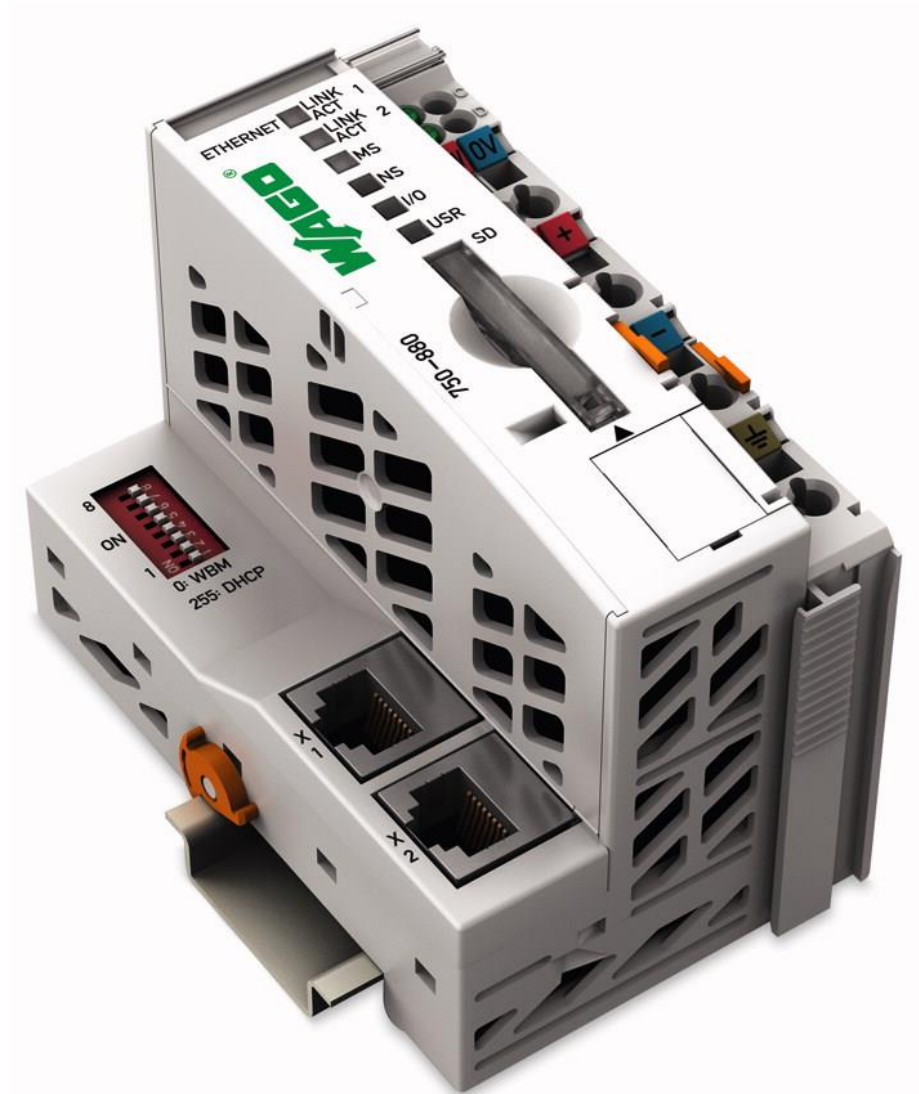


# Labor


## Grundlagen der Gebäudeautomation



BILDQUELLE: WAGO KONTAKTECHNIK GMBH & CO. KG

## Versuch 3

# *Programmierung mit Strukturierten Text sowie EnOcean-Anwendungen*

 <b>h_da</b> HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES <b>fbeit</b> FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK	<p style="text-align: center;"><b>Labor</b></p> <p style="text-align: center;">Grundlagen der Gebäudeautomation (GGA)</p> <p style="text-align: center;"><b>Versuch 3</b></p>	<p style="text-align: center;">Autor</p> <p style="text-align: center;"><b>Prof. Dr. Sven Rogalski</b></p>
--	---	--


## Lernziele

Die Studierenden sollen einerseits den Aufbau einfacher, modularer CoDeSys-Anwenderprogramme in der Programmiersprache Strukturierter Text festigen sowie die Verwendung des CoDeSys-Tasksystems und den Einsatz „Globaler Variablen“ vertiefen. Zusätzlich wird der Einsatz der EnOcean-Technologie in Kombination mit dem WAGO-I/O-System anhand ausgewählter Beispiele in Gebäuden trainiert.

