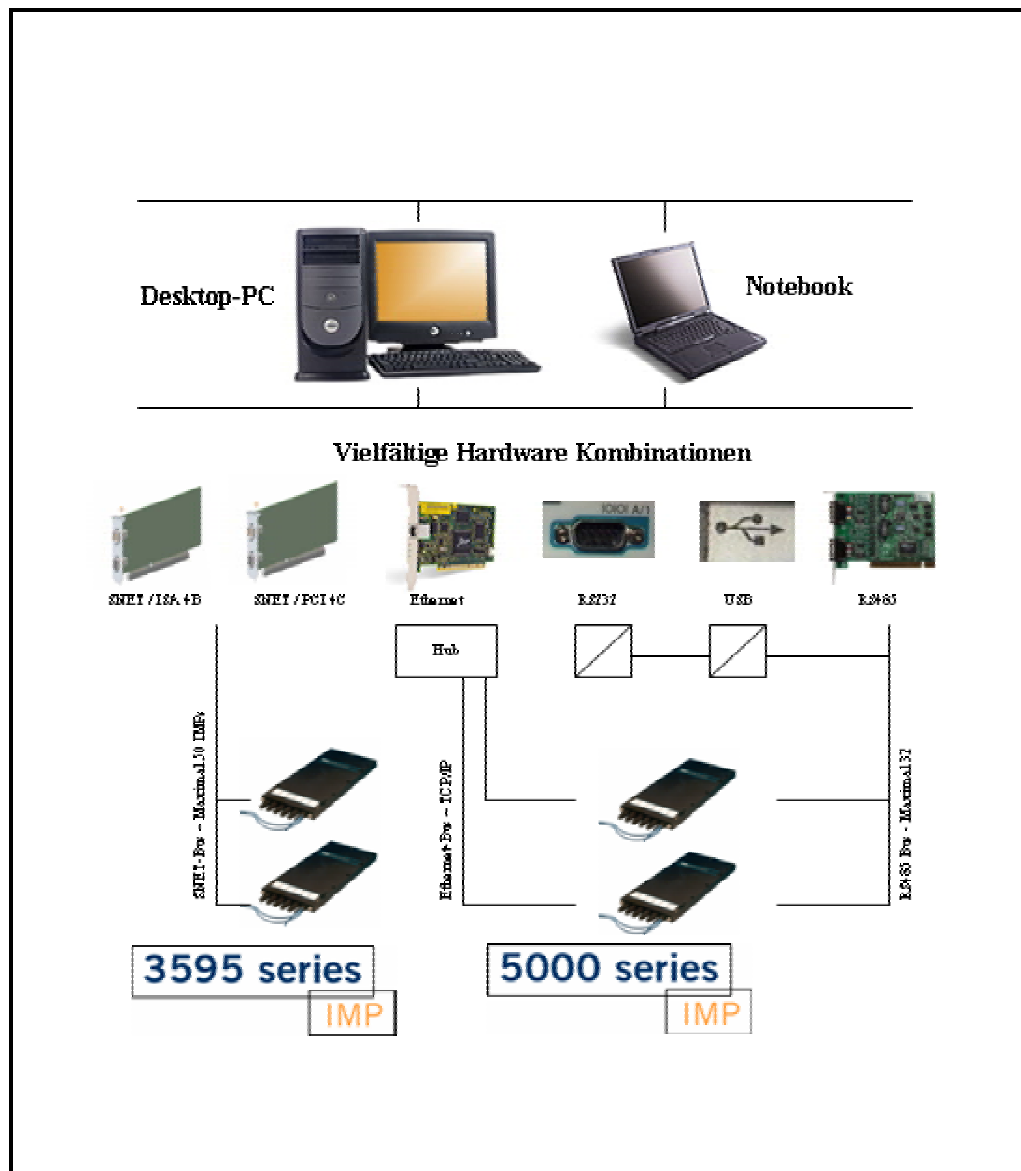


IMP Universal Treiber

(C) PMS / MESTEC 2004



Bedienungshandbuch zu IMP Universal Treiber

Letzte Änderung am 23.Juli 2004

Autor – Dipl. Phys. Christian Frank

© MESTEC 2004

Inhalt

Übersicht	1
Produkt.....	1
Funktion.....	1
Anwendung.....	1
Fehlerbehandlung	2
Registrierung	2
IMP Modul Übersicht.....	2
IMP Serie 3595.....	2
IMP Serie 5000.....	3
IMP 5000 Gateway Modus.....	3
Installation	4
Lieferumfang	4
System Voraussetzung	5
Hardware Voraussetzung.....	5
Software Voraussetzung	5
Installieren	6
Funktions-Referenz	7
ImpLab_GetVersion	7
ImpLab_Reset.....	7
ImpLab_SetMeasSrate.....	7
ImpLab_SetIfcSnet.....	8
ImpLab_GetIfcSnet	8
ImpLab_SetIfcSeriell.....	9
ImpLab_SetIfcEthernet.....	9
ImpLab_SetImp	10
ImpLab_GetImp	11
ImpLab_SetImpChan.....	11
ImpLab_Download.....	12
ImpLab_StartMeasure	13
ImpLab_StopMeasure.....	13
ImpLab_ReadDataCopy	13
ImpLab_ReadDataChannels	14
ImpLab_GetChanIndex	14
ImpLab_Disconnect.....	15
ImpLab_IsMeasureRunning	15
ImpLab_GetRegistrationCode	15
ImpLab_SetRegistrationKey	16
ImpLab_GetRegistrationState	16
ImpLab_GetErrorReport.....	16
ImpLab_GetErrorReportSize.....	17
ImpLab_ResetErrorReport.....	17
ImpLab_GetErrorString.....	17
ImpLab_SetDebugMode.....	18
Anwendung	19

