TELETASK

PROJECT, COMPACT & MICROS
DOMOTICS SYSTEEM

'WINDOWS'
SOFTWARE LIBRARY

TDS_OCX_EN.doc - 04/07/03

TELETASK DOMOTIC SYSTEMS WINDOWS OCX V2.73

Platform: To be used on all IBM compatible personal computers with Microsoft WINDOWS 95/98/ME or WINDOWS NT/2000 operating system.

It is the purpose to generate a start-up environment for software developers, who are interested in generating their own PC-based control environment for the TELETASK domotics systems. In this way the developer can create his own user interface and connected solutions and services using a standard personal computer.

This situation should stimulate creative minds in working with their own control environment as an exclusive add-on to a TELETASK system. In this way the developer can generate solutions which are not available at TELETASK but only on his own to be commercialized platform.

Applications: creating mouse- or touch screen controlled PC-screens with ground floors of the particular environment where the TELETASK system is installed. This can be in a very exclusive private villa, but it even can be a central controller which is connected to several houses in a larger project (as there are houses for elderly people). The application list is very large and limited to your imagination and applications.

1. LICENCE AGREEMENT.

De TELETASK ‘TDS Windows Library’- license is free of charge and is only to be used by developers who are the owner of at least one TELETASK central unit.

Due to the unlimited freedom of the developer, it is impossible for TELETASK to give free support or to carry any responsibility on the use and the results of the use of this libraries in any way. Claims in any way for any reason are not allowed.
By using this libraries, the developer and the user agree with this statement.

2. BUILD-UP OF THE EXAMPLE ROUTINES.

The routines are based on the following parts:

- The Windows Custom Control TELETASK.OCX wherein the control functions for the TELETASK systems are included.
  TDS272.OCX is a 32-bit ActiveX Control Interface.
- The example program TeletaskExample272.EXE is a simple example of a graphical user interface which uses the available control functions on a VISUAL BASIC 6.0 based compiler.

Remark: These parts are usable on personal computers which run on a Microsoft WINDOWS 95/98/NE or WINDOWS NT/2000 operating system

2.1 Functions

The functions in the OCX use a parameter *"Fnc"* and can have following values

FNC_RELAY = 1	(control or get the status of a relay)
FNC_DIMMER = 2	(control or get the status of a dimmer)
FNC_MOTOR = 6	(control or get the status of a Motor: On/Off)
FNC_MTRUPDOWN = 55	(control or get the status of a Motor: Op/Down)
FNC_LOCMOOD = 8	(control or get the status of a Local Mood)
FNC_TIMEDMOOD = 9	(control or get the status of a Timed Local Mood)
FNC_GENMOOD = 10	(control or get the status of a General Mood)
FNC_FLAG = 15	(control or get the status of a Flag)
FNC_PROCES = 3	(control or get the status of a Process function)
FNC_REGIME = 14	(control or get the status of a Regime function)
FNC_SERVICE = 53	(control or get the status of a Service function)
FNC_MESSAGE = 54	(control or get the status of a Messages or Alarms)
FNC_COND = 60	(get the status of a Condition)
FNC_TPKEY = 52	(simulate a key press on an interface)
FNC_GETSENSTARGET = 21	(get the status of a Sensor setting)
FNC_GETSENSVAL = 20	(get the status of a Sensor value)
FNC_SETTEMPUP = 21	(control the Sensor Function: Up)
FNC_SETTEMPDOWN = 22	(control the Sensor Function: Down)
FNC_SETTEMPFROST = 24	(control the Sensor Function: Go to frost)
FNC_SETTEMPNIGHT = 25	(control the Sensor Function: Go to night)
FNC_SETTEMPDAY = 26	(control the Sensor Function: Go to day)
FNC_SETTEMPSETNIGHT = 27	(control the Sensor Function: Set night preset)
FNC_SETTEMPSETDAY = 29	(control the Sensor Function: Set day preset)
FNC_AUDIOSTATE = 31	(get status of an Audio Zone)
FNC_AUDIOUP = 32	(Control Audio Function: Vol. Up)
FNC_AUDIODOWN = 33	(Control Audio Function: Vol. Down)
FNC_AUDIOON = 36	(Control Audio Function: On)
FNC_AUDIOOFF = 37	(Control Audio Function: Off)
FNC_AUDIOFM = 38	(Control Audio Function: Tuner)
FNC_AUDIOFM2 = 47	(Control Audio Function: Tuner Extra)
FNC_AUDIOCD = 39	(Control Audio Function: CD)
FNC_AUDIOCD2 = 48	(Control Audio Function: CD Extra)
FNC_AUDIOTAPE = 40	(Control Audio Function: Tape)
FNC_AUDIOTAPE2 = 49	(Control Audio Function: Tape Extra)
FNC_AUDIOVIDEO = 41	(Control Audio Function: Video)
FNC_AUDIOVIDEO2 = 43	(Control Audio Function: Video Extra)

