



TeWinCC Dokumentation

Copyright AVI GmbH

Version 2.2 - 2007

Inhaltsverzeichnis

1. TeWinCC	1
1.1. Leistungsumfang von TeWinCC	2
2. Verwendung	3
2.1. Konfiguration	3
2.2. Projektdatenbank	6
2.2.1. V-Variablen	10
2.2.2. Rohdatenvariablen	10
2.2.2.1. R-Variablen	11
2.2.2.2. B-Variablen	15
2.2.2.3. P-Variablen	20
2.2.3. Alarmvariablen	24
2.2.3.1. Bitmeldeverfahren	29
2.2.3.2. Meldenummernverfahren	31
2.3. Export	32
3. System Informationen	34
3.1. Technische Voraussetzungen	34
3.2. Installation und Deinstallation	34
3.3. Hardwaremäßiger Softwareschutz	34
3.3.1. Umfang der Lizenzen	35
3.3.2. Treiber	35
3.3.3. Hilfsprogramme	35
3.3.4. Updates	35
3.4. Einschränkungen der Demo-Version	35
3.5. Bezugsquellen	36
4. Referenz	37

Abbildungsverzeichnis

1.1. Einordnung in TeRANiS	1
2.1. Konfiguration der Exportparameter	4
2.2. Konfiguration der Exportspalten	6
2.3. Konfiguration eines Exportparameters	7
2.4. Anlegen einer Exportspalte	8
2.5. Projektdatenbankausschnitt mit V-Variablen	10
2.6. Exportierte V-Variablen im WinCC-Variablenhaushalt	10
2.7. Konfiguration von Rohdatenvariablen	11
2.8. Schema der Rohdatenkommunikation mit R-Variablen	12
2.9. Konfiguration einer Rohdatenvariable für R-Variablen	13
2.10. Projektdatenbankausschnitt mit R-Variablen	14
2.11. Rohdatenvariable 'SPS1_STATUS'	15
2.12. Exportierte R-Variablen im WinCC-Variablenhaushalt	15
2.13. Schema der Rohdatenkommunikation mit B-Variablen	16
2.14. Konfiguration einer Rohdatenvariable für B-Variablen	16
2.15. Projektdatenbankausschnitt mit B-Variablen	17
2.16. Erzeugte Rohdatenvariable	18
2.17. Lesen einer B-Variablen	18
2.18. Schreiben einer B-Variablen	19
2.19. Schema der Rohdatenkommunikation mit P-Variablen	20
2.20. Konfiguration einer Rohdatenvariable für P-Variablen	21
2.21. Projektdatenbankausschnitt mit P-Variablen	23
2.22. Rohdatenvariable 'SPS1_ALARM'	23
2.23. Exportierte P-Variablen im WinCC-Variablenhaushalt	23
2.24. Alarmkonfiguration	24
2.25. Alarmkonfiguration Meldenummernverfahren	26
2.26. Alarm 8 Konfiguration	27
2.27. Projektdatenbank Bitmeldeverfahren mit V-Variablen	29
2.28. Variablenhaushalt Bitmeldungen mit V-Variablen	29
2.29. Projektdatenbank Bitmeldeverfahren mit P-Variablen	29
2.30. Variablenhaushalt Bitmeldeverfahren mit P-Variablen	30
2.31. Projektdatenbank Bitmeldeverfahren mit R-Variablen	30
2.32. Variablenhaushalt Bitmeldeverfahren mit R-Variablen	30
2.33. Projektdatenbank Bitmeldeverfahren mit übergeordneter Meldevariablen	31
2.34. Variablenhaushalt Bitmeldeverfahren mit übergeordneter Meldevariablen	31
2.35. Projektdatenbank Meldenummernverfahren	32
2.36. Variablenhaushalt Meldenummernverfahren	32
2.37. Exportieren der Variablen und Meldungen	32

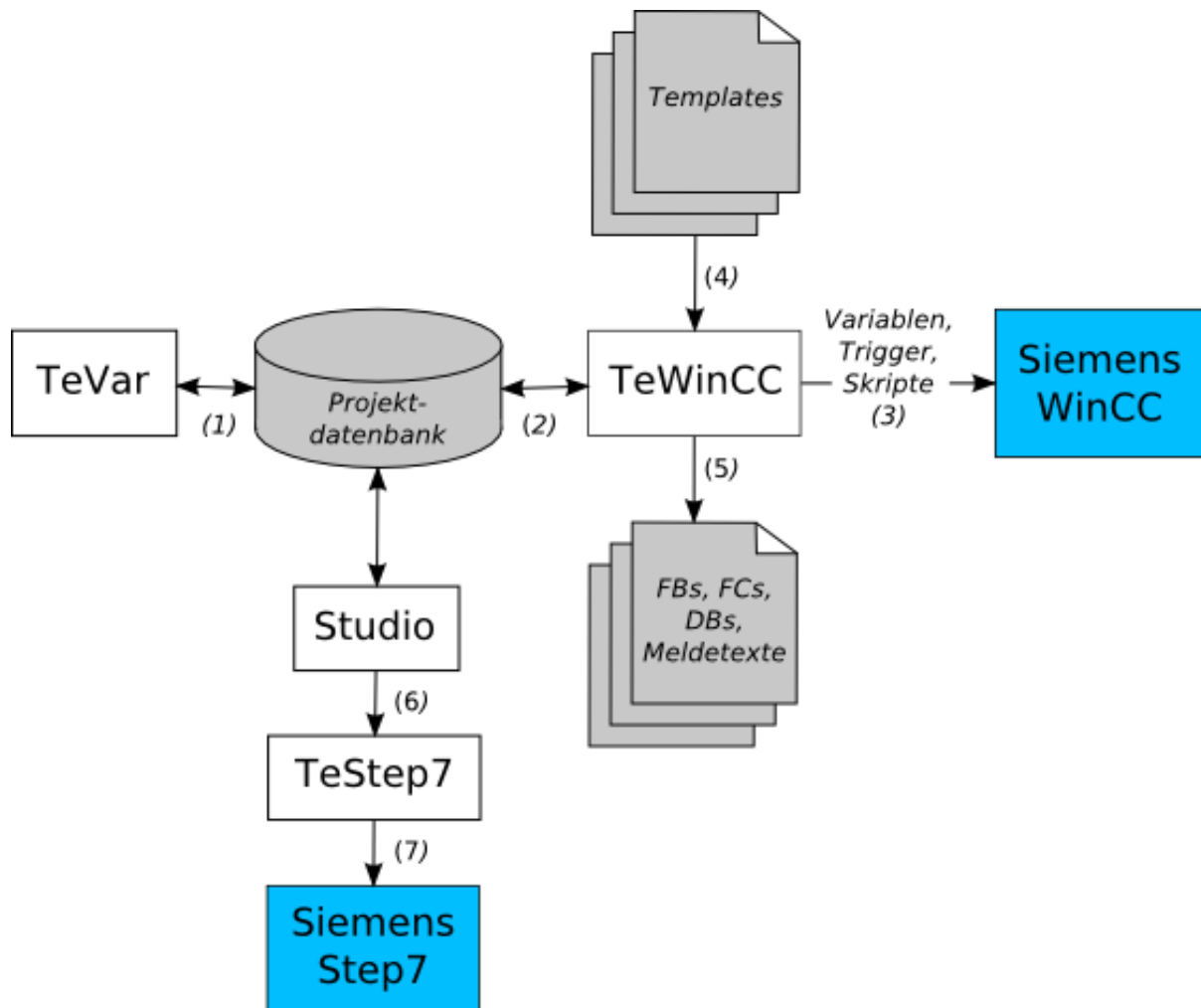
Tabellenverzeichnis

2.1. Filterbare Spalten in der Tabelle 'RAWDATA'	5
2.2. Unterstützte Exportparameter	8
2.3. Unterstützte Datentypen	9
2.4. Alarmparameter	25

Kapitel 1. TeWinCC

In der folgenden Abbildung wird die Einordnung von TeWinCC im Umfeld des TeRANiS-Systems dargestellt. Dieser Zusammenhang wird im folgenden beschrieben.

Abbildung 1.1. Einordnung in TeRANiS



Mit dem Werkzeug TeVar wird der Inhalt einer TeRANiS-Projekt-datenbank gepflegt (1). Diese Datenbank beinhaltet unter anderem besonders gekennzeichnete Variablen für die Verwendung in einem WinCC-Projekt. Der Abgleich zwischen der Projekt-datenbank und dem WinCC-Variablenhaushalt erfolgt mit TeWinCC (2). Neben dem Export von Variablen-Definitionen aus der Projekt-datenbank automatisiert dieses Werkzeug auch die Generierung der für eine Rohdatenkommunikation benötigten Variablen, Trigger und Skripte (3). Des Weiteren werden für die Rohdatenkommunikation zwischen SPS und WinCC Datenbausteine sowie mit Hilfe von TeList-Templates (4) die benötigten Funktionsbausteine bzw. Funktionen generiert (5). Der Transfer der generierten Bausteine nach Step7 sowie deren Übersetzung muss im TeRANiS-Studio manuell gestartet werden (6). Dieser Vorgang wird durch TeStep7 koordiniert (7).

Neben dem beschriebenen Variablenexport können durch TeWinCC auch Meldungen für das Alarm-Logging von WinCC exportiert werden. Der Export der Meldungen erfolgt als CSV-Datei, welche manuell im WinCC-Alarm-Logging importiert werden muss. Optional können Alarmer mit Hilfe von

Alarm-8-Bausteinen an WinCC übertragen werden, die dafür benötigten Datenbausteine und Funktionen können ebenfalls durch TeWinCC erzeugt werden.

1.1. Leistungsumfang von TeWinCC

- Export von Variablen und Alarmen nach WinCC
- Automatische Konvertierung von Adressen in die von WinCC geforderte Syntax
- Generierung aller benötigten Variablen, Skripte und Quellen für die Rohdatenkommunikation

Kapitel 2. Verwendung

Mit TeVar werden Variablen und ihre mögliche Zuordnung zu Rohdatenvariablen und Alarmen in einer Projektdatenbank verwaltet. Dabei werden die folgenden Variablenarten unterschieden:

- V-Variablen werden im WinCC-Variablenhaushalt als externe Variablen angelegt. Sie besitzen den in der TeVar-Datenbank definierten Datentyp, die vorgegebene Initialisierung und Adresse. V-Variablen können in WinCC gelesen und geschrieben werden.
- R-Variablen werden im WinCC-Variablenhaushalt als interne Variablen mit dem in der TeVar-Datenbank definierten Datentyp sowie der vorgegebenen Initialisierung angelegt. Mit Hilfe der Rohdatenkommunikation werden Werte aus einer SPS auf diese internen Variablen abgebildet. Auf die im WinCC-Projekt als interne Variablen angelegten R-Variablen kann nur lesend zugegriffen werden, da die Rohdatenkommunikation für R-Variablen nur von der SPS in Richtung WinCC erfolgt. R-Variablen werden durch die Rohdatenkommunikation bei Änderung aktualisiert.
- P-Variablen werden wie R-Variablen im WinCC-Variablenhaushalt als interne Variablen angelegt. Die Aktualisierung dieser Variablen erfolgt durch die zyklische Ausführung von WinCC-Skripten, welche die Daten von Rohdatenvariablen auf P-Variablen abbilden. Auf P-Variablen kann in WinCC nur lesend zugegriffen werden.
- B-Variablen bilden zusammenhängende Bereiche in Datenbausteinen. Im WinCC-Variablenhaushalt werden für diese Bereiche Rohdatenvariablen angelegt. Das Lesen und Schreiben dieser Rohdatenvariablen muss durch geeignete Skripte in WinCC erfolgen.
- Rohdatenvariablen dienen der Kommunikation zwischen SPS und WinCC. R-, B- und P-Variablen werden solchen Rohdatenvariablen zugeordnet. Die Konfiguration der Rohdatenvariablen erfolgt in einem Konfigurationsdialog in TeWinCC und wird in der Projektdatenbank in der Tabelle 'RAWDATA' gespeichert.
- Alarmvariablen werden für das WinCC-Alarm Logging konfiguriert. Eine Alarmvariable kann zusätzlich als V-, R- oder P-Variable definiert werden.

