Table of Contents

1 Overview 1

1.1 Introduction 1
1.2 Real-time Application Systems 2
1.3 Real-time Executive 2
1.4 RTEMS Application Architecture 4
1.5 RTEMS Internal Architecture 4
1.6 User Customization and Extensibility 5
1.7 Portability 5
1.8 Memory Requirements 6
1.9 Audience 6
1.10 Conventions 7
1.11 Manual Organization 7

2 Key Concepts 11

2.1 Introduction 11
2.2 Objects 11
2.3 Communication and Synchronization 13
2.4 Time 13
2.5 Memory Management 14

3 Initialization Manager 17

3.1 Introduction 17
3.2 Background 17
3.2.1 Initialization Tasks 17
3.2.2 The System Initialization Task 18
3.2.3 The Idle Task 18
3.2.4 Initialization Manager Failure 18
3.3 Operations 19
3.3.1 Initializing RTEMS 19
3.3.2 Shutting Down RTEMS 20
3.4 Directives 20

3.4.1 INITIALIZE_EXECUTIVE - Initialize RTEMS 21
3.4.2 INITIALIZE_EXECUTIVE_EARLY - Initialize **RTEMS** and do NOT Start Multitasking 22
3.4.3 INITIALIZE_EXECUTIVE_LATE - Complete Initialization and Start Multitasking 23
3.4.4 SHUTDOWN_EXECUTIVE - Shutdown RTEMS 24

4 Task Manager 25

4.1 Introduction 25
4.2 Background 26
4.2.1 Task Definition 26
4.2.2 Task Control Block 26
4.2.3 Task States 27
4.2.4 Task Priority 27
4.2.5 Task Mode 28
4.2.6 Accessing Task Arguments 29
4.2.7 Floating Point Considerations 29
4.2.8 Building a Task's Attribute Set 30
4.2.9 Building a Mode and Mask 31
4.3 Operations 31
4.3.1 Creating Tasks 31
4.3.2 Obtaining Task IDs 32
4.3.3 Starting and Restarting Tasks 32
4.3.4 Suspending and Resuming Tasks 32
4.3.5 Delaying the Currently Executing Task 33
4.3.6 Changing Task Priority 33
4.3.7 Changing Task Mode 33
4.3.8 Notepad Locations 33
4.3.9 Task Deletion 34
4.4 Directives 34
4.4.1 TASK_CREATE - Create a task 35
4.4.2 TASK_IDENT - Get ID of a task 37
4.4.3 TASK_START - Start a task 38
4.4.4 TASK_RESTART - Restart a task 39
4.4.5 TASK_DELETE - Delete a task 40

4.4.6 TASK_SUSPEND - Suspend a task 41
4.4.7 TASK_RESUME - Resume a task 42
4.4.8 TASK_SET_PRIORITY - Set task priority 43
4.4.9 TASK_MODE - Change current task's mode 44
4.4.10 TASK_GET_NOTE - Get task notepad entry 46
4.4.11 TASK_SET_NOTE - Set task notepad entry 47
4.4.12 TASK_WAKE_AFTER - Wake up after interval 48
4.4.13 TASK_WAKE_WHEN - Wake up when specified 49

5 Interrupt Manager 51

5.1 Introduction 51
5.2 Background 51
5.2.1 Processing an Interrupt 51
5.2.2 RTEMS Interrupt Levels 53
5.2.3 Disabling of Interrupts by RTEMS 53
5.3 Operations 53
5.3.1 Establishing an ISR 53
5.3.2 Directives Allowed from an ISR 53
5.4 Directives 54
5.4.1 INTERRUPT_CATCH - Establish an ISR 55

6 Clock Manager 57

6.1 Introduction 57
6.2 Background 57
6.2.1 Required Support 57
6.2.2 Time and Date Data Structures 58
6.2.3 Timeslicing 58
6.2.4 Delays 59
6.2.5 Timeouts 59
6.3 Operations 59
6.3.1 Announcing a Tick 59
6.3.2 Setting the Time 59
6.3.3 Obtaining the Time 59
6.4 Directives 60
6.4.1 CLOCK_SET - Set system date and time 61

