

InputDeviceD

COLLABORATORS

	<i>TITLE :</i> InputDeviceD		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		March 29, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	InputDeviceD	1
1.1	InputDeviceD.doc	1
1.2	input.device/InputDeviceAllgemeines()	1
1.3	input.device/INDCMD_AddHandler()	1
1.4	input.device/INDCMD_RemHandler()	2
1.5	input.device/INDCMD_WriteEvent()	3
1.6	input.device/INDCMD_SetThresh()	3
1.7	input.device/INDCMD_SetPeriod()	4
1.8	input.device/INDCMD_SetMPort()	4
1.9	input.device/INDCMD_SetMType()	5

Chapter 1

InputDeviceD

1.1 InputDeviceD.doc

input.device

```
INDCMD_AddHandler()   INDCMD_RemHandler()   INDCMD_SetMPort()
INDCMD_SetMType()     INDCMD_SetPeriod()   INDCMD_SetThresh()
INDCMD_WriteEvent()   InputDeviceAllgemeines()
```

1.2 input.device/InputDeviceAllgemeines()

STRUKTUREN

```
struct pOS_Device
struct pOS_InputDevice
struct pOS_InputEvent
```

INCLUDES

```
Device/Input.h
Device/IEvent.h
pExec/Types.h
pExec/Device.h
proto/pExec2.h
```

ENUMERATION

```
enum pOS_InputIOReqCommands
```

1.3 input.device/INDCMD_AddHandler()

PROTOTYP

```
INDCMD_AddHandler
```

FUNKTION

Hinzufügen eines Input-Handlers

PARAMETER pOS_IOStdReq

io_Message.mn_ReplyPort : Zeiger auf den Message-Port, der den IO-Request nach Ausführung des Kommandos empfangen soll.

io_Device,

io_Unit : Zeiger auf das Device und die Unit; durch pOS_OpenDevice() initialisiert.

io_Command : INDCMD_AddHandler

io_Data : Zeiger auf eine initialisierte Interrupt-Struktur, die den Input-Handler beschreibt.
 pOS_InputEvent* (*is_Code) (_R_A0 const pOS_Interrupt ← *,
 _R_A1 pOS_InputEvent*);

BEMERKUNG

Kann mit dem Kommando INDCMD_RemHandler wieder entfernt werden.

SIEHE AUCH

INDCMD_RemHandler

AMIGA FUNKTION

IND_ADDHANDLER

1.4 input.device/INDCMD_RemHandler()

PROTOTYP

INDCMD_RemHandler

FUNKTION

Entfernen eines Input-Handlers

PARAMETER pOS_IOStdReq

io_Message.mn_ReplyPort : Zeiger auf den Message-Port, der den IO-Request nach Ausführung des Kommandos empfangen soll.

io_Device,

io_Unit : Zeiger auf das Device und die Unit; durch pOS_OpenDevice() initialisiert.

io_Command : INDCMD_RemHandler

io_Data : Zeiger auf eine Interrupt-Struktur des Handlers

SIEHE AUCH

INDCMD_AddHandler

AMIGA FUNKTION

IND_REMHANDLER

1.5 input.device/INDCMD_WriteEvent()

PROTOTYP

INDCMD_WriteEvent

FUNKTION

Erzeugen eines Input-Events

PARAMETER pos_IOSdReq

io_Message.mn_ReplyPort : Zeiger auf den Message-Port, der den IO-Request nach Ausführung des Kommandos empfangen soll.

io_Device,

io_Unit : Zeiger auf das Device und die Unit; durch pos_OpenDevice() initialisiert.

io_Command : INDCMD_WriteEvent

io_Flags : IOREQB_Quick, falls Quick-IO erwünscht ist

io_Data : Zeiger auf die initialisierte InputEvent-Struktur, die im System verschickt werden soll.
ie_NextEvent zeigt auf den nächsten IE oder NULL
ie_TimeStamp wird auf die aktuelle Systemzeit gesetzt

SIEHE AUCH

%

AMIGA FUNKTION

IND_WRITEEVENT

1.6 input.device/INDCMD_SetThresh()

wird noch erweitert

PROTOTYP

INDCMD_SetThresh

FUNKTION

Setzen der Verzögerungsdauer zwischen ersten Tastendruck und den folgenden, ab dem ein Repeat der Taste verwendet wird.

