InitParams

InitParams structure

```
typedef struct _InitParams
 {
   unsigned int VerHi; // version of DLL interface - High DWORD
   unsigned int VerLo; // version of DLL interface - Low DWORD
   // callback procedures
   void (_stdcall *GetPoint)(HOBJ StId, char *Adr, PointParPtr *PtParPtrPtr);
   void (_stdcall *GetFastPoint)(HOBJ StId, unsigned int FastAdr, PointParPtr *PtParPtrPtr);
   void (_stdcall *GetStation)(HOBJ LnID, char *Adr, StationParPtr *StParPtrPtr);
   void (_stdcall *PointNewValue)(HOBJ PtID, KomUniVal *Val, unsigned int Stat);
   void (_stdcall *PointOldValue)(HOBJ PtID, KomUniVal *Val, unsigned int Stat, int TransId);
   KomUniVal(_stdcall *PointLastValidValue)(HOBJ PtID);
   void (_stdcall *SetStationStatus)(HOBJ StID, TStVal Stat);
   unsigned int (_stdcall *WriteToLine)(HOBJ StID, void *Data, unsigned int Len);
   unsigned int (_stdcall *ReadFromLine)(HOBJ StID, void *Data, unsigned int MaxData);
   void (_stdcall *SetReceiveState)(HOBJ StID);
   void (_stdcall *SetTransmitState)(HOBJ StID);
   void (_stdcall *ChangeLineMode)(HOBJ LnID, unsigned int LineMode);
   void (_stdcall *TraceBufferHex)(HOBJ LnID, void *Data, unsigned int Len, TColor *Color);
   void (_stdcall *TraceBufferHexFile)(HOBJ LnID, char *FileName, unsigned int MaxFileSize, void *Data,
unsigned int Len);
   void (_stdcall *TraceBufferASCII)(HOBJ LnID, void *Data, unsigned int Len, TColor *Color);
   void (_stdcall *TraceBufferASCIIFile)(HOBJ LnID, char *FileName, unsigned int MaxFileSize, void *Data,
unsigned int Len);
   void (_stdcall *TraceComText)(HOBJ LnID, TColor *Color, char *Str);
   void (_stdcall *TraceComTextFile)(HOBJ LnID, char *FileName, unsigned int MaxFileSize, char *Str);
   void (_stdcall *TraceComError)(HOBJ LnID, char *Str);
   void (_stdcall *TraceComErrorFile)(HOBJ LnID, char *FileName, unsigned int MaxFileSize, char *Str);
   void (_stdcall *PointsApply)(HOBJ StID, void *AplPar);
   void (_stdcall *StationsApply)(HOBJ LnID, unsigned int Prot, void *AplPar);
   void (_stdcall *MessageBox)(char *Title, char *Message);
   unsigned int (_stdcall * CreateThread)(void *Proc, void *Par);
   void (_stdcall *DeleteThread)(unsigned int ThrID);
   void * (_stdcall *GetMem)(unsigned int Len);
   void (_stdcall *FreeMem)(void *Adr);
   D2Time (_stdcall *ActTime)(void);
   D2Time (_stdcall *TimeFromItems)(unsigned int Year, unsigned int Month, unsigned int Day, unsigned int Hour,
unsigned int Min, unsigned int Sec, unsigned int Msec);
   Bool32 (_stdcall *TimeToItems)(D2Time Tm, unsigned int *Year, unsigned int *Month, unsigned int *Day,
unsigned int *Hour, unsigned int *Min, unsigned int *Sec, unsigned int *Msec);
   LinListH (_stdcall *CreateList)(void);
   void (_stdcall *DeleteList)(LinListH List);
   void (_stdcall *InsertEl)(LinListH List, void *DataPtr, void *SortProc);
   void (_stdcall *InsertElPrio)(LinListH List, void *DataPtr, unsigned int HighPriority);
   void * (_stdcall *FindEl)(LinListH List, void *DataPtr, void *FindProc);
   void (_stdcall *DeleteEl)(LinListH List, void *DataPtr, void *FindProc);
   void * (_stdcall *FindFirstElAndDelete)(LinListH List);
   void * (_stdcall *FindFirstEl)(LinListH List);
   void * (_stdcall *FindNextEl)(LinListH List);
   void * (_stdcall *FindPreviousEl)(LinListH List);
   unsigned int (_stdcall *GetNumber)(LinListH List);
   void (_stdcall *StPerf_CommActivity)(HOBJ StID);
   void (_stdcall *StPerf_TimeSynchro)(HOBJ StID);
   void (_stdcall *StPerf_IncFramesIn)(HOBJ StID);
   void (_stdcall *StPerf_IncFramesOut)(HOBJ StID);
   void (_stdcall *StPerf_IncRetryErr)(HOBJ StID);
   void (_stdcall *LnPerf_IncTxBytes)(HOBJ LnID, unsigned int Increment);
   void (_stdcall *LnPerf_IncRxBytes)(HOBJ LnID, unsigned int Increment);
   void (_stdcall *LnPerf_IncRetryErr)(HOBJ LnID);
   void (_stdcall *LnPerf_IncFramesIn)(HOBJ LnID);
   void (_stdcall *LnPerf_IncFramesOut)(HOBJ LnID);
   unsigned int Prot; // Komunikacny protokol
   void (_stdcall *GetRdStateAndInstance)(Bool32 * RdHot, Bool32 *Instance);
   void (_stdcall *GetOemProtName)(unsigned int spProtocol, char *ProtName);
 } InitParams;
```

The structure *InitParams* is transferred into the protocol DLL by calling the function Init (required). The structure contains a number of the protocol interface and addresses of call-back functions.

Related pages:

D2000 KomAPI - interface structures