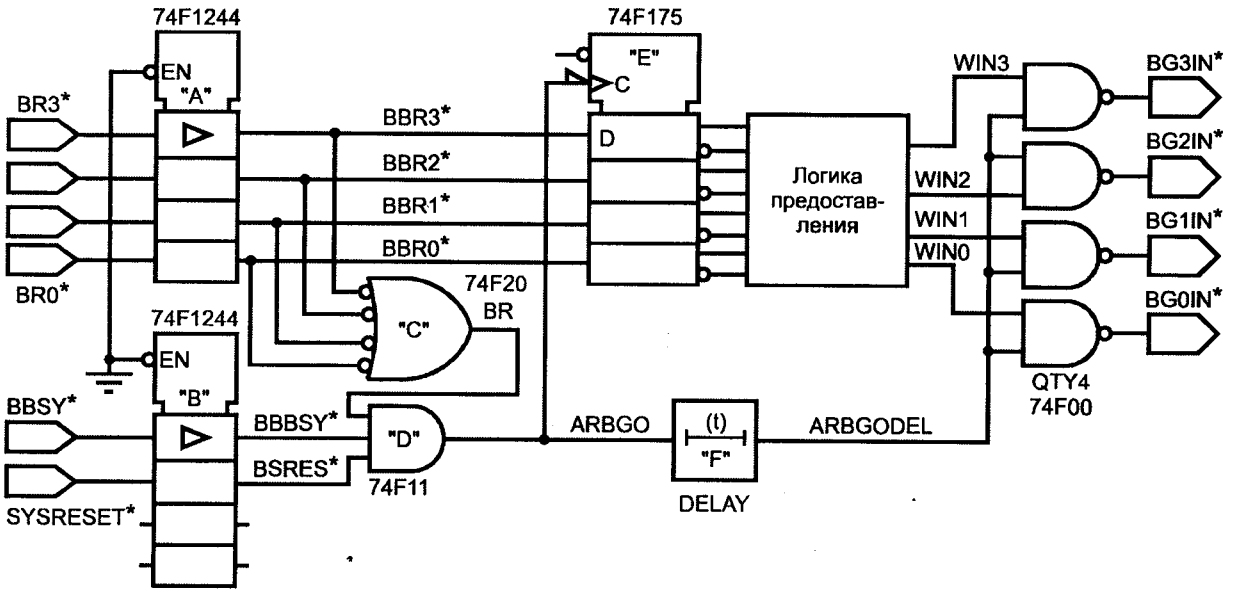


Рисунок D.9 — Асинхронная схема арбитра магистрали VME

а) нет смысла быстро отменять BGxIN*, поскольку запросчику необходимо сохранять BBSY* в низком состоянии минимум 90 нс, что является относительно длительным временем;

б) если предположить что «F» — это однокаскадная гибридная активная RLC линия задержки (не многоотводное или другое сложное составное устройство), то она будет работать как фильтр для любой помехи, которая может быть наведена на BBSY* на объединительной плате и иметь длительность вплоть до половины ее задержки.

D.7 Выводы

В качестве основных направлений конструирования сложных систем могут быть даны два основных правила, которые позволят свести к минимуму возможность ошибки метастабильности:

(1) Уменьшить количество сопряжений синхронизации;

(2) Увеличить отношение t'/τ в формуле (D.1).

Чтобы следовать первому правилу, необходимо тщательно проанализировать архитектуру системы с тем, чтобы найти структуру с минимальным количеством асинхронных сопряжений (которые требуют ресинхронизацию).

После сведения их количества до минимума единственным способом уменьшения частоты появления ошибки метастабильности является увеличение отношения t'/τ в формуле (D.1). Этого можно достигнуть двумя способами:

- заданием для устройства синхронизации большего времени разрешающего его метастабильные состояния (а именно, использовать многокаскадное устройство синхронизации или маскировать выходные сигналы с помощью задержек);
- использованием технологии, обеспечивающей наименьшее τ.