Funktionsreihenfolge	19
C++	19
Beispiel "CallImpLab.EXE"	20
LabView	21
Beispiel "IMPDEMO.LLB"	21
Anhang	22
Fehlercodes	22
TCP/IP Dll	23
Solartron DLL	23
Glossar	24
Index	25

Übersicht

Produkt

Der IMP Universal Treiber ist eine 32-Bit DLL für Windows zur Konfiguration und Datenerfassung der SOLARTRON IMP Module.

Untertstützt werden die Serien 3595 und 5000.

Der Treiber arbeitet mit SNET, Ethernet und Seriellem Interface.

Funktion

Der Treiber bietet einheitliche Funktionsaufrufe für die Parametrierung der Messung mit den verschiedenen IMP Typen der Serien 3595 und 5000.

Dabei muss sich der Anwender nicht um die unterschiedlichen Interface Typen kümmern. Die Funktionsaufrufe sind einfach in die Anwendung einzubinden.

Das Auslesen der Daten erfordert ebenfalls nur geringen Programmieraufwand.

Mit den Set.. Funktionen werden für die Interface-Ports, die IMPs und die IMP Kanäle Datenstrukturen angelegt.

Erst mit der Download Funktion werden die Solartron bzw. TCPIP DLL geladen und die Hardwareverbindung getestet. Ebenso werden die Parameter an die IMP-Module gesendet.

Falls der Download fehlerfrei war, wird der Start durchgeführt.

Ab Start können die Meßdaten mit der Funktion ReadDataCopy Impweise abgeholt werden.

Anwendung

Der Treiber kann in beliebige 32-Bit Windows Programme eingebunden werden.

Beispiele sind LabView, LabWindows, MFC C++, C#, Delphi usw.

Für LabView und C++ sind ausführliche Beispiele vorhanden.

Fehlerbehandlung

Zusätzlich zum Rückgabewert der Treiberfunktionen wird jeder interne Fehler in einem Reportstring protokolliert:

Die Fehlerstrings werden, jeweils durch ein Zeilenende-Zeichen getrennt, aneinandergelängt und können mit der Funktion "ImpLab_GetErrorReport()" ausgelesen werden.

Der Reportstring wird durch die Funktion "ImpLab_ResetErrorReport()" gelöscht.

Bei gesetzter Debug Option (Funktion "ImpLab_SetDebug()") werden alle Fehler als Meldungbox angezeigt.

Zu den Rückgabewerten der Funktionen kann mit der Funktion "ImpLab_GetErrorString()" der Fehlerstring angefordert werden.

Eine Liste der Fehlercodes findet sich im Anhang.

Registrierung

Der Treiber ist durch einen computerbezogenes Passwort geschützt.

Nach der Erstinstallation läuft der Treiber als Testversion für 10 Tage, danach startet er als Demoversion ohne Hardwarezugriff.

Die Freischaltung zur Vollversion erfordert die Software Registrierung mit einem Registrierschlüssel oder die Verwendung eines Hardware Dongles.

Die interne Abfrage der Registrierung wird durch die Treiberfunktion "ImpLab_GetRegistryState()" aktiviert. Wird diese Funktion nicht aufgerufen, arbeitet der Treiber immer im Demo Modus.

IMP Modul Übersicht

IMP Serie 3595

Die Serie 3595 besteht aus unterschiedlichen Modultypen nach folgender Tabelle.

Typ	Kanalzahl	Anwendung
1A	20	Analogeingang - Spg., Strom, Thermopaar - Kanalumschalter mit Halbleiter
1B	10	Analogeingang - Spg., Widerst, DMS - Kanalumschalter mit Relais
1C	20	Analogeingang - Spg., Strom, Thermopaar - Kanalumschalter mit Relais Iso max. 200V
1D	4	Analogausgang - Spg., Strom
1E	20	Analogeingang - Spg., Strom, Thermopaar - Kanalumschalter mit Relais Iso max. 500V
1H	20	Analogeingang - Spg, Widerst, Thermopaar, PT100, Status Ein/Aus (2K) - Kanalumschalter mit Relais Iso max. 200V
1J	20	Analogeingang - Spg, Widerst, Thermopaar, PT100, Status Ein/Aus (2K) - Kanalumschalter mit Relais Iso max. 500V
2A	20	Digital Ein/Aus - Status, Frequenz, Periode

2B	32	Switch - Status
----	----	-----------------

IMP Serie 5000

Die Serie IMP 5000 besteht aus nur einem Modultyp.

