

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      MIDIIHandleReply(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)
```