В качестве пути избежания конфликтов доступа между процессором и его сопроцессором предлагается использование фазовой автоподстройки для синхронизации отдельных генераторов. Тот же подход может быть использован для принудительной фиксации фазовых соотношений между тактовыми сигналами с различными частотами, что также позволяет устранить ошибки метастабильности.

ПРИЛОЖЕНИЕ Е
(справочное)

Допустимые поднаборы возможностей

Е.1 Введение

В этом приложении обобщены поднаборы возможностей, которые поддерживают различные функциональные блоки. Кроме того, способность к взаимодействию поднаборов возможностей сведена в таблицы. Для упрощения составления таблиц допустимым поднаборам присвоены мнемонические обозначения, определенные в таблицах.

Е.2 Допустимые поднаборы функциональных блоков шины пересылки данных

Возможности шины пересылки данных определены для трех из четырех типов функциональных блоков этой шины: задатчика, исполнителя и адресного монитора. Эти возможности относятся к способности блоков формировать циклы, отвечать на циклы или отслеживать циклы, которые обеспечивают доступ к байтовым ячейкам в различных адресных диапазонах или выполняют специфические типы пересылок данных. Однако не все из описанных возможностей являются необязательными. Как указано в разделе 2 настоящего стандарта, одни блоки должны обеспечивать некоторые возможности, в то время как другие — нет. В этом разделе обобщены допустимые поднаборы возможностей шины пересылки данных.

Е.2.1 Допустимые поднаборы возможностей адресации

В разделе 2 настоящего стандарта определены возможности адресации с точки зрения формата адреса и способности функциональных блоков исполнять циклы Только Адрес (ADO).

Е.2.1.1 *Формат адреса*

Задатчик обозначается А32, А24 и/или А16 в зависимости от того, может ли он формировать циклы шины с адресами указанной ширины. Исполнитель подобным же образом обозначается в зависимости от того, может ли он отвечать на циклы этой ширины адреса.

Эти три независимых адресных пространства идентифицируются кодами модификатора адреса, которые возбуждаются на линиях модификатора адреса (АМ) в начале каждого цикла. Коды АМ, кроме того, делят адресное пространство на пространство Супервизор/Пользователь и Программа/Данные. Далее, функциональное различие между циклами блочной пересылки и одиночной пересылки также содержится в сигналах линий АМ. Группа кодов сигналов АМ также зарезервирована для определения пользователем.

Настоящий стандарт требует, чтобы все исполнители декодировали сигналы линий АМ и не отвечали на цикл, если они не находятся в запрашиваемом адресном пространстве или не могут выполнять запрашиваемую операцию. Кроме того, стандарт разрешает исполнителю отвечать на более чем одно адресное пространство. Эта возможность часто используется модулями памяти для ответа во всех пространствах: Супервизора, Пользователя, Программы и Данных. Стандарт требует, чтобы задатчики А32 также имели возможности А24 и А16, а задатчики А24 также имели возможность А16.

Е.2.1.2 *Циклы Только Адрес*

Задатчик обозначается ADO, если он может формировать цикл Только Адрес, т.е. цикл, в котором данные не передаются. В таком цикле задатчик предоставляет код модификатора адреса, адрес и строб адреса, но не утверждает ни одного строба данных. Он удерживает достоверной адресную информацию и сохраняет строб адреса истинным в течение установленного минимального времени, а затем отменяет их.

По отношению к исполнителю возможность ADO описывает его способность к правильной реакции, следующей за циклом, который был закончен без утверждения какого-либо строба данных. Если для задатчиков эта возможность необязательна, то все исполнители должны включать возможность ADO.

В таблице Е.1 определены мнемонические обозначния, которые описывают допустимые поднаборы возможностей адресации для задатчиков, исполнителей и адресных мониторов.