## Allgemeine Hinweise

- Arbeiten Sie die Aufgaben dieses Versuchs vollständig durch und lösen Sie diese bereits zuhause!
- Die in den Aufgaben zu erstellenden Schaltlogiken/SPS-Programme sind handschriftlich auf dem Papier vorzubereiten und werden nach Eingabe in den jeweiligen CoDeSys-Editor mit den Laborgeräten getestet! Dokumentieren Sie Ihre Lösungen individuell, d.h. **jedes Gruppenmitglied hat eine Lösung!**
- Jede der Aufgaben ist in einem neuen CoDeSys-Projekt zu bearbeiten.
- Bevor Sie mit einer neuen Aufgabe beginnen, lassen Sie die Programmfunktionalität der vorangegangenen durch das Laborpersonal abnehmen, um Folgefehler zu vermeiden!
- Sämtliche Ergebnisse Ihrer Arbeit sind in einem von Ihnen anzufertigenden Laborbericht zu dokumentieren! Dafür steht Ihnen im Labor auch ein Drucker zur Verfügung, über den Sie Ihre in CoDeSys erstellten Schaltlogiken ausdrucken und danach handschriftlich kommentieren können.
- Der **Laborbericht** wird bewertet und berücksichtigt neben den dokumentierten Ergebnissen auch die Qualität der Versuchsvorbereitung und Versuchsdurchführung!

**Achtung: Eine fehlende oder mangelhafte Vorbereitung Ihrerseits führt unweigerlich zum Ausschluss von diesem Laborversuch!**

 <p>h_da HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES fbeit FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK</p>	<p style="text-align: center;"><b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b></p>	<p style="text-align: center;">Autor <b>Prof. Dr. Sven Rogalski</b></p>
--	---	---

## Aufgabe 1: Erstellung der FB "Temperaturkonverter" und "Spannungsauswertung" in der Programmiersprache ST


In dieser Aufgabe sollen die beiden Funktionsbausteine aus dem Versuch 2, "*Temperaturkonverter*" (Aufgabe 3) und "*Spannungsauswertung*" (vgl. Aufgabe 4) in der Programmiersprache Strukturierter Text (ST) umgesetzt werden. Danach sind diese innerhalb eines CoDeSys-Projektes in zwei verschiedenen Programmen, die jeweils in einer eigenen Task laufen, auszuführen.

### Aufgabe 1.1: Funktionsbaustein "Temperaturkonverter"

- Erzeugen Sie ein neues CoDeSys-Projekt mit einem Namen Ihrer Wahl auf Grundlage der bestehenden Kommunikationsparameter und der Hardwarekonfiguration "*Vorlage\_HardwareKonfig.csv*" (vgl. Aufgabe 1 vom Versuch 2).
- Legen Sie im Projekt einen neuen Funktionsbaustein namens "*FB\_Temp\_ST*" in der Programmiersprache **Strukturierter Text (ST)** an und realisieren Sie dort eine Schaltlogik gemäß der Aufgabe 3 aus dem Versuch 2!
- Löschen Sie den Baustein "*PLC\_PRG*" und erstellen Sie einen neuen Programm-Baustein namens "**Temperaturumrechnung**" zur Programmierung in ST. Fügen Sie dort dann Ihren neuen Funktionsbaustein "*FB\_Temp\_ST*" ein und verschalten Sie alle zum Funktionsbaustein gehörenden Ein- und Ausgänge sinnvoll!


### Aufgabe 1.2: Funktionsbaustein "Spannungsauswertung"

- Legen Sie im gleichen Projekt einen zweiten Funktionsbaustein namens "*FB\_VoltConvert\_ST*" in der Programmiersprache **Strukturierter Text (ST)** an und realisieren Sie dort eine Schaltlogik gemäß der Aufgabe 4 aus dem Versuch 2!
- Erstellen Sie einen zweiten Programm-Baustein namens "**Spannungsauswertung**" zur Programmierung in ST. Fügen Sie dort Ihren neuen Funktionsbaustein "*FB\_VoltConvert\_ST*" ein und verschalten Sie alle zum Funktionsbaustein gehörenden Ein- und Ausgänge sinnvoll!

