

# .NET Programmierung für industriennahe Anwendungen

Wintersemester 2022

(c) 2022

Roland J. Graf / Gerald Lochner

## Labor 03: Industrie 4.0, Assets und Digitaler Zwilling (Digital Twin)

### 1.1. Themen

- Vertiefung zu C#/.NET Programmierung
- Implementieren des Modbus Clients
- Erstellen eines digitalen Zwillings auf Basis von Asset Administration Shell (AAS)

### 1.2. Projektaufgaben

Im dritten Schritt bereiten wir die Anbindung an den Modbus Server und die Definition der digitalen Zwillinge vor. D

#### 1.2.1. Erstellung der digitalen Zwillinge

- Erstellung jeweils einen digitalen Zwillings für jede der **mind. 5 Maschinen**, die jeweils ein bestimmtes *Produkt/Bauteil* (z.B. Motor, Schraube, Kabelbaum, Schalter, Getriebe,...) herstellen.
- Erstellen Sie ein Submodel "*General Information*" für die Beschreibung der Machine (Hersteller, Produktbezeichnung, Logo des Herstellers, Produktionsjahr, ...)
- Erstellen Sie mind. ein weiteres Submodel "*Operations*" für die Darstellung von Prozessdaten der jeweiligen Maschine. Diese sollen mit dem im Modbus Server definierten **mindestens 5 Produktions-/Maschinenparameter bzw. Kenngrößen** übereinstimmen (z.B. die Anzahl der jeweils produzierten *Produkte/Bauteile*, die aktuelle Geschwindigkeit, die Anzahl der produzierten Bauteile pro Minute, Fehlerrate, Herstellungstoleranzen, Temperaturen, Wartungsintervalle, Status der Maschine [processing, failed, maintenance,...] usw. ...) übereinstimmen. Achten Sie darauf, das die Parameter semantisch korrekt beschrieben sein (Name, Beschreibung, Einheiten, Datentypen, Wertebereiche, Einheiten, ...). Operationen zum Starten und Stoppen der Maschine und eine beliebige weitere Operation (z. B. Umrüsten für ein neues Produkt).
- Verwenden Sie zur Erstellung AAS mit Hilfe des AASX Package Explorers. In Moodle Kurs gibt es dazu eine dedizierte Anleitung.
- Weiterführender Lesestoff

- <http://www.admin-shell-io.com/>
- Siehe PDFs von IDTA und ZVEI im Moodle Kurs

### 1.2.2. Erstellung eines Modbus Clients

- Erstellen Sie einen Modbus Client um alle Daten von den fiktiven Produktionsanlagen zu lesen und zu schreiben.
- Stellen Sie sicher, dass verschiedene Datentypen gelesen und geschrieben werden können und korrekt am Modbus Server ankommen.
- Erstellen Sie ein Command Line Utility zum Testen der Implementierung.
- Prüfen Sie auf Fehler bzw. Ausnahmesituationen. Stellen Sie dazu beispielsweise sicher, dass eine Verbindung zwischen Client und Server wiederhergestellt wird, sollte die TCP Verbindung abbrechen.
- Erstellen Sie für die Modbus Client Implementierung eine eigene Assembly, da dieser in Einheit 4 wiederverwendet wird.
- Einbindung der EasyModbusTCP Bibliothek  
⇒ siehe [easymodbustcp.net](http://easymodbustcp.net), [Modbus Specifications](#)



Die Einbindung des EasyModbusTCP Assemblies erfolgt innerhalb von Visual Studio über NuGet; zusätzlich muss auch noch das Assembly System.IO.Port eingebunden werden, das von EasyModbusTCP benötigt wird.

### 1.2.3. Allgemeine Anforderungen:

- Es gelten die gleichen Regeln wie für Aufgabe 02 für Entwicklung.
- Beschreiben Sie die Elemente für ihre digitalen Zwillinge mind. auf Englisch.
- Achten Sie auf eine durchgängige Benennung von Elementen.

## 1.3. Projektabgaben

- Abgabebedingungen und Formate: siehe Moodle-Kurs