FNC_AUDIOAUX = 42	(Control Audio Function: Aux)
FNC_AUDIOAUX2 = 35	(Control Audio Function: Aux Extra)
FNC_AUDIOKEY = 44	(Control Audio Function: Key)

2.2 Methods

InitComm (Port As Integer, Baudrate As Integer) As Integer

The communication between the PC and the TELETASK Domotics System (=TDS) have to start with this function, before any other function can be used.

Input:
- Port (= 1,2,3 of 4) is the used communication port.
- Baud (= 2400, 4800, 9600 of 19200) is the desired baud rate, which has to be the same as the baud rate on the central unit (default 19200).

Output:
- Return value:
   0: Communication successfully opened
  -1: Problem on opening
  -2: Invalid Com port
  -3: Invalid baud rate

EndComm () As Integer

The communication between the PC and the TDS must be closed before shutting down your application. If you do not provide this function in your program, the COM port will not be available for other applications after closing your application.

GetVersion () As String

This functions returns a string with the name and version of the OCX

FunctionLog (Fnc As Integer, State As Integer) As Integer

This function will open/close a channel for the function
Example: If you call this function with the parameter Fnc=FNC_RELAY and State=1, then will afterwards all changes on relays occur as ‘event’! In case you set State=0 no more events will occur from relays.

Input:
- Fnc (FNC_DIMMER, FNC_RELAY, FNC_MOTOR, FNC_MTRUPDOWN, FNC_LOCMOOD, FNC_TIMEDMOOD, FNC_GENMOOD, FNC_FLAG , FNC_PROCES , FNC_REGIME , FNC_SERVICE, FNC_MESSAGE, FNC_COND , FNC_GETSENSVAL, FNC_GETSENSTARGET, FNC_AUDIOSTATE ,FNC_TPKEY)
- State (0 of 1)

Output:
- Return value:
   0 = Message successfully transmitted
  -1 = Communication not opened
  -2 = No Answer

Remark: Use only the indicated functions. After a PROSOFT download all channels will be closed

FunctionSet (Fnc As Integer, Outp As Integer, State As Integer) As Integer

With this function you set the indicated function to a state that correspond to the value in State
Example: If you set Fnc=FNC_FLAG, Outp=3, State=255 the central unit will set the flag with number 3 ON.

Input:
- fnc (FNC_RELAY, FNC_DIMMER, FNC_MOTOR, FNC_MTRUPDOWN, FNC_LOCMOOD, FNC_TIMEDMOOD, FNC_GENMOOD, FNC_FLAG, FNC_PROCES, FNC_REGIME, FNC_SERVICE, FNC_MESSAGE, FNC_SETTEMPUP, FNC_SETTEMPDOWN, FNC_SETTEMPVRIES, FNC_SETTEMPNIGHT, FNC_SETTEMPDAY, FNC_SETTEMPSETNIGHT, FNC_SETTEMPSETDAY, FNC_AUDIOUP, FNC_AUDIODOWN, FNC_AUDIOON, FNC_AUDIOOFF, FNC_AUDIOFM, FNC_AUDIOFM2, FNC_AUDIOCD, FNC_AUDIOCD2, FNC_AUDIOTAPE, FNC_AUDIOTAPE2, FNC_AUDIOVIDEO, FNC_AUDIOVIDEO2, FNC_AUDIOAUX, FNC_AUDIOAUX2, FNC_AUDIOKEY)
- Outp (dependent on the function)

- State (dependent on the function)

Output:
- Return value: 0 = Message successfully transmitted
  -1 = Communication not opened
  -2 = No Answer

Remark 1:
for Fnc = FNC_RELAY, FNC_DIMMER, temperature functions & audio functions
-> Outp = 1 to Maximum
for Fnc = FNC_REGIME (this Event occurs on a change of regime)
-> Outp= 0 Automatic workday / weekend-day
1 Workday
2 Weekend-day
3 Simulation
4 Manual
for Fnc = FNC_TPKEY
-> Outp = Touch panel number you want to simulate
All other Fnc
-> Outp = 0 to Maximum -1