Anwendungsbeispiele:

R-Variablen sollten immer dann verwendet werden, um Statusinformation aus der SPS an den WinCC-Rechner zu übertragen. Da R-Variablen auf interne Variablen des WinCC-Variablenhaushalts abgebildet werden, kann auf diese Variablen wie bei V-Variablen in beliebiger Weise lesend zugegriffen werden. R-Variablen besitzen den Vorteil, dass deren Übermittlung an den WinCC-Rechner mit Hilfe des BSEND-Funktionsbausteins ereignisorientiert erfolgt und damit eine geringere Kommunikationslast als bei V-Variablen entsteht. Der Nachteil von R-Variablen besteht darin, dass die Kommunikation nur von der SPS in Richtung WinCC erfolgt.

P-Variablen kommen dann zum Einsatz wenn durch Einschränkungen der SPS (keine Verfügbarkeit des BSEND-Funktionsbausteins) der Einsatz von R-Variablen nicht möglich ist.

B-Variablen kommen dann zur Anwendung, wenn sich eine am WinCC-Rechner ausgeführte Bedienungshandlung auf der SPS auswirken soll, z.B. durch das Setzen eines bestimmten Bits in einem Datenbaustein. Da der lesende und schreibende Zugriff auf B-Variablen immer mit Skripten erfolgen muss, wird diese Variablenart vor allem bei automatisch dynamisierten WinCC-Bildern verwendet.

2.1. Konfiguration

Auf der ersten Seite des Wizard können die Exportparameter konfiguriert werden. Nach der Eingabe aller Einstellungen kann über die Schaltfläche **Weiter** > zur nächsten Seite gewechselt werden.

Abbildung 2.1. Konfiguration der Exportparameter

Diese Wizard-Seite enthält die folgenden Einstellungen:

WinCC Station

Der Name der in WinCC geöffneten Station. Über diesen Namen wird die für die Konfiguration des Exports zu verwendende Spalte der Projektdatei ermittelt. Heißt die Station z.B. 'VISU',

so wird in der Projektdatenbank die Spalte 'WINCC_VISU' für den Export verwendet.

TeRANiS Projekt

TeRANiS Projekt mit den zu exportierenden Variablendaten. Das zu verwendende Projekt kann über die Schaltfläche **Öffnen** ausgewählt werden.

Exportspalte

Mit dieser Auswahlliste kann die für den Export zu verwendende Spalte ausgewählt werden. Zur Auswahl stehen alle Spalten der Projektdatenbank deren Bezeichnung mit der Zeichenkette 'WINCC' beginnt, bei Verwendung einer nicht in der Auswahlliste aufgeführten Spalte, muß diese von Hand eingetragen werden. Die Exportspalte hat die Aufgabe die zu exportierenden Variablen zu kennzeichnen und dient der variablenspezifischen Konfiguration des Exports. Zum Anlegen und Einstellen von Exportspalten gelangt man über die Schaltfläche **Konfigurieren**

Externe Variablen

Sollen die in der ausgewählten Exportspalte mit 'V' gekennzeichneten Variablen exportiert werden, dann muß die Option **V-Variablen** ausgewählt werden. Sollen diese Variablen als CSV-Datei für den manuellen Import nach WinCC exportiert werden, dann muß die Option **Variablen als CSV-Datei exportieren** aktiviert werden. Um die durch die Exportspalte definierte Variablenauswahl weiter einzuschränken, kann ein zusätzlicher **Filter**, z.B. 'name like "160FA%"' angegeben werden, um nur die Variablen mit dem Präfix '160FA' zu exportieren.

Rohdatenvariablen

Der Export der Rohdatenvariablen kann über das jeweilige Kontrollkästchen für **R-Variablen**, **B-Variablen** und **P-Variablen** separat aktiviert werden. Um die durch die Exportspalte definierte Variablenauswahl weiter einzuschränken kann ein zusätzlicher **Filter** angegeben werden. Dieser Filter bezieht sich auf die Tabelle 'RAWDATA' der Projektdatenbank. Es können die folgenden Spalten verwendet werden:

Tabelle 2.1. Filterbare Spalten in der Tabelle 'RAWDATA'

Spaltenname	Beschreibung
NAME	Bezeichnung der Rohdatenvariable, z.B. werden durch den Filter 'name like "SPS1_ANALOG*"' nur die Rohdatenvariablen mit dem Präfix 'SPS1_ANALOG*' exportiert (z.B. 'SPS1_ANALOG_1' und 'SPS1_ANALOG_2').
RAWDATAID	Identifikationsnummer der Rohdatenvariable, durch den Filter 'rawdataid = 101' wird nur die Rohdatenvariable mit der ID 101 exportiert.

Werden die Optionen **Rohdatenkommunikation für WINCC 1/2** aktiviert, dann werden alle für die Rohdatenkommunikation zwischen SPS und WinCC-Station benötigten Funktionen, Funktions- und Datenbausteine neu erzeugt. Zur Konfiguration der Rohdatenvariablen gelangt man über die Schaltfläche **Konfigurieren**.

Meldungen

Der Export der zu den Alarmvariablen in der Projektdatenbank abgelegten Meldetexte kann über die Option **Meldetexte als CSV-Datei exportieren** aktiviert werden. Durch Angabe eines zusätzlichen Filters können die zu exportierenden Meldungen weiter eingeschränkt werden, z.B. werden mit dem Filter 'name like "160FA%"' nur die Meldungen der Alarmvariablen exportiert, deren Name mit dem Präfix '160FA' beginnt. Erfolgt die Signalisierung von Alarmen mit Hilfe von

Alarm-8-Bausteinen, dann wird die Generierung der dafür benötigten Funktionen und Datenbausteine durch die Option **Alarm-8-Kommunikation** aktiviert.

Nur neue oder geänderte Variablen und Alarmer exportieren

Ist diese Option aktiviert, dann werden nur die Variablen der Projektdatenbank exportiert, die nach dem letzten gespeicherten Exportzeitpunkt geändert wurden (siehe folgende Option).

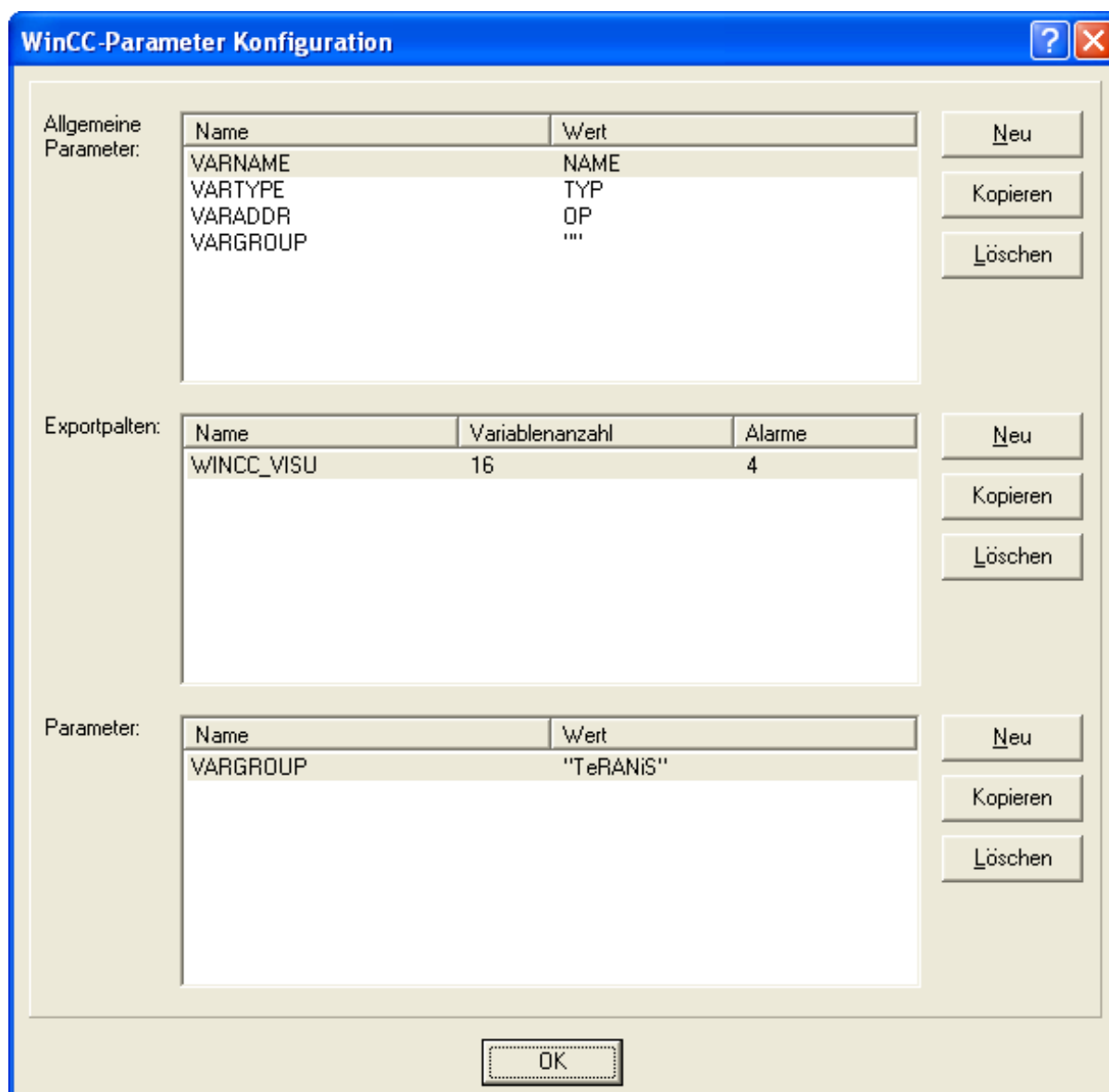
Exportzeitpunkt in der Projektdatenbank aktualisieren

Bei Aktivierung dieser Option wird der Exportzeitpunkt des aktuellen Exports als Zeitstempel in der Projektdatenbank gespeichert. Der Zeitstempel wird in der Konfigurationstabelle der Projektdatenbank für jede WinCC-Station gespeichert. Für eine WinCC-Station mit der Bezeichnung 'VISU' wird dieser z.B. unter dem Schlüssel 'WINCC_VISU_LastExport' gespeichert.

