Version 1.0
August 2001
Telephone Directory : 5092

LONMARK® Functional Profile: Telephone Directory

NOTE: When time permits, this Profile will be modified to have the proper format.

Overview

This document describes the functional profile of the Telephone Directory Object.

The Telephone Directory object is used to store and retreive arrays of ASCII strings that are characterized as telephone numbers (including characters used for control) used in dialing a data modem. The ASCII arrays are configured using data file transfer or configuration network variables and are retrieved in realtime operation using an index value passed via a network variable.

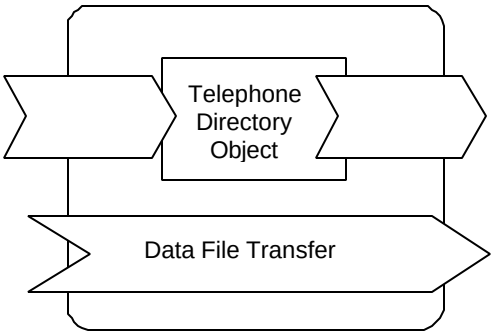
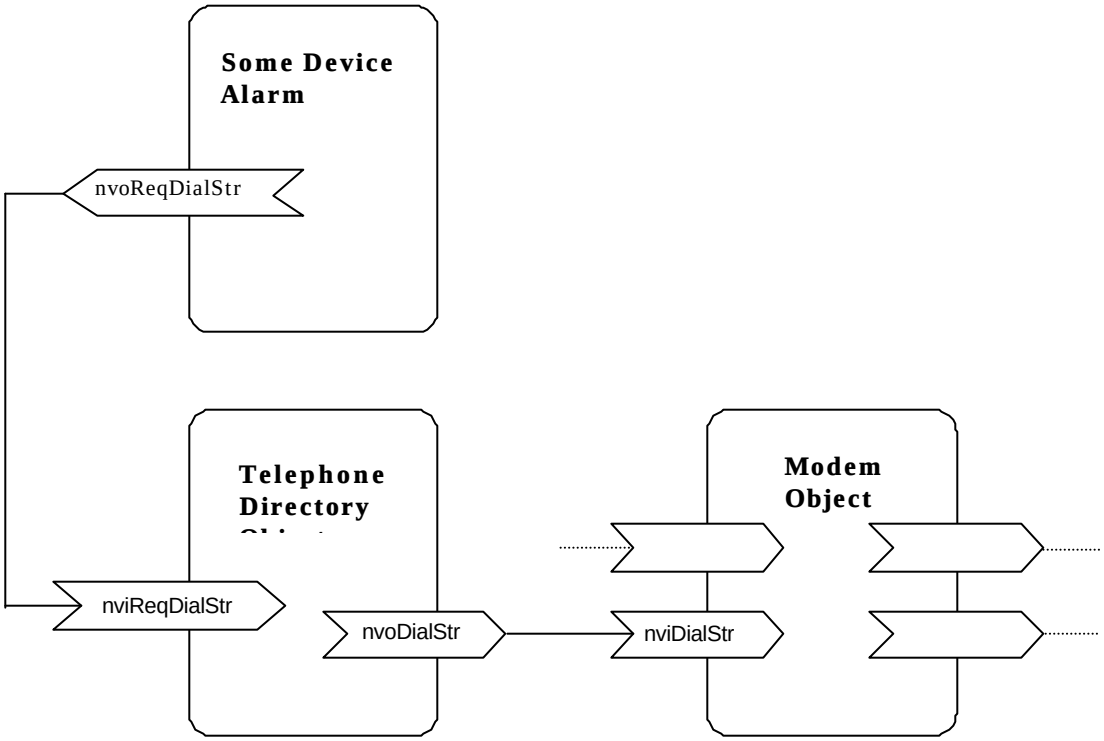


Figure 1 Telephone Directory

Example Usage

The output of the Telephone Directory object (which emits telephone numbers) is typically bound to a Modem object. The input to the Telephone Directory object is typically driven by an object requiring the services of the Modem object to accomplish some connectivity task, e.g. notification of an alarm to a remote site. Usage of the Telephone Directory object allows the alerting object to initiate dialing of the Modem object to a specific telephone number by simply emitting an array index from 0 to 255.