Typ	Kanalzahl	Anwendung
1KE	20	Analogeingang - Spg, Widerst, Thermopaar, PT100, Status Ein/Aus (2K)- Kanalumschalter mit Relais Iso max. 500V

Mit der Serie 5000 ist ein Universal Meßmodul hinzugekommen, das im Gegensatz zur 3595 Serie keine spezielles Interfacekarte benötigt.

Damit kann die Messung über jeden PC erfolgen, der einen LAN-Anschluß (Ethernet) oder ein seriellen Anschluß besitzt.

IMP 5000 Gateway Modus

Besonders vorteilhaft ist der Gateway Modus, bei dem das erste IMP 5000 direkt über LAN an den PC verbunden wird. Dieses Imp arbeitet zusätzlich zur Messung als Ethernet / RS485 Gateway, an dem die weiteren IMPs mit einer verdrehten 2-Draht Leitung verbunden sind.

Installation

Lieferumfang

Der Treiber wird als ZIP-Datei oder mit Installations CD geliefert.

Auf der Installations CD bzw. nach Entpacken der ZIP-Datei finden Sie diese Dateien:

Datei	Typ	Funktion
Setup.EXE	EXE	Installationsprogramm
*.exe, *.msi, *.ini		Weitere Files zum Installationsprogramm
\program files	Header	Verzeichnis mit den Installationsfiles des Programmordners
\system32	Header	Verzeichnis mit den Installationsfiles des Windows Systemordners
\Addon	Header	Verzeichnis mit Zusatz Informationen und Tools zu den IMP Modulen

Nach der Installation befinden sich folgende Dateien auf dem PC:

Datei	Typ	Funktion
implab.dll	DLL	Datei mit der Treiberfunktionalität
Implab2.dll	DLL	Zusatzfunktionen zum Treiber
implab.lib	LIB	Datei zur Einbindung in den Compiler
implab_exp.h	Header	Datei mit den Definitionen, Konstanten, den Funktionsnamen und Funktionsparametern
globaldef.h	Header	Datei mit Konstanten und TypDefinitionen
impdef.h	Header	Datei mit Konstanten und TypDefinitionen
Tcpip.lib	LIB	Datei zur Einbindung in den Compiler
Tcpip.dll	DLL	Datei mit den TCP/IP Funktionen des Treibers
tcpip_exp.h	Header	Datei mit den Definitionen, Konstanten, den Funktionsnamen und Funktionsparametern

System Voraussetzung

Der Treiber setzt ein Windows Betriebssystem voraus wie Windows 98, Windows 2000, Windows NT, XP.

Hardware Voraussetzung

Der Treiber untertstützt die IMP Serien 3595 und 5000.

Für die 3595 IMPs wird eine SNET Karte ISA oder PCI benötigt mit dem zugehörigen Solartron IMP Treiber.

Für die 5000 IMPs wird ein Ethernet LAN Anschluß oder eine serielle Schnittstelle RS485 benötigt.

Der RS485 Bus kann auch mit einem RS232/RS485 Konverter oder einem USB / RS485 Konverter an einem Standard PC mit RS232 und/oder USB Schnittstelle zur Verfügung gestellt werden.

Software Voraussetzung

Für die 3595 IMPs wird der Solartron IMP Hardware Treiber "impdrv.dll" benötigt, der mit der SNET Karte ausgeliefert wird.

Für die 5000 IMPs mit Ethernet Anschluß wird ein TCP/IP Treiber und das Windows Winsock Interface benötigt:

Der TCPIP Treiber ist Bestandteil des Universal Treibers.

Das Winsock Interface ist Bestandteil des Windows Betriebssystems.

Für die 5000 IMPs mit Serielltem Anschluß wird kein zusätzlicher Treiber benötigt.

Installieren

Die Installation erfolgt durch Aufruf der Datei "SETUP.EXE" auf der IMP Universal CD.

Danach startet das Installations Programm Installshield .

Für die Installation muss die Lizenz.. zugestimmt werden und die gestartet.

Es werden keine weiteren Informationen abgefragt.

Nach der Installation befinden sich die Dateien wie unter "Lieferumfang" aufgeführt im benutzerdefinierten Programmverzeichnis bzw. im Windows System Verzeichnis.

Funktions-Referenz

ImpLab_GetVersion

Liest die Versionsnummer und den Versionsstring aus.

Aufruf	ImpLab_GetVersion (char* pszVersion, int nLen);
Return Typ	int
Return-Wert	Versionsnr

Funktionsparameter:

Typ	Wert
char-Zeiger - pszVersion	Versionsstring
Int - nLen	max. Anzahl an Zeichen

ImpLab_Reset

Initialisiert den Treiber:

Löscht alle Interface, IMP und Kanal Strukturen.

Setzt alle Variablen auf Ihre Default Werte

Entlädt die Solartron DLL und die TCPIP DLL, falls diese geladen wurden.

Aufruf	ImpLab_Reset (void);
Return	--
Return-Wert	--

Funktionsparameter:

Keine

ImpLab_SetMeasSrate

Setzt die Basis Abtastrate für alle IMPs in 1/1000 Sekunden.

Aufruf	ImpLab_SetMeasSrate (int nMilliSeconds);
Return Typ	int
Return-Wert	Status:

	0 = OK <> 0 = Fehler
--	-------------------------

Funktionsparameter:

Typ	Wert
Int - nMilliseconds	Basis Abtastrate der Messung in 1/1000 Sekunden Bereich: 100 ms = 10 Messungen / Sekunde bis 10000 ms = 1 Messungen / 10 Sekunden

ImpLab_SetIfcSnet

Meldet die SNET Karte als Interface an und setzt die Parameter

Aufruf	ImpLab_SetIfcSnet (int nSnetAdress, int nPowerOffAtExit);
Return Typ	Int
Return-Wert	Interface-Index, ab 0

Funktionsparameter:

Typ	Wert
Int - nSnetAdress	PC-Basis Adresse der SNET-Karte, Wert innerhalb CMinBoardAdress und CmaxBoardAdress Schrittweite CBoardAdressStep
Int - nPowerOffAtExit	Flag für die Abschaltung der Spannungsversorgung an die IMPs bei Stop der Messung 0 – keine Abschaltung 1 - Abschaltung

ImpLab_GetIfcSnet

Auslesen der Setup Parameter der SNET Karte.

Aufruf	ImpLab_GetIfcSnet (int *pnSnetAdress, int *pnPowerOffAtExit);
Return Typ	Int
Return-Wert	Status: 0 = OK <> 0 = Fehler

Funktionsparameter:

Typ	Wert
Int* - pnSnetAdress	Zeiger auf Integer Variable. Erhält den Wert der PC-Basis Adresse der SNET- Karte. Siehe auch ImpLab_SetIfcSnet()

Int* - pnPowerOffAtExit	Zeiger auf Integer Variable. Erhält den Wert des Flag für die Abschaltung der Spannungsversorgung an die IMPs bei Stop der Messung 0 – keine Abschaltung 1 - Abschaltung
-------------------------	--

ImpLab_SetIfcSeriell

Meldet das Serielle Interface an mit Übergabe der Parameter.

Gilt nur für IMPs der Serie 5000.

Aufruf	ImpLab_SetIfcSeriell (int nComPort, int nBaudrate, int nSerialFormat, int nPowerOffAtExit);
Return Typ	Int
Return-Wert	Interface-Index, ab 0

Funktionsparameter:

Typ	Wert
Int - nComPort	PC-Port Nr. der seriellen Schnittstelle Bereich cComPortMin .. cComPortMax
Int - nBaudrate	Taktrate der Seriellen Übertragung. (muss mit IMP Konfiguration übereinstimmen!) Bereich siehe cBaudrate...
Int - nSerialFormat	Format der seriellen Übertragung (muss mit IMP Konfiguration übereinstimmen!) Bereich cSER_FORMAT_ASCII, cSER_FORMAT_RTU
Int - nPowerOffAtExit	Flag für Stop der Messung bei Stop der Messung 0 – keine Abschaltung 1 - Abschaltung

ImpLab_SetIfcEthernet

Meldet das Ethernet Interface an und setzt die Parameter

Gilt nur für IMPs der Serie 5000.

Aufruf	ImpLab_SetIfcEthernet (int nTCPPort, char *szTCPAddress, char *szTCPSubnet, int nPowerOffAtExit);
Return Typ	Int
Return-Wert	Interface-Index, ab 0

Funktionsparameter:

Typ	Wert
Int - nTCPPort	Port Nr. der TCP/IP Verbindung. (muss mit IMP Konfiguration übereinstimmen!) Default: CnDefTCPPort
Char* - szTCPAdress	Adresse der TCP/IP Verbindung im Format "a.b.c.d". (muss mit IMP Konfiguration übereinstimmen!) Default: CszDefTCPAdress
Char* - szTCPSubnet	Netzmaske der TCP/IP Verbindung im Format "a.b.c.d". (muss mit IMP Konfiguration übereinstimmen!) Default: CszDefTCPSubnet
Int - nPowerOffAtExit	Flag für Stop der Messung bei Stop der Messung 0 – keine Abschaltung 1 - Abschaltung

ImpLab_SetImp

Anmelden des IMPs für die Messung und Setzen der IMP Parameter

Aufruf	ImpLab_SetImp (int nIfcIndex, int nImpAdress, int nIntegrationTime , int nTempUnitSelect, int nTCCheckOnOff);
Return Typ	Int
Return-Wert	IMP Index. Wird bei weiteren Zugriffen auf dieses IMP als Parameter erwartet. > 0 = OK <= 0 = Fehler

Funktionsparameter:

Typ	Wert
Int - nIfcIndex	Index auf das Interface, der bei ImpLab_SetIfc... zurückgegeben wurde
Int - nImpAdress	Adresse des IMPs im Bereich 1..50
Int - nIntegrationTime	Der Wert im Bereich 0..5 legt die interne Integrationszeit des ADC fest: 0 – 20,00 ms (50 Hz) 1 – 16,67 ms (60 Hz) 2 – 5,00 ms (400 Hz) 3 – 4,17 ms 4 – 1,25 ms 5 – 1,04 ms
Int - nTempUnitSelect	Der Wert legt die Temperatureinheit bei allen Temperaturmessungen fest 0 – Einheit °C 1 - Einheit °F
Int - nTCCheckOnOff	Dieses Flag schaltet eine zusätzliche