Remark 2:
for Fnc=FNC_DIMMER & FNC_MOTOR
-> State = 0 to 255 (always use the result from function ConvPercToDimVal, may never be 3!!!) *for Fnc=FNC_SETTEMPSETDAY & SETTEMPNIGHT*
-> State = 0 to 255 (to calculate with functions ConvTempToSensVal, ConvHumToSensVal or ConvLuxToSensVal)
for Fnc = FNC_AUDIOKEY
-> State = 1 to 9 (preset)
for Fnc = FNC_TPKEY
-> State bit 3-0 = Key number (0 to 7)
-> State bit 7-8 = 00 Normal Short Press, 01 Key Depressed, 10 Key Released
for all other Fnc
-> State = 0 or 255 (or 1) = OFF or ON

FunctionGet(fnc As Integer, Outp As Integer) As Integer

This function ask for te state of the indicated fnc
Example. If you call this function with Fnc=GET_SENSORVAL, Outp=3 the central unit will transmit the value of sensor 3, and a event will ocour. (This will alwayw happen regardless if the log-channel is open or not!)

Input:
- fnc (FNC_RELAY, FNC_DIMMER, FNC_MOTOR, FNC_MTRUPDOWN, FNC_LOCMOOD, FNC_TIMEDMOOD, FNC_GENMOOD, FNC_FLAG, FNC_PROCES, FNC_REGIME, FNC_SERVICE, FNC_MESSAGE, FNC_COND, FNC_GETSENSVAL, FNC_GETSENSTARGET, FNC_SETTEMPVRIES, FNC_SETTEMPNIGHT, FNC_SETTEMPDAY, FNC_SETTEMPSETNIGHT, FNC_SETTEMPSETDAY, FNC_AUDIOSTATE)
- Outp (dependent on the function, look remarks at FunctionSet)

Output:
- Return value: 0 = Message successfully transmitted
  -1 = Communication not opened
  -2 = No Answer

ConvDimValToPerc(Value As Integer) As Double
This function converts an integer value (derived from the event ‘Report’) to a Percent

ConvPercToDimVal(Value As Integer) As Integer
This function converts a percentage to an integer required for the Method ‘FunctionSet’

ConvSensValToTemp(Value As Integer) As Double
This function converts an integer value (derived from the event ‘Report’ with met Fnc= FNC_GETSENSVAL, FNC_GETSENSTARGET, FNC_SETTEMPSETNIGHT, FNC_SETTEMPSETDAY) to degrees Celsius

ConvSensValToHum(Value As Integer) As Double

This function converts an integer value (derived from the event 'Report' with met Fnc= FNC_GETSENSVAL, FNC_GETSENSTARGET, FNC_SETTEMPSETNIGHT, FNC_SETTEMPSETDAY) to % humidity.

ConvSensValToLux(Value As Integer) As Double

This function converts an integer value (derived from the event 'Report' with Fnc= FNC_GETSENSVAL, FNC_GETSENSTARGET, FNC_SETTEMPSETNIGHT, FNC_SETTEMPSETDAY) to a lux value

ConvTempToSensVal(Value As Double) As Integer

This function converts degrees Celsius to an integer required for the Method 'FunctionSet' with Fnc= FNC_SETTEMPSETNIGHT, FNC_SETTEMPSETDAY)

ConvHumToSensVal(Value As Double) As Integer

This function converts % humidity to an integer required for the Method 'FunctionSet' with Fnc= FNC_SETTEMPSETNIGHT, FNC_SETTEMPSETDAY)

ConvLuxToSensVal(Value As Double) As Integer

This function converts a lux value to an integer required for the Method 'FunctionSet' with Fnc= FNC_SETTEMPSETNIGHT, FNC_SETTEMPSETDAY)

2.3 Events

ErrMsg(Err As Integer, Msg As String)

This event occurs when there is something wrong:

Err	Msg
0	All OK
2	Invalid Command
255	Communication problem

Report(Fnc As Integer, Outp As Integer, State As Integer)

This event occurs when something changes in the central unit. (on condition that the log channel is open)

Param:
- fnc (FNC_RELAY, FNC_DIMMER, FNC_MOTOR, FNC_MTRUPDOWN, FNC_LOCMOOD, FNC_TIMEDMOOD, FNC_GENMOOD, FNC_FLAG, FNC_PROCES, FNC_REGIME, FNC_SERVICE, FNC_MESSAGE, FNC_COND, FNC_GETSENSVAL, FNC_GETSENSTARGET, FNC_SETTEMPVRIES, FNC_SETTEMPNIGHT, FNC_SETTEMPDAY, FNC_SETTEMPSETNIGHT, FNC_SETTEMPSETDAY, FNC_AUDIOSTATE)
- Outp (dependent on the function, see the remarks in FunctionSet)
- State(dependent on the function)