Object Details

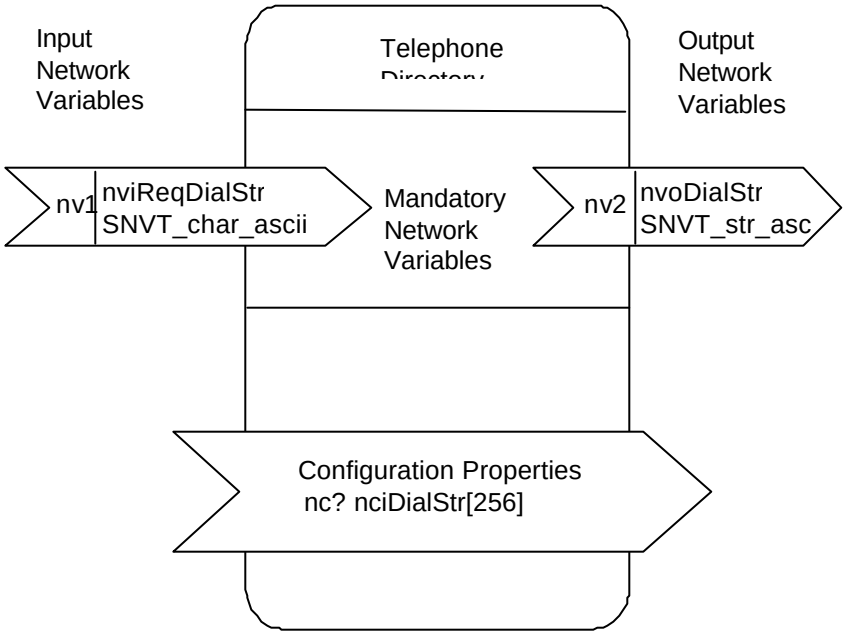


Figure 2 Object Details

Table 1 SNVT Details

NV # (M/O)*	Name	In/ Out	SNVT Type (SNVT Index)	Class	Description
1 (M)	nviReqDialStr	In	SNVT_char_ascii	I/O	An index value in to this NV will cause the phone number from that array index to be sent to nvoDialStr.
2(M)	nvoDialStr	Out	SNVT_str_asc	I/O	Phone number output per nviReqDialStr input
3(O)	nciDialStr[256]	In	SNVT_str_asc	config	An array of phone number entries. Dimensioned up to a maximum of 256 NULL terminated strings by 31 characters (this length includes the NULL).

* M = mandatory, O = optional

Table 2 SCPT Details

SCPT Index (M/O)*	Name	Association **	Description
(O)	SCPTdialStr	NV0	Phone numbers to be configured into the Telephone Directory

* M = mandatory, O = optional

** NV index = 0 means configuration property applies to the object as a whole.

Request Dial String Input

```
network input SNVT_char_ascii nviReqDialStr;
```

This input network variable requests the output of a telephone number, from the requested index.

Valid Range

0-255 or Maximum Entry index. A Telephone Directory object implementation may support anywhere from 1 to 256 entries maximum.

Default Value

N/A

Dial String Output

```
network output SNVT_str_asc nvoDialStr;
```

This output provides the phone number string from the Telephone Directory entry corresponding to the index number received on nviReqDialStr.

Valid Range

A NULL terminated ASCII string up to 31 bytes in length (including the NULL).

All ASCII characters are valid.

When Transmitted

Transmitted upon receipt of a valid nviReqDialStr input, unless requested entry index is not supported or entry is blank (first character of entry is a NULL), in which case this variable is ***not*** transmitted.