2.2. Projektdatenbank

Die Konfiguration der Exportspalten kann über die Schaltfläche Konfigurieren neben der Auswahlliste der Exportspalten geöffnet werden.

Abbildung 2.2. Konfiguration der Exportspalten

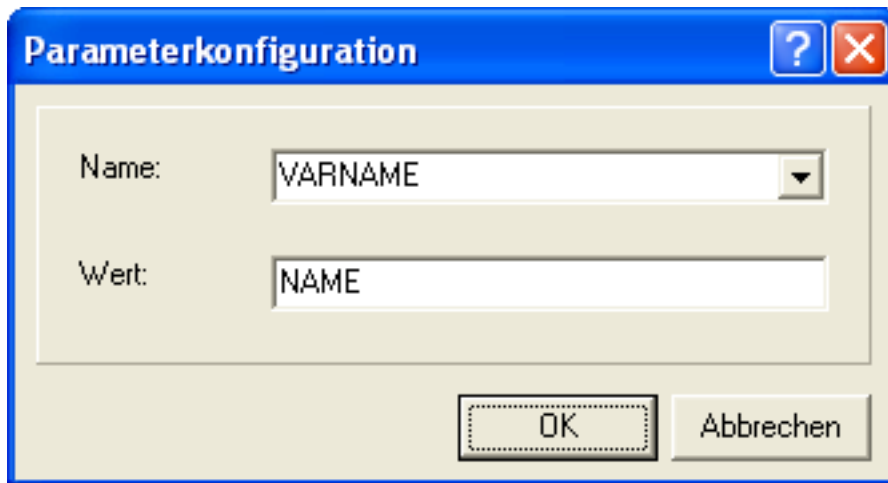


Der Dialog enthält die folgenden Einstellungen.

Allgemeine Parameter

In dieser Liste können die für alle Exportspalten gültigen Exportparameter konfiguriert werden. Mit der Schaltfläche **Neu** können neue Parameter angelegt werden, mit **Kopieren** wird der aktuell ausgewählte Parameter kopiert und mit der Schaltfläche **Löschen** kann der ausgewählte Parameter gelöscht werden. Durch Doppelklick auf einen Parameter kann dieser bearbeitet werden, es erscheint der folgende Dialog.

Abbildung 2.3. Konfiguration eines Exportparameters



Die verfügbaren Parameter zeigt die folgende Tabelle.

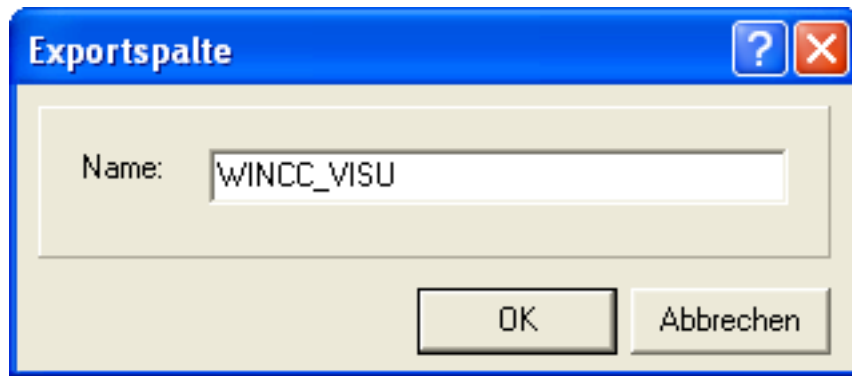
Tabelle 2.2. Unterstützte Exportparameter

Parametername	Beschreibung
VARNAME	Bezeichnung der zu exportierenden Variable, standardmäßig wird der Eintrag aus der Spalte 'NAME' der Projektdatenbank verwendet
VARTYPE	Datentyp, standardmäßig wird der Datentyp aus der Spalte 'TYP' verwendet
VARADDR	Adresse, standardmäßig wird der Operand aus der Spalte 'OP' verwendet
VARGROUP	Variablengruppe, welcher die zu exportierenden Variablen zugeordnet werden sollen
VARLENGTH	Länge der Variablen in Byte, standardmäßig wird der Länge der Variablen aus ihrem Typ ermittelt

Exportspalten

In dieser Liste werden alle angelegten Exportspalten angezeigt. Zusätzlich wird die Anzahl der in jeder Spalte als zu exportierende Variable bzw. Alarmvariable angezeigt. Über die Schaltfläche **Neu** können neue Exportspalten angelegt werden, mit **Kopieren** wird die aktuell ausgewählte Exportspalte kopiert und mit der Schaltfläche **Löschen** kann die ausgewählte Spalte gelöscht werden. Der Name einer Exportspalte sollte mit dem Präfix 'WINCC_' beginnen und danach den Namen des betreffenden WinCC-Projektes enthalten. Dadurch kann beim Start von TeWinCC automatisch aus einem geöffneten WinCC-Projekt die zu verwendende Exportspalte ermittelt werden.

Abbildung 2.4. Anlegen einer Exportspalte



Parameter

Hier können die unter 'Allgemeine Parameter' gemachten Einstellungen für jede Exportspalte separat konfiguriert werden. Z.B. wird in der abgebildeten Konfiguration dem Parameter 'VAR-GROUP' der Wert 'TeRANiS' zugewiesen. Durch diese Konfiguration werden alle in der Spalte 'WINCC_VISU' als zu exportieren gekennzeichneten Variablen im WinCC-Variablenhaushalt der Gruppe 'TeRANiS' zugeordnet.

Die folgende Tabelle zeigt die von TeWinCC unterstützten Datentypen. Dabei sind die Einschränkungen bei der Verwendung von Strukturen und Zeichenketten zu beachten!

Tabelle 2.3. Unterstützte Datentypen

Bezeichnung	Datentyp	Bemerkung
boolesch	BOOL	
Bitkette (8 Bit)	BYTE	
Bitkette (16 Bit)	WORD	
Bitkette (32 Bit)	DWORD	
Ganzzahl (16 Bit)	INT	
Ganzzahl (32 Bit)	DINT	
Gleitpunktzahl (32 Bit)	REAL	
Struktur	STRUCT	Strukturen werden bis zu einer Größe von 16 bzw. 32 Bit unterstützt und im WinCC-Variablenhaushalt als WORD bzw. DWORD Variablen angelegt.
Zeichenkette	STRING	Der Name eines String-Datentyps muß dessen Länge beinhalten, z.B. muß der Datentyp für Stringvariablen mit 8 Zeichen den Namen STRING8 haben. Diese Konvention ist z.Zt. für die Generierung von Datenbausteinen einzuhalten.

2.2.1. V-Variablen

V-Variablen sind die in WinCC-Projekten ursprünglich verwendeten Variablen. Sie stellen den einfachsten Weg dar, um Daten zwischen einer SPS und einem WinCC-Rechner auszutauschen. V-Variablen können zum Lesen und Schreiben von beliebigen Variablen mit elementarem Datentyp verwendet werden. Diese Variablenart besitzt den Nachteil, dass sie durch den WinCC-Rechner zyklisch gelesen werden und dadurch im Gegensatz zu einer ereignisorientierten Kommunikation eine höhere Kommunikationslast entsteht. Da jeder V-Variablen der Projektdatenbank eine externe Variable im WinCC-Variablenhaushalt entspricht sollte deren Verwendung sparsam erfolgen. Im Gegensatz dazu werden mehrere R-, B- bzw. P-Variablen zu einer Rohdatenvariablen zusammengefasst, welche im WinCC-Variablenhaushalt durch eine externe Variable repräsentiert wird.

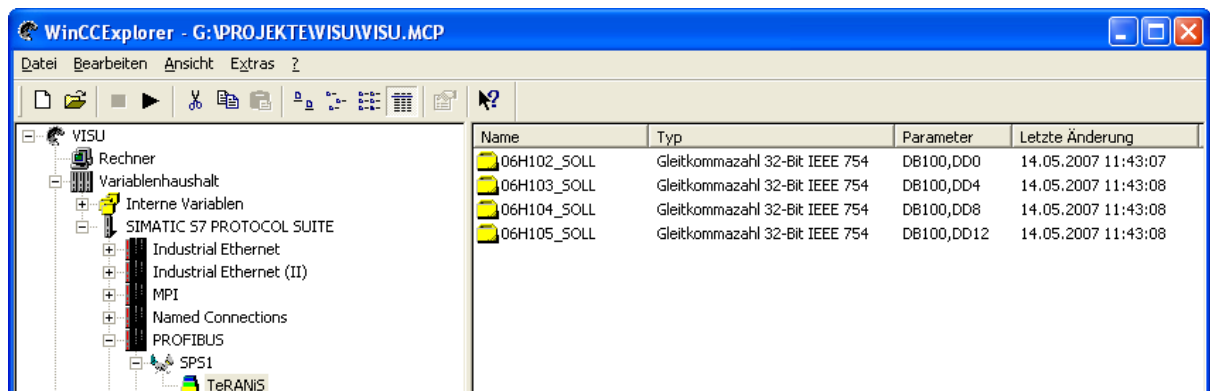
Beispiel:

Abbildung 2.5. Projektdatenbankausschnitt mit V-Variablen

+ ID	RESOURCE	NAME	TYP	OP	COMMENT	WINCC_VISU
1	SPS1	06H102_SOLL	REAL	DB100.DBD0	Sollwert	V
2	SPS1	06H103_SOLL	REAL	DB100.DBD4	Sollwert	V
3	SPS1	06H104_SOLL	REAL	DB100.DBD8	Sollwert	V
4	SPS1	06H105_SOLL	REAL	DB100.DBD12	Sollwert	V

In der ersten Abbildung werden vier V-Variablendefinitionen einer Projektdatenbank dargestellt. TeWinCC erzeugt daraus im WinCC-Variablenhaushalt die in der unteren Abbildung dargestellten externen Variablen. Dabei werden die exportierten V-Variablen einer Ressource, einer gleichnamigen Verbindung im Variablenhaushalt zugeordnet. Hier im Beispiel der Ressource 'SPS1'. Diese Verbindung muss im WinCC-Projekt manuell angelegt werden.

