

CA3000 Open API

Continental **Access**
A Napco Security Group Company



CardAccess® 3000 



The following pages contain detailed programming instructions. This API is for experienced programmers only.

The `_Initca3000` function initializes the `ca3000` API for use by the host application.

`int _Initca3000();`

Parameters:.

Return Values:

If the function succeeds, the return value is 0.

If the function fails, the return value is a negative number.

See Also:

`_DeInitca3000`, `_GetConnectStr`, `_GetErrorString`

The `_DeInitca3000` function de-initializes the `ca3000` API. This function must be called before unloading the API dll to avoid memory leaks.

`int _Initca3000();`

Parameters:

N/A

Return Values:

If the function succeeds, the return value is 0.

If the function fails, the return value is a negative number.

See Also:

`_Initca3000`

The `_GetErrorString` function returns the actual error text if a function returns an error code.

```
int _GetErrorString(Char* ES;  
                    DWORD ESLen);
```

Parameters:

ES

Returns the last error string. *Required field.*

ESLen

Length of the allocated space for *ES*. *Required field.*

Return Values:

N/A

The `_GetConnectStr` return the connection string that will be used by the API.

```
int _GetConnectStr (char* CS,  
                   DWORD CSLen,  
                   BOOL Net);
```

Parameters:

CS

**Returns the connection string that will be used by the API.
*Required field.***

CSLen

Length of the allocated space for *CS*. *Required field.*

Net

Specifies if the calling application is a .Net application or not.

Return Values:

If the function succeeds, the return value is 0.

If the function fails, the return value is a negative number.

See Also:

_Initca3000, _GetErrorString

The `_BadgeOperation` function adds/deletes/modifies Badges.

```
Int _BadgeOperation(WORD Facility,  
                    __int64 Badge,  
                    char* FirstName,  
                    char* LastName,  
                    char* MI,  
                    WORD AGNo1,  
                    WORD AGNo2,  
                    WORD AGNo3,  
                    WORD AGNo4,  
                    WORD AGNo5,  
                    WORD AGNo6,  
                    WORD Operation);
```

Parameters:

Facility

Facility number. Value between 0 – 10. *Required field.*

Badge

Badge number. Value any positive integer. *Required field.*

FirstName

First name.

LastName

Last name. *Required field.*

MI

Middle initial.

AGNo1 - AGNo6

Access groups 1 through 6.

Operation

Intended operation. *Required field.*

0 = Add

1 = Delete

2 = Modify

Return Values:

If the function succeeds, the return value is 1.

If the function fails, the return value is a negative number.

See Also:

[_DeInitca3000](#), [_GetErrorString](#)

The `_InlistQuery` function returns the list of badges that are IN based on the Panel and Reader.

```
int _InlistQuery (WORD Panel,  
                 WORD Reader,  
                 __int64* BadgeList,  
                 WORD* ResultSize);
```

Parameters:

Panel

Panel number. *Required field.*

Reader

Reader number. *Required field.*

BadgeList

Pointer to an array. *Required field.*

ResultSize

Length of the allocated space for *BadgeList*. If the size is smaller than the list, the function will return the actual size of the result set. *Required field.*

Return Values:

If the function succeeds, the return value is 0.

If the function fails, the return value is a negative number.

See Also:

`_DeInitca3000`, `_GetErrorString`

The `_DoorOperation` function allows manual door control.

**`Int _DoorOperation(WORD Panel,`
**`WORD Door,`
**`WORD Time,`
`WORD Operation);`******

Parameters:

Panel

Panel number. *Required field.*

Door

Reader number (1- 8). *Required field.*

Time

Duration (seconds: 0-59; minutes: 60 – 120 -> 1-60). *Required field.*

Operation

1 = Lock door

2 = Pulsed unlock

3 = Hold unlocked

Return Values:

If the function succeeds, the return value is 0.

If the function fails, the return value is a negative number.

See Also:

The `_RelayOperation` triggers a relay or a group of relays.

```
Int _RelayOperation(WORD Panel,  
                     WORD Operation,  
                     WORD* Relays,  
                     WORD RelayCount);
```

Parameters:

Panel

Panel number. *Required field.*

Operation Required field.

1 = Pulse On

2 = Hold On

3 = Turn Off

Relays

List of relays that will be acted upon. *Required field.*

RelayCount

Number of relays in *Relays*. *Required field.*

Return Values:

If the function succeeds, the return value is 0.

If the function fails, the return value is a negative number.

See Also:

The `_ReaderOperation` function enables or disables a reader.

```
int _ReaderOperation(WORD Panel,  
                        WORD Reader,  
                        BOOLEAN Enabled);
```

Parameters:

Panel

Panel number. *Required field.*

Reader

Reader number. *Required field.*

Enable

Value *True* or *False* to enable/disable reader. *Required field.*

Return Values:

If the function succeeds, the return value is 0.

If the function fails, the return value is a negative number.

See Also:

The `_TAQuery` function returns the Time and Attendance record for a badge.

```
int _TAQuery(WORD Facility,
             __int64 Badge,
             SYSTEMTIME StartTime,
             SYSTEMTIME EndTime,
             SYSTEMTIME* TimeList,
             WORD* ResultSize);
```

Parameters:

Facility

Panel number. *Required field.*

Badge

Badge number. *Required field.*

StartTime

Start of time range to report on. *Required field.*

EndTime

End of time range to report on. *Required field.*

TimeList

List of time pairs for T&A activity with in *StartTime* and *EndTime* range. *Required field.*

ResultSize

Site of allocated space for *TimeList*. If the size is smaller than the actual result set then the actual result size is reported. *Required field.*

Return Values:

If the function succeeds, the return value is 0.

If the function fails, the return value is a negative number.

See Also: