



IIoT Libraries SL

Das Softwarepaket ‚IIoT Libraries SL‘ enthält Bibliotheken zum Thema ‚IIoT Kommunikation‘ und Bibliotheken zum Lesen und Schreiben von Datenstrukturen.

Produktbeschreibung

Das Produkt ‚IIoT Libraries SL‘ enthält zahlreiche Bibliotheken zur Unterstützung unterschiedlicher Kommunikationsprotokolle (wie z.B. MQTT, HTTPS..) und enthält Werkzeuge zum Kodieren und Dekodieren von Datenstrukturen (wie z.B. JSON, XML..). Außerdem werden Bibliotheken mit Bausteinen zur direkten Kommunikation mit Cloud-Diensten einzelner Anbieter(AWS, Azure,...) bereitgestellt. Die im CODESYS Package enthaltenen Komponenten eignen sich hervorragend zur Kommunikation im IIoT Umfeld.

Bibliotheken zur Kommunikation

Web Client SL

Die Bibliothek ‚Web Client SL‘ enthält Funktionsbausteine zur Kommunikation mit einem Webserver via http bzw. https.

- Mit http-GET können Daten von einem Webserver abgefragt werden.
- Mit http-POST und http-PUT können Daten von der Steuerung zu einem Webserver übertragen werden.
- Mit http-DELETE können Daten auf einem Webserver gelöscht werden.
- Mit http-HEAD kann der http-Header abgefragt werden.

Der http-Header einer Anfrage kann dabei beliebig erweitert werden. Zusätzlich sind Funktionsbausteine zur http-Basic-Authentifizierung, Digest Access Authentifizierung und zur Authentifizierung über OAuth (Version 1a und 2) enthalten.

MQTT Client SL

MQTT (Message Queuing Telemetry Transport) ist ein offenes IoT Protokoll, das die Übertragung von Telemetriedaten in Form von Nachrichten zwischen Geräten ermöglicht. Die Kommunikation der Geräte erfolgt dabei immer über einen MQTT Broker (z.B. Mosquitto <https://mosquitto.org/>). Das Abonnieren und Verschicken von Nachrichten erfolgt auf Basis von ‚Topics‘. Ein ‚Topic‘ entspricht einem Pfad (z.B. device1/temperature). Das Abonnieren von Nachrichten erfolgt unter Angabe eines Topic-Filters, Wildcards sind hierbei auch erlaubt (+ für eine Ebene bzw. # für mehrere Ebenen) Das Format einer Nachricht ist nicht festgelegt, d.h. es kann ein JSON-String oder eine beliebige Datenstruktur übertragen werden.

Mit der Bibliothek ‚MQTT Client SL‘ können Nachrichten von einer CODESYS Steuerung an einen MQTT Broker geschickt werden und Nachrichten auf Basis von ‚Topics‘ abonniert werden.

Mail Service SL

Die Bibliothek enthält Funktionsbausteine zum Senden, Empfangen und Löschen von Emails über das SMTP- und POP3-Protokoll. Die Kommunikation mit dem Mailserver kann dabei verschlüsselt (TLS) oder unverschlüsselt erfolgen. Ein Beispielprojekt veranschaulicht die Verwendung der Funktionsbausteine.

SNMP Service SL

Die Bibliothek unterstützt folgende SNMP Funktionen:

- SNMP GET: Abfrage eines Wertes (Steuerung ist Manager)
- SNMP GET_NEXT: Abfrage von Tabellen (Steuerung ist Manager)
- SNMP Agent: Abfrage von Werten der Steuerung (Steuerung ist Agent)
- SNMP TRAP: Senden und Empfangen von TRAP/INFORM-Telegrammen
- SNMP SET: Setzen von Werten

Unterstützte SNMP Versionen

- SNMP V1
- SNMP V2c
- SNMP V3

SNTP Service SL

Die Bibliothek ‚SNTP Service SL‘ enthält Funktionsbausteine zum Aufbau von SNTP Client - und Serverapplikationen (SNTP V3, SNTP V4). Mit dem Funktionsbaustein SNTPGetUTCTime kann die UTC-Zeit von einem NTP/SNTP-Server abgefragt werden. Der Funktionsbaustein SNTPServer dient als einfacher SNTP-Server auf einer CODESYS Steuerung.

Folgende Funktionsbausteine (FBs) sind in der Bibliothek enthalten:

- SNTPGetUTCTime: FB zum Abfragen der Uhrzeit via SNTP (SNTP Client)
- SNTPServer: FB zum Verteilen der lokalen Uhrzeit via SNTP (SNTP Server)

AWS IoT Core Client SL

‚AWS IoT Core‘ ist eine verwaltende Cloud-Plattform von Amazon, mit der verbundene Geräte einfach und sicher mit Cloud-Anwendungen und anderen Geräten zusammenarbeiten können. Die Bibliothek ‚AWS IoT Core Client SL‘ stellt Funktionsbausteine zum Senden und Empfangen von Nachrichten zur Verfügung. Die Kommunikation erfolgt dabei verschlüsselt über das Protokoll MQTT. Nachrichten werden im AWS Umfeld typischerweise im JSON Format übertragen. Zum Parsen und Erzeugen von JSON Dateien kann die Bibliothek ‚JSON Utilities‘ verwendet werden.

Azure IoT Hub Client SL

Der Cloud-Dienst ‚Azure IoT Hub‘ von Microsoft ist ein Service zur direkten Anbindung von IoT-Geräten (weitere Informationen siehe <https://azure.microsoft.com/de-de/services/iot-hub/>). Die Bibliothek ‚Azure IoT Hub Client SL‘ stellt Funktionsbausteine zum Senden und Empfangen von Nachrichten zur Verfügung. Ein Beispielprojekt veranschaulicht die Verwendung der Bibliothek.

JSON Web Token SL

Die Bibliothek ‚JSON Web Token SL‘ enthält einen Funktionsbaustein zum Erstellen von JWT (JSON Web Token) auf der Steuerung. Dabei werden die Algorithmen HS256, HS384, HS512 und RS256 unterstützt.

Web Socket Client SL

Das WebSocket-Protokoll ermöglicht eine bidirektionale Kommunikation zwischen einem Client und einem Web Socket Server über das Internet. Die Kommunikation erfolgt dabei in der Regel über die TCP-Portnummer 443 (oder 80 bei ungesicherten Verbindungen). Die Bibliothek ‚Web Socket Client SL‘ stellt Funktionsbausteine zur Kommunikation über das WebSocket-Protokoll zur Verfügung.

OpenWeather Client SL

Mit der OpenWeather Client SL Library ist es möglich Wetterdaten von einem API-Server abzufragen, der von OpenWeather zur Verfügung gestellt wird.

Unterstützte Dienste sind:

- Abfrage des aktuellen Wetters für einen bestimmten Ort
- Abfrage einer stündlichen Wettervorhersage für die nächsten 48 Stunden für einen bestimmten Ort
- Abfrage von Ortsinformationen über
 - Stadtname
 - Postleitzahl
 - GPS-Koordinaten

Um die Library verwenden zu können, benötigen Sie einen API-Schlüssel von OpenWeather.

Bibliotheken zum Lesen und Schreiben von Datenstrukturen

CSV Utility SL

Die Bibliothek ‚CSV Utility SL‘ stellt Funktionsbausteine zum Lesen und Schreiben von CSV Dateien zur Verfügung. Die Bibliothek besteht aus den Komponenten CSVReader (Lesen von CSV Dateien) und CSVWriter (Schreiben von CSV Dateien). Sämtliche Funktionsbausteine können klassisch oder objektorientiert verwendet werden. In einem Beispielprojekt wird die Verwendung der Bibliothek ausführlich demonstriert.

INI File Utility SL

„INI File Utility SL“ ist eine Bibliothek zum Lesen und Schreiben von INI-Dateien. Die Bibliothek enthält Funktionsbausteine zum Lesen, Schreiben und Suchen von Werten in INI-Dateien. Die Verwendung der Funktionsbausteine wird in einem Beispielprojekt demonstriert.

JSON Utilities SL

Die Bibliothek ‚JSON Utilities SL‘ bietet folgende Funktionalitäten:

- Erstellen von komplexen JSON Dateien

- Lesen von JSON Dateien (UTF-8 und UTF-16)
- Lesen von JSON Bytearrays
- Schreiben von JSON Dateien (UTF-8 und UTF-16)
- Schreiben von JSON Bytearrays
- Suche nach Schlüssel, Werten, Kind-Elementen und Eltern-Elementen

Die Daten werden in einem Array von Strukturen gespeichert. Der Datenzugriff und die Suche nach Elementen erfolgt über Methoden des Funktionsbausteins `JSONData`. Das einfache Erstellen einer JSON-Datei wird über den Funktionsbaustein `JSONBuilder` und seine Methoden realisiert. Die Funktionsbausteine `JSONByteArrayReader` und `JSONFileReader` lesen Daten im JSON-Format ein und speichern die Werte im Funktionsbaustein `JSONData`. Das Schreiben von Dateien und Bytearrays erfolgt über die Funktionsbausteine `JSONFileWriter` und `JSONByteArrayWriter`.

