

# Bedienungsanleitung



## Inhalt

Einleitung.....	2
Modus Lesen und Schreiben .....	3
Controlbyte.....	3
Datensatzauswahl .....	3
Alle Codes im Überblick.....	4
Beispiel Rezeptverwaltung .....	5
Rezeptur in Datenbank speichern .....	7
Rezeptur aus Datenbank lesen.....	8
Rezeptur löschen.....	9
Anzeige Datenbaustein .....	10
WhereClause – Datenbaustein.....	12
Modus Ringspeicher .....	14
Modus QDAS .....	15
Kontakt.....	16

## Einleitung

**HSDBASE** wurde konzipiert um es für jeden Anwender möglich zu machen, Daten aus S7-Steuerungen mitzuschreiben. Auf die einfache Anwendbarkeit wurde bei der Entwicklung dieser Software das Hauptaugenmerk gelegt.

Zur schnellen Einarbeitung gibt es im Programm eine integrierte Popup-Hilfe, die direkt bei der Verwendung von **HSDBASE** jede Schaltfläche erklärt.

Alle grundlegenden Informationen zu **HSDBASE** und den Modi zum Datenloggen (*Zyklisch, Trigger, Änderung, Analyzer*) sind im Programm in der Popup-Hilfe bereits enthalten.

Diese Kurzanleitung beschreibt deshalb lediglich die Modi *Lesen und Schreiben, Ringspeicher*, sowie die Datenerfassung im *QDAS Datenformat*.

Weitere Informationen finden Sie auf unserer Homepage unter FAQ und **HSDBASE** Beispiele.

## Modus Lesen und Schreiben

### Hinweis:

Dieser Modus ist nicht zum Datenloggen gedacht, sondern nur zum Austausch von Daten in beide Richtungen.

Verwenden Sie zum Datenloggen, besonders bei schnellen Aufzeichnungen, einen der anderen Modi, da hier deutlich höhere Datenraten erzielt werden können.

### Controlbyte

Alle Lese- und Schreibaufträge werden von der Steuerung über ein Controlbyte angestoßen.

Sollen Daten geschrieben bzw. gelesen werden, muss die Steuerung das entsprechende Kommando in dieses Controlbyte schreiben. **HSDBASE** liest das Controlbyte zyklisch entsprechend der eingestellten Zeit. Wird ein Wert größer 100 gelesen, arbeitet **HSDBASE** dieses Kommando ab und schreibt anschließend als Quittierung einen Antwortcode (kleiner 100) in das Controlbyte.

### Datensatzauswahl

In **HSDBASE** befindet sich in der Variablen-tabelle eine Spalte *Wert ändern*. Nur die angewählten Werte in dieser Spalte werden von **HSDBASE** in der Steuerung überschrieben. Die nicht angewählten Variablen werden zur Auswahl der Datensätze verwendet.

Um zu bestimmen, welcher Datensatz aus der Datenbank gelesen (bzw. aktualisiert) werden soll, muss die Auswahlvariable in **HSDBASE** unter den Variablen angelegt werden und das Häkchen in der Spalte *Wert ändern* muss **abgewählt** sein. Der Datensatz kann auch anhand einer Kombination von mehreren Variablen bestimmt werden. Vor jeder Schreib- / Leseoperation werden die Werte aus der Steuerung ausgelesen. Dann werden nur die Datensätze bearbeitet, bei denen die Auswahlvariablen in Steuerung und Datenbank übereinstimmen.

Werden in der Datenbank mehrere Datensätze ausgewählt, so werden alle diese Datensätze aktualisiert/gelöscht. Bei einem Lesebefehl werden die Werte des ersten ausgewählten Datensatzes verwendet.

Zum besseren Verständnis wird empfohlen sich das folgende Beispiel in diesem Handbuch anzuschauen.

**Alle Codes im Überblick**

<b>Kommandocodes</b>	
101	Aus Datenbank in SPS (Datensatz bleibt erhalten) - z.B. Rezeptur laden
102	Aus Datenbank in SPS (Datensatz wird gelöscht) - z.B. Auftrag laden
103	Abfrage Datensatz vorhanden - z.B. für Abfrage ob überschrieben werden soll
111	Aus SPS in Datenbank (Überschreiben, wenn vorhanden) - z.B. Rezeptur speichern
112	Aus SPS in Datenbank (Doppelt anlegen, wenn vorhanden) - z.B. Auftrag anlegen
113	Datensatz löschen - z.B. Rezept oder Auftrag löschen
121	AnzeigeDB mit verfügbaren Rezepten/Aufträgen füllen
<b>Antwortcodes</b>	
1	Daten wurden in SPS geschrieben
2	Daten wurden in SPS geschrieben und Datensatz wurde gelöscht
3	Datensatz wurde erstellt
4	Datensatz wurde aktualisiert
5	Datensatz wurde gelöscht
6	Datensatz vorhanden
7	AnzeigeDB wurde gefüllt
<b>Fehlercodes</b>	
11	Unbekannter Befehl
12	Fehler Datensatz nicht vorhanden
13	Fehler beim Lesen aus Datenbank
14	Fehler Datenbank/Tabelle/Datei nicht gefunden
15	Fehler keine Kopfzeile vorhanden
16	Fehler Anzahl Spalten Kopfzeile falsch
17	Fehler Anzahl Spalten Datenzeile falsch
18	Fehler beim Schreiben in Datenbank/Tabelle/Datei
19	Fehler beim Konvertieren / Datensatz wurde nicht komplett geschrieben
20	AnzeigeDB nicht vorhanden
21	AnzeigeDB zu klein
22	AnzeigeDB nicht angewählt
23	WhereClauseDB nicht vorhanden
24	WhereClauseDB zu klein

## Beispiel Rezeptverwaltung

Folgendes Beispiel zeigt eine einfache Rezeptur mit 3 Parametern. Dieses Beispiel lässt sich einfach nachstellen um alle Kommandos zu testen.

Legen Sie folgenden Datenbaustein an:

Name	Data type	Offset	Start value	Retain	Accessible f...	Write...	Visible in ...	Setpoint	Supervis...	Comment
Static										
Controlbyte	Byte	0.0	16#0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
RecipeName	String[10]	2.0	"	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
Parameter1	Bool	14.0	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
Parameter2	Int	16.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
Parameter3	Real	18.0	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

Legen Sie in **HSDBASE** eine neue Tabelle an. Stellen Sie als Modus *Lesen und Schreiben* ein. Anschließend geben sie die Adresse des Controlbytes ein:

**Capture**

Mode: Read and Write  ID column

min cycle(ms): 500 ID name: Id

Control byte address: **DB2.DBB0** Control byte status: 0  DisplayDB

**Save type**

Save as: MySQL Write new table every: Once

Table name: Recipe

Server: localhost User-ID: root

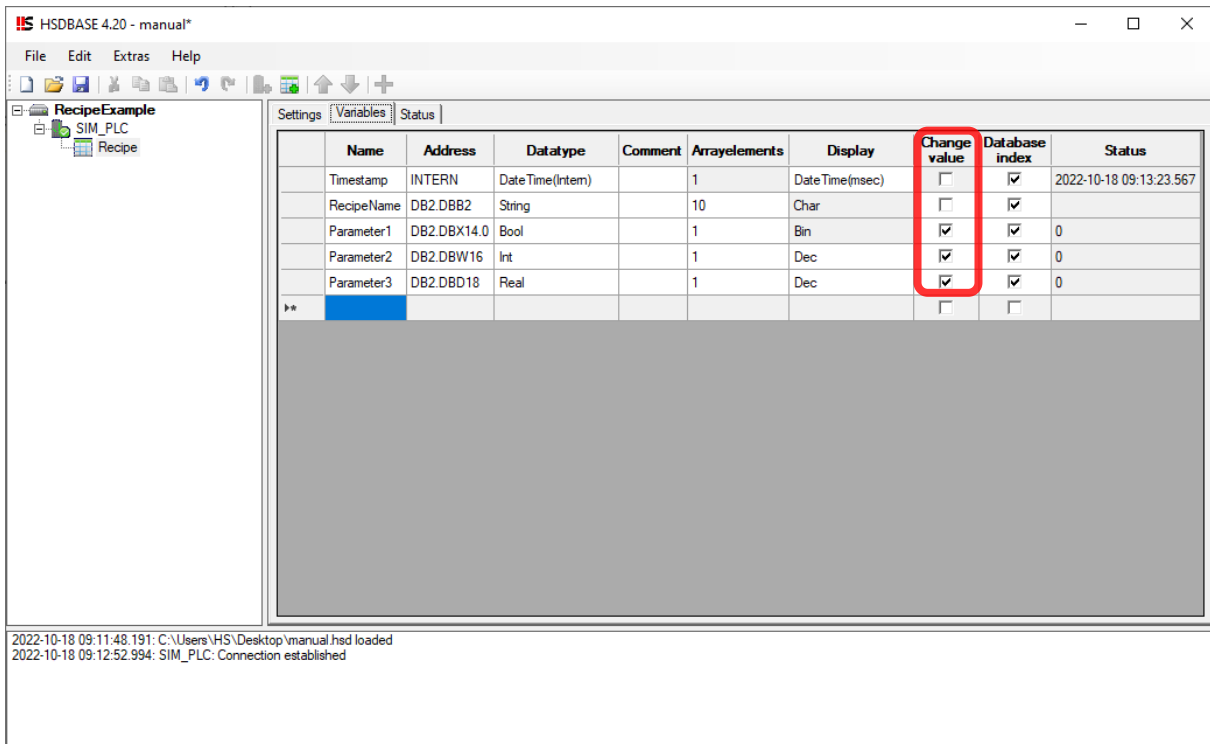
Database: SIM\_PL\_C Password:

alternative storage location

Write new database/folder depending on variable  Write new table depending on variable

2022-10-18 09:11:48.191: C:\Users\HS\Desktop\manual.hsd loaded  
 2022-10-18 09:12:52.994: SIM\_PL\_C. Connection established

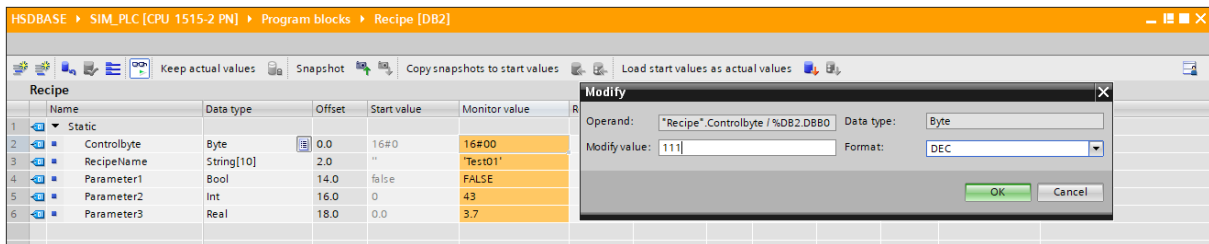
Beim Anlegen der Variablen ist zu beachten, dass die Häkchen in der Spalte *Wert ändern* richtig gesetzt sind. Bei Rezeptnamen (Auswahlvariablen) nicht gesetzt, bei den Parametern gesetzt, da diese von **HSDBASE** entsprechend des Datenbankeintrags überschrieben werden:



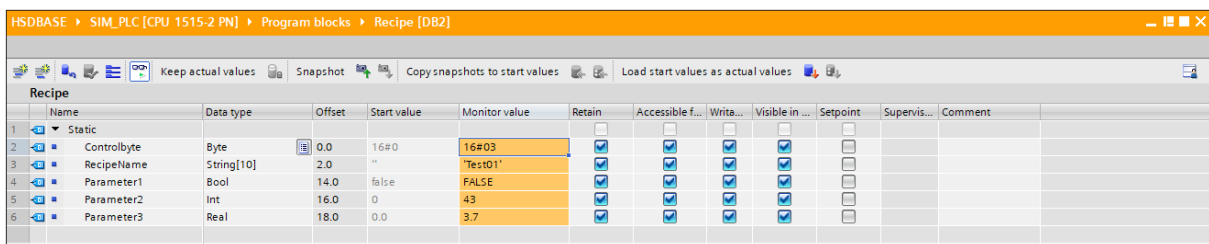
Starten Sie nun die Erfassung, damit **HSDBASE** auf die Kommandos der SPS reagiert.

## Rezeptur in Datenbank speichern

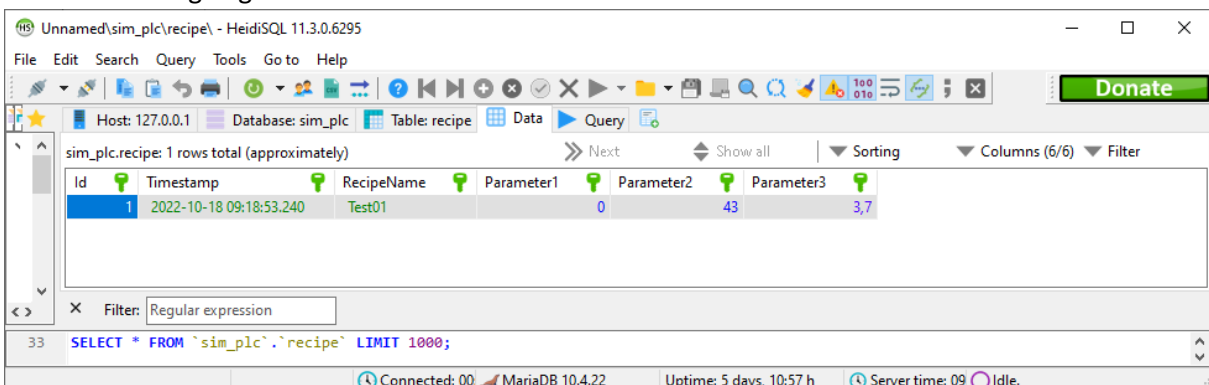
Steuern Sie den Rezeptnamen und die Parameter auf beliebige Werte und dann das Controlbyte auf 111 (Aus SPS in Datenbank):