 <p>h_da HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES fbeit FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK</p>	<p style="text-align: center;"><b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b></p>	<p style="text-align: center;">Autor <b>Prof. Dr. Sven Rogalski</b></p>
--	---	---

### Aufgabe 1.3: Taskkonfiguration

- a) Erstellen Sie eine neue Task mit dem Namen „*Temperatur*“, die alle 6 Sekunden ausgeführt werden soll und hängen Sie an diese Ihr Programm "*Temperaturumrechnung*" an.
- b) Erstellen Sie eine zweite Task mit dem Namen „*Spannung*“, die alle 2.000 Millisekunden ausgeführt werden soll und hängen Sie an diese Ihr Programm "*Spannungsauswertung*" an.
- c) Verbinden Sie das LLT-Experimentiermodell mit dem LN-Schulungsboard so, dass Sie Ihre beiden Programme ausführen und hieran testen können.
- d) Testen Sie die Programmfunktionalität und zeigen Sie das Ergebnis Ihrem Dozenten!
- e) Exportieren Sie Ihre beiden neuen Funktionsbausteine in Ihren Ordner "**Eigene Bausteine**"!
- f) Weshalb benötigt Ihr CoDeSys-Projekt jetzt keinen Baustein "*PLC\_PRG*" mehr?
- g) In welcher Frequenz wird der Temperaturwert und in welcher Frequenz der Spannungswert aktualisiert? Was passiert wenn die beiden Programme "*Temperaturumrechnung*" und "*Spannungsauswertung*" gleichzeitig aufgerufen werden? Begründen Sie Ihre Antwort!
- h) Warum blinkt die Lampe auf dem LLT-Experimentiermodell beim Erreichen des Alarmwerts nicht in der verlangten Frequenz von 500 ms? Lösen Sie dieses Problem!
- i) Dokumentieren Sie alle Ergebnisse zur Aufgabe 1 in Ihrem Laborbericht!

 <p><b>h_da</b> HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES <b>fbeit</b> FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK</p>	<p align="center"><b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b></p>	<p align="center">Autor <b>Prof. Dr. Sven Rogalski</b></p>
--	--	--


## Aufgabe 2: Einfache Programmierbeispiele zur EnOcean-WAGO-Kommunikation

In dieser Aufgabe, die sich in verschiedene Teilbereiche untergliedert, wird anhand kleinerer Programmierbeispiele die Kommunikation zwischen dem WAGO-Controller und unterschiedlichen Arten von EnOcean-Geräten vertieft. Hierbei ist auf verschiedene Bausteine der von WAGO angebotenen EnOcean-Bibliothek zurückzugreifen.

### Aufgabe 2.1: Kommunikations- und Detektorbaustein anlegen

Als Grundvoraussetzung für alle nachfolgenden Aufgaben, soll hier in einem ersten Schritt die Kommunikation zwischen dem WAGO-Controller und der *Funkempfänger EnOcean Busklemme (750-642)* auf Ihrem LN-Schulungsboard hergestellt werden. Außerdem ist im CoDeSys-Programm eine Funktionalität zu definieren, durch welche die Geräte-/Sender-ID eines zu verwendenden EnOcean-Sensors erkannt wird.

- Erstellen Sie hierzu ein neues Projekt unter einem geeigneten Namen und wählen Sie für die Programmierung von „PLC\_PRG“ eine Sprache Ihrer Wahl aus. Fügen Sie zudem einen neuen Programm-Baustein in Ihr Projekt ein (**Programmiersprache CFC**) und geben Sie diesem den Namen „**EnOcean\_PRG**“.
- Fügen Sie in das Programm „*EnOcean\_PRG*“ den EnOcean-Kommunikationsbaustein „**FbEnOceanReceive**“ ein, den Sie in der EnOcean-Bibliothek „*EnOcean\_05.lib*“ finden.
  - Geben Sie dem Baustein einen geeigneten Instanznamen.
  - Legen Sie am Bausteineingang „*bModule\_750\_642*“ fest, dass sich Ihr Kommunikationsbaustein auf die erste EnOcean-Busklemme bezieht, indem Sie als Bytenummer die Zahl „1“ eintragen.
  - Erzeugen Sie eine Variable mit dem Namen „*typKlemme\_1*“ vom Datentyp „*typEnOcean*“ und verbinden Sie diese mit dem gleichnamigen Bausteinausgang.
- Erklären Sie, wofür die Variable „*typKlemme\_1*“ vom Typ „*typEnOcean*“ benötigt wird!
- Fügen Sie in Ihr Programm „*EnOcean\_PRG*“ jetzt einen sog. EnOcean-Detektorbaustein vom Typ „**FbShow\_ID\_ByClick**“ hinzu, den Sie ebenfalls in der EnOcean-Bibliothek „*EnOcean\_05.lib*“ finden.
  - Geben Sie dem Baustein einen geeigneten Instanznamen und ordnen Sie Ihre vom Typ „*typEnOcean*“ erzeugte Variable dem Bausteineingang „*typEnOcean*“ zu.
  - Legen Sie am Bausteineingang „*bRF\_Type*“ durch Eingabe der RORG-Nummer fest, dass Sie Datentelegramme vom Typ *Repeated Switch Communication (RORG=F6)* identifizieren wollen. **Hinweis:** Schauen Sie ggf. in der Bausteinbeschreibung nach!


