CC Pilot™ XS

Backlight interface description

CC SYSTEMS

Table of Contents

Introduction

Purpose

This document describes the software interface to the backlight device.

References

-

History

Rev	Date	Author	Remarks
1.0	2005-09-31	Göran Nordin	First version.
1.1	2008-04-08	Fredrik Lans	Revision

Backlight device interface

IOCTLs are used to communicate with the backlight device. The name of the backlight device is “BKL1” in Unicode character set.

Summary of IOCTLs

IOCTL_BACKLIGHT_GET_RESOLUTION	Returns the number of steps of the backlight element.
IOCTL_BACKLIGHT_GET	Returns the current level of the backlight element in number of steps.
IOCTL_BACKLIGHT_SET	Sets the level of the backlight element. The level is the number of steps.

IOCTL_BACKLIGHT_GET_RESOLUTION

Description

Returns the number of steps of the backlight element. The windows DeviceIoControl function passes the IOCTL to the backlight device.

Include files

#include "Backlight.h"

Parameters supplied to DeviceIoControl

hDevice	Handle to the backlight device. To obtain a device handle, call the CreateFile function with *lpFileName* parameter set to TEXT("BKL1:").
dwIoControlCode	Set to IOCTL_BACKLIGHT_GET_RESOLUTION.
lpInBuffer	Set to NULL.
nInBufferSize	Set to zero.
lpOutBuffer	Pointer to a unsigned long where the number of steps of the backlight element will be returned.
nOutBufferSize	Set to sizeof(unsigned long).
lpBytesReturned	Pointer to a variable that receives the size, in bytes, of the data stored into the buffer pointed to by *lpOutBuffer*.
lpOverlapped	Ignored; set to NULL

Return value

TRUE if operation succeeded otherwise FALSE. If operation failed then "GetLastError" can be used to get more information of the error.

Restrictions

-

Example

```
HANDLE hDevice;

if ((hDevice =
  CreateFile(
  TEXT("BKL1:"),
  GENERIC_READ | GENERIC_WRITE,
  FILE_SHARE_READ | FILE_SHARE_WRITE,
  NULL,
  OPEN_EXISTING,
  0,
  NULL)) != INVALID_HANDLE_VALUE)
{
  unsigned long bytesReturned, resolution;

  if (DeviceIoControl(
    hDevice,
    IOCTL_BACKLIGHT_GET_RESOLUTION,
    NULL,
    0,
    &resolution,
    sizeof(resolution),
    &bytesReturned,
    NULL))
  {
    printf("Resolution in steps is %lu\n", resolution);
  }
  else
  {
    printf(
      "!!ERROR, %lu when calling \"DeviceIoControl\"\n",
      GetLastError());
  }
}
else
{
  printf(
    "!!ERROR, %lu when calling \"CreateFile\"\n",
    GetLastError());
}
```

IOCTL_BACKLIGHT_GET

Description

Returns the current level of the backlight element in number of steps. The windows DeviceIoControl function passes the IOCTL to the backlight device.

Include files

#include "Backlight.h"

Parameters supplied to DeviceIoControl

hDevice	Handle to the backlight device. To obtain a device handle, call the CreateFile function with *lpFileName* parameter set to TEXT("BKL1:").

dwIoControlCode	Set to IOCTL_BACKLIGHT_GET.
lpInBuffer	Set to NULL.
nInBufferSize	Set to zero.
lpOutBuffer	Pointer to a unsigned long where the current level of the backlight element in number of steps will be returned.
nOutBufferSize	Set to sizeof(unsigned long).
lpBytesReturned	Pointer to a variable that receives the size, in bytes, of the data stored into the buffer pointed to by *lpOutBuffer*.
lpOverlapped	Ignored; set to NULL

Return value

TRUE if operation succeeded otherwise FALSE. If operation failed then "GetLastError" can be used to get more information of the error.

Restrictions

-

Example

```
HANDLE hDevice;

if ((hDevice =
  CreateFile(
  TEXT("BKL1:"),
  GENERIC_READ | GENERIC_WRITE,
  FILE_SHARE_READ | FILE_SHARE_WRITE,
  NULL,
  OPEN_EXISTING,
  0,
  NULL)) != INVALID_HANDLE_VALUE)
{
  unsigned long bytesReturned, level;

  if (DeviceIoControl(
    hDevice,
    IOCTL_BACKLIGHT_GET,
    NULL,
    0,&level,
    sizeof(level),
    &bytesReturned,
    NULL))
  {
    printf("Level in steps is %lu\n", level);
  }
  else
  {
    printf(
      "!!ERROR, %lu when calling \"DeviceIoControl\"\n",
      GetLastError());
  }
}
else
{
  printf(
    "!!ERROR, %lu when calling \"CreateFile\"\n",
    GetLastError());
}
```

IOCTL_BACKLIGHT_SET

Description

Sets the level of the backlight element. The level is the number of steps. The windows DeviceIoControl function passes the IOCTL to the backlight device.

Include files

#include "Backlight.h"

Parameters supplied to DeviceIoControl

hDevice	Handle to the backlight device. To obtain a device handle, call the CreateFile function with *lpFileName* parameter set to TEXT("BKL1:").
dwIoControlCode	Set to IOCTL_BACKLIGHT_SET.
lpInBuffer	Pointer to a unsigned long where the level in number of steps is stored.
nInBufferSize	Set to sizeof(unsigned long).
lpOutBuffer	Set to NULL.
nInBufferSize	Set to zero.
lpBytesReturned	Pointer to a variable that receives the size, in bytes, of the data stored into the buffer pointed to by *lpOutBuffer*.
lpOverlapped	Ignored; set to NULL

Return value

TRUE if operation succeeded otherwise FALSE. If operation failed then "GetLastError" can be used to get more information of the error.

Restrictions

-

Example

```
  HANDLE hDevice;

  if ((hDevice =
    CreateFile(
    TEXT("BKL1:"),
    GENERIC_READ | GENERIC_WRITE,
    FILE_SHARE_READ | FILE_SHARE_WRITE,
    NULL,
    OPEN_EXISTING,0,
    NULL )) != INVALID_HANDLE_VALUE )

  {
    unsigned long bytesReturned, level = 10;
    if ( !DeviceIoControl(
      hDevice,
      IOCTL_BACKLIGHT_SET,
      &level,
      sizeof(level),
      NULL,
      0,
      &bytesReturned,
      NULL))
    {
      printf(
        "!!ERROR, %lu when calling \"DeviceIoControl\"\n",
        GetLastError());
    }
  }
  else
  {
    printf(
      "!!ERROR, %lu when calling \"CreateFile\"\n",
      GetLastError());
  }
```