	Testmessung zur Erkennung offener Thermokanäle ein. (Verlangsamt die Messrate) 0 – Aus 1 – Ein
--	---

ImpLab_GetImp

Auslesen der IMP Setup Parameter

Aufruf	ImpLab_GetImp (int nIfcIndex, int nImpAdress, int *pnIntegrationTime , int *pnTempUnitSelect, int *pnTCCheckOnOff);
Return Typ	Int
Return-Wert	Status: 0 = OK < 0 = Fehler

Funktionsparameter:

Typ	Wert
Int - nIfcIndex	Index auf das Interface, der bei ImpLab_SetIfc... zurückgegeben wurde.
Int - nImpIndex	Index auf IMP, der bei ImpLab_SetImp zurückgegeben wurde
Int* - pnIntegrationTime	Zeiger auf Integer. Der erhaltene Wert liefert die Integrationszeit des ADC
Int* - pnTempUnitSelect	Zeiger auf Integer. Der erhaltene Wert liefert die Temperatureinheit
Int* - pnTCCheckOnOff	Zeiger auf Integer. Der erhaltene Wert liefert das Flag zur Erkennung offener Thermokanäle

ImpLab_SetImpChan

Anmelden eines IMP-Kanals und setzen der Parameter.

Aufruf	ImpLab_SetImpChan (int nIfcIndex, int nImpIndex, int nImpChan, int nModeCode, int nUnitConversMode, float fUnitConvConst, float fUnitConvSlope);
Return Typ	Int
Return-Wert	IMP Kanal Index. Wird bei weiteren Zugriffen auf diesen Kanal als Parameter erwartet. > 0 = OK <= 0 = Fehler

Funktionsparameter:

Typ	Wert
Int - nIfcIndex	Index auf das Interface, der bei ImpLab_SetIfc... zurückgegeben wurde
Int - nImpIndex	Index auf IMP, der bei ImpLab_SetImp zurückgegeben wurde
Int - nImpChan	Terminal Nummer des IMP Kanals im Bereich 1..20
Int - nModeCode	Funktionscode für diesen Kanal. 0 – Kanal inaktiv >0 .. 1000 hex - siehe IMP Manual und Datei "IMPDEF.H"
Int – nUnitConversMode	Der Wert legt die Linearisierung des Meßwertes fest 0 – Meßwert normal 1 - Meßwert mit Linearisierung berechnen (siehe unten)
Float - fUnitConvConst	Linearisierungskonstante
Float - fUnitConvSlope	Linearisierungsfaktor Rückgabe - Meßwert = Linearisierungskonstante + Linearisierungsfaktor * Original-Meßwert

ImpLab_Download

Sendet alle Parameter an die Hardware:

Lädt die Solartron-DLL "IMPDRV.DLL"

Überprüft, ob die angemeldeten SNET Karten vorhanden sind

Überprüft, ob die angemeldeten IMPs am SNET-Bus vorhanden sind

Sendet die Setup Parameter an die IMPs.

Gibt die IMPs zur Messung frei

Aufruf	ImpLab_Download (int nMode);
Return Typ	Int
Return-Wert	Status 0 = OK <> 0 = Fehler

Funktionsparameter:

Typ	Wert
Int - nMode	Modus für Download. Wertebereich: CDownloadModeIfc = nur Interface anmelden cDownloadModeImp = nur IMPs anmelden (nur nach Aufruf von ImpLab_Download (CDownloadModeIfc)) cDownloadModeAll = Alles anmelden

ImpLab_StartMeasure

Startet die Messung bei allen IMPs

(Snet: Senden des Kommandos "CO;TR")

Aufruf	ImpLab_StartMeasure (void);
Return Typ	Int
Return-Wert	Status 0 = OK <> 0 = Fehler

Funktionsparameter:

Keine

ImpLab_StopMeasure

Stoppt die Messung bei allen IMPs.

(Snet: Senden des Kommandos "HA")

Aufruf	ImpLab_StopMeasure (void);
Return Typ	Int
Return-Wert	Status 0 = OK <> 0 = Fehler

Funktionsparameter:

Keine

ImpLab_ReadDataCopy

Kopiert die Daten eines IMPs in den Datenspeicher, falls neue Daten verfügbar sind.

