

# TSEP POSEIDON

*Ihr Remote System mit SCPI-99-Parser.*



**TSEP**

Innovation made measurable.

# TSEP Poseidon

*Unser SCPI 488.2 konformer Parser.*

## Highlights



Leistungsstarke Befehlsausführung und Datenübertragung



Generierung von Code-Skeletten und Dokumentation mit XML



Standard-Datenkanäle: TCP/IP, RS232 und HiSlip (nur für LXI-Mitglieder)



Innovative und effiziente SCPI-Parameter-Architektur



Multi-Instanz-Konzept für Parser und Datenkanäle



Plattformübergreifend kompatibel (Windows und Linux)

In der heutigen Testautomatisierung besteht ein Bedarf an schneller Befehls- und Abfrageausführung von Messgeräten. TSEP's Poseidon optimiert den zeitaufwändigen Prozess der Kommunikation des Messgerätes mit den Clients des Testsystems. Das innovative Design zielt auf die Optimierung der Befehlsausführung auf allen systematischen Ebenen. Die Optimierungsmöglichkeiten des Poseidon beruhen auf der baumartigen Datenstruktur der SCPI-Befehle und der Priorisierung der SCPI-Befehle während der Laufzeit auf Basis einer Zugriffsstatistik.

Parallel dazu bietet Poseidon ein einfach zu integrierendes, flexibles und zuverlässiges Framework. Die Generierung, Integration und Dokumentation von eigenen SCPI-Befehlen und Parsern kann mit einem XML-Generator erfolgen.

# Effizienz und Innovation

*TSEP bündelt das Wissen und die Erfahrung aus über 30 Jahren in der T&M-Industrie und entwickelte das SCPI 488.2-konforme Parse.*

## Effizientes Multi-Instanz-Konzept

Mit Poseidon können Prüf- und Messgeräte einfach, effektiv und kostengünstig mit einem SCPI-konformen Parser ausgestattet werden. Poseidon enthält einen SCPI 488-konformen Parser, der die entsprechenden Befehle übersetzt und an die Gerätefirmware weitergibt. Poseidon ist speziell auf eine schnelle Befehls- und Abfrageausführung und Datenübertragung ausgelegt. Auf Intel I5-Rechnern sind Antwortzeiten (z.B. \*IDN?) von weniger als 400µs mit TCP/IP und localhost loop-back kein Problem. Bei High-End-Geräten mit I7 und entsprechender CPU-Taktung können diese Zeiten noch einmal minimiert werden. Im Labor wurden \*IDN?-Reaktionszeiten von weniger als 100µs gemessen. TSEP verbessert kontinuierlich die Leistung von Poseidon, da die Faktorleistung ein wichtiges Schlüsselmerkmal ist.

Die Modularität von TSEP Poseidon spiegelt sich in der Architektur wider. Die drei Hauptkomponenten sind: die Kanäle, die Parser und ein Kernmodul. Der Kommunikationskanal implementiert die Hardware-Kommunikationsschnittstelle. Das Multi-Instanz-Kanalkonzept kann mehrere aktive Verbindungen gleichzeitig handhaben. Kunden können auch unabhängige Kanäle definieren und implementieren. Der Parser definiert die einzelnen SCPI-Befehle, deren Syntax und Verarbeitung. Poseidon unterstützt mehrere Parser zur gleichen Zeit. Dies erlaubt dem Kunden, seine SCPI-Befehlsstruktur modular aufzubauen und in anderen Gerätevarianten wiederzuverwenden. Alle Kanäle übertragen ihre Daten an den Kern von Poseidon und werden dann vorverarbeitet und an den entsprechenden Parser weitergeleitet. Die Ergebnisse des Parsers werden dann über den Core an die Gerätefirmware kommuniziert.

## Innovative SCPI-Parameter-Architektur

Für Poseidon wurde ein vollständig objektorientierter Ansatz für die Verarbeitung der SCPI-Befehls- und Abfrageparameter gewählt. Innerhalb von Poseidon werden diese Parameter immer über objektorientierte Klassen abgebildet. Dieser Ansatz vereinfacht die Verarbeitung und Erweiterung von Parametern drastisch. Poseidon stellt die notwendigen Basisparameter wie numeric, float oder string zur Verfügung. Darüber hinaus werden einige zusätzliche Parameter definiert. So sind z.B. Arrays von Parametern möglich, was den Aufwand für die Definition von Konfigurationsbefehlen (Antennenparameter, Kalibrierdaten, etc.) reduziert. Auch die Verarbeitung von Sequenzcontainern ist möglich.

Die mitgelieferten Parameter reichen in der Regel nicht aus, um alle Kundenbedürfnisse abzudecken, so dass der Kunde die Parameter jederzeit erweitern und zur Laufzeit in den Poseidon laden kann. Es gibt mehrere Beispiele für benutzerdefinierte Parameter, die die Verwendung durch den Kunden zeigen. Bei der Generierung des Quellcode-Skeletts integriert der Generator alle verfügbaren Parameter in den Befehl, was dem Entwickler bei der Implementierung hilft. Innerhalb des Befehls ist der Zugriff auf die Parameter über einen Casting-Mechanismus einfach möglich. Der Zugriff auf die Parameterdaten erfolgt über einen Get/Set-Mechanismus, der mit einer entsprechenden Signatur versehen ist.

# Poseidon Struktur

*Definieren Sie Ihren eigenen SCPI-Parser für Ihr Bussystem und setzen Sie auf die plattformunabhängige Systemrobustheit.*

## Allgemeines

Poseidon basiert auf einem objekt-orientierten Ansatz, der konsequent umgesetzt wurde. Alle notwendigen Schnittstellen werden über Vererbung an die entsprechende Gerätefirmware weitergegeben. Da SCPI-Befehle und Abfragen immer auf Gerätedaten referenzieren, wurde ein Ansatz zum Zugriff auf diese Daten definiert und implementiert. So können gerätespezifische Daten einfach an die entsprechenden Befehle weitergegeben werden.

## Parser

Die eigentliche Verarbeitung der SCPI-Befehle erfolgt im entsprechenden Parser. Innerhalb von Poseidon kann eine unbegrenzte Anzahl von Parsern definiert werden. Dies bedeutet, dass Funktionalitäten gekapselt und in anderen Geräten wiederverwendet werden können. TSEP stellt mehrere Parser mit Teilen des SCPI 488-2 Befehlssatzes zur Verfügung. Die Befehlssyntax basiert auf SCPI 488-2, enthält aber auch spezifische Erweiterungen, die separat ein- oder ausgeschaltet werden können. Das Parsen von Befehlen in Poseidon ist sehr effektiv und effizient, so dass eine optimale Verarbeitungszeit erreicht wird. Darüber hinaus hat TSEP einige Optimierungsverfahren in die Verarbeitung eingebaut. Darüber hinaus ist die SCPI-Befehlsverarbeitung für Multicore- und Multi-Thread-CPU's ausgelegt, so dass eine parallele Verarbeitung möglich ist, was wiederum zur Optimierung der Verarbeitungszeit beiträgt. Poseidon unterstützt auch die Verarbeitung von überlappenden Befehlen. Diese Befehle müssen nicht abgeschlossen sein, bevor der nächste Befehl ausgeführt wird. Diese Befehle können praktisch parallel ablaufen. Die Implementierung dieser Befehle unterscheidet sich nur geringfügig von den "nicht überlappenden" Befehlen, lediglich die Möglichkeit, auf einen Abbruch während der Ausführung zu reagieren, muss programmatisch hinzugefügt werden. Die Parser erlauben auch die Verwendung von Alias-Befehlen. Alias-Befehle sind Befehle, die eine andere Syntax verwenden, aber die gleichen Funktionalitäten und Parameter haben. Solche Konstruktionen sind notwendig, um ältere Befehle (Kompatibilitätsmodus) von bestehender Messsoftware zu unterstützen.

## Kanäle

Poseidon definiert Kanäle als Kommunikation zwischen dem Parser und der Kommunikationshardware. Diese Kanäle sind die Implementierung des Datenzugriffs von und auf die Hardware. Derzeit unterstützt Poseidon die Standardschnittstellen TCP/IP, HiSlip und RS232. Es ist jedoch jederzeit möglich, dass ein Kunde seinen eigenen Kanal erstellt. Die Erstellung eines eigenen Kanals kann mit Hilfe einer detaillierten Beschreibung erfolgen. In Poseidon können sowohl Single-Instance Channels als auch Multi-Instance Channels definiert und implementiert werden. Singleinstance-Kanäle sind typischerweise Kanäle, die nur eine Verbindung pro Hardware-Ressource erlauben. Als Beispiel kann hier die RS232-Schnittstelle genannt werden. Es können natürlich mehrere RS232-Hardwaremodule unterstützt werden. Dazu müssen die einzelnen Singleinstance Channels mehrfach (d.h. für jede Hardware) in Poseidon registriert werden. Multiinstanz-Kanäle können mehrere aktive Verbindungen zur gleichen Zeit verarbeiten. Als Beispiel für solche Kanäle können die TCP/IP- und die HiSlip-Kanäle angeführt werden. Diese Channels können auch mehrfach in Poseidon registriert werden und z.B. über verschiedene TCP/IP Ports kommunizieren. Channels können zur Laufzeit geladen oder aus Poseidon entfernt werden

## Core

Der Core von Poseidon verwaltet und bearbeitet die Kommunikation zwischen dem Client und der Firmware. Der Core kann nicht nur synchrone Befehle verarbeiten, sondern unterstützt auch asynchrone Befehle. Dabei laufen diese Befehle in einem eigenen Thread und sind somit in sich abgeschlossen und unabhängig ausführbar. Die Implementierung dieser asynchronen Befehle unterscheidet sich nicht von den synchronen Befehlen.

Der Kern unterstützt außerdem die folgenden Funktionen:

- ✓ SCPI-Statusregister;
- ✓ Kompatibilitätsmodi um Funktionen von anderen Geräteherstellern zu nutzen;
- ✓ Speed-Up-Modus, verbesserte Befehls
- ✓ Verarbeitung

## XML-Generator

Die SCPI-Befehle werden innerhalb von Poseidon über XML-basierte Definitionen erstellt und verwaltet. Mit diesem Generator werden nicht nur die notwendigen Quellcode-Teile erstellt, sondern auch die notwendige Dokumentation generiert. In den XML-Dateien wird nicht nur die Befehlssyntax definiert, sondern es werden auch Attribute und Beschreibungen hinterlegt. Somit sind alle befehlspezifischen Definitionen zentral abgelegt. Mit Hilfe des Generators werden die notwendigen Quellcode-Skelette zum Einbinden in die Firmware erzeugt. Der Kunde muss sich nur noch um die Implementierung der eigentlichen Funktionalität kümmern. Alle SCPI-Befehle sind als C++-Klassen implementiert. Alle notwendigen Header und das spezifische Parser Framework werden automatisch erstellt. Die in XML definierten Beschreibungen sind ebenfalls in den Source Code Skeletons enthalten. Auch die Client-Seite der SCPI-Kommunikation kann mit dem Generator erstellt werden.

Analog zur Serverseite (Messgerät) werden auch hier die Quellcode-Skelette erzeugt. Der Generator kann auch zur Erstellung der Dokumentation für die in den Parsern enthaltenen Befehle verwendet werden. Der Generator generiert die entsprechende Dokumentation auf Basis der XML-Definitionsdatei und einer Word-Vorlage. TSEP stellt eine Vorlage zur Verfügung, die zur Erstellung eines eigenen Dokuments verwendet werden kann. Der Kunde kann diese Vorlage jedoch jederzeit ändern und an seine Wünsche und sein CI anpassen.