 <p><b>h_da</b> HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES <b>fbeit</b> FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK</p>	<p style="text-align: center;"><b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b></p>	<p style="text-align: center;">Autor <b>Prof. Dr. Sven Rogalski</b></p>
--	---	---

- Bestimmen Sie für Ihren Baustein, dass die Sender-ID erst dann angezeigt wird, wenn der Sensor mindesten zweimal hintereinander kurzzeitig aktiviert wurde und definieren Sie eine Variable, über die Sie sich die ID eines zu identifizierenden EnOcean kompatiblen Gerätes anzeigen lassen.
- e) Erklären Sie was RORG bedeutet und worin der Unterschied zu ORG besteht!
- f) Welche Eingänge des Bausteins „*FbEnoceanReceive*“ sind mit einem Standardwert belegt und wie lautet dieser?
- g) Fügen Sie das Programm „*EnOcean\_PRG*“ in den Programm-Baustein „*PLC\_PRG*“ zur Ausführung ein und testen Sie dieses. Dokumentieren Sie alle Ergebnisse im Laborbericht!

## Aufgabe 2.2: EnOcean-Schalter

Nachdem Sie den Kommunikations- und der Detektorbaustein in Ihr Programm eingefügt haben, soll die Geräte-ID des an Ihrem Laborarbeitsplatz befindlichen **EnOcean-Schalters** (Marke: *OPUS*) erkannt werden. Außerdem ist hierfür ein geeigneter Funktionsbaustein zu definieren, um mit diesem Schalter Lampe und Lüfter auf dem LLT-Experimentiermodell schalten zu können.

- a) Ermitteln Sie die Geräte-ID des EnOcean-Schalters (Marke: *OPUS*) über Ihren Detektorbaustein und notieren Sie sich diese.
- b) Fügen Sie in Ihr Programm „*EnOcean\_PRG*“ den Funktionsbaustein „***FbF602xx\_RockerSwitch\_2\_Rocker***“ ein, den Sie in der EnOcean-Bibliothek „*Enocean\_05.lib*“ unter „*Enocean Equipment Profiles EEP*“ (*RORG=F6*) finden.
  - Geben Sie dem Baustein einen geeigneten Instanznamen.
  - Weisen Sie den Bausteineingängen *typEnocean*, *bTYPE* und *dwID* die richtigen Parameter zu und legen Sie einen geeigneten Wert für *tTimeOut* fest.  
**Hinweis:** Beachten Sie, dass *brf\_Type* und *bTYPE* unterschiedliche Parameter haben.
  - Verschalten Sie die Bausteinausgänge so, um mit dem ersten Taster die Lampe und mit dem zweiten Taster den Lüfter des LLT-Experimentiermodells anzusteuern.
- c) Verbinden Sie Ihr LN-Schulungsboard mit dem LLT-Experimentiermodell, so dass die Lampe und der Lüfter auf dem LLT-Experimentiermodell über den EnOcean-Schalter angesteuert werden können. Zeigen Sie das Ergebnis Ihrem Dozenten!
- d) Worin liegt der Unterschied zwischen den beiden Datentypen *brf\_Type* und *bTYPE*?
- e) Was bewirkt die Eingabe „*T#2s*“ am Bausteineingang *tTimeOut*?
- f) Welchen Funktionsbaustein müssten Sie in Ihr Programm einfügen, wenn Sie einen EnOcean-Schalter mit vier Wippen/Tastern benutzen würden?
- g) Dokumentieren Sie Ihre Ergebnisse im Laborbericht!

 <b>h_da</b> HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES <b>fbeit</b> FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK	<b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b>	Autor <b>Prof. Dr. Sven Rogalski</b>
--	--	---

### Aufgabe 2.3: EnOcean-Fensterkontakt

In dieser Aufgabe soll als erstes die Geräte-ID des an Ihrem Laborarbeitsplatz befindlichen **EnOcean-Tür-/Fensterkontakts** (Marke: *OPUS*) erkannt werden. Danach ist über diesen in Abhängigkeit zum zuvor programmierten Schalter die Lampe auf dem LLT-Experimentiermodell anzusteuern.

**Hinweis:** Sofern der Fensterkontakt nicht ansprechend reagiert, kann es daran liegen, dass er über seine Solarzellen nicht ausreichend geladen wurde (14 Stunden Betriebszeit bedürfen 2-3 Stunden Helligkeit von 100 lx). Achten Sie daher bei Ihrer Versuchsdurchführung darauf, dass die Solarzellen des Fensterkontakts nicht verdeckt/verschattet sind.

- a) Ermitteln Sie die Geräte-ID des EnOcean-Tür-/Fensterkontakts (Marke: *OPUS*) über Ihren Detektorbaustein und notieren Sie sich diese.


**Hinweis:** Beachten Sie hierbei, dass es sich beim Telegrammtyp des Fensterkontakts (*bRF\_Type*) um eine *1 Byte Communication (RORG=D5)* handelt.

- b) Fügen Sie in Ihr Programm „*EnOcean\_PRG*“ den Funktionsbaustein „***FbD500xx\_ContactsAndSwitches***“ ein, den Sie in der EnOcean-Bibliothek „*Enocean\_05.lib*“ unter „*Enocean Equipment Profiles EEP*“ finden.

- Geben Sie dem Baustein einen geeigneten Instanznamen.
- Weisen Sie den Bausteineingängen *typEnocean*, *bTYPE* und *dwID* die richtigen Parameter zu und legen Sie einen geeigneten Wert für *tTimeOut* fest.

**Hinweis:** Beachten Sie, dass *bRF\_Type* und *bTYPE* unterschiedliche Parameter haben.

- c) Verschalten Sie die Bausteinausgänge so, dass die Lampe des LLT-Experimentiermodells nur dann leuchtet, wenn sie durch den EnOcean-Schalter zuvor angeschaltet wurde **UND** der Fensterkontakt das Signal TRUE liefert.
- d) Testen Sie die Funktionalität Ihres Programms und zeigen Sie das Ergebnis Ihrem Dozenten!
- e) Dokumentieren Sie Ihre Ergebnisse im Laborbericht!


 <b>h_da</b> HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES <b>fbeit</b> FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK	<b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b>	Autor <b>Prof. Dr. Sven Rogalski</b>
--	--	---

## Aufgabe 2.4: EnOcean-Fenstergriff

Aufgabe ist es, als erstes die Geräte-ID des an Ihrem Laborarbeitsplatz befindlichen **EnOcean-Fenstergriffs** (Marke: *HOPPE*) zu erkennen. Danach muss über diesen, in Abhängigkeit zum zuvor programmierten Schalter, der Lüfter auf dem LLT-Experimentiermodell angesteuert werden.

- Ermitteln Sie die Geräte-ID des EnOcean-Fenstergriffs (Marke: *OPUS*) über Ihren Detektorbaustein und notieren Sie sich diese.  
**Hinweis:** Beachten Sie hierbei, dass es sich beim Telegrammtyp des Fenstergriffs (*bRF\_Type*) um eine *Repeated Switch Communication* (*RORG=F6*) handelt.
- Fügen Sie in Ihr Programm „*EnOcean\_PRG*“ den Funktionsbaustein „**FbF610xx\_MechanicalHandle**“ ein, den Sie in der EnOcean-Bibliothek „*Enocean\_05.lib*“ unter „*Enocean Equipment Profiles EEP*“ finden.
  - Geben Sie dem Baustein einen geeigneten Instanznamen.
  - Weisen Sie den Bausteineingängen *typEnocean*, *bTYPE* und *dwID* die richtigen Parameter zu. **Hinweis:** Beachten Sie, dass *bRF\_Type* und *bTYPE* unterschiedliche Parameter haben.
- Verschalten Sie die BausteinAusgänge so, dass der Lüfter des LLT-Experimentiermodells sich nur dann dreht, wenn er über den EnOcean-Schalter zuvor angeschaltet wurde **UND** vom Fenstergriff ein TRUE-Signal für die Griffstellung „offen“ **ODER** „gekippt“ geliefert wird.
- Testen Sie die Funktionalität Ihres Programms und zeigen Sie das Ergebnis Ihrem Dozenten!
- Dokumentieren Sie Ihre Ergebnisse im Laborbericht!