For Fnc=FNC_DIMMER & FNC_MOTOR

State = 0 to 255 (you can afterwards calculate the % value with ConvDimValToPerc)

For Fnc=FNC_SETTEMPSETDAY & SETTEMPNIGHT

State = 0 tot 255 (you can calculate the real value u with ConvSensValToTemp,
ConvSensValToHum of ConvSensValToLux)

For Fnc=FNC_AUDIOSTATE

State =		
	36 ON	37 OFF
	38 FM	39 CD
	40 TAPE	41 VIDEO
	42 AUX	

For Fnc=FNC_TPKEY

State bit 5-0 = Key number
State bit 6 = 1 Key down, 0 Key Up
State bit 7 = 1 Remote Audio Key, 0 normal

For All other

State = 0 OR 255 (or. 1) = OFF or ON

3. Communication

This section describes what you have to know if you write your own interface but you won't use the OCX. If you use the OCX you don't have to read this section.

The TDS CENTRAL is connected to a CCT (custom command terminal), which has to be operated by the user and can send commands to the TDS CENTRAL. The communication between the CCT and the TDS CENTRAL is a standard RS-232 interface. The cable will be connected to the TDS CENTRAL by means of a 9 pin female SubD type connector. The interface will use only 3 of the 9 pins:

Pin #	Description
3	Transmit data
2	Receive data
5	Signal Ground

This interface protocol will have 1 stop bit and no parity by default. Possible baud rates are: 2400, 4800, 9600 & 19200 (Default)

All commands and messages in both directions will use the same frame format:

STX (02h)	Length	Command Number	Parameter 1		Parameter n	ChkSm

The length does not include the ChkSm-byte.
The ChkSm is calculated on Command Number + Command Parameters + Length + STX
After the ChkSm the central unit send an acknowledge byte 0A (hex). If no acknowledge byte is send the command is not handled.

3.1 Command details

Function Set

- Description: This command allows the CCT to set individual functions. See "methods" for detailed descriptions
- Command number: 01h
- Length: 6
- Direction: From TDS to CCT.
- Parameter 1 = Fnc
- Parameter 2 = Outp
- Parameter 3 = State

Function Get

- Description: When the TDS receives this command it reports the level of the specified load. See methods for detailed descriptions
- Command number: 02h
- Length: 5
- Direction: From CCT to TDS
- Parameter 1 = fnc
- Parameter 2 = Outp

Function Log On/Off

- Description: When the TDS receives this command it (de-)activates it's channel for reporting the function!
- Command number: 03h
- Length: 5

- Direction: From CCT to TDS
- Parameter 1 = fnc
- Parameter 2 = state

Event Report

- Description: TDS sends this command it the level of the specified load has changed
- Command number: 08h
- Length: 5
- Direction: From TDS to CCT
- Parameter 1 = Fnc
- Parameter 2 = Outp
- Parameter 3 = State

3.2 Sensor values

Temperatures

To change from byte to °C: (byte/2)-40
To change from °C to byte: (temp+40)x2

Humidity

Byte = % humidity

Lux values

To change from byte to lux = $(10^{(byte / 40)}) - 1$
To change from lux to byte = Log10(lux + 1) * 40

4. The included demo program.

To be able to test the demo program, your pc has to be connected with the TDS by means of a standard serial port. After starting up the program you must click on the '*initcomm 1*'or *'initcomm 2'* button depending on the port you are using (default baud rate is 19200).
The program sources are included on your CD-ROM and have to be opened in Visual Basic version 6.0 or later.

On screen you will find several buttons:

"Log" : For every function you can activate the log with this function
"Get" : A request to get the actual state of the function, a bit later you will receive a event
"Set" : Activate the function

All the received messages are put in a list.

REMARK: There are some actions(commands) which are not able to be controlled in an OFF status or which have no result. Please keep in mind these (logic) features of the system.

If you need support, it will be on the level of your Visual Basic compiler. If you need support concerning the TELETASK softtools, this will be charged at normal cost. For more information please contact your local distributor.

CHANGES since Version 2.71

- The functionality *FNC_TPKEY* is changed in the FunctionSet command
- There is an acknowledge byte (0A) send after a command is send to the central unit.