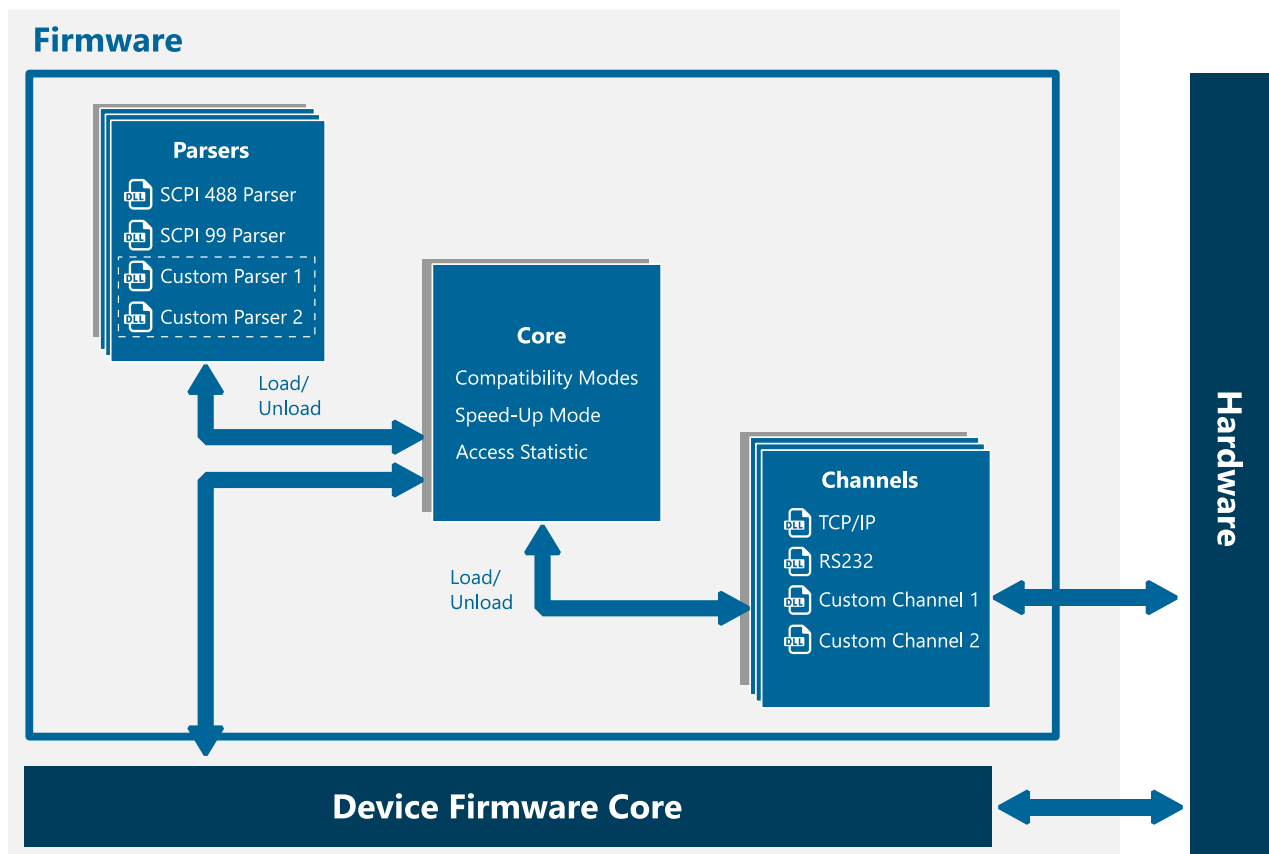


Abb. 1: Poseidon kommuniziert mit der Firmware und der Hardware des Geräts.