Aufruf	ImpLab_ReadDataCopy (int nIfcIndex, int nImpIndex, int nNumChannels, float *pfDataValue, short *psDataValid);
Return Typ	Int
Return-Wert	Status der Daten 0 - keine Daten bereit 1 - Daten bereit, in DatenListe kopiert <0 - Error

Funktionsparameter:

Typ	Wert
Int - nIfcIndex	Index auf das Interface, der bei ImpLab_SetIfc... zurückgegeben wurde
Int - nImpIndex	Index auf IMP, der bei ImpLab_SetImp zurückgegeben wurde
Int - nNumChannels	Anzahl der zu kopierenden Kanäle, ab 1

float* - pfDataValue	Zeiger auf Float Speicher. Erhält die Meßwerte.
short* - psDataValid	Zeiger auf Short Speicher. Erhält die Gültigkeitswerte der Messung

ImpLab_ReadDataChannels

(Funktion ist erst ab Version 2 verfügbar)

Kopiert die Daten aller Kanäle des letzten Meßzyklus in den Datenspeicher, falls neue Daten verfügbar sind.

Die Daten werden aus einem Ringspeicher gelesen, der automatisch von einem unabhängigen Meßthread geschrieben wird.

Die Anordnung der Meßkanäle entspricht der Anmelde Reihenfolge bei ImpLab_SetImp (...).

Die Kanalnummern sind aufsteigend angeordnet.

Die Anordnung der Meßkanäle kann über die Funktion ImpLab_GetChanIndex (...) gelesen werden.

Aufruf	
Return Typ	Int
Return-Wert	Status der Daten 0 - keine Daten bereit > 0 - Daten bereit, Meßindex ab 1 <0 - Error

Funktionsparameter:

Typ	Wert
Int - nMaxChannels	Max. Anzahl der zu kopierenden Werte
float* - pfTimeValue	Zeiger auf Float Speicher der Größe 1. Erhält die Meßzeit
float* - pfDataValue	Zeiger auf Float Speicher der Größe nMaxChannels. Erhält die Meßwerte.
short* - psDataValid	Zeiger auf Short Speicher der Größe nMaxChannels. Erhält die Gültigkeitswerte der Messung

ImpLab_GetChanIndex

(Funktion ist erst ab Version 2 verfügbar)

Liest die Anordnung der Meßkanäle im Ringpuffer aus. Dies wird für die Funktion ImpLab_ReadDataChannels () gebraucht.

Aufruf	
Return Typ	Int
Return-Wert	Status 0 – OK <0 - Error

Funktionsparameter:

Typ	Wert
Int - nIfcIndex	Index auf das Interface, der bei ImpLab_SetIfc... zurückgegeben wurde
Int - nImpIndex	Index auf IMP, der bei ImpLab_SetImp zurückgegeben wurde
Int - nImpChanIndex	Index auf IMP Kanal, der bei ImpLab_SetImpChan zurückgegeben wurde
Int* - pnDataIndex	Zeiger auf Integer Erhält den Index-Wert im Datenfeld. Ab 0

ImpLab_Disconnect

Trennt alle Interface Verbindungen und setzt Power Off bzw. Messtakt aus.

Aufruf	ImpLab_Disconnect (void);
Return Typ	Int
Return-Wert	Status 0 = OK <> 0 = Fehler

Funktionsparameter:

Keine

ImpLab_IsMeasureRunning

Zeigt Status der Messung an

Aufruf	ImpLab_IsMeasureRunning (void);
Return Typ	Int
Return-Wert	Status der Messung 0 - Messung steht 1 - Messung läuft

Funktionsparameter:

Keine

ImpLab_GetRegistrationCode

Liest den Registrierungscode und die Computer-ID aus.

Achtung!

Der Registrierungscode ändert mit jedem Aufruf dieser Funktion. Die Registrierungsschlüssel sind nur für den zuletzt angeforderten Registrierungscode gültig.

Aufruf	ImpLab_GetRegistrationCode (int *pnRegCode, int *pnCompID);
Return Typ	Int

Return-Wert	Statusrückmeldung 0 = OK <> 0 = Fehler
-------------	--

Funktionsparameter:

Typ	Wert
Int*- pnRegCode	Zeiger auf Integer Variable. Erhält nach Aufruf den Wert des Registrierungscode
Int* - pnCompID	Zeiger auf Integer Variable. Erhält nach Aufruf den Wert der Computer-ID

ImpLab_SetRegistrationKey

Setzt den Registrierungsschlüssel zum letzten Registrierungsaufruf.

Aufruf	ImpLab_SetRegistrationKey (int nCodeVal, int nDataVal);
Return Typ	Int
Return-Wert	Statusrückmeldung 0 = OK <> 0 = Fehler

Funktionsparameter:

Typ	Wert
Int - nCodeVal	Code
Int - nDataVal	Wert

ImpLab_GetRegistrationState

Liest den Status der Registrierung aus mit Zusatzinformationen

Aufruf	ImpLab_GetRegistrationState (int*pnRunsleft, int *pnDaysleft, int *pnCompNo);
Return Typ	Int
Return-Wert	Registrierungs Status: 0 = Demo Modus 1 = Test Modus 2 = Vollversion

Funktionsparameter:

Typ	Wert
Int* - pnRunsleft	Im Testmodus: Verbleibende Starts
Int* - pnDaysleft	Im Testmodus: Verbleibende Tage
Int* - pnCompNo	Computer-ID

ImpLab_GetErrorReport

Kopiert den vollständigen Fehlerreport.

Der Fehlerreport enthält alle Fehlerstrings hintereinander, mit Zeilenumbruch getrennt.