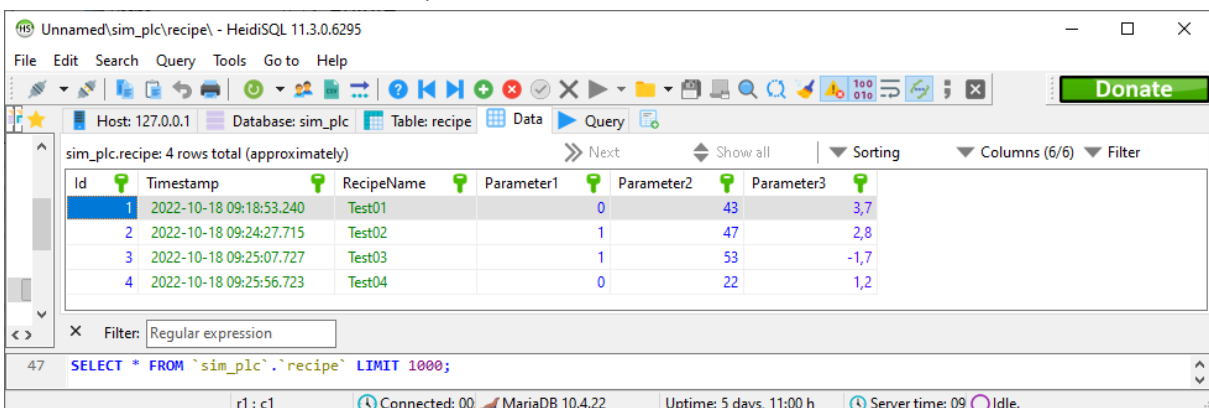
Als Antwortcode schreibt **HSDBASE** eine 3 (Datensatz wurde erstellt) in das Controlbyte:



Das Rezept befindet sich nun in der Datenbank. Datenbank und Tabelle wurden von **HSDBASE** automatisch angelegt:

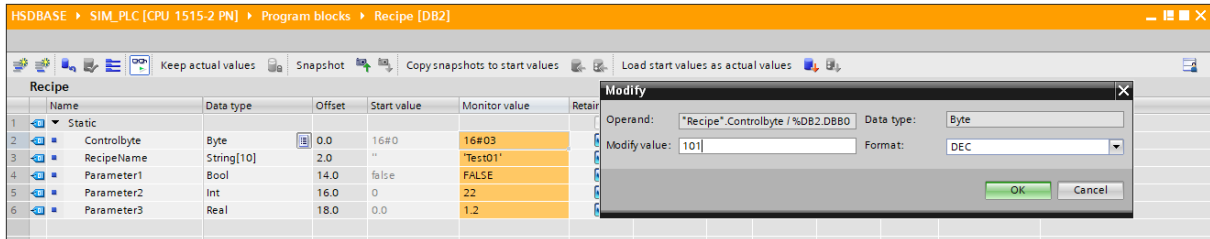


Dieser Schritt kann nun beliebig oft mit verschiedenen Rezeptnamen und Werten wiederholt werden. Wird eine bereits verwendete Rezeptnummer benutzt, wird der Datensatz aktualisiert:

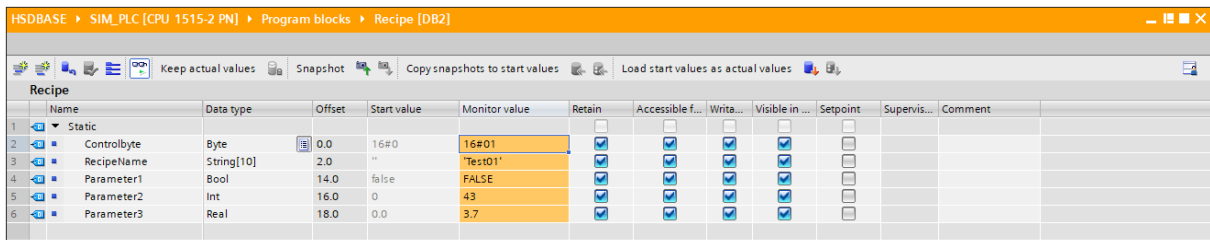


### Rezeptur aus Datenbank lesen

Nun soll das Rezept mit dem Namen „Test01“ von der Datenbank zurück in die SPS übertragen werden. Hierzu muss erst der Rezeptname in der Steuerung eingestellt werden und dann das Controlbyte auf 101 (Aus Datenbank in SPS) gesteuert werden:



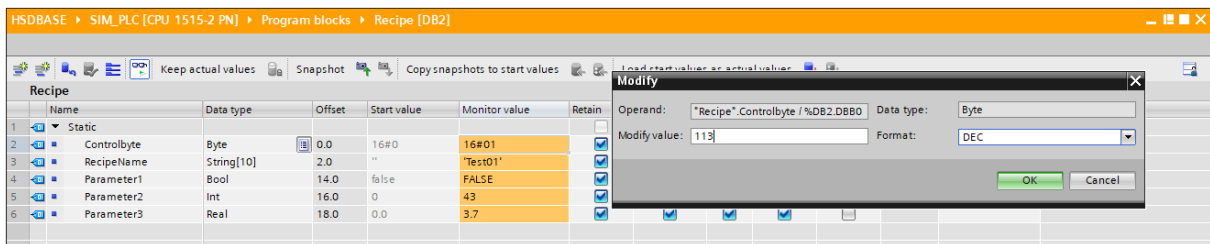
Die Parameter werden mit denen aus der Datenbank überschrieben und der Antwortcode 1 (Daten wurden in SPS geschrieben) wird von HSDBASE gesetzt:



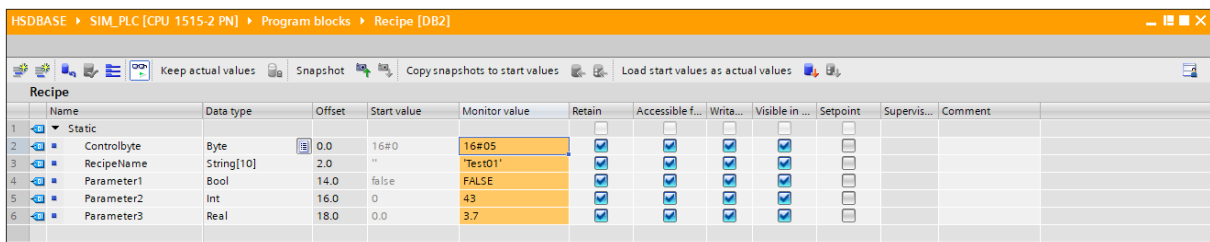


### Rezeptur löschen

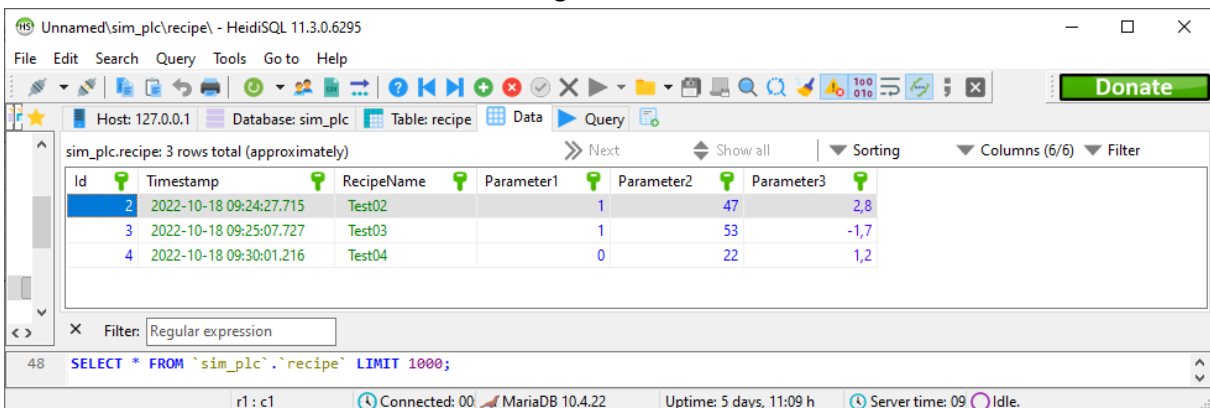
Nun soll das Rezept („Test01“) gelöscht werden. Hierzu muss das Controlbyte auf 113 (Datensatz löschen) gesetzt werden:



Als Antwortcode erhalten Sie 5 (Datensatz wurde gelöscht):