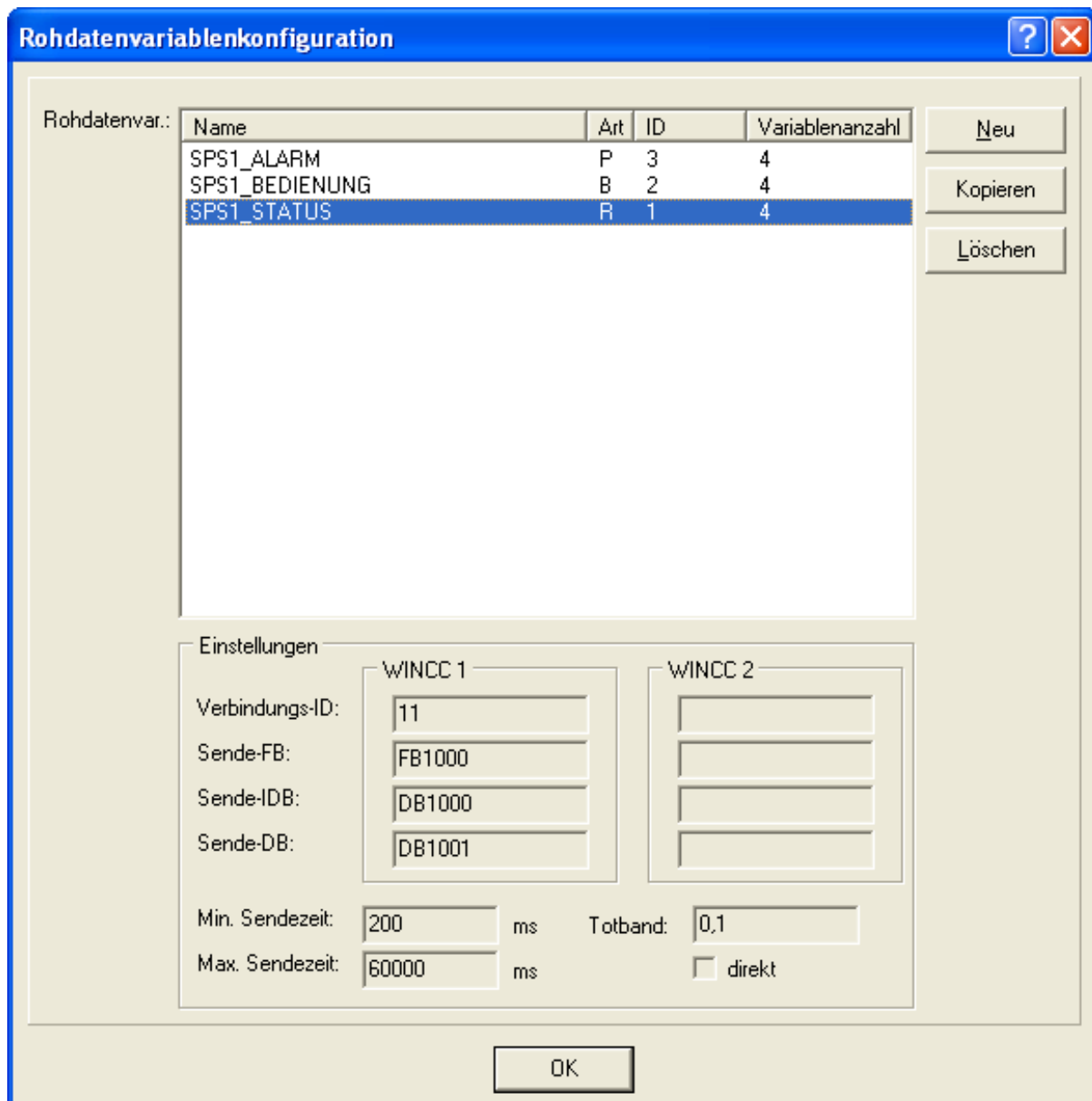
Abbildung 2.6. Exportierte V-Variablen im WinCC-Variablenhaushalt



2.2.2. Rohdatenvariablen

Die in der Projektdatenbank definierten R-, B- und P-Variablen werden bestimmten Rohdatenvariablen zugeordnet. Das Anlegen und Konfigurieren dieser Variablen erfolgt in einem separaten Dialog, welcher über die Schaltfläche Konfigurieren neben der Exportkonfiguration für Rohdatenvariablen geöffnet werden kann.

Abbildung 2.7. Konfiguration von Rohdatenvariablen



Rohdatenvariablen

In diesem Listenfeld sind alle vorhandenen Rohdatenvariablen aufgeführt. Dabei wird zu jeder Variable ihre Art, ID sowie die Anzahl der ihr zugeordneten R-, B- bzw. P-Variablen dargestellt. Über die Schaltfläche **Neu** kann eine neue Rohdatenvariable angelegt werden. Mit der Schaltfläche **Kopieren** wird die aktuell selektierte Rohdatenvariable kopiert. Die Schaltfläche **Löschen** erlaubt das Entfernen der gerade selektierten Rohdatenvariable.

Einstellungen

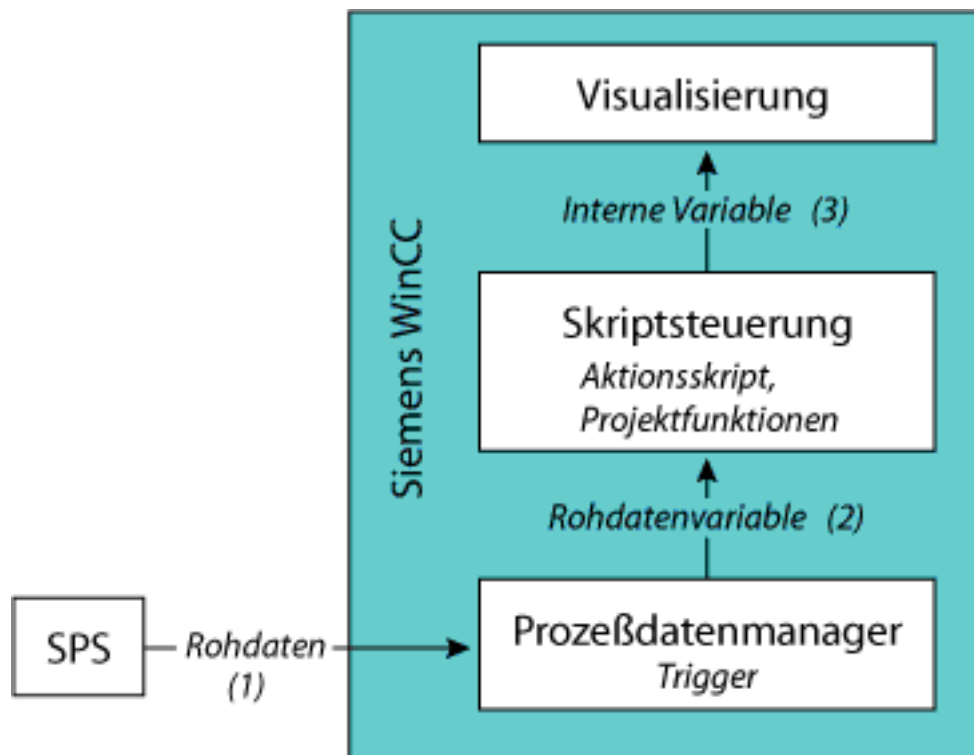
In diesem Bereich werden alle Konfigurationseinstellungen zu der aktuell selektierten Variable angezeigt. Dabei unterscheidet sich die Anzahl und Art der angezeigten Einstellungen von der Art der ausgewählten Rohdatenvariable. Die einzelnen Eigenschaften werden im Folgenden beschrieben.

2.2.2.1. R-Variablen

R-Variablen werden im WinCC-Variablenhaushalt als interne Variablen mit dem in der Projektdatenbank definierten Datentyp angelegt. Mit Hilfe der Rohdatenkommunikation werden Werte aus der SPS auf diese internen Variablen abgebildet.

Aus den Informationen, der in der Projektdatenbank definierten R- und Rohdatenvariablen, werden alle benötigten Variablen, Skripte, Trigger, Funktionsbausteine und Datenbausteine durch TeWinCC erzeugt. Im folgenden wird die Rohdatenkommunikation mit R-Variablen beschrieben.

Abbildung 2.8. Schema der Rohdatenkommunikation mit R-Variablen



Die durch TeWinCC generierte Rohdatenkommunikation ist eine ereignisorientierte Kommunikation, d.h. nur bei Änderung des Inhaltes einer Rohdatenvariablen in der SPS wird die entsprechende Rohdatenvariable an den Prozessdatenmanager von WinCC gesendet (1). Die vom Prozessdatenmanager empfangene Rohdatenvariable ist als Trigger für ein Aktionsskript definiert. Das verknüpfte Aktionsskript wird durch die Skriptsteuerung ausgeführt (2) und bildet die Rohdaten der Rohdatenvariable auf interne Variablen ab (3).

Die den internen Variablen zugewiesenen Werte können in der Visualisierung lesend verwendet werden.

Die Parametrierung von Rohdatenvariablen erfolgt in einem separaten Konfigurationsdialog. Dieser Dialog kann in TeWinCC über die Schaltfläche **Konfigurieren** neben der Exportkonfiguration für Rohdatenvariablen geöffnet werden. Zum Editieren der Eigenschaften einer Rohdatenvariablen kann in der dargestellten Auswahlliste durch Doppelklick auf eine Rohdatenvariable deren Konfiguration geöffnet werden. Wurde die ausgewählte Rohdatenvariable für die Verwendung mit R-Variablen konfiguriert erscheint der folgende Dialog.

Abbildung 2.9. Konfiguration einer Rohdatenvariable für R-Variablen

Rohdatenvariablenkonfiguration

Name:

ID:

Art: R-Variable B-Variable P-Variable

	WINCC 1	WINCC 2
Verbindungs-ID:	<input type="text" value="11"/>	<input type="text"/>
Sende-FB:	<input type="text" value="FB1000"/>	<input type="text"/>
Sende-IDB:	<input type="text" value="DB1000"/>	<input type="text"/>
Sende-DB:	<input type="text" value="DB1001"/>	<input type="text"/>

Min. Sendezeit: ms Totband:

Max. Sendezeit: ms direkt

Name

Eindeutige Bezeichnung der Rohdatenvariablen.

ID

Eindeutige Identifikationsnummer der Rohdatenvariablen. Diese ID muss in TeVar für die Zuordnung von Variablen zu Rohdatenvariablen in der Spalte RAWDATA eingetragen werden.

Art

Unterscheidung nach R-, B- und P-Variable.

Verbindungs-ID

Hexadezimale Verbindungs-ID, der in Step7 projektierten Verbindung zum ersten bzw. zweiten WinCC Rechner.

Sende-FB

Diese FBs enthalten die Anweisungen für das Rangieren und Senden der Rohdaten an den ersten bzw. zweiten WinCC Rechner. Die Erzeugung dieser FBs erfolgt mit Hilfe eines TeList-Templates. Dieses Template muss in der Datei 'BSendBlock.lst' im Ordner '..\Bibliothek\WinCC\' des TeRANiS-Projektes abgelegt sein.

Sende-IDB

Instanzdatenbausteine der Sende-FBs. Aus den Daten dieser DBs kann der aktuelle Verbindungszustand und die Dienstqualität der Rohdatenkommunikation ermittelt werden.

Sende-DB

In diese DBs werden durch die Sende-FBs die zu sendenden R-Variablen für den ersten und zweiten WinCC Rechner rangiert. Zusätzlich wird in diesen DBs ein Sendepuffer verwaltet, welcher zyklisch mit den rangierten Daten verglichen wird. Bei Änderung der rangierten Daten gegenüber dem Sendepuffer und Ablauf der minimalen Sendezeit wird der Sendepuffer aktualisiert und an die Prozessdatenmanager der beteiligten WinCC Rechner als Rohdatenvariable übermittelt.