Aufruf	ImpLab_GetErrorReport (char *pszError, int nByteSize);
Return	Int
Return-Wert	Anzahl kopierte Bytes

Funktionsparameter:

Typ	Wert
Char* – pszError	Zeiger auf String
Int - nByteSize	Max. Anzahl Bytes zu kopieren

ImpLab_GetErrorReportSize

Liefert die Speichergröße zum Kopieren des Fehlerreports.

Aufruf	ImpLab_GetErrorStringSize (void);
Return	Int
Return-Wert	Anzahl Bytes

Funktionsparameter:

keine

ImpLab_ResetErrorReport

Setzt Fehlerreport zurück, d.h. der Report-String wird komplett gelöscht.

Aufruf	ImpLab_ResetErrorReport (void);
Return	--
Return-Wert	--

Funktionsparameter:

keine

ImpLab_GetErrorString

Liefert einen Fehlerstring zum Fehlerindex (+ oder - identisch)

Aufruf	ImpLab_GetErrorString (int nErrorIndex, char* pszError, int nStringSize);
Return	--
Return-Wert	--

Funktionsparameter:

Typ	Wert
int – nErrorIndex	Fehler Index, der von einer

	ImpLab_..Funktion zurückgeliefert wurde.
Char* – pszError	Zeiger auf String
Int - nStringSize	Max. Anzahl Bytes zu kopieren

ImpLab_SetDebugMode

Fehlerausgabe über Messagebox ein- oder ausschalten

Aufruf	ImpLab_SetDebugMode (int nMode);
Return Typ	--
Return-Wert	--

Funktionsparameter:

Typ	Wert
Int - nMode	Modus für Fehlermeldungen. Wertebereich: 0 – keine Ausgabe der Fehlermeldungen 1 - Ausgabe aller Fehlermeldungen in einer Messagebox mit Bestätigung.

Anwendung

Funktionsreihenfolge

Die Einbindung der Funktionen sollte nach der folgenden Reihenfolge ablaufen:

- Treiber-Reset
- Treiber Versions- und Registrierungsabfrage
- Setzen der Meßparameter
- Setzen der Interface Parameter
- Setzen der IMP Parameter
- Setzen der Kanal Parameter
- Download der Parameter an die Hardware
- Start der Messung
- Auslesen der Meßdaten
- Stop der Messung

Zuordnung

Die Zuordnung der IMP Module zu den Interfaces und die Zuordnung der IMP Kanäle zu den IMP Modulen erfolgt über die Rückgabe Codes beim Setzen der Interface bzw. Setzen der IMPs.

Somit können IMP Module und Kanäle auf einfache Weise zugeordnet werden.

C++

Zur Einbindung des Treibers in ein MFC bzw. C++ Programm wird die Dll über die LIB Datei und die Header Datei definiert.

Die Einbindung kann statisch mit Programmstart oder dynamisch zur Laufzeit erfolgen.

Alle Funktionsaufrufe wurden über die "extern "C" " Anweisung exportiert und führen das Makro AFX... aus.

Die Einbindung erfolgt in den Projekteinstellungen im Feld "Link / Input" durch einen zusätzlichen Eintrag "IMPLAB.LIB" unter "Object/Library Modules".

Die Datei "IMPLAB.DLL" kann sich entweder im EXE Verzeichnis oder im Windows-System32-Verzeichnis befinden.

Beispiel “CallImpLab.EXE”

Das Beispiel-Programm zeigt die Einbindung in ein MFC C++ Programm.

Dieses Programm wurde mit dem Projekt-Wizard als MFC-Appwizard (EXE) mit Namen CallImpLab erstellt, mit Auswahl Single Document und Baseclass = CformView.

Der Wizard erstellt u.a. die Datei “CallImpLabView.CPP” für die Ansichtsklasse “CCallImpLabView”. Die Treiberaufrufe wurden in der Datei “ImpLabImp.CPP” zusammengefasst. Dort werden die Treiberfunktionen über die Header Datei “implab_exp.h” eingebunden.

Projekteinstellungen

General / MFC = Use MFC in a shared DLL

C++ Code Generation / Calling Convention = _cdecl

Link / Object-Library Modules = implab.lib

Init

In der Funktion OnInitialUpdate wird der Treiber initialisiert durch Aufruf der Funktion ImpInit(). Diese setzt den Treiber zurück, ruft die Versionsinfo ab und setzt die Fehlerausgabe (optional).

Das Dialogfenster enthält zwei Buttons, mit denen die Messung gestartet und gestoppt wird.

Start

Der Startbutton ist mit der Funktion “OnStart2()” verknüpft, die wiederum die Funktion “ImpDownloadAndStart()” aufruft und den 1 Sekunden Timer freigibt.

Setup

An diese Funktion wird die Snet Adresse übergeben. Dort wird der Treiber Reset aufgerufen, das SNET Interface gesetzt und im weiteren das IMP mit Adresse 1 und die Kanäle 1 bis 4 mit der Funktion “Thermoelement Typ K” gesetzt.