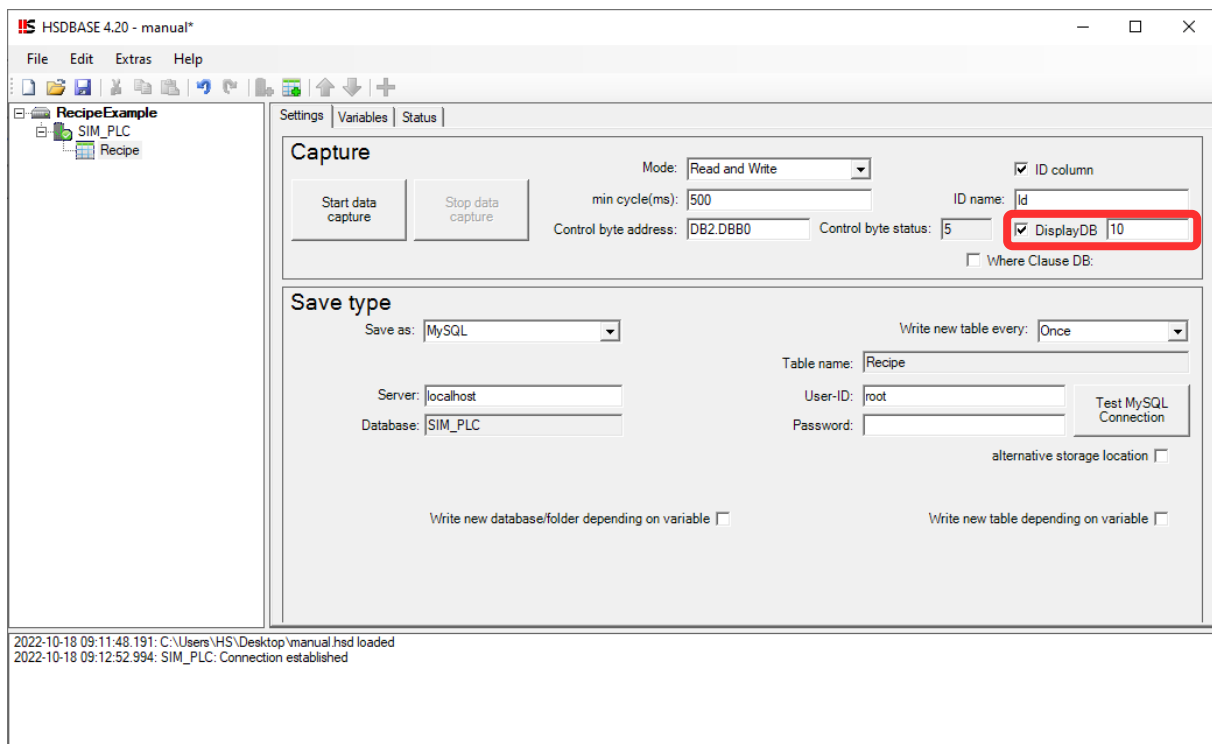


Die Tabelle in der Datenbank sieht nun wie folgt aus:



## Anzeige Datenbaustein

Um abzulesen welche Rezepturen in der Datenbank vorhanden sind, kann ein Anzeige-Datenbaustein erstellt werden. Dieser wird dann von **HSDBASE** beschrieben.



Der Anzeige DB muss wie folgt aufgebaut sein:

Name	Data type	Offset	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Supervis...	Comment
1	Static									
2	nr_of_datasets	0.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
3	start_datatset	4.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
4	datasets_in_db	8.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
5	recipes	Array[1..25] of String[10]	12.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

Die ersten drei Einträge müssen vom Typ DInt sein. Der Array kann beliebig lang sein, muss aber dem Datentyp des verwendeten Bezeichners der Datensätze entsprechen.

Wird das Controlbyte auf 121 (AnzeigeDB füllen) gesetzt, schreibt **HSDBASE** die Anzahl aller Datensätze in der Datenbank in das erste Datendoppelwort. Im dritten Datendoppelwort steht die Anzahl der Datensätze, die in den Datenbaustein geschrieben wurden. Der Rest des DBs wird mit den Bezeichnern der Datensätze aufgefüllt. **HSDBASE** bestimmt automatisch wie viele Bezeichner in den Baustein passen. Sollten mehr Datensätze in der Datenbank vorhanden sein, als in den DB passen, kann der Beginn des Anzeigebereichs über das zweite Datendoppelwort vorgegeben werden. Sind weniger Datensätze vorhanden, wird der restliche DB mit w#16#00 gefüllt.

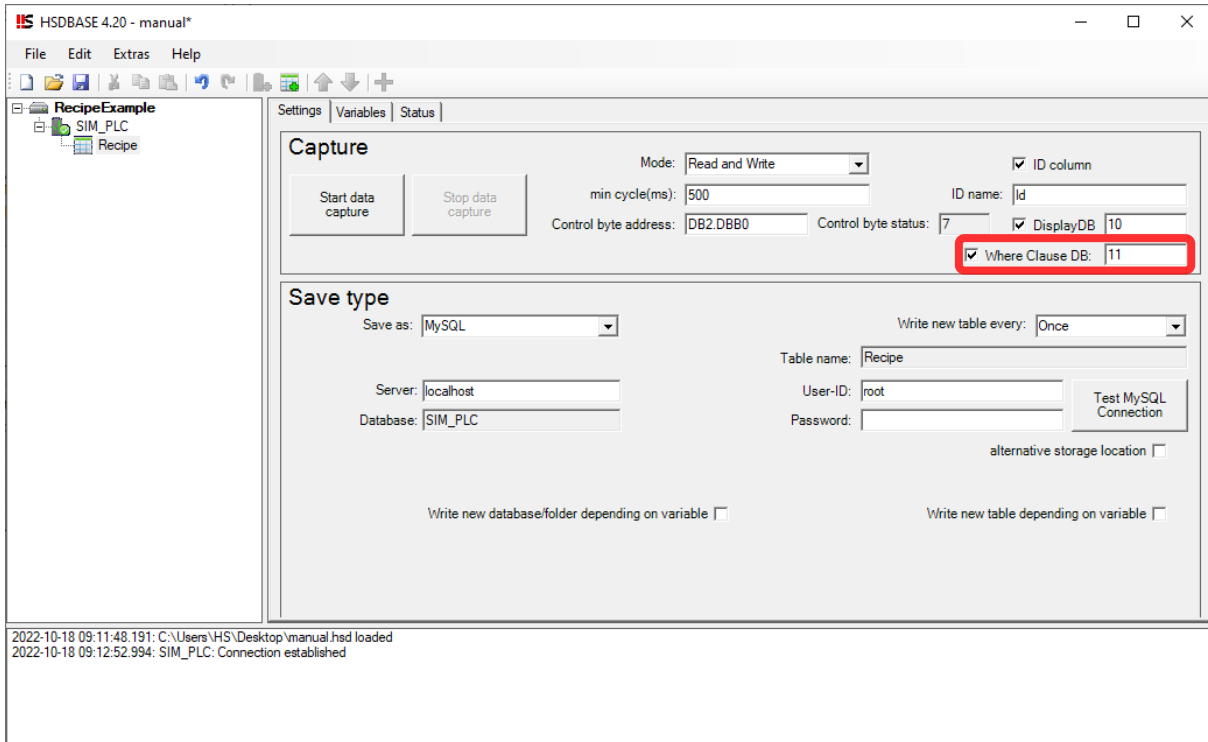
Nach dem Steuern des Controlbytes zeigt der Anzeige DB nun die vorhandenen Datensätze an:

The screenshot shows the 'DisplayDB' window in the HSDBASE software. The window title is 'HSDBASE - SIM\_PLG [CPU 1515-2 PN] - Program blocks - DisplayDB [DB10]'. The table below represents the data shown in the interface:

Name	Data type	Offset	Start value	Monitor value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Supe...	Comment
Static											
nr_of_datasets	Dint	0.0	0	3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
start_datatset	Dint	4.0	0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
datasets_in_db	Dint	8.0	0	3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
recipes	Array[1..25] of String[10]	12.0									
recipes[1]	String[10]	12.0	"	'Test02'	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
recipes[2]	String[10]	24.0	"	'Test03'	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
recipes[3]	String[10]	36.0	"	'Test04'	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
recipes[4]	String[10]	48.0	"	"	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

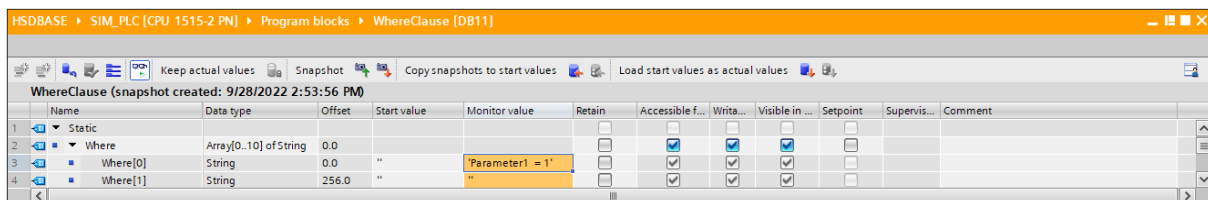
### WhereClause Datenbaustein

Um die Möglichkeit zu schaffen, aus einer Datenbank nur Datensätze auszuwählen, die bestimmte Bedingungen erfüllen, gibt es in **HSDBASE** die Möglichkeit in einem weiteren DB Bedingungen einzugeben, die bei der SQL-Abfrage von **HSDBASE** in einer „Where Clause“ angehängt werden.



Der WhereClause-DB besteht aus beliebig vielen Strings. Mit diesen Strings wird der interne SQL-Befehl in **HSDBASE** hinter dem Schlüsselwort WHERE erweitert. Ist der erste String leer, findet die Abfrage ohne „WhereClause“ statt. Bei der Erstellung der Strings ist die entsprechende SQL-Syntax der gewählten Datenbank zu beachten. Eine Überprüfung findet nicht statt.

Einfache Bedingung für das vorherige Beispiel:



Nach Steuern des Controlbytes auf 121 (Anzeige BD füllen) wird der AnzeigeDB nur mit den Datensätzen gefüllt, die die eingestellten Bedingungen erfüllen.

The screenshot shows the 'DisplayDB' configuration window in the HSDBASE software. The window title is 'HSDBASE - SIM\_PL\_C [CPU 1515-2 PN] - Program blocks - DisplayDB [DB10]'. The interface includes a toolbar with icons for 'Keep actual values', 'Snapshot', 'Copy snapshots to start values', and 'Load start values as actual values'. Below the toolbar is a table with the following columns: Name, Data type, Offset, Start value, Monitor value, Retain, Accessible f..., Writa..., Visible in..., Setpoint, Supervis..., and Comment. The table contains the following data:

	Name	Data type	Offset	Start value	Monitor value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Supervis...	Comment
1	Static											
2	nr_of_datasets	Dint	0.0	0	2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
3	start_datatset	Dint	4.0	0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
4	datasets_in_db	Dint	8.0	0	2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
5	recipes	Array[1..25] ...	12.0			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
6	recipes[1]	String[10]	12.0	"	Test02'	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
7	recipes[2]	String[10]	24.0	"	Test03'	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
8	recipes[3]	String[10]	36.0	"	"	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

## Modus Ringspeicher

In diesem Modus werden die Daten im Arbeitsspeicher des Rechners, auf dem **HSDBASE** läuft gesammelt und erst auf Befehl in die Datenbank geschrieben. Der Ringspeicher wird wie im Modus Lesen und Schreiben über ein Controlbyte gesteuert.