Min. Sendezeit

Minimale Zeitspanne zwischen zwei Sendevorgängen, vor Ablauf dieser Zeit wird die Rohdatenvariable nicht gesendet. Z.B. wird die Änderung des Inhaltes der Rohdatenvariable 'SPS1_STATUS' höchstens alle 200ms an die WinCC Rechner gesendet.

Max. Sendezeit

Maximale Zeitspanne zwischen zwei Sendevorgängen, nach Ablauf dieser Zeit wird die Rohdatenvariable in jedem Fall durch die SPS gesendet. Z.B. beträgt für die Rohdatenvariable 'SPS1_STATUS' diese Zeit 1min. Ändert sich der Inhalt dieser Rohdatenvariable nicht, so wird die Variable nach Ablauf einer Minute an die WinCC Rechner gesendet.

Totband

Minimale Änderung des Wertes einer R-Variablen, die zum Senden der Rohdatenvariable führt. Liegt die Wertänderung unter dem Wert des eingestellten Totbandes wird die Rohdatenvariable nicht gesendet.

direkt

Ist der Wert dieses Feldes 1, so besteht die Rohdatenvariable aus einem zusammenhängenden Bereich eines Datenbausteins. Ist der Wert dieses Feldes 0, so werden die einer Rohdatenvariablen zugeordneten Elemente vor dem Senden zunächst in einen Datenbaustein ('db1/2') rangiert. B-Variablen können nur zu Rohdatenvariablen mit dem Wert 1 für 'direct' zugeordnet werden. Die Grösse einer als 'direct' gekennzeichneten Rohdatenvariablen ist auf 400Byte beschränkt. Soll ein Datenbausteinbereich mit einer Grösse von mehr als 400Byte als Rohdatenvariable verwendet werden, so muss dieser Bereich auf mehrere Rohdatenvariablen aufgeteilt werden. Dafür muss die Spalte 'doffset' angepasst werden. Für nicht als 'direct' gekennzeichnete Rohdatenvariablen beträgt die maximale Grösse 64KB.

Beispiel:

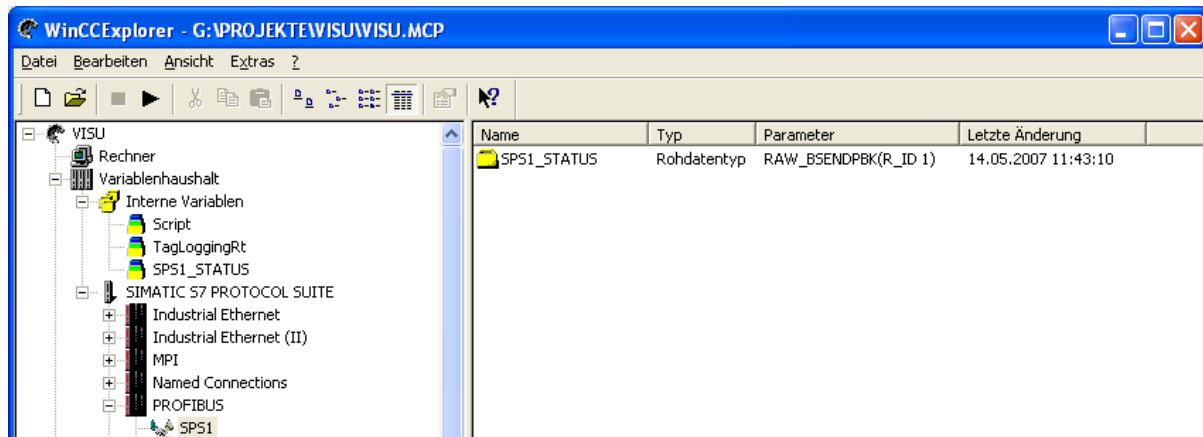
Abbildung 2.10. Projektdatenbankausschnitt mit R-Variablen

+ ID	RESOURCE	NAME	TYP	OP	COMMENT	WINCC_VISU	RAWDATA
▶ 5	SPS1	06H102_STAT	WORD	DB102.DBW0	Status	R	1
6	SPS1	06H103_STAT	WORD	DB103.DBW0	Status	R	1
7	SPS1	06H104_STAT	WORD	DB104.DBW0	Status	R	1
8	SPS1	06H105_STAT	WORD	DB105.DBW0	Status	R	1

Die Abbildung 'Projektdatenbankausschnitt mit R-Variablen' zeigt die Definitionen von vier R-Variablen. Diese sind der Rohdatenvariablen mit der Nummer 1 ('SPS1_STATUS') zugeordnet. Die Definition der Rohdatenvariablen wird in der ersten Abbildung gezeigt.

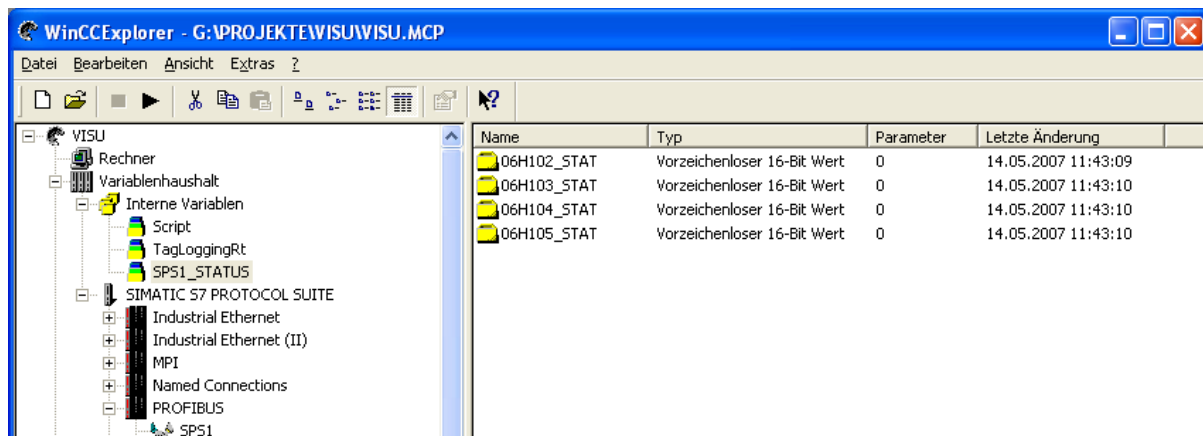
Beim Export der Rohdatenvariablen mit TeWinCC wird im WinCC-Variablenhaushalt unter der Verbindung 'SPS1' die Rohdatenvariable 'SPS1_STATUS' angelegt. Die Verbindung 'SPS1' muss manuell angelegt werden und als Rohdatenverbindung parametrisiert werden.

Abbildung 2.11. Rohdatenvariable 'SPS1_STATUS'



Die Abbildung 'Exportierte R-Variablen im WinCC-Variablenhaushalt' zeigt die, für die definierten R-Variablen erzeugten internen Variablen im Variablenhaushalt von WinCC. Dabei werden die erzeugten internen Variablen einer zur Rohdatenvariable gleichnamigen Variablengruppe zugeordnet. Hier im Beispiel ist das die Gruppe 'SPS1_STATUS'.

Abbildung 2.12. Exportierte R-Variablen im WinCC-Variablenhaushalt



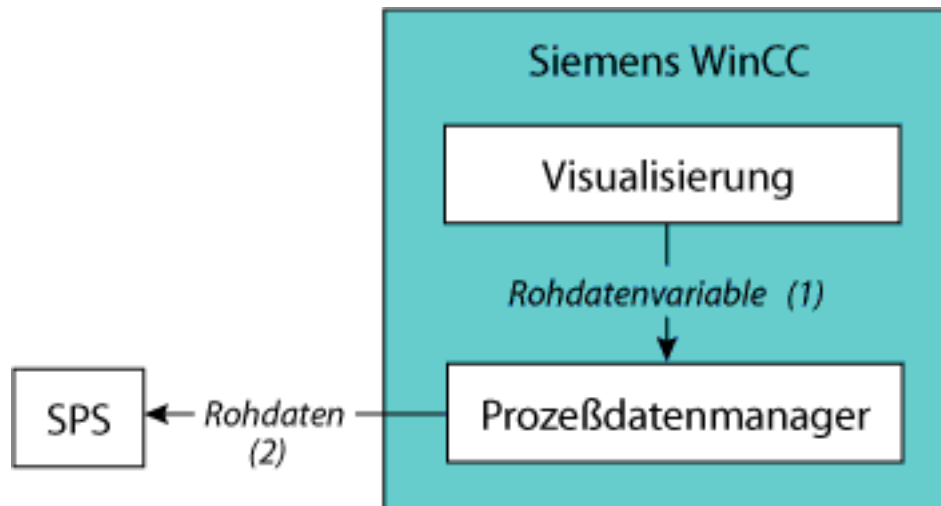
Für die Abbildung der Daten aus der Rohdatenvariable auf interne Variablen wird ein Aktionskript mit dem Namen der zugordneten Rohdatenvariable generiert. Für dieses Aktionskript wird die betreffende Rohdatenvariable als Trigger definiert, sodass bei jeder Änderung der Rohdatenvariable deren Inhalt durch das Skript auf die internen Variablen abgebildet wird. Für diese Abbildung werden zusätzlich die Projektfunktionen 'GetDword', 'GetFloat' und 'GetWord' in der WinCC-Skriptsteuerung erzeugt, welche eine Wandlung des in der SPS verwendeten Big-Endian Formates in das durch Intel-Hardware verwendete Little-Endian Format durchführen.

Die durch TeWinCC generierten Quellen für die Kommunikations- und Datenbausteine werden im Verzeichnis '..\WinCC\AWL\Rohdaten\' des TeRANiS-Projektes der entsprechenden Resource abgelegt. Diese Bausteine müssen nun noch mit Hilfe von TeRANiS-Studio in das Step7 - Projekt übertragen werden. Die für den Aufruf der Funktionsbausteine benötigten Instanzdatenbausteine müssen manuell erzeugt werden. Aus den Aktualwerten eines solchen Instanzdatenbausteins kann der aktuelle Zustand der Rohdatenkommunikation abgeleitet werden. Hierbei sind vor allem die Elemente 'ALL-SEND' und 'LASTERROR' zu beachten.

2.2.2.2. B-Variablen

Im Gegensatz zu R-Variablen bilden mehrere B-Variablen immer einen zusammenhängenden Bereich eines Datenbausteins. Im WinCC-Variablenhaushalt werden für diese Bereiche Rohdatenvariablen angelegt. Das Lesen und Schreiben, der von B-Variablen gebildeten Rohdatenvariablen, muss in WinCC über Skripte erfolgen.

Abbildung 2.13. Schema der Rohdatenkommunikation mit B-Variablen



Die Parametrierung von Rohdatenvariablen erfolgt in einem separaten Konfigurationsdialog. Dieser Dialog kann in TeWinCC über die Schaltfläche **Konfigurieren** neben der Exportkonfiguration für Rohdatenvariablen geöffnet werden. Zum Editieren der Eigenschaften einer Rohdatenvariablen kann in der dargestellten Auswahlliste durch Doppelklick auf eine Rohdatenvariable deren Konfiguration geöffnet werden. Wurde die ausgewählte Rohdatenvariable für die Verwendung mit B-Variablen konfiguriert erscheint der folgende Dialog.