6.4.2 CLOCK_GET - Get system date and time information 62
6.4.3 CLOCK_TICK - Announce a clock tick 63

7 Timer Manager 65

7.1 Introduction 65
7.2 Background 65
7.2.1 Timers 65
7.2.2 Timer Service Routines 66
7.3 Operations 66
7.3.1 Creating a Timer 66
7.3.2 Obtaining Timer IDs 66
7.3.3 Initiating an Interval Timer 66
7.3.4 Initiating a Time of Day Timer 67
7.3.5 Canceling a Timer 67
7.3.6 Resetting a Timer 67
7.3.7 Deleting a Timer 67
7.4 Directives 67
7.4.1 TIMER_CREATE - Create a timer 68
7.4.2 TIMER_IDENT - Get ID of a timer 69
7.4.3 TIMER_CANCEL - Cancel a timer 70
7.4.4 TIMER_DELETE - Delete a timer 71
7.4.5 TIMER_FIRE_AFTER - Fire timer after interval 72
7.4.6 TIMER_FIRE_WHEN - Fire timer when specified 73
7.4.7 TIMER_RESET - Reset an interval timer 74

8 Semaphore Manager 75

8.1 Introduction 75
8.2 Background 75
8.2.1 Nested Resource Access 76
8.2.2 Priority Inversion 76
8.2.3 Priority Inheritance 77
8.2.4 Building an Attribute Set 77
8.2.5 Building an Option Set 78
8.3 Operations 78
8.3.1 Creating a Semaphore 78

8.3.2 Obtaining Semaphore IDs 79
8.3.3 Acquiring a Semaphore 79
8.3.4 Releasing a Semaphore 80
8.3.5 Deleting a Semaphore 80
8.4 Directives 80
8.4.1 SEMAPHORE_CREATE - Create a semaphore 81
8.4.2 SEMAPHORE_IDENT - Get ID of a semaphore 83
8.4.3 SEMAPHORE_DELETE - Delete a semaphore 84
8.4.4 SEMAPHORE_OBTAIN - Acquire a semaphore 85
8.4.5 SEMAPHORE_RELEASE - Release a semaphore 87

9 Message Manager 89

9.1 Introduction 89
9.2 Background 89
9.2.1 Messages 89
9.2.2 Message Queues 90
9.2.3 Building an Attribute Set 90
9.2.4 Building an Option Set 91
9.3 Operations 91
9.3.1 Creating a Message Queue 91
9.3.2 Obtaining Message Queue IDs 91
9.3.3 Receiving a Message 92
9.3.4 Sending a Message 92
9.3.5 Broadcasting a Message 92
9.3.6 Deleting a Message Queue 93
9.4 Directives 93
9.4.1 MESSAGE_QUEUE_CREATE - Create a queue 94
9.4.2 MESSAGE_QUEUE_IDENT - Get ID of a queue 96
9.4.3 MESSAGE_QUEUE_DELETE - Delete a queue 97
9.4.4 MESSAGE_QUEUE_SEND - Put message at rear of a queue 98
9.4.5 MESSAGE_QUEUE_URGENT - Put message at front of a queue 99
9.4.6 MESSAGE_QUEUE_BROADCAST - Broadcast N messages to a queue 100
9.4.7 MESSAGE_QUEUE_RECEIVE - Receive message from a queue 101
9.4.8 MESSAGE_QUEUE_FLUSH - Flush all messages on a queue 103

10 Event Manager 105

10.1 Introduction 105
10.2 Background 105
10.2.1 Event Sets 105
10.2.2 Building an Event Set or Condition 106
10.2.3 Building an Option Set 106
10.3 Operations 107
10.3.1 Sending an Event Set 107
10.3.2 Receiving an Event Set 107
10.3.3 Determining the Pending Event Set 108
10.3.4 Receiving all Pending Events 108
10.4 Directives 108
10.4.1 EVENT_SEND - Send event set to a task 109
10.4.2 EVENT_RECEIVE - Receive event condition 110

11 Signal Manager 113

11.1 Introduction 113
11.2 Background 113
11.2.1 Definitions 113
11.2.2 A Comparison of ASRs and ISRs 114
11.2.3 Building a Signal Set 114
11.2.4 Building an ASR's Mode 114
11.3 Operations 115
11.3.1 Establishing an ASR 115
11.3.2 Sending a Signal Set 116
11.3.3 Processing an ASR 116
11.4 Directives 117
11.4.1 SIGNAL_CATCH - Establish an ASR 118
11.4.2 SIGNAL_SEND - Send signal set to a task 119