 <p><b>h_da</b> HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES <b>fbeit</b> FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK</p>	<p style="text-align: center;"><b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b></p>	<p style="text-align: center;">Autor <b>Prof. Dr. Sven Rogalski</b></p>
--	---	---


## Aufgabe 2.5: EnOcean-Raumtemperaturregler

Es muss als erstes die Geräte-ID des an Ihrem Laborarbeitsplatz befindlichen **EnOcean-Temperaturreglers mit Handrad** erkannt werden. Anschließend ist über diesen der thermische Stellantrieb anzusteuern.

- a) Speichern Sie Ihr Projekt unter einem neuen Namen und löschen Sie alle im Programm-Baustein „EnOcean\_PRG“ eingefügten Funktionsbausteine, bis auf den Kommunikations- und den Detektorbaustein.
- b) Ermitteln und notieren Sie die Geräte-ID des EnOcean-Temperaturreglers, unter Verwendung des hierfür an Ihrem Laborarbeitsplatz befindlichen Magneten sowie mit Hilfe Ihres Detektorbausteins.

**Hinweis:** Beachten Sie hierbei, dass es sich beim Telegrammtyp des Temperaturreglers (*bRF\_Type*) um eine *4 Byte Communication (RORG=A5)* handelt.

- c) Fügen Sie in Ihr Programm „EnOcean\_PRG“ den Funktionsbaustein „**FbA510xx\_RoomOperatingPanel**“ ein, den Sie in der EnOcean-Bibliothek „EnOcean\_05.lib“ unter „EnOcean Equipment Profiles EEP“ finden.
  - Geben Sie dem Baustein einen geeigneten Instanznamen.
  - Weisen Sie den Bausteineingängen *typEnOcean*, *bTYPE* und *dwID* die richtigen Werte zu.  
**Hinweis:** Beachten Sie, dass *bRF\_Type* und *bTYPE* unterschiedliche Parameter haben.
  - Legen Sie fest, dass der regelbare Stellbereich für die Temperatur bei +/- 4°C liegen soll.
  - Definieren Sie für die Bausteinausgänge „*rTemperature*“ und „*rSetpointCorrection*“ geeignete Variablen. Legen Sie außerdem am Bausteinausgang „xDB0\_Bit0“ eine Variable namens „*xlch\_bin\_aktiv*“ fest, mit der Sie überprüfen können, ob der Temperaturregler mit Ihrem Controller kommuniziert.
- d) Realisieren Sie eine Schaltlogik, die bei einer Verstellung des „Setpoint“ in eine negative Richtung den thermischen Stellantrieb an Ihrem Laborarbeitsplatz öffnet, wohingegen der bei positiver oder keiner Verstellung des „Setpoint“ das Ventil geschlossen wird/bleibt.
- e) Testen Sie die Funktionalität Ihres Programms und zeigen Sie das Ergebnis Ihrem Dozenten!
- f) Um welche Art der Raumklimatisierung könnte es sich bei Ihrer realisierten Schaltlogik handeln, wenn der Stellantrieb öffnet?
- g) Was bewirkt der Bausteineingang „*rMaxSetpointCorrection*“?
- h) Dokumentieren Sie Ihre Ergebnisse im Laborbericht!

 <p><b>h_da</b> HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES <b>fbeit</b> FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK</p>	<p align="center"><b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b></p>	<p align="center">Autor <b>Prof. Dr. Sven Rogalski</b></p>
--	--	--

## Aufgabe 2.6: EnOcean-Helligkeitssensor

Zunächst erfolgt das Erkennen der Geräte-ID des an Ihrem Laborarbeitsplatz befindlichen **EnOcean-Helligkeitssensors**. Im Anschluss daran ist über diesen die Lampe auf dem LLT-Experimentiermodell so anzusteuern, dass sie einschaltet, wenn ein Helligkeitswert von 600 Lux unterschritten wurde.

- a) Ermitteln Sie die Geräte-ID EnOcean-Helligkeitssensor über Ihren Detektorbaustein und notieren Sie sich diese.


**Hinweis:** Beachten Sie hierbei, dass es sich beim Telegrammtyp des Temperaturreglers (*bRF\_Type*) um eine *4 Byte Communication (RORG=A5)* handelt.

- b) Fügen Sie in Ihr Programm „*EnOcean\_PRG*“ den Funktionsbaustein „**FbA506xx\_LightSensor**“ ein, den Sie in der EnOcean-Bibliothek „*Enocean\_05.lib*“ unter „*Enocean Equipment Profiles EEP*“ finden.