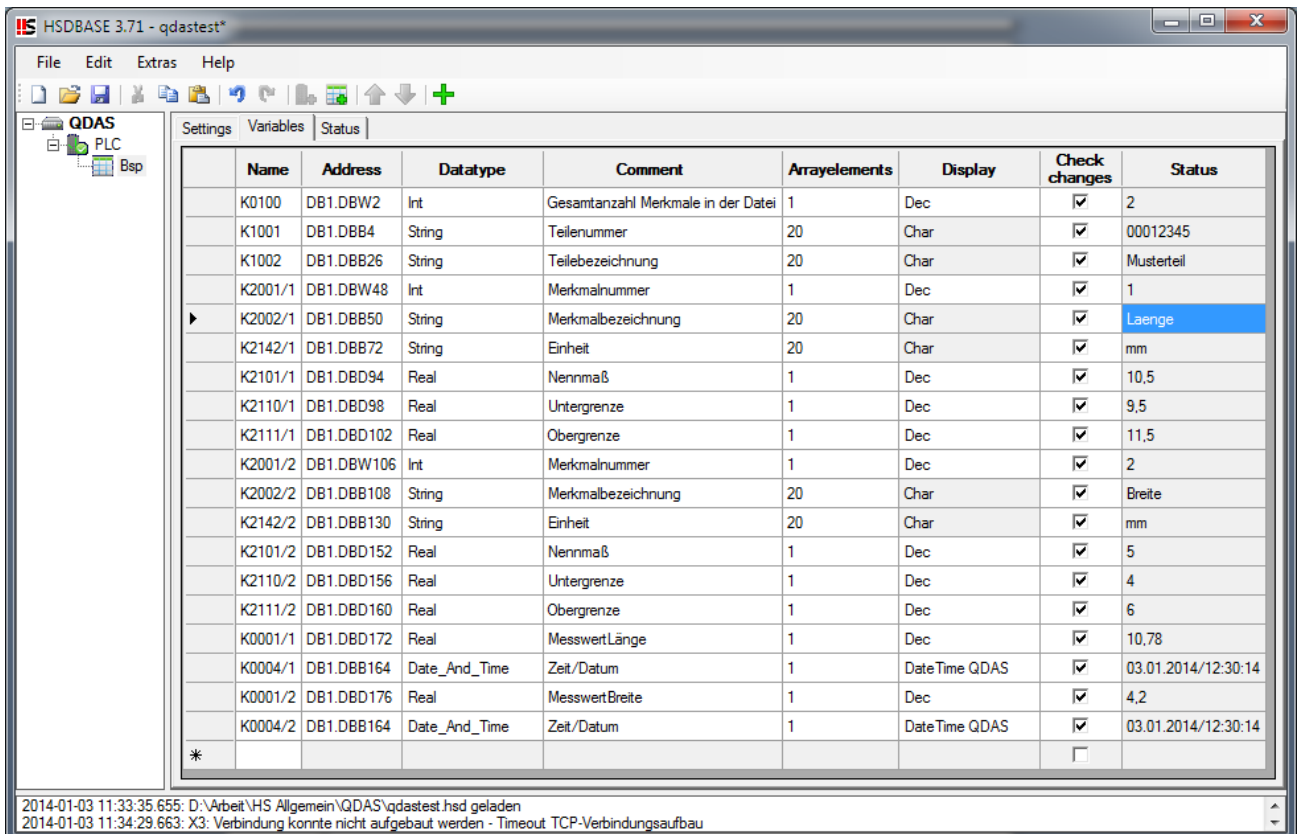
### Alle Codes im Überblick:

	<b>Kommandocodes</b>
101	Ringspeicher löschen
102	Ringspeicher füllen (wird nicht quittiert)
103	Ringspeicher abspeichern
	<b>Antwortcodes</b>
1	Ringspeicher wurde gelöscht
2	Ringspeicher wurde gespeichert (und gelöscht)
11	unbekannter Befehl

# Modus QDAS

Der folgende Abschnitt zeigt anhand eines Beispiels, wie **HSDBASE** konfiguriert werden muss, um Daten im QDAS-Datenformat DFQ abzuspeichern. Diese können dann bspw. mithilfe des QDAS-Uploadtools automatisch in die QDAS-Datenbank eingelesen werden.

Die für QDAS benötigten K-Nummern sind in der Spalte Name einzutragen. Pro Trigger wird eine separate DFQ-Datei mit fortlaufender Nummer generiert.



## Kontakt

Für noch offene Fragen sind wir jederzeit für Sie erreichbar.

HS Automation Software  
Im Öderich 2  
D-73663 Berglen  
Tel.: + 49 (0) 7195 - 588 639  
E-Mail: [info@hs-automation-software.de](mailto:info@hs-automation-software.de)  
Web: [www.hs-automation-software.de](http://www.hs-automation-software.de)