

# 10

## *MIDI Driver API*

### Driver Functions

#### Clock Functions

##### Functions to set clock behavior

kern\_return\_t     **MIDISetClockMode**(port\_t *driverPort*, port\_t *ownerPort*, short *synchUnit*, int *mode*)

kern\_return\_t     **MIDISetClockQuantum**(port\_t *driverPort*, port\_t *ownerPort*, int *interval*)

##### Functions to set and get clock time

kern\_return\_t     **MIDISetClockTime**(port\_t *driverPort*, port\_t *ownerPort*, int *time*)

kern\_return\_t     **MIDIGetClockTime**(port\_t *driverPort*, port\_t *ownerPort*, int \**time*)

kern\_return\_t     **MIDIGetMTCTime**(port\_t *driverPort*, port\_t *ownerPort*, short \**format*, short \**hours*, short \**minutes*, short \**seconds*, short \**frames*)

##### Functions to start and stop the clock

kern\_return\_t     **MIDIStartClock**(port\_t *driverPort*, port\_t *ownerPort*)

kern\_return\_t     **MIDIStopClock**(port\_t *driverPort*, port\_t *ownerPort*)

### Data Sending Function

##### Send data via the MIDI driver

kern\_return\_t     **MIDISendData**(port\_t *driverPort*, port\_t *ownerPort*, short *unit*, MIDIRawEvent \**data*, unsigned int *count*)

### Driver Ownership Functions

##### Acquire and release ownership of the MIDI driver

kern\_return\_t     **MIDIBecomeOwner**(port\_t *driverPort*, port\_t *ownerPort*)

kern\_return\_t     **MIDIReleaseOwnership**(port\_t *driverPort*, port\_t *ownerPort*)

# Ignore MIDI Codes Function

Request that the driver ignore certain MIDI codes

kern\_return\_t      **MIDISetSystemIgnores**(port\_t *driverPort*, port\_t *ownerPort*, short *unit*, unsigned  
int *ignoreBits*)

# Queue Management Functions

Query about and manage data flow in the queue

kern\_return\_t      **MIDIClearQueue**(port\_t *driverPort*, port\_t *ownerPort*, short *unit*)  
kern\_return\_t      **MIDIFlushQueue**(port\_t *device\_port*, port\_name\_t *ownerPort\_port*, short *unit*)  
kern\_return\_t      **MIDIGetAvailableQueueSize**(port\_t *driverPort*, port\_t *ownerPort*, short *unit*, int  
\**theSize*)

# Reply Handling Functions

Handle replies from the MIDI driver to a client

kern\_return\_t      **MIDIAwaitReply**(port\_t *reply\_port*, MIDIReplyFunctions \**handlers*, int *timeout*)  
kern\_return\_t      **MIDIHandleReply**(msg\_header\_t \**msg*, MIDIReplyFunctions \**handlers*)

# Request Functions

Request services from the MIDI driver

kern\_return\_t      **MIDIRequestData**(port\_t *driverPort*, port\_t *ownerPort*, short *unit*, port\_t  
   *replyPort*)  
kern\_return\_t      **MIDIRequestAlarm**(port\_t *driverPort*, port\_t *ownerPort*, port\_t *replyPort*, int  
   *time*)  
kern\_return\_t      **MIDIRequestExceptions**(port\_t *driverPort*, port\_t *ownerPort*, port\_t *replyPort*)  
kern\_return\_t      **MIDIRequestQueueNotification**(port\_t *driverPort*, port\_t *ownerPort*, short *unit*,  
   port\_t *replyPort*, int *size*)

# Serial Port Ownership Functions

Acquire and release ownership of the serial ports

kern\_return\_t **MIDIClaimUnit**(port\_t *driverPort*, port\_t *ownerPort*, short *unit*)  
kern\_return\_t **MIDIReleaseUnit**(port\_t *driverPort*, port\_t *ownerPort*, short *unit*)