- Geben Sie dem Baustein einen geeigneten Instanznamen.
- Weisen Sie den Bausteineingängen *typEnocean*, *bTYPE* und *dwID* die richtigen Parameter zu und legen Sie einen geeigneten Wert für *tTimeOut* fest.

**Hinweis:** Berücksichtigen Sie, bei der Bestimmung des *bTYPE*-Parameters, dass Ihr Helligkeitssensor einen Arbeitsbereich von 300 bis 30.000 Lux hat.

- c) Verschalten Sie den Bausteinausgang „*rIllumination*“, so dass in Abhängigkeit von der Helligkeit am Sensor die Lampe auf dem LLT-Experimentiermodells angesteuert wird.
- d) Testen Sie die Funktionalität Ihres Programms und zeigen Sie das Ergebnis Ihrem Dozenten!
- e) Was für einen Arbeitsbereich haben EnOcean-Helligkeitssensoren mit einem *bTYPE* von *16#02*?
- f) Dokumentieren Sie Ihre Ergebnisse im Laborbericht!

 <p>h_da HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES fbeit FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK</p>	<p style="text-align: center;"><b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b></p>	<p style="text-align: center;">Autor <b>Prof. Dr. Sven Rogalski</b></p>
--	---	---

## Aufgabe 3: Raumautomatisierung mittels EnOcean


Diese Aufgabe dient dem Aufbau und der Simulation verschiedener Szenarien innerhalb der Raumautomatisierung.

**Achtung:** Die Intervallzeiten bei der Definition der verschiedenen Tasks müssen zwischen **10-30 Millisekunden** liegen, da sonst die Gefahr besteht, dass die Funkt-Telegramme der EnOcean-Sensoren nicht richtig erkannt werden!

### Aufgabe 3.1: Wohnzimmer

In diesem Raum wird der thermische Stellantrieb einer Fußbodenheizung über eine EnOcean-Raumtemperaturregelung angesteuert, ähnlich der Aufgabe 2.5. Hierbei ist jedoch zwischen Sommer- und Winterbetrieb der Heizungssteuerung zu unterscheiden. Befindet sich die Heizungsanlage im **Winterbetrieb**, dann darf das Ventil nur dann öffnen, wenn die zulässige Temperaturdifferenz überschritten ist und die Stellung des EnOcean-Fenstergriffs einer Balkontür nicht „offen“ oder „gekippt“ ist. Im **Sommerbetrieb** soll der thermische Stellantrieb jedoch unabhängig von der Temperaturdifferenz und der Fenstergriffstellung geschlossen bleiben, allerdings nur solange bis die Außentemperatur nicht unter 5°C fällt. Ist dies der Fall, dann muss sich die Automatisierungslogik der Heizung nach dem Modus des Winterbetriebs richten.


- a) Erstellen Sie ein neues Projekt unter einem geeigneten Namen und legen Sie einen Programm-Baustein namens „*Wohnzimmer\_PRG*“ an!
- b) Schreiben Sie nun ein Programm in der Programmiersprache Ihrer Wahl, durch das der oben beschriebene Steuerungsablauf realisiert wird. Beachten Sie dabei:
  - Bei Verstellung des „Setpoint“ über den EnOcean-Temperaturreger in eine positive Richtung soll der thermische Stellantrieb an Ihrem Laborarbeitsplatz öffnen, wohingegen der bei negativer oder keiner Verstellung des „Setpoint“ das Ventil geschlossen wird/bleibt.
  - Die Außentemperatur soll in diesem Beispiel den vom **Potenzimeter** (LN-Schulungsboard) gelieferten Spannungswert repräsentieren. Überlegen Sie sich eine Logik, wie Sie diesen Spannungswert in einen sinnvollen Wertebereich zur Außentemperaturdarstellung umwandeln. Setzen Sie die Logik in Form einer eigenen **Funktion** um, die Sie in Ihr Programm einfügen.
  - Das Umschalten zwischen Winter- und Sommerbetrieb soll über den *Schalter S1* am LN-Schulungsboard erfolgen. Ist  $S1 = TRUE$ , so steht dies für den Winterbetrieb der Heizung.
- c) Erstellen Sie eine neue **Task** mit dem Namen „*Wohnung*“ und legen Sie selbst ein Intervall (zwischen 10-30 ms) für dessen Ausführung sowie dessen Priorität fest.
- d) Testen Sie die Funktionalität Ihres Programms und zeigen Sie das Ergebnis Ihrem Dozenten!
- e) Dokumentieren Sie Ihre Ergebnisse im Laborbericht!