Таблица Е.1 — Допустимые поднаборы возможностей адресации

Мнемоническое обозначение функционального блока	Возможности адресации			
	A16	A24	A32	ADO
Поднаборы задатчика				
MA16	×			
MADO16	×			×
MA24	×	×		
MADO24	×	×		×
MA32	×	×	×	
MADO32	×	×	×	×

Окончание таблицы Е.1

Мнемоническое обозначение функционального блока	Возможности адресации			
	A16	A24	A32	ADO
Поднаборы исполнителя				
SADO16	×			×
SADO24	×	×		×
SADO32	×	×	×	×
Поднаборы адресного монитора				
LMA16	×			
LMA24	×	×		
LMA32	×	×	×	

В таблице Е.2 показано, как взаимодействуют допустимые поднаборы возможностей. Необходимо отметить, что конструкция модуля будет определять, как описанная возможность взаимодействия сконфигурирована: статически (с помощью перемычек) или динамически, через биты управляющего регистра.

Таблица Е.2 — Возможность взаимодействия допустимых поднаборов адресации

Мнемоническое обозначение функционального блока	Возможности взаимодействия с задатчиком					
	MA16	MA24	MA32	MADO16	MADO24	MADO32
Поднаборы исполнителя						
SADO16	Есть	Есть	Есть	Есть	Есть	Есть
SADO24	Есть	Есть	Есть	Есть	Есть	Есть
SADO32	Есть	Есть	Есть	Есть	Есть	Есть
Поднаборы адресного монитора						
LMA16	Есть	Есть	Есть	Есть	Есть	Есть
LMA24	Есть	Есть	Есть	Есть	Есть	Есть
LMA32	Есть	Есть	Есть	Есть	Есть	Есть

Е.2.2 Допустимые поднаборы возможностей пересылки данных

В разделе 2 настоящего стандарта определяются возможности пересылки данных с точки зрения формата данных, способности выполнять циклы невыровненных пересылок (UAT), циклы блочных пересылок (BLT) и циклы чтения—модификации—записи (RMW).

Е.2.2.1 *Формат данных*

Задатчик обозначается D32, D16 и/или D08(EO) в соответствии с тем, может ли он инициировать циклы для 32-битных данных, 16-битных данных и 8-битных данных.

Исполнитель обозначается D32, D16, D08(EO) и/или D08(O) в соответствии с тем, может ли он отвечать на циклы для 32-битных данных, 16-битных данных, 8-битных данных по четным и нечетным адресам или 8-битных данных только по нечетным адресам.

Стандарт требует, чтобы задатчики D32 и исполнители D32 также включали возможности D16 и D08(EO), задатчики и исполнители D16 также включали возможность D08(EO). Задатчик D08(EO) может получать доступ к байтам как по четным, так и по нечетным адресам, но одновременно только к одному байту.

Е.2.2.2 *Возможность невыровненной пересылки*

Обозначения форматов, описанные выше, применимы к «выровненным» пересылкам данных, т.е. к пересылкам, где 16-битные данные адресуются по четному адресу, а 32-битные данные — по адресу, кратному четырем. Задатчики D32 далее будут обозначаться как UAT, если они могут формировать циклы, которые включают:

(1) три младше—адресных байта 32-битных данных;

(2) три старше—адресных байта 32-битных данных;

(3) два средних байта 32-битных данных.

Различие между этими пересылками данных определяется задатчиком с помощью сигналов на линиях DS1*, DS0*, LWORD* и A01. В соответствии с требованиями стандарта возможность UAT является необязательной для задатчиков и обязательной для всех исполнителей D32. Кроме того, стандарт требует, чтобы все другие типы исполнителей полностью декодировали сигналы указанных четырех линий. Если исполнитель выбран (посредством сигналов линий AM и адреса) для пересылки данных формата, который он не может обработать, то он может ответить либо установкой сигнала BERR*, либо игнорировать пересылку, что приводит к тому же результату, как только шинный таймер выдаст сигнал BERR*.

Е.2.2.3 *Возможность блочной пересылки*

Задатчик обозначается BLT, если он может формировать цикл, который включает более одной пересылки данных по последовательно возрастающим адресам. Задатчик сигнализирует о блочной пересылке по линиям модификатора адреса и затем удерживает истинным адресный строб в течение нескольких пересылок данных. Исполнитель обозначается BLT, если он может отвечать на цикл блочной пересылки.

Модулей «Только BLT» не существует — требуется, чтобы все задатчики BLT, исполнители BLT и адресные мониторы BLT поддерживали также циклы одиночной пересылки.