Receipt of another nviReqDialStr input causes this output to be transmitted, containing the phone number string from the Telephone Directory entry corresponding to the index received on nviReqDialStr, even if the index received is the same index as received on the previous update to nviReqDialStr.

Update Rate

Updated only on an update to nviReqDialStr input.

Default Service Type

The default service type is acknowledged.

Configuration Properties

Dial String

```
network input config SNVT_str_asc nciDialStr[];
```

This input configuration network property provides the array of data storage. This array consists of entries containing 31 bytes of storage and may be 1 to 256 elements in depth.

Valid Range

A NULL terminated ASCII string up to 31 bytes in length.

MANDATORY support for the following ASCII characters is required:

Digits 0-9 plus * and # – telephone dialing digits.

OPTIONAL support for the following ASCII characters is specified (if the corresponding function is to be implemented, it MUST be done so using the following specified character):

Letter T (upper or lower case, first character in string only) – Tone dial.

Letter P (upper or lower case, first character in string only) – Pulse dial.

Comma (,) – pause dialing for 2 seconds for each comma encountered.

Letter W (upper or lower case) – Pause dialing until a dial tone is detected.

Exclamation Point (!) – Hookflash, go onhook for 0.5 seconds, then back offhook.

Support of any other characters/functions in the dial string are considered optional. If an optional character is encountered in the string which is not supported by a particular implementation (including but not limited to punctuation such as space, dash, left and right parentheses), that character must be ignored, rather than causing an error condition.

Default Value

The NULL character.

SCPT Reference

```
SCPTdialStr (?)
```

Two optional methods are specified to configure the telephone numbers into the Telephone Directory. One of these methods ***must*** be implemented. The configuration network variable method requires a separate configuration network variable array element for each entry in the Telephone Directory. If this quantity of network variables cannot be implemented in the device containing the Telephone Directory, then file transfer must be used.

Data Transfer

```
Const char SCPT_FILE2[]=      // Configuration parameter template
                              // File type 2

"1.1;"                        // Current version number for file type 2
"1,4-8,3\x80,7,31,16;"        // SCPTDialString
;


far eprom struct
{                             // Configuration parameter value file
char PhoneNumber[31];
} SCPT_FILE1[N_ENTRIES] = {
NULL,                         // Phone number entry 1
.
.
.
NULL                          // Phone number entry N_ENTRIES-1
};

define N_FILES 2              // Directory size

typedef struct {
unsigned long     size;       // File size
unsigned long     type;       // File type
void *            pData       // Pointer to data
} FileDescriptor;

const struct  {               // File directory table
int   Version;
int   NumFiles;               // Number of files
FileDescriptor    Files[N_FILES];
} FileDirectory = {
0x20,                         // Version number
{ sizeof(SCPT_FILE2), 2, &SCPT_FILE2},    //Type 2 template file
{ sizeof(SCPT_FILE1), 1, &SCPT_FILE1},    //Type 1 value file
}

FILE values 1 1
"0123456789abcdE"
"0123456789abcdE"
\x00
```

Power-up State

The only network output variable (nvoDialStr) is NOT written by default at power-up.

Boundary and Error Conditions

If a value of nviReqDialStr is received by a Telephone Directory object that is either greater than the largest index value supported (Maximum Entry Index), or refers to a supported entry that is blank (the first character of the entry is a NULL), the nvoDialStr output must not be written.

Additional Considerations

A Telephone Directory object implementation may support anywhere from 1 to 256 entries maximum (Maximum Entry Index). The first entry supported must be referenced by index 0, the next by index 1, and so forth.

The storage used for the Telephone Directory entries must be non-volatile.

Browsing the phonebook should be done strictly via the Configuration Property mechanism, not via the network variable interface nviReqDialStr and nvoDialStr, as simultaneous input requests from multiple devices could cause race conditions and reception of incorrect data. The nviReqDialStr input network variable should be bound to only one output network variable to eliminate the race condition possibility.