Abbildung 2.14. Konfiguration einer Rohdatenvariable für B-Variablen

Name

Eindeutige Bezeichnung der Rohdatenvariablen.

ID

Eindeutige Identifikationsnummer der Rohdatenvariablen. Diese ID muss in TeVar für die Zuordnung von Variablen zu Rohdatenvariablen in der Spalte RAWDATA eingetragen werden.

Art

Unterscheidung nach R-, B- und P-Variable.

Startoffset im DB

Offset im DB an dem die die Daten der ersten dieser Rohdatenvariable zugeordneten B-Variable beginnen. Durch diese Einstellmöglichkeit können in einem DB mehrere Rohdatenvariablenvariablen definiert werden, die sich nur im Startoffset unterscheiden.

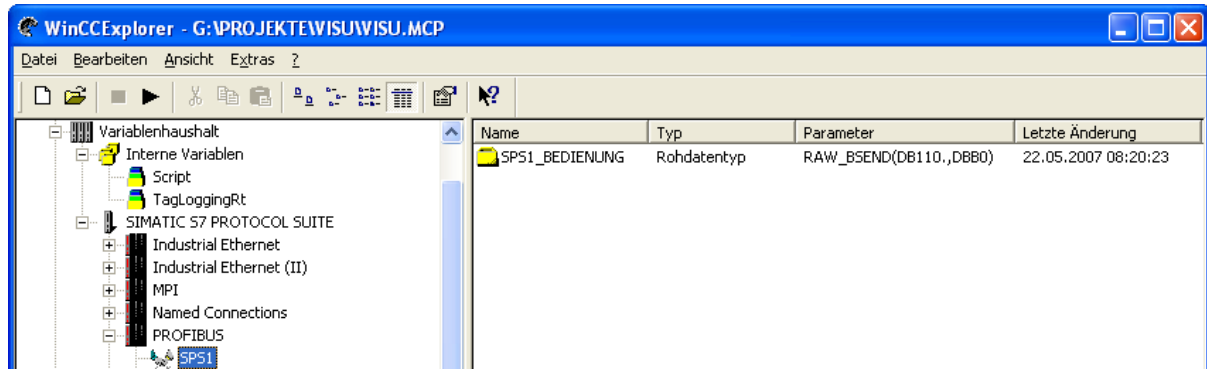
Beispiel:

Abbildung 2.15. Projektdatenbankausschnitt mit B-Variablen

+ ID	RESOURCE	NAME	TYP	OP	COMMENT	WINCC_VISU	RAWDATA
▶ 9	SPS1	06H102_EIN	BOOL	DB110.DBX0.0	HAND Bedienung	B	2
10	SPS1	06H103_EIN	BOOL	DB110.DBX10.0	HAND Bedienung	B	2
11	SPS1	06H104_EIN	BOOL	DB110.DBX20.0	HAND Bedienung	B	2
12	SPS1	06H105_EIN	BOOL	DB110.DBX30.0	HAND Bedienung	B	2

Auf der Grundlage der in den obigen Abbildungen dargestellten Variablendefinition wird im WinCC-Variablenhaushalt die in der folgenden Abbildung dargestellte Rohdatenvariable 'SPS1_BEDIENUNG' in der Verbindung 'SPS1' angelegt. Die Verbindung 'SPS1' muss manuell angelegt werden und als Rohdatenverbindung parametrisiert werden.

Abbildung 2.16. Erzeugte Rohdatenvariable

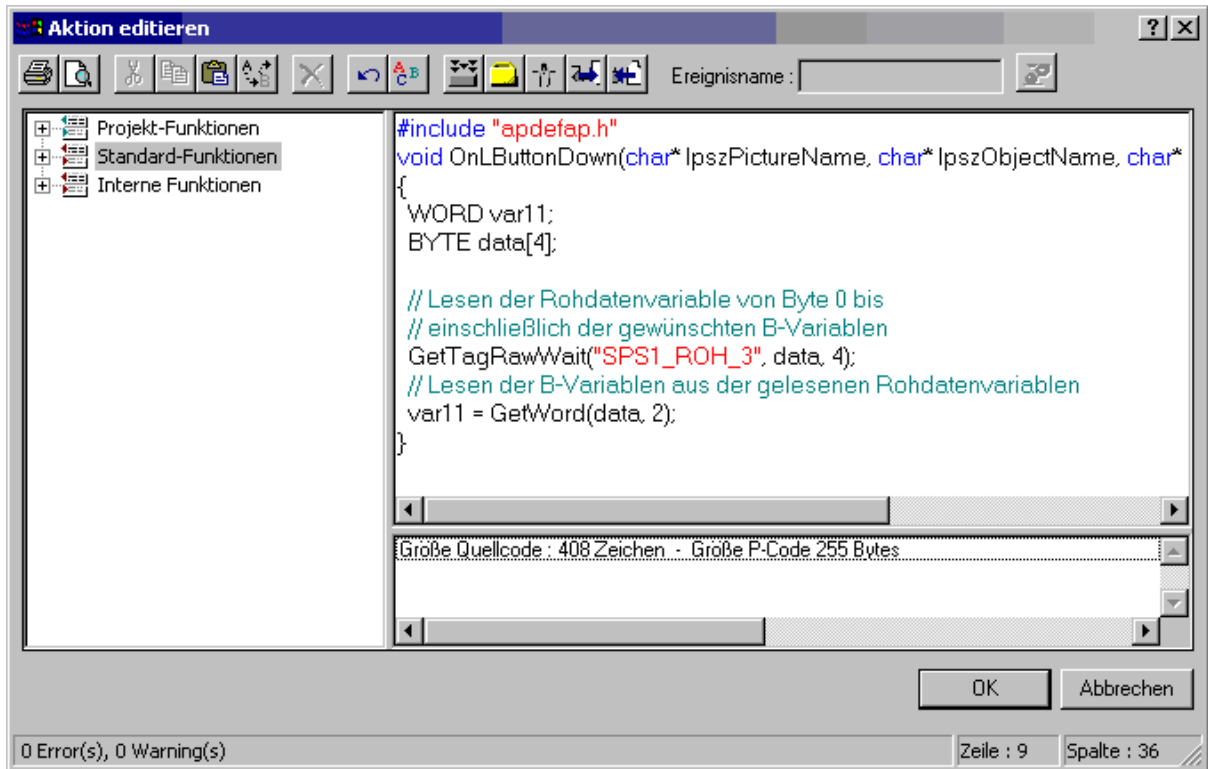


Lesen einer B-Variablen aus einer Rohdatenvariablen:

Abbildung 'Lesen einer B-Variablen' zeigt den Code für das Lesen, der in Abbildung 'Projektdatenbankausschnitt mit B-Variablen' dargestellten B-Variablen 'Var11' aus der Rohdatenvariablen 'SPS1_ROH_3'. Da 'Var11' Byte 2 und 3 von 'SPS1_ROH_3' belegt sind, müssen die ersten 4 Byte (Byte 0 bis 3) dieser Rohdatenvariable mit der Funktion 'GetTagRawWait' gelesen werden. Das Auslesen von 'Var11' aus den Rohdaten erfolgt mit dem Offset von 'Var11' in der Rohdatenvariable, mit den Argumenten 'data' (Rohdaten) und durch die Funktion 'GetWord'.

Zum Lesen von Variablen aus Rohdaten werden durch TeWinCC die Funktionen 'GetWord', 'GetDword' und 'GetFloat' im WinCC-Projekt angelegt. Diese Funktionen haben die Aufgabe, die typspezifische Anzahl Bytes ab einem vorgegebenen Offset aus einer Rohdatenvariablen zu lesen und deren Reihenfolge der Bytes vom Big-Endian Format auf Little-Endian Format zu ändern.

Abbildung 2.17. Lesen einer B-Variablen

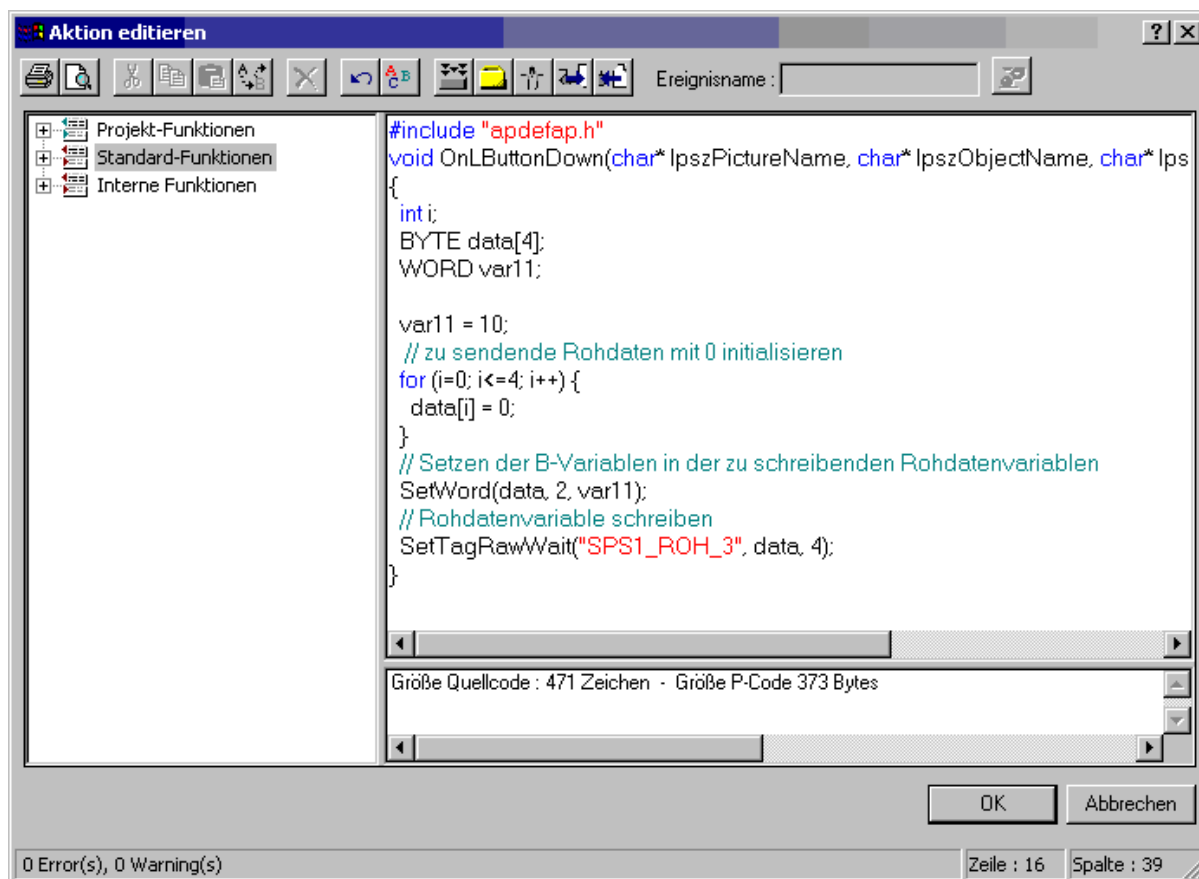


Schreiben einer B-Variablen in eine Rohdatenvariablen:

Abbildung 'Schreiben einer B-Variablen' zeigt den Code für das Schreiben, der in Abbildung 2 dargestellten B-Variablen 'Var11' in die Rohdatenvariable 'SPS1_ROH_3'. Dafür wird zunächst ein vier Byte grosser Bereich ('data') mit 0 initialisiert. Die Funktion 'SetWord' beschreibt dann das dritte und vierte Byte der Variable 'data' mit dem Inhalt von 'var11'. Die Funktion 'SetTagRawWait' beschreibt die ersten vier Byte der Rohdatenvariable 'SPS1_ROH_3' mit dem Inhalt von 'data'. In diesem Beispiel wurde zunächst die Variable 'data' mit 0 initialisiert. Es ist auch möglich, zunächst in diese Variable mit der Funktion 'GetTagRawWait' den aktuellen Inhalt der Rohdatenvariable 'SPS1_ROH_3' einzulesen.

Zum Schreiben von Variablen in Rohdaten werden durch TeWinCC die Funktionen 'SetWord', 'SetDword' und 'SetFloat' im WinCC-Projekt angelegt. Diese Funktionen haben die Aufgabe, die typspezifische Anzahl Bytes ab einem vorgegebenen Offset in eine Rohdatenvariable zu schreiben und deren Reihenfolge vom Little-Endian Format auf Big-Endian Format zu konvertieren.

Abbildung 2.18. Schreiben einer B-Variablen

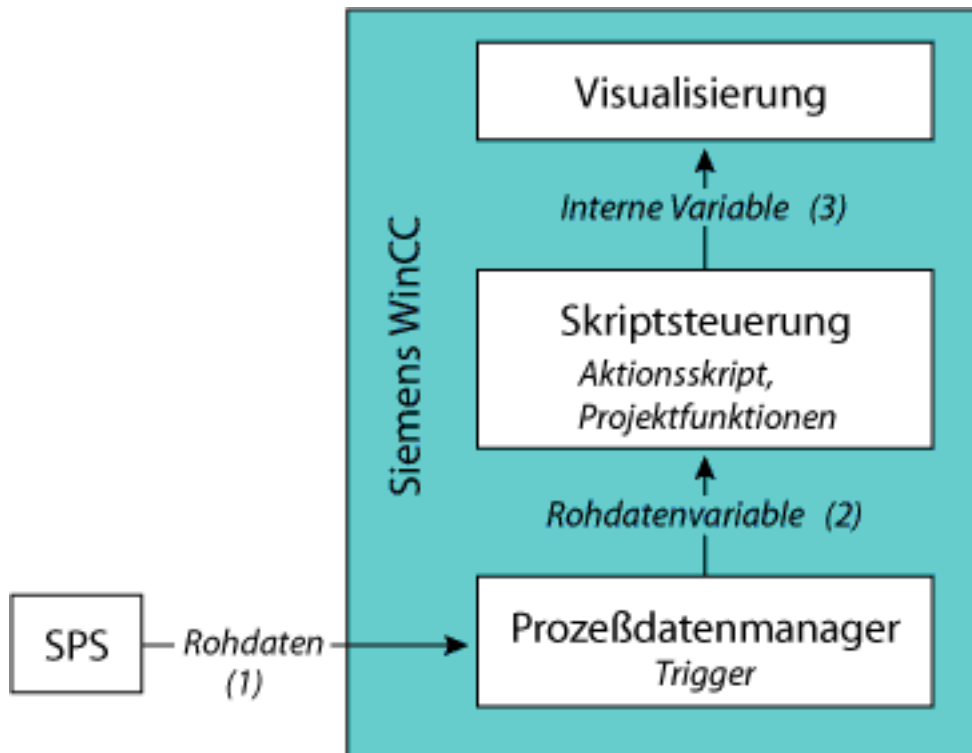


2.2.2.3. P-Variablen

P-Variablen werden im WinCC-Variablenhaushalt als interne Variablen mit dem in der Projektdatenbank definierten Datentyp angelegt. Mit Hilfe eines WinCC-Aktionsskriptes werden Werte einer Rohdatenvariablen auf diese internen Variablen abgebildet.

Aus den Informationen, der in der Projektdatenbank definierten P- und Rohdatenvariablen, werden alle benötigten Variablen, Skripte, Trigger, Funktionen und Datenbausteine durch TeWinCC erzeugt. Im folgenden wird die Rohdatenkommunikation mit P-Variablen beschrieben.

Abbildung 2.19. Schema der Rohdatenkommunikation mit P-Variablen



Die durch TeWinCC generierte Rohdatenvariable wird vom WinCC-Prozessdatenmanager zyklisch gelesen (1). Die vom Prozessdatenmanager empfangene Rohdatenvariable ist als Trigger für ein Aktionsskript definiert. Das verknüpfte Aktionsskript wird durch die Skriptsteuerung ausgeführt (2) und bildet die Rohdaten der Rohdatenvariable auf interne Variablen ab (3).

Die den internen Variablen zugewiesenen Werte können in der Visualisierung lesend verwendet werden.

Die Parametrierung von Rohdatenvariablen erfolgt in einem separaten Konfigurationsdialog. Dieser Dialog kann in TeWinCC über die Schaltfläche **Konfigurieren** neben der Exportkonfiguration für Rohdatenvariablen geöffnet werden. Zum Editieren der Eigenschaften einer Rohdatenvariablen kann in der dargestellten Auswahlliste durch Doppelklick auf eine Rohdatenvariable deren Konfiguration geöffnet werden. Wurde die ausgewählte Rohdatenvariable für die Verwendung mit P-Variablen konfiguriert erscheint der folgende Dialog.

Abbildung 2.20. Konfiguration einer Rohdatenvariable für P-Variablen

Rohdatenvariablenkonfiguration

Name:

ID:

Art: R-Variable B-Variable P-Variable

Rangier FC:
 direkt

Rangier DB:

Totband:

OK

Name

Eindeutige Bezeichnung der Rohdatenvariablen.

ID

Eindeutige Identifikationsnummer der Rohdatenvariablen. Diese ID muss in TeVar für die Zuordnung von Variablen zu Rohdatenvariablen in der Spalte RAWDATA eingetragen werden.

Art

Unterscheidung nach R-, B- und P-Variable.

Rangier FC

Dieser FC rangiert die der Rohdatenvariable zugeordneten P-Variablen in den Rangier-DB.

Rangier DB

Dieser DB enthält die durch den Rangier-FC rangierten P-Variablen.

Totband

Minimale Änderung des Wertes einer R-Variablen, die zum Senden der Rohdatenvariable führt. Liegt die Wertänderung unter dem Wert des eingestellten Totbandes wird die Rohdatenvariable nicht gesendet.

direkt

Ist der Wert dieses Feldes 1, so besteht die Rohdatenvariable aus einem zusammenhängenden Bereich eines Datenbausteins. Ist der Wert dieses Feldes 0, so werden die einer Rohdatenvariablen

zugeordneten Elemente in den Rangier-DB kopiert. Die Grösse einer als 'direct' gekennzeichneten Rohdatenvariablen ist auf 400Byte beschränkt. Soll ein Datenbausteinbereich mit einer Grösse von mehr als 400Byte als Rohdatenvariable verwendet werden, so muss dieser Bereich auf mehrere Rohdatenvariablen aufgeteilt werden. Dafür muss die Spalte 'dboffset' angepasst werden. Für nicht als 'direct' gekennzeichnete Rohdatenvariablen beträgt die maximale Grösse 64KB.

Beispiel:

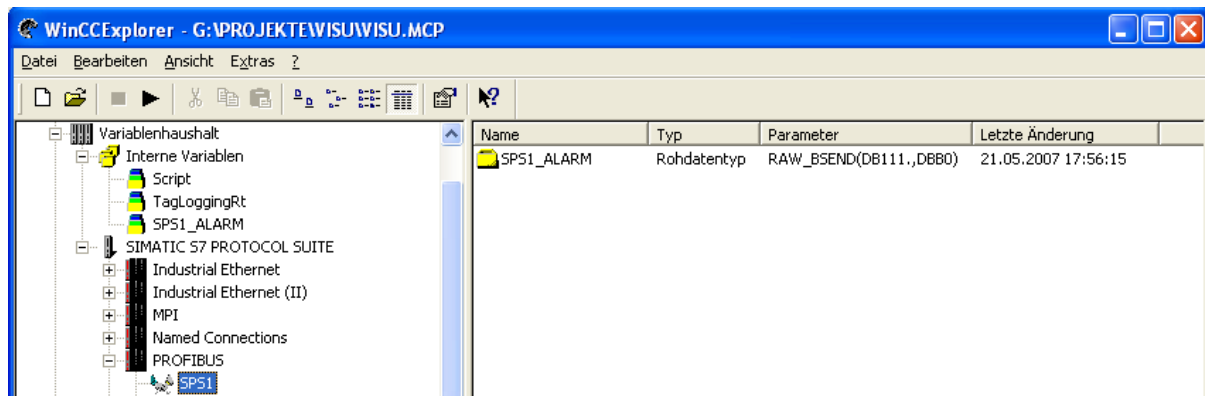
Abbildung 2.21. Projektdatenbankausschnitt mit P-Variablen

+ ID	RESOURCE	NAME	TYP	OP	COMMENT	WINCC_VISU	RAWDATA
▶ 13	SPS1	06H102_STOER	BOOL	DB111.DBX0.0	06H102 Sammelstörung	PA(CLASS:1,TYPE:1)	3
14	SPS1	06H103_STOER	BOOL	DB111.DBX10.0	06H103 Sammelstörung	PA(CLASS:1,TYPE:1)	3
15	SPS1	06H104_STOER	BOOL	DB111.DBX20.0	06H104 Sammelstörung	PA(CLASS:1,TYPE:1)	3
16	SPS1	06H105_STOER	BOOL	DB111.DBX30.0	06H105 Sammelstörung	PA(CLASS:1,TYPE:1)	3

Die Abbildung 'Projektdatenbankausschnitt mit P-Variablen' zeigt die Definitionen von vier P-Variablen. Diese sind über die Datenbankspalte 'RAWDATA' der Rohdatenvariablen mit der Nummer 3 ('SPS1_ALARM') zugeordnet. Der Eintrag 'P' in der Spalte 'WINCC_VISU' kennzeichnet die dargestellten Variablen als P-Variablen, zusätzlich sind in diesem Beispiel die Variablen durch die Kennzeichnung mit 'A(CLASS:1,TYPE:1)' als Alarmvariablen konfiguriert. Die Definition der Rohdatenvariablen 'SPS1_ALARM' wird in der ersten Abbildung gezeigt.

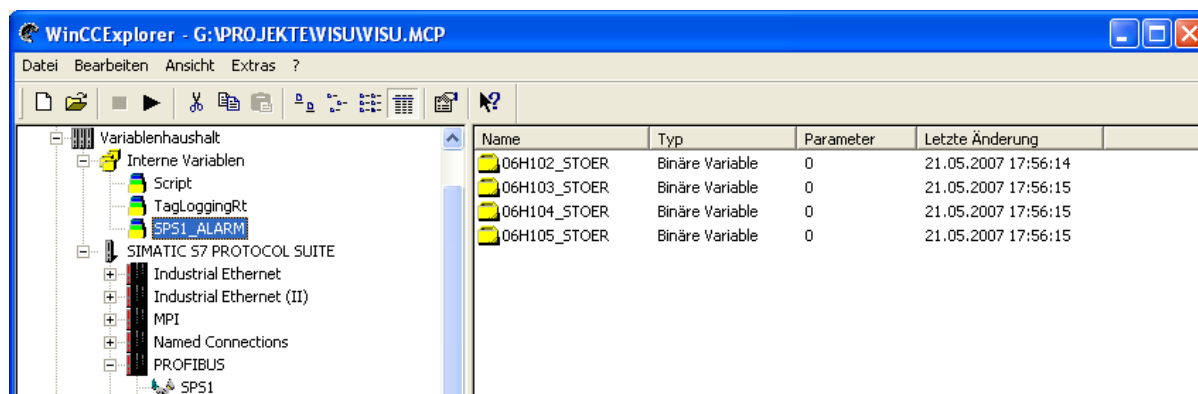
Beim Export der Rohdatenvariablen mit TeWinCC wird im WinCC-Variablenhaushalt unter der Verbindung 'SPS1' die Rohdatenvariable 'SPS1_ALARM' angelegt. Die Verbindung 'SPS1' muss manuell angelegt werden und als Rohdatenverbindung parametrisiert werden.

Abbildung 2.22. Rohdatenvariable 'SPS1_ALARM'



Die Abbildung 'Exportierte P-Variablen im WinCC-Variablenhaushalt' zeigt die, für die definierten P-Variablen erzeugten internen Variablen im Variablenhaushalt von WinCC. Dabei werden die erzeugten internen Variablen einer zur Rohdatenvariable gleichnamigen Variablengruppe zugeordnet. Hier im Beispiel ist das die Gruppe 'SPS1_ALARM'.

Abbildung 2.23. Exportierte P-Variablen im WinCC-Variablenhaushalt



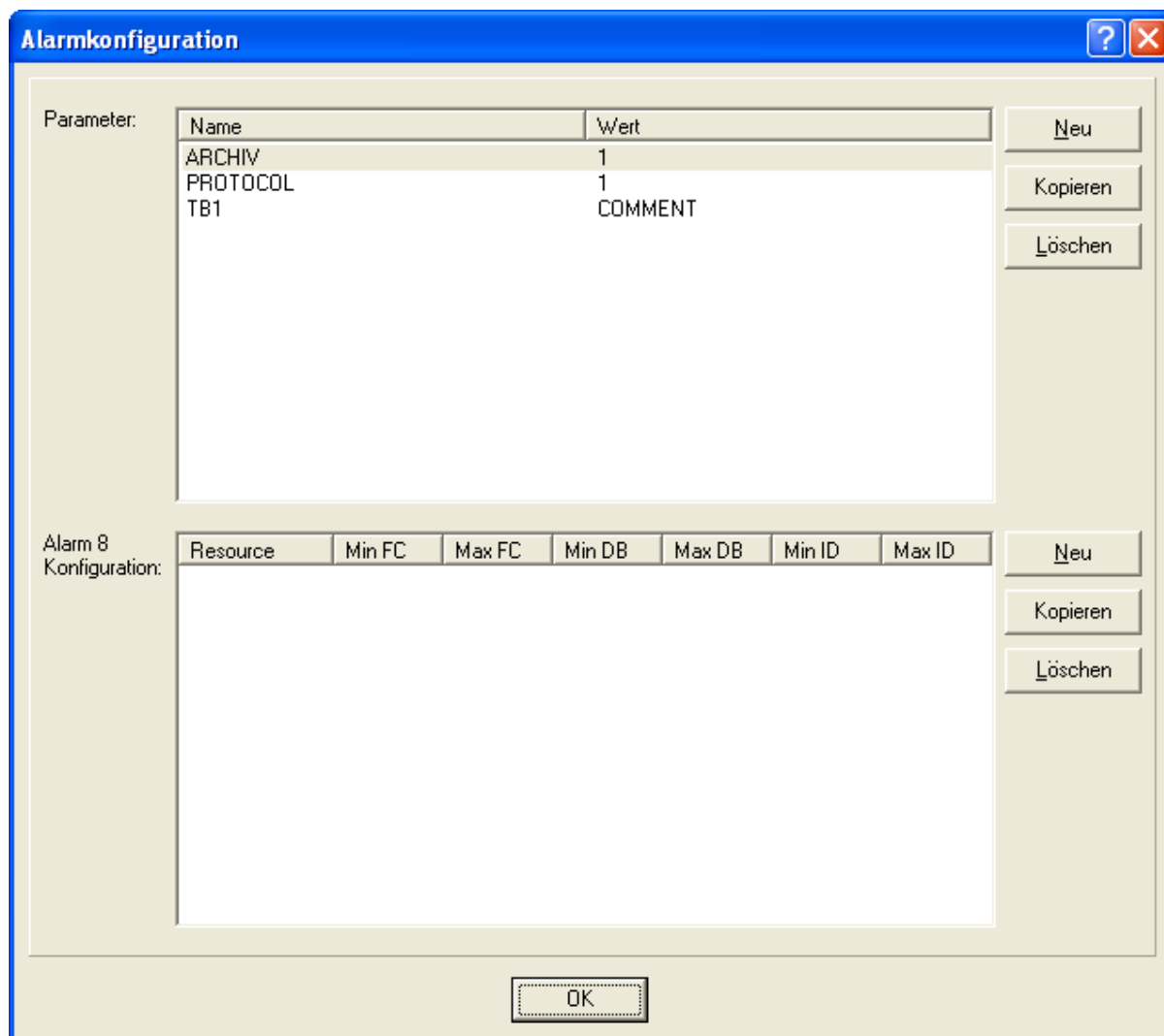
Für die Abbildung der Daten aus der Rohdatenvariable auf interne Variablen wird ein Aktionskript mit dem Namen der zugeordneten Rohdatenvariable generiert. Dieses Aktionskript wird zyklisch ausgeführt, sodass bei jeder Änderung der Rohdatenvariable deren Inhalt durch das Skript auf die internen Variablen abgebildet wird. Für diese Abbildung werden zusätzlich die Projektfunktionen 'GetDword', 'GetFloat' und 'GetWord' in der WinCC-Skriptsteuerung erzeugt, welche eine Wandlung des in der SPS verwendeten Big-Endian Formates in das durch Intel-Hardware verwendete Little-Endian Format durchführen.

2.2.3. Alarmvariablen

TeWinCC unterstützt das Bitmelde- und das Meldenummernverfahren (Alarm 8). Sollen Variablen als Alarme exportiert werden, dann müssen die betreffenden Variablen in der entsprechenden Exportspalte mit 'A' gekennzeichnet werden, dabei muss die zu meldende Variable den Datentyp BOOL haben. Für jede Einzelmeldung kann eine eigene Meldevariable verwendet werden. In diesem Fall muß die Meldevariable zusätzlich als V-, P- oder R-Variable gekennzeichnet werden bzw. einer Alarm-8-Konfiguration zugeordnet sein. Alternativ ist es möglich mehrere Meldungen einer übergeordneten Meldevariable z.B. mit dem Datentyp WORD zuzuordnen. Die einzelnen Meldungen werden dann durch ein Meldebit eindeutig. Die übergeordnete Meldevariable muß als V-, P- oder R-Variable gekennzeichnet werden. Die genannten Alternativen werden anhand von Beispielen im nächsten Kapitel beschrieben. In diesem Kapitel wird die allgemeine Konfiguration des Meldungsexports erklärt.

Die allgemeine Konfiguration des Alarmexports erfolgt über einen separaten Dialog in TeWinCC. Über die Schaltfläche Konfigurieren neben der Exportkonfiguration für Meldungen kann dieser geöffnet werden.

Abbildung 2.24. Alarmkonfiguration



Parameter

Die in der Abbildung dargestellte Parameterierung definiert z.B. dass Meldungen archiviert ('ARCHIV:1') und protokolliert ('PROTOCOL:1') werden. Zusätzlich soll der Text aus der Spalte COMMENT im WinCC-AlarmLogging dem Textblock 1 (TB1:COMMENT) zugewiesen werden. An dieser Stelle können die Alarmparameter der folgenden Tabelle angegeben werden.

Tabelle 2.4. Alarmparameter

Parametername	Beschreibung
MSGID	Nummer der Meldung, standardmäßig wird der Wert der Spalte 'ID' (Sonderfunktionen, z.B. für die Verwendung von Alarm-8 siehe unten)
CLASS	Klasse der Meldung als Nummer
TYPE	Art der Meldung als Nummer
SQUIT	Meldung ist einzelquittierpflichtig [0,1]
HORN	Meldung aktiviert zentralen Melder [0,1]
ARCHIV	Meldung wird archiviert [0,1]

Parametername	Beschreibung
PROTOCOL	Meldung wird protokolliert [0,1]
NEG	Meldung wird mit fallender Flanke ausgewertet, standardmäßig wird der Wert der Spalte 'NEG' ausgewertet [0,1]
LOCKED	Meldung ist beim Anlauf gesperrt [0,1]
TB1 ... TB10	Textblock 1-10
QUITVAR	Quittvariable
QUITBIT	Quittierbit innerhalb der Quittvariable
STATEVAR	Zustandsvariable
STATEBIT	Zustandsbit innerhalb der Zustandsvariable
AG	AG-Nummer
CPU	AG-Subnummer
INFOTEXT	Infotext
ACTIONTYPE	Aktionstyp
FUNCTION	Funktionsname für LoopIn-Alarm
PICTURE	Bildname für LoopIn-Alarm
NORMDLL	NormierDll
GROUPID	Gruppenkennung
GROUP	Gruppenname
CREATORID	Erzeugerkennung
VAR	Variablenbezeichner für die Meldevariable (Hier kann eine Rohdatenvariable für Eventgesteuertes Melden angegeben werden, außerdem sollte dann eine NormierDll angegeben sein).

Diese allgemeingültige Parametrierung kann für jede Meldung in einer Exportspalte der Projektdatenbank angepaßt werden. Dazu können in Klammern im Format 'PARAMETER:WERT' durch Komma getrennt einzelnen Parametern Werte zugewiesen werden. Z.B. wird durch die Definition 'A(CLASS:1,TYPE:1)' eine Meldung der Klasse 1 und Art 1 zugeordnet.

Alarm-8-Konfiguration

TeWinCC unterstützt die Generierung der Alarmkommunikation zwischen WinCC und einer SPS mit Hilfe des 'ALARM 8' Bausteins. Für jede Resource kann eine Alarm 8 Konfiguration angegeben werden. über die Schaltfläche **Neu** kann eine neue Konfiguration angelegt werden, mit der Schaltfläche **Kopieren** kann die aktuell ausgewählte Konfiguration kopiert werden, **Löschen** entfernt die ausgewählte Konfiguration. Nach einem Doppelklick auf eine Konfiguration können in dem sich öffnenden Dialog die Einstellungen editiert werden.

Abbildung 2.25. Alarmkonfiguration Meldenummernverfahren