In Verbindung mit der Bibliothek ‚Web Client SL‘ kann auf einfache Weise ein einfacher REST-Client implementiert werden.

XML Utility SL

Die Bibliothek ‚XML Utility SL‘ enthält Funktionsbausteine zum Lesen und Schreiben von XML-Dateien bzw. XML-Strings auf der Steuerung. Die einzelnen Elemente werden dabei in einem Array von Strukturen gespeichert. Darüber hinaus sind in der Bibliothek auch Funktionsbausteine zum Suchen von XML-Elementen enthalten. Die Verwendung der Funktionsbausteine wird anhand von zwei Beispielapplikationen demonstriert.

Wichtiger Hinweis zur Version 1.9.0.0

Der Firmenname der Bibliotheken wurde auf ‚CODESYS‘ geändert. Beim Upgrade auf Version 1.9.0.0 müssen die alten Bibliotheken (< 1.9.0.0) über den Bibliotheksmanager entfernt und die neuen Bibliotheken eingefügt werden.

Die Beispielprojekte werden ab Version 1.9.0.0 in den Ordner ‚IIoT Libraries SL‘ der CODESYS-Installation (CODESYS Root-Verzeichnis) installiert.

Die Bibliotheksdokumentation ist jetzt in der CODESYS Onlinehilfe integriert. Siehe <https://content.helpme-codesys.com/en/libs/index.html> Die CHM Dateien werden nicht mehr mit dem Package ausgeliefert.

Allgemeine Informationen

Lieferant:

CODESYS GmbH
 Memminger Straße 151
 87439 Kempten
 Deutschland

Support:

Technischer Support ist bei diesem Produkt nicht enthalten. Um technischen Support zu erhalten, erwerben Sie bitte ein CODESYS Support Ticket.

<https://support.codesys.com>

Artikelname:

IIoT Libraries SL

Artikelnummer:

2311000000

Vertrieb/Bezugsquelle:

CODESYS Store
<https://store.codesys.com>

Lieferumfang:

- CODESYS Package mit Bibliotheken und Beispielprojekten zum Thema IIoT
- SL Lizenz

Systemvoraussetzungen und Einschränkungen

Programmiersystem	CODESYS Development System V3.5.18.0
Laufzeitsystem	CODESYS Control V3.5.18.0
Unterstützte Plattformen/ Geräte	Hinweis: Verwenden Sie das Projekt Device Reader, um die von der Steuerung unterstützten Funktionen zu ermitteln.
Zusätzliche Anforderungen	Die Bibliotheken Web Client SL, MQTT Client SL, Mail Service SL, AWS IoT Core Client SL, Azure IoT Hub Client SL benötigen u.a. die Bibliotheken SysSocket2 3.5.16.0, Net Base Services, CmpCrypto, CmpTls 3.5.15.0 und CmpX509Cert. Die Bibliotheken CSV Utility SL, INI File Utility SL, JSON Utilities SL und XML Utility SL benötigen u.a. die Bibliothek SysFile.

Einschränkungen

Die Eingänge itfAsyncProperty und das AsyncProperty werden erst ab Net Base Services V3.5 SP17 Patch 4 unterstützt. In älteren Versionen kann es zu Deadlocks bei einem Reset der Steuerung kommen. In diesem Fall darf das AsyncProperty nicht verwendet werden.

Aufgrund eines Fehlers in der Bibliothek Net Base Service in der Version 3.5.19.0 können keine TLS Verbindungen aufgebaut werden. Bitte verwenden Sie eine andere Version der Bibliothek Net Base Services.

Lizenzierung

Einzelgerätelizenz: Die Lizenz kann auf der Steuerung, auf dem das CODESYS Laufzeitsystem installiert ist, verwendet werden.

Die Lizenzaktivierung erfolgt auf einem softwarebasierten Lizenz-Container (Soft-Container), der fest an die Steuerung gebunden ist. Alternativ kann die Lizenz auf einem CODESYS Key (USB-Dongle) hinterlegt werden. Durch Umstecken des CODESYS Keys kann die Lizenz auf einer anderen Steuerung genutzt werden.

Hinweis: Ohne Lizenz läuft die Software im Demomodus für 30 Minuten. Danach ist ein manueller Neustart erforderlich.

Erforderliches Zubehör

-

Bitte beachten Sie: Technische Änderungen, Druckfehler und Irrtümer vorbehalten. Es gilt der Inhalt der aktuellen Online-Version dieses Dokuments.

Erstellungsdatum: 29.04.2025