 <p><b>h_da</b> HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES <b>fbeit</b> FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK</p>	<p align="center"><b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b></p>	<p align="center">Autor <b>Prof. Dr. Sven Rogalski</b></p>
--	--	--

### Aufgabe 3.2: Wohnküche

In der Wohnküche wird ein EnOcean-Fensterkontaktschalter verwendet, um bei gleichzeitigem Betrieb der Dunstabzugshaube und einer raumluftunabhängigen Feuerstätte (z.B. Ofen, Kamin, Gastherme, Kohleheizung) das Entstehen eines Unterdrucks zu verhindern, durch den giftige Gase wie z.B. Kohlenmonoxid aus der Feuerstätte in den Raum gesaugt werden könnten. Der Motor der Dunstabzugshaube darf nur bei geöffnetem Küchenfenster laufen oder bei offener bzw. angekippter Balkontür im Wohnzimmer, um somit die Frischluftzufuhr zu gewährleisten.

- a) Erstellen Sie einen Programm-Baustein namens „*Kueche\_PRG*“ in Ihrem bestehenden Projekt und schreiben Sie ein Programm in der Programmiersprache Ihrer Wahl, durch das der oben beschriebene Steuerungsablauf realisiert wird. Beachten Sie dabei:
  - Der Motor der Dunstabzugshaube wird über den Lüfter des LLT-Experimentiermodells simuliert. Er soll durch einen Taster des EnOcean-Schalters ein-/ausgeschaltet werden, unter Berücksichtigung des vom Fenster/von der Balkontür gemeldeten Zustands!
  - Für die Kommunikation zwischen zwei Programmen, wie im Fall des EnOcean-Fenstergriffs, benötigen Sie **globale Variablen!**
- b) Fügen Sie Ihren Programmbaustein „*Kueche\_PRG*“ zu Ihrer bestehenden Task „*Wohnung*“ hinzu (vgl. Aufgabe 3.1c).
- c) Testen Sie die Funktionalität Ihres Programms und zeigen Sie das Ergebnis Ihrem Dozenten!
- d) Wofür werden globale Variablen benötigt und worin unterscheiden Sie sich zu Systemvariablen?
- e) Dokumentieren Sie Ihre Ergebnisse im Laborbericht!

 <p><b>h_da</b> HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES <b>fbeit</b> FACHBEREICH ELEKTROTECHNIK UND INFORMATIONSTECHNIK</p>	<p align="center"><b>Labor</b> Grundlagen der Gebäudeautomation (GGA) <b>Versuch 3</b></p>	<p align="center">Autor <b>Prof. Dr. Sven Rogalski</b></p>
--	--	--

### Aufgabe 3.3: Außenbeleuchtung Balkon

Der zur Wohnung gehörende Balkon soll immer dann beleuchtet werden, wenn die Außenbeleuchtung eingeschaltet wurde und der EnOcean-Helligkeitssensor, infolge eines zu geringen Lux-Wertes, das Einschaltsignal freigibt.

- a) Erstellen Sie einen Programm-Baustein namens „*Balkon\_PRG*“ in Ihrem bestehenden Projekt und schreiben Sie ein Programm in der Programmiersprache Ihrer Wahl, durch das der oben beschriebene Steuerungsablauf realisiert wird. Beachten Sie dabei:
  - Die Balkonbeleuchtung soll durch die Lampe auf dem LLT-Experimentiermodell simuliert werden und bei einem Helligkeitswert von unter 500 Lux einschalten.
  - Das Ein-/Ausschalten der Beleuchtung muss über den noch nicht belegten Taster des EnOcean-Schalters an Ihrem Laborarbeitsplatz erfolgen!
- b) Fügen Sie Ihren Programmbaustein „*Balkon\_PRG*“ zu Ihrer bestehenden Task „Wohnung“ hinzu (vgl. Aufgabe 3.1c).
- c) Testen Sie die Funktionalität Ihres Programms und zeigen Sie das Ergebnis Ihrem Dozenten!
- d) Begründen Sie, warum Sie die Vergabe Ihrer Prioritäten und Intervallzeiten zur Ausführung der drei Task in Ihrem Projekt so gewählt haben!
- e) Erklären Sie, wie man die drei Programm-Bausteine *Wohnzimmer\_PRG*, *Kueche\_PRG* und *Balkon\_PRG* zur Ausführung bringen kann, ohne sie an alle an die Task „Wohnung“ anzuhängen!
- f) Dokumentieren Sie Ihre Ergebnisse im Laborbericht!