Е.2.2.4 *Возможность Чтение—Модификация—Запись*

Задатчик обозначается RMW, если он может формировать неделимый цикл чтения—модификации—записи. Исполнитель обозначается RMW, если он может отвечать на этот цикл. Такой цикл состоит из одного цикла считывания, за которым следует цикл записи по одному и тому же адресу, причем адресный строб остается утвержденным во время обеих пересылок.

Неделимость более протяженной или обобщенной группы пересылок должна предусматриваться при проектировании задатчиков путем удержания управления шины пересылки данных (например удержанием в низком состоянии сигнала BBSY*) до тех пор, пока пересылка этой группы не будет закончена.

В таблице Е.3 определены мнемонические обозначения, которые описывают допустимые поднаборы возможностей пересылки данных задатчиками.

В таблице Е.4 определены мнемонические обозначения, которые описывают допустимые поднаборы возможностей пересылки данных исполнителями.

В таблице Е.5 определены мнемонические обозначения, которые описывают допустимые поднаборы возможностей пересылки данных адресными мониторами.

В таблице Е.6 показано, как взаимодействуют допустимые поднаборы возможностей пересылки данных.

Таблица Е.3 — Допустимые поднаборы возможностей пересылки данных задатчиком

Мнемоническое обозначение	Возможности пересылки данных					
	D08(EO)	D16	D32	UAT	BLT	RMW
MD8	×					
MBLT8	×				×	
MRMW8	×					×
MALL8	×				×	×
MD16	×	×				
MBLT16	×	×			×	
MRMW16	×	×				×
MALL16	×	×			×	×
MD32	×	×	×			
MBLT32	×	×	×		×	
MRMW32	×	×	×			×
MALL32	×	×	×		×	×
MD32+UAT	×	×	×	×		
MRMW32+UAT	×	×	×	×		×

Таблица Е.4 — Допустимые поднаборы возможностей пересылки данных исполнителем

Мнемоническое обозначение	Возможности пересылки данных						
	D08(O)	D08(EO)	D16	D32	UAT	BLT	RMW
SD8(O)	×						
SRMW8(O)	×						×
SD8		×					
SBLT8		×				×	
SRMW8		×					×
SALL8		×				×	×
SD16		×	×				
SBLT16		×	×			×	
SRMW16		×	×				×
SALL16		×	×			×	×
SD32		×	×	×	×		
SBLT32		×	×	×		×	
SRMW32		×	×	×	×		×
SALL32		×	×	×		×	×

Таблица Е.5 — Допустимые поднаборы возможностей обнаружения пересылки данных адресным монитором

Мнемоническое обозначение	Возможности обнаружения пересылки данных					
	D08(EO)	D16	D32	UAT	BLT	RMW
LMBLT32	×	×	×		×	
LMRMW32	×	×	×	×		×
LMALL32+UAT	×	×	×		×	×

Таблица Е.6 — Возможности взаимодействия поднаборов пересылки данных

Тип задатчика	Тип взаимодействующего исполнителя	Байтовые ячейки, доступные в цикле одиночной пересылки									
		0	1	2	3	0—1	1—2	2—3	0—2	1—3	0—3
MD8	SD8(O)		×		×						
или	SD8	×	×	×	×						
MBLT8	SD16	×	×	×	×						
	SD32	×	×	×	×						
MD16	SD8(O)		×		×						
или	SD8	×	×	×	×						
MBLT16	SD16	×	×	×	×	×		×			
	SD32	×	×	×	×	×		×			
MD32	SD8(O)		×		×						
или	SD8	×	×	×	×						
MBLT32	SD16	×	×	×	×	×		×			
	SD32	×	×	×	×	×		×			×
MD32+UAT	SD8(O)		×		×						
	SD8	×	×	×	×						
	SD16	×	×	×	×	×		×			
	SD32	×	×	×	×	×	×	×	×	×	×

Примечания

1 Только исполнители, имеющие возможность BLT, могут отвечать на циклы блочной пересылки. Однако, как показано в таблице Е.6, задатчики типа MBLTxx могут пересылать данные на исполнители типа SDxx, используя циклы одиночной пересылки. Подобным образом задатчики типа MDxx могут пересылать данные на исполнители типа SBLTxx, используя циклы одиночной пересылки.