AKTIVIEREN SIE IHR “  
T&M GERÄT MIT LEICHTIGKEIT  
” ZUM SCPI-KONFORMEN  
KOMMUNIZIEREN

# Variable Bereitstellungsmodelle

*Sie liefern die Plattform und Ihre Anforderungen, und wir bereiten für Sie TSEP Poseidon vor.*

Poseidon wurde vollständig in C++ erstellt und basiert auf dem C++17-Standard. Für die Generierung der Projektdaten (Visual Studio Solution/Projektdateien, Makefiles oder Eclipse-Projekte) wurde CMake verwendet, um den unterschiedlichen Entwicklungsumgebungen Rechnung zu tragen. Das macht es einfach, neue Entwicklungsumgebungen einzuführen oder Änderungen an bestehenden Projekten vorzunehmen.

Poseidon ist auf allen Windows (z.B. 7 und 10) und Linux (z.B. Ubuntu 16.04 und Ubuntu 20.04) Systemen verfügbar. Der Quellcode ist für beide Plattformen identisch (Common Source), was die Wartung und Integration vereinfacht.

## Zukünftige Features

*TSEP treibt die Entwicklung von TSEP Poseidon weiter voran, um allen Kundenanforderungen von heute und morgen gerecht zu werden.*

### **HiSlip 2.0-Kanal**

Als aktives Mitglied der IVI Foundation verfolgt TSEP die Entwicklung des neuen Sicherheitsstandards HiSlip 2.0. Sobald der HiSlip 2.0-Standard eingeführt ist, wird TSEP einen entsprechenden Kanal für Poseidon bereitstellen.

### **Sicherer TCP/IP-Kanal**

Sicherheit für TCP/IP ist heutzutage ein wichtiges Thema. TSEP befasst sich mit diesem Thema und wird einen sicheren TCP/IP-Kanal für den Poseidon entwickeln.

### **VISA-Bibliothek**

TSEP wird eine proprietäre VISA-Bibliothek für die Kommunikation mit beliebigen Messgeräten anbieten. Um die Nutzung dieses Clients zu vereinfachen und zu standardisieren, wird TSEP den Client entsprechend den VISA-Standards erweitern.

### **Python-Plugins**

Integrieren Sie neue benutzerdefinierte SCPI-Befehle und -Kanäle mit einem einfachen und schnellen Python-Plugin-System, so dass kein Neuaufbau der Firmware erforderlich ist. Die Plugins werden in Python erstellt, so dass nur eine geringe Einarbeitungszeit und ein geringes Qualifikationsniveau erforderlich sind. Dadurch wird die Zeit bis zur Einsatzbereitschaft drastisch verkürzt.

# Bestellinformationen

## Binär Lizenzen

Bestellref.	Beschreibung
PS-B	Firmen Binär Lizenz
PS-SC	Source Code Lizenz

## Allgemeine Lizenzen

Bestellref.	Beschreibung
PS-SUP-1	Support für 1 Jahr
PS-SUP-3	Support für 3 Jahre
PS-UPD-1	Updates für 1 Jahr
PS-UPD-3	Updates für 3 Jahre
PS-DL	Consulting

## Kontakt

Technical Software Engineering Plazotta GmbH

Hopfenstr. 30  
85283 Wolnzach  
Tel: +49 8442 96240 0  
E-Mail: [info@tsep.com](mailto:info@tsep.com)  
[www.tsep.com](http://www.tsep.com)

## Über TSEP

Technical Software Engineering Plazotta GmbH

TSEP ist ein weltweit tätiges Systemhaus. Seit über 30 Jahren haben wir uns erfolgreich auf die Entwicklung systemnaher Soft- und Hardware in den Bereichen Automotive, Telekommunikation, Nachrichtentechnik und Messtechnik spezialisiert.

### TSEP Poseidon

[www.tsep.com/de/products/poseidon/](http://www.tsep.com/de/products/poseidon/)