Download und Start

Im folgenden ruft diese Funktion nach fertiger Parameterübergabe die Treiberfunktion “ImpLab_Download()” auf und startet die Messung mit “ImpLab_StartMeasure()”.

Abfrage und Anzeige der Meßdaten

In der Funktion “ImpReadout()” wird die Treiberfunktion “ImpLab_ReadDataCopy()” aufgerufen, die prüft, ob neue Daten vorliegen und falls ja, diese in den Zielspeicher umkopiert. Im Treiber Callback “OnTimer()” der Klasse “CCallImpLabView” werden diese Werte dann in die Member-Variablen “m_strData1” bis ..“m_strData4” eingetragen und angezeigt.

Der Button Stop ist mit der Funktion “OnStart3()” verknüpft, die wiederum die Funktion “ImpStop()” aufruft und den Timer beendet.

Bei Stop wird die Treiberfunktion “ImpLab_StopMeasure()” aufgerufen. Der Solartron Treiber wird mit der Treiberfunktion “ImpLab_Disconnect()” entladen.

LabView

Zur Einbindung des Treibers in ein Virtuelles Instrument (VI) in LabView wird die Dll dynamisch als ... über den DLL-Namen eingebunden.

Als Aufruf Konvention wird Pascal gewählt.

Die Beispiel Vis wurden mit LabView 6.1 erstellt und getestet.

Beispiel "IMPDEMO.LLB"

Anhang

Fehlercodes

Für die Treiberfunktionen, die als Rückgabewert den Status der Funktionsausführung liefern, gelten die Fehlercodes aus "ImpLab_Error.H"

- 102 "SNET kann nicht geöffnet werden";
- 104 "SNET Basisadresse unterhalb Gültigkeitsbereich";
- 105 "SNET Basisadresse oberhalb Gültigkeitsbereich";
- 106 "SNET Basisadresse nicht Vielfaches von 512";
- 107 "SNET Klasse konnte nicht angelegt werden";
- 108 "Kein freier Index für SNET Klasse";
- 109 "SNET Index ausserhalb Bereich";
- 110 "Nullzeiger auf SNET Klasse";
- 111 "SNET Parameter außerhalb Wertebereich";
- 201 "IMP kann nicht geöffnet werden";
- 202 "Keine freier Index für IMP-Klasse";
- 204 "Snet Connect nicht möglich";
- 205 "IMP Parameter ausserhalb Wertebereich";
- 206 "IMP nicht initialisiert";
- 207 "IMP Typ wird nicht unterstützt";
- 208 "Stream Adresse ausserhalb 0..3";
- 209 "Fehler in Funktion -ImpConnect-";
- 210 "IMP Adresse ausserhalb 1..50";
- 211 "Imp nicht am Bus gefunden";
- 212 "IMP Index ausserhalb Bereich";
- 213 "Nullzeiger auf IMP Klasse";
- 301 "Nullzeiger auf IMP-Kanal-Klasse";
- 302 "IMP Kanalparameter ausserhalb Wertebereich";
- 303 "IMP Kanal Adresse ausserhalb Bereich";
- 305 "IMP Kanal nicht initialisiert";

401 "Solartron DLL nicht geladen";
403 "Solartron DLL Funktionszeiger nicht gesetzt";
405 "Solartron DLL nicht entladen";
501 "Setup String Überlauf";
502 "Setup String Teilungsfehler";
601 "Aufruf Nullzeiger";
602 "Funktionsaufruf hier nicht erlaubt";
701 "String wurde nicht fehlerfrei gesendet";
702 "Indexüberlauf bei Zugriff auf IMP-Datentabelle";
703 "Timeout bei Teststream";
801 "Messung läuft";
802 "Messung läuft nicht";
888 "Passwort nicht gesetzt";
889 "Registrierung nicht gesetzt";

TCP/IP DII

Datei "test_dll1.dll"

Funktionen zur Kommunikation mit den IMP 5000 Modulen über die Ethernet Schnittstelle.

Verwendet die Windows WINSOCK DLL.

Wird vom IMP Universaltreiber dynamisch beim Download der Parameter bzw. Start der Messung eingebunden.

Solartron DLL

Datei "impdrv.dll" und weitere vxd bzw. sys Dateien

Basisfunktionen zum Zugriff auf das SNET PC Interface.

Wird vom IMP Universaltreiber dynamisch beim Download der Parameter bzw. Start der Messung eingebunden.

Header Datei "IMPDEF.H"

Treiber Spezifische Typen und Konstanten.

Die Konstanten können als Werte an die Funktionsaufrufe übergeben werden.

Header Datei "GLOBALDEF.H"

Allgemeine Typen Definitionen

Glossar

LabView

Grafischer Compiler von National Instruments

DLL

Data Link Library zur Einbindung in Windows Programme

IMP

Isolated Measurement Pod, Präzisions Feldbus Modul

Index

3

3595 1–2, 4

5

5000 1–2, 4, 8, 21

E

Ethernet 1–2, 4, 8, 21

I

IMP 1–6, 8–11, 13–14, 19, 21

L

LabView 1, 20

R

RS485 2, 4

T

TCPIP 4–6

Treiber 1, 3–6, 19, 21