2 На циклы Чтение—Модификация—Запись могут отвечать только исполнители, имеющие возможность RMW.

3 В таблице Е.6 поля, содержащие «x», представляют пересылки данных, на которые реагирует адресуемый исполнитель, устанавливая DTACK* в низкое состояние. Наоборот, пустые поля описывают пересылки данных, на которые адресуемому исполнителю запрещено отвечать установкой DTACK* в низкое состояние. Эти циклы заканчиваются либо адресацией исполнителя, либо установкой шинным таймером сигнала BERR* в низкое состояние.

Е.3 Взаимодействие функциональных блоков во время арбитража шины

Арбитраж шины определяется двумя функциональными блоками: арбитром и запросчиком. В разделе 3 настоящего стандарта описываются три возможности, связанные с этими блоками. Для арбитра определены три возможности: одноуровневый — SGL, приоритетный — PRI и круговой — RRS. Для запросчика определены три возможности: освобождающий шину по запросу — ROR, освобождающий по исполнении пересылки — RWD и паритетный (справедливый) — FAIR.

Е.3.1 Возможности арбитра

В разделе 3 настоящего стандарта представлена концепция центрального арбитра, который получает запрос на управление шиной от запросчиков и предоставляет им управление. Поскольку вопрос «справедливости» относительно приоритета подлежит обсуждению и весьма противоречив с точки зрения разнообразных нужд прикладного применения того или иного вида арбитража или какого-либо компромисса между ними, то алгоритм предоставления шины арбитром определен недостаточно точно. В стандарте дан пример строго приоритетного арбитра (PRI) и строго кругового (RRS). Однако общий протокол обмена сигналами для запроса, предоставления и взятия управления шиной является фиксированным. Несовместимость получается в результате сложного отношения к запросу шины обоих типов арбитров.

Третий тип арбитра SGL использует тот же протокол обмена сигналами, но только распознает запросы и предоставляет управление шиной по запросу уровня три.

Е.3.2 Возможности запросчика

Метод, который использует запросчик для освобождения шины пересылки данных, в стандарте определен недостаточно полно посредством двух примеров:

(1) запросчик, который отслеживает запросы шины от других запросчиков и освобождает шину только тогда, когда существует такой запрос (запросчик ROR);

(2) запросчик, который не делает указанную выше операцию, а просто освобождает шину, когда закончит ее использование (запросчик RWD).

Наиболее «справедливое» использование шины (чем может быть позволено арбитром чередованием четырех уровней запроса/предоставления) осуществляется с помощью запросчика типа FAIR. Он требует большей дисциплины в случае, когда запросчик может повторно запросить шину пересылки данных после ее освобождения.

В таблице Е.7 показано, как различные запросчики взаимодействуют с различными арбитрами.

Таблица Е.7 — Возможность взаимодействия арбитров и запросчиков. **FAIR SGL**

Тип арбитра	Возможность обрабатывать запросы от запросчиков		
	ROR	RWD	FAIR
SGL	Есть (см. примечание)	Есть (см. примечание)	Есть (см. примечание)
PRI	Есть	Есть	Есть
RRS	Есть	Есть	Есть

Примечание — Арбитр типа SGL отслеживает только линию BR3* и предоставляет шину, управляя только линией BG3IN*.

Е.4 Возможность взаимодействия функциональных блоков шины приоритетных прерываний

Шина приоритетных прерываний определяет два функциональных блока: прерыватель и обработчик прерываний. В разделе 4 настоящего стандарта описываются возможности, связанные с этими блоками. Возможности, определенные для обработчика прерываний, относятся к формату информации статуса/идентификации, который он запрашивает. Возможности, определенные для прерывателя, относятся к формату информации статуса/идентификации, который он обеспечивает, и к протоколу освобождения линии запроса прерывания.