12 Partition Manager 121

12.1 Introduction 121
12.2 Background 121
12.2.1 Definitions 121

12.2.2 Building an Attribute Set 122
12.3 Operations 122
12.3.1 Creating a Partition 122
12.3.2 Obtaining Partition IDs 122
12.3.3 Acquiring a Buffer 123
12.3.4 Releasing a Buffer 123
12.3.5 Deleting a Partition 123
12.4 Directives 123
12.4.1 PARTITION_CREATE - Create a partition 124
12.4.2 PARTITION_IDENT - Get ID of a partition 126
12.4.3 PARTITION_DELETE - Delete a partition 127
12.4.4 PARTITION_GET_BUFFER - Get buffer from a partition 128
12.4.5 PARTITION_RETURN_BUFFER - Return buffer to a partition 129

13 Region Manager 131

13.1 Introduction 131
13.2 Background 131
13.2.1 Definitions 131
13.2.2 Building an Attribute Set 132
13.2.3 Building an Option Set 132
13.3 Operations 133
13.3.1 Creating a Region 133
13.3.2 Obtaining Region IDs 133
13.3.3 Adding Memory to a Region 134
13.3.4 Acquiring a Segment 134
13.3.5 Releasing a Segment 134
13.3.6 Obtaining the Size of a Segment 134
13.3.7 Deleting a Region 135
13.4 Directives 135
13.4.1 REGION_CREATE - Create a region 136
13.4.2 REGION_IDENT - Get ID of a region 138
13.4.3 REGION_DELETE - Delete a region 139
13.4.4 REGION_EXTEND - Add memory to a region 140
13.4.5 REGION_GET_SEGMENT - Get segment from a region 141

13.4.6 REGION_RETURN_SEGMENT - Return segment to a region 144
13.4.7 REGION_GET_SEGMENT_SIZE - Obtain size of a segment 145

14 Dual-Ported Memory Manager 147

14.1 Introduction 147
14.2 Background 147
14.3 Operations 148
14.3.1 Creating a Port 148
14.3.2 Obtaining Port IDs 148
14.3.3 Converting an Address 148
14.3.4 Deleting a DPMA Port 148
14.4 Directives 149
14.4.1 PORT_CREATE - Create a port 150
14.4.2 PORT_IDENT - Get ID of a port 151
14.4.3 PORT_DELETE - Delete a port 152
14.4.4 PORT_EXTERNAL_TO_INTERNAL - Convert external to internal address 153
14.4.5 PORT_INTERNAL_TO_EXTERNAL - Convert internal to external address 154

15 I/O Manager 155

15.1 Introduction 155
15.2 Background 155
15.2.1 Device Driver Table 155
15.2.2 Major and Minor Device Numbers 156
15.2.3 Device Driver Environment 156
15.2.4 Device Driver Interface 156
15.2.5 Device Driver Initialization 157
15.3 Operations 157
15.4 Directives 157
15.4.1 IO_INITIALIZE - Initialize a device driver 158
15.4.2 IO_OPEN - Open a device 159
15.4.3 IO_CLOSE - Close a device 160
15.4.4 IO_READ - Read from a device 161
15.4.5 IO_WRITE - Write to a device 162

15.4.6 IO_CONTROL - Special device services 163

16 Fatal Error Manager 165

16.1 Introduction 165
16.2 Background 165
16.3 Operations 166
16.3.1 Announcing a Fatal Error 166
16.4 Directives 166
16.4.1 FATAL_ERROR_OCCURRED - Invoke the fatal error handler 167

17 Scheduling Concepts 169

17.1 Introduction 169
17.2 Scheduling Mechanisms 170
17.2.1 Task Priority 170
17.2.2 Preemption 170
17.2.3 Timeslicing 171
17.2.4 Manual Round-Robin 171
17.2.5 Dispatching Tasks 171
17.3 Task State Transitions 172