PARAMETER pos_IOSdReq

io_Message.mn_ReplyPort : Zeiger auf den Message-Port, der den IO-Request nach Ausführung des Kommandos empfangen soll.

io_Device,

io_Unit : Zeiger auf das Device und die Unit; durch pos_OpenDevice() initialisiert.

io_Command : INDCMD_WriteEvent

io_Flags : IOREQB_Quick, falls Quick-IO erwünscht ist

io_Time.tv_Secs,

io_Time.tv_Micro : Sekunden- und Mikrosekundenanteil der Verzögerungsdauer

SIEHE AUCH
INDCMD_SetPeriod

AMIGA FUNKTION
CMD_SETTHRESH

1.7 input.device/INDCMD_SetPeriod()

PROTOTYP
INDCMD_SetPeriod

FUNKTION
Setzen den Zeitintervall für Tastaturwiederholungen

PARAMETER pOS_TimerIO
io_Message.mn_ReplyPort : Zeiger auf den Message-Port, der den
IO-Request nach Ausführung des Kommandos
empfangen soll.

io_Device,
io_Unit : Zeiger auf das Device und die Unit; durch
pOS_OpenDevice() initialisiert.
io_Command : INDCMD_SetPeriod

io_Flags : IOREQB_Quick, falls Quick-IO erwünscht ist
io_Time.tv_Secs,
io_Time.tv_Micro : neuer Sekunden- und Mikrosekundenanteil des
Zeitintervalls

SIEHE AUCH
INDCMD_SetThresh

AMIGA FUNKTION
CMD_SETPERIOD

1.8 input.device/INDCMD_SetMPort()

PROTOTYP
INDCMD_SetMPort

FUNKTION
Setzt den aktuellen Mausport

PARAMETER pOS_IOStdReq
io_Message.mn_ReplyPort : Zeiger auf den Message-Port, der den
IO-Request nach Ausführung des Kommandos
empfangen soll.

io_Device,
io_Unit : Zeiger auf das Device und die Unit; durch
pOS_OpenDevice() initialisiert.

```
io_Command      : INDCMD_SetMPort

io_Flags        : IOREQB_Quick, falls Quick-IO erwünscht ist
io_Length       : 1
io_Data         : Zeiger auf eine Byte-Variable, die entweder
                  0 oder 1 enthält und damit anzeigt, an welchem
                  Gameport die Maus angeschlossen ist.
```

SIEHE AUCH

INDCMD_SetMType, INDCMD_SetMTrig

AMIGA FUNKTION

CMD_SETMPORT

1.9 input.device/INDCMD_SetMType()

wird noch erweitert

PROTOTYP

INDCMD_SetMType

FUNKTION

Setzt den aktuellen Maus-Controller-Type

PARAMETER pOS_IOStdReq

io_Message.mn_ReplyPort : Zeiger auf den Message-Port, der den
IO-Request nach Ausführung des Kommandos
empfangen soll.

io_Device,

io_Unit : Zeiger auf das Device und die Unit; durch
pOS_OpenDevice() initialisiert.

io_Command : INDCMD_SetMType

io_Flags : IOREQB_Quick, falls Quick-IO erwünscht ist

io_Length : 1

io_Data : Adresse der Byte-Variablen, die den Gameport-
Controller-Typ enthält.

SIEHE AUCH

INDCMD_SetMPort, INDCMD_SetMTrig

AMIGA FUNKTION

CMD_SETMTYPE

END