Е.4.1 Формат информации статуса/идентификации

Различие между возможностями D32, D16 и D08 также отражено в формате информации статуса/идентификации, которая посылается в процессе цикла подтверждения прерывания. Обработчик прерываний имеет право запросить любой из трех форматов, а прерыватель может ответить любым форматом, но соотношение между ними не такое же, как в других циклах шины. Прерывателю разрешено отвечать только информацией статуса/идентификации, т.е. форматом, запрашиваемым обработчиком прерываний или меньшим. Если он отвечает меньшим форматом, то оконечные нагрузки шины гарантируют, что логическое состояние старших линий данных считывается обработчиком прерываний определенно, т.е. они находятся в высоком состоянии или в "1". Если прерыватель сталкивается с циклом подтверждения прерывания, который запрашивает вектор меньшей величины, чем та, которую он может обеспечить, то ему не разрешено отвечать на этот цикл, а требуется, чтобы он передал цикл дальше по последовательной цепочке подтверждения прерывания. Если далее по цепочке нет активных прерывателей, то в результате появится ошибка шины (BERR*), как только выйдет время шинного таймера.

В таблице Е.8 показано, как взаимодействуют прерыватели и обработчики прерываний с различными возможностями пересылки информации статуса/идентификации.

Таблица Е.8 — Возможность взаимодействия прерывателей и обработчиков прерываний

Тип прерывателя	Возможность взаимодействия с обработчиком прерываний		
	D08	D16	D32
Только D08	Есть	Есть	Есть
Только D16	Нет	Есть	Есть
D08 и D16	Есть	Есть	Есть
Только D32	Нет	Нет	Есть
D32 и D16 и D08	Есть	Есть	Есть

Е.4.2 Снятие запросов прерывания

Прерыватели обозначаются ROAK, если они снимают свой запрос прерывания во время цикла подтверждения прерывания, или RORA, если они снимают свой запрос, когда осуществляется доступ к внутримодульному регистру в процессе исполнения программы обслуживания прерывания. Единственным следствием, относящимся к возможности взаимодействия, является то, что программа обслуживания, которая обрабатывает прерыватель RORA, должна осуществить доступ к регистру и сбросить запрос, прежде чем повторно разрешить прерывание этого уровня.

ПРИЛОЖЕНИЕ F
(справочное)

Библиография

[1] МЭК 297-1 (1982) Размеры механических конструкций серии 482,6 мм (19 дюймов). Часть 1. Панель и стойка

[2] МЭК 297-3 (1984) Размеры механических конструкций серии 482,6 мм (19 дюймов). Часть 3. Блочные каркасы вдвижные частичные каркасы

[3] МЭК 603-2 (1980) Соединители для печатных плат на частоты до 3 МГц. Часть 2. Соединители, состоящие из двух частей, с общими монтажными характеристиками, предназначенные для применения с печатными платами, имеющими шаг координатной сетки 2,54 мм (0,1 дюйма)

[4] МЭК 822 (1988) Шина МЭК 822 VSB — Параллельная подсистемная шина МЭК 821 VME

[5] МЭК 823 (1990) Микропроцессорная системная шина (VMS bus) — Последовательная подсистемная шина для магистрали МЭК 821 (VME bus)

УДК 681.327.2:006.354 ОКС 35.020 П01 ОКСТУ 4012

Ключевые слова: магистраль микропроцессорных систем, обмен информацией, модуль, шина, цикл, диаграммы, протокол магистрали, линии шины, линии адресации

Редактор *В. П. Огурцов*
Технический редактор *Н. С. Гришанова*
Корректор *С. И. Фирсова*
Компьютерная верстка *Т. Ф. Кузнецовой*

Изд. лиц. № 02354 от 14.07.2000. Сдано в набор 15.06.2000. Подписано в печать 12.10.2000. Усл. печ. л. 24,65. Уч.-изд. л. 23,20. Тираж 180 экз. С 6035. Зак. 1644

ИПК Издательство стандартов,107076, Москва, Колодезный пер., 14.
Набрано в Калужской типографии стандартов на ПЭВМ.
Калужская типография стандартов, 248021, Калуга, ул. Московская, 256.
ПЛР № 040138