



PEM-Lite++

Programmierbibliothek für TranSpec[®] - Spektrometer

Zur Erstellung eigener Applikationen für die allgemeine Spektroskopie und speziell zur Messung von Plasma-Emissionen mit Hilfe unserer TranSpec-Spektrometer bieten wir unsere leistungsfähige, aber dennoch einfach zu handhabende Programmierbibliothek **PEM-Lite++** an.

In PEM-Lite++ ist die gesamte Spektrenerfassung, d.h., das Scannen der Diodenzeile, eventuelle Mittelung der Rohdaten, Dunkelstrom-Korrektur und die anschließende Spektrennormierung vollständig in einfachen Funktionsaufrufen gekapselt, so dass Sie sich ganz auf die Erstellung Ihrer Applikation konzentrieren können. Natürlich haben Sie Zugriff auf alle gemessenen Spektren, inklusive der Rohdaten.

- Laufzeit-Lizenzierte Dynamische Link Library (DLL) mit Standard-C Aufrufen
Kompatibel zu gängigen C/C++ Compilern, Visual Basic und VBA (Excel), LabView
- Ausführliche Überprüfung von Parametern und aktuellem Mess-Status
Sie können mit PEM-Lite++ praktisch nichts falsch machen!
- Unterstützung von optionalem, externen I/O-Modul mit 8x TTL und 4x Analogausgabe
- Detailliertes, gedrucktes Benutzerhandbuch, HTML-Hilfe und PDF-Dokument
- Beispielprogramm als Windows Konsolenapplikation, mit C/C++ Quellcode
- Auf der nächsten Seite finden Sie ein kleines Programmierbeispiel

Technische Spezifikationen auf der nächsten Seite ►



PEM-Lite++ Programmierbibliothek - Technische Daten

Januar 2011, bezogen auf Version 1.0, alle Angaben ohne Gewähr, technische Änderungen vorbehalten

Hardware- und Softwarevoraussetzung

- PC/Laptop mit wenigstens Pentium-4, Dual-Core empfohlen
- Windows XP, Windows Vista oder Windows 7 (jeweils in der 32-Bit Version)
- CD-ROM Laufwerk für die Installation
- C/C++ Entwicklungssystem (MS Visual Studio empfohlen), Delphi, Visual Basic oder VBA (Excel), LabView
- TranSpec Spektrometer (USB 2.0 Anschluss am PC/Laptop benötigt)

Programmierbeispiel

Nachfolgend ein kleines Programmbeispiel, um die Leistungsfähigkeit und einfache Handhabung von PEM-Lite++ zu demonstrieren. Als exemplarische Aufgabe programmieren wir die vollautomatische Messung eines 10fach gemittelten Emissions-Spektrums mit Analogausgabe eines Emissions-Trendwertes:

```
// Schritt 1: Spektrometer öffnen und initialisieren
PEMLITE_SPECHARDWARE sSpecHardwareInfo;
PEMLite_OpenSpectrometer( PEMLITE_TRANSPEC_19Z, &sSpecHardwareInfo );

// Schritt 2: Messparameter festlegen
PEMLITE_MEASPARA sMeasPara;
sMeasPara.dIntegrationTime = 20.0;           // 20 ms Integrationszeit
sMeasPara.bEnableAverage = 1;             // Spektrenmittelung an
sMeasPara.lNumberAverage = 10;          // 10fach messen und mitteln
PEMLite_SetMeasPara( &sMeasPara );       // Messparameter an Spektrometer senden

// Schritt 3: Externes Digital/Analog-Modul öffnen und initialisieren
PEMLite_USB3110_OpenDevice( PEMLITE_DIGITAL-8OUT0IN, PEMLITE_ANALOG_UNIPOLAR );

// Schritt 4: Messung eines gemittelten Emissionsspektrums
PEMLITE_SPECSTATUS sSpecStatus;
PEMLite_RunMeasSpectrum ();               // Messung starten

PEMLite_GetSpecStatus( &sSpecStatus );    // Warten bis Messung beendet ist
while ( sSpecStatus.bRunSpectrum )
    PEMLite_GetSpecStatus( &sSpecStatus );

// Schritt 5: Die Messung ist beendet, wir rufen das Spektrum ab
PEMLITE_SPECDATA sSpecData;
PEMLite_GetSpectrumData( PEMLITE_SPECTRUM_EMISSION, &sSpecData );

// Schritt 6: Berechnung und Ausgabe des Emissions-Trendwertes bei z.B. 254 nm als analoger Spannungswert:
double dTraceValue;
PEMLite_GetTraceValue( 254.0 , &dTraceValue );

// Trendwert auf +10 Volt normieren und am Kanal 0 ausgeben
....
PEMLite_USB3110_SetAnalogOut( 0 , dTraceValueAsVolt );
```

Hinweis TranSpec ist ein in Deutschland eingetragenes Warenzeichen des Ing.-Büros für Angewandte Spektrometrie, Dipl.-Ing. (FH) Th. Fuchs. Alle sonstigen Produktnamen sind möglicherweise Warenzeichen oder eingetragene Warenzeichen ihrer jeweiligen Hersteller.