18 Rate Monotonic Manager 177

18.1 Introduction 177
18.2 Background 177
18.2.1 Definitions 178
18.2.2 Rate Monotonic Scheduling Algorithm 178
18.2.3 Schedulability Analysis 179
18.2.3.1 Assumptions 180
18.2.3.2 Processor Utilization Rule 180
18.2.3.3 Processor Utilization Rule Example 181
18.2.3.4 First Deadline Rule 181
18.2.3.5 First Deadline Rule Example 181
18.2.3.6 Relaxation of Assumptions 183
18.2.4 Further Reading 183
18.3 Operations 184
18.3.1 Creating a Rate Monotonic Period 184

18.3.2 Manipulating a Period 184
18.3.3 Obtaining a Period's Status 184
18.3.4 Canceling a Period 185
18.3.5 Deleting a Rate Monotonic Period 185
18.3.6 Examples 185
18.3.6.1 Simple Periodic Task 186
18.3.6.2 Task with Multiple Periods 186
18.4 Directives 188
18.4.1 RATE_MONOTONIC_CREATE - Create a rate monotonic period 189
18.4.2 RATE_MONOTONIC_IDENT - Get ID of a period 190
18.4.3 RATE_MONOTONIC_CANCEL - Cancel a period 191
18.4.4 RATE_MONOTONIC_DELETE - Delete a rate monotonic period 192
18.4.5 RATE_MONOTONIC_PERIOD - Conclude current/Start next period 193

19 Board Support Packages 195

19.1 Introduction 195
19.2 Reset and Initialization 195
19.2.1 Interrupt Stack Requirements 196
19.2.2 Processors with a Separate Interrupt Stack 197
19.2.3 Processors without a Separate Interrupt Stack 197
19.3 Device Drivers 197
19.3.1 Clock Tick Device Driver 198
19.4 User Extensions 198
19.5 Multiprocessor Communications Interface (MPCI) 199
19.5.1 Tightly-Coupled Systems 199
19.5.2 Loosely-Coupled Systems 199
19.5.3 Systems with Mixed Coupling 200
19.5.4 Heterogeneous Systems 200

20 User Extensions Manager 201

20.1 Introduction 201
20.2 Background 201
20.2.1 Extension Sets 202
20.2.2 TCB Extension Area 202
20.2.3 Extensions 203

20.2.3.1 TASK_CREATE Extension 203
20.2.3.2 TASK_START Extension 204
20.2.3.3 TASK_RESTART Extension 204
20.2.3.4 TASK_DELETE Extension 204
20.2.3.5 TASK_SWITCH Extension 205
20.2.3.6 TASK_BEGIN Extension 205
20.2.3.7 TASK_EXITTED Extension 206
20.2.3.8 FATAL Error Extension 206
20.2.4 Order of Invocation 206
20.3 Operations 207
20.3.1 Creating an Extension Set 207
20.3.2 Obtaining Extension Set IDs 207
20.3.3 Deleting an Extension Set 208
20.4 Directives 208
20.4.1 EXTENSION_CREATE - Create a extension set 209
20.4.2 EXTENSION_IDENT - Get ID of a extension set 210
20.4.3 EXTENSION_DELETE - Delete a extension set 211

21 Configuring a System 213

21.1 Configuration Table 213
21.2 CPU Dependent Information Table 214
21.3 Initialization Task Table 214
21.4 Driver Address Table 216
21.5 User Extensions Table 217
21.6 Multiprocessor Configuration Table 219
21.7 Multiprocessor Communications Interface Table 221
21.8 Determining Memory Requirements 221
21.9 Sizing the RTEMS RAM Workspace 222

22 Multiprocessing Manager 225

22.1 Introduction 225
22.2 Background 226
22.2.1 Nodes 226
22.2.2 Global Objects 226
22.2.3 Global Object Table 227

22.2.4 Remote Operations 227
22.2.5 Proxies 228
22.2.6 Multiprocessor Configuration Table 229
22.3 Multiprocessor Communications Interface Layer 229
22.3.1 INITIALIZATION 230
22.3.2 GET_PACKET 231
22.3.3 RETURN_PACKET 231
22.3.4 RECEIVE_PACKET 231
22.3.5 SEND_PACKET 232
22.3.6 Supporting Heterogeneous Environments 232
22.4 Operations 233
22.4.1 Announcing a Packet 233
22.5 Directives 234
22.5.1 MULTIPROCESSING_ANNOUNCE - Announce the arrival of a packet 235

23 Directive Status Codes 237

24 Example Application 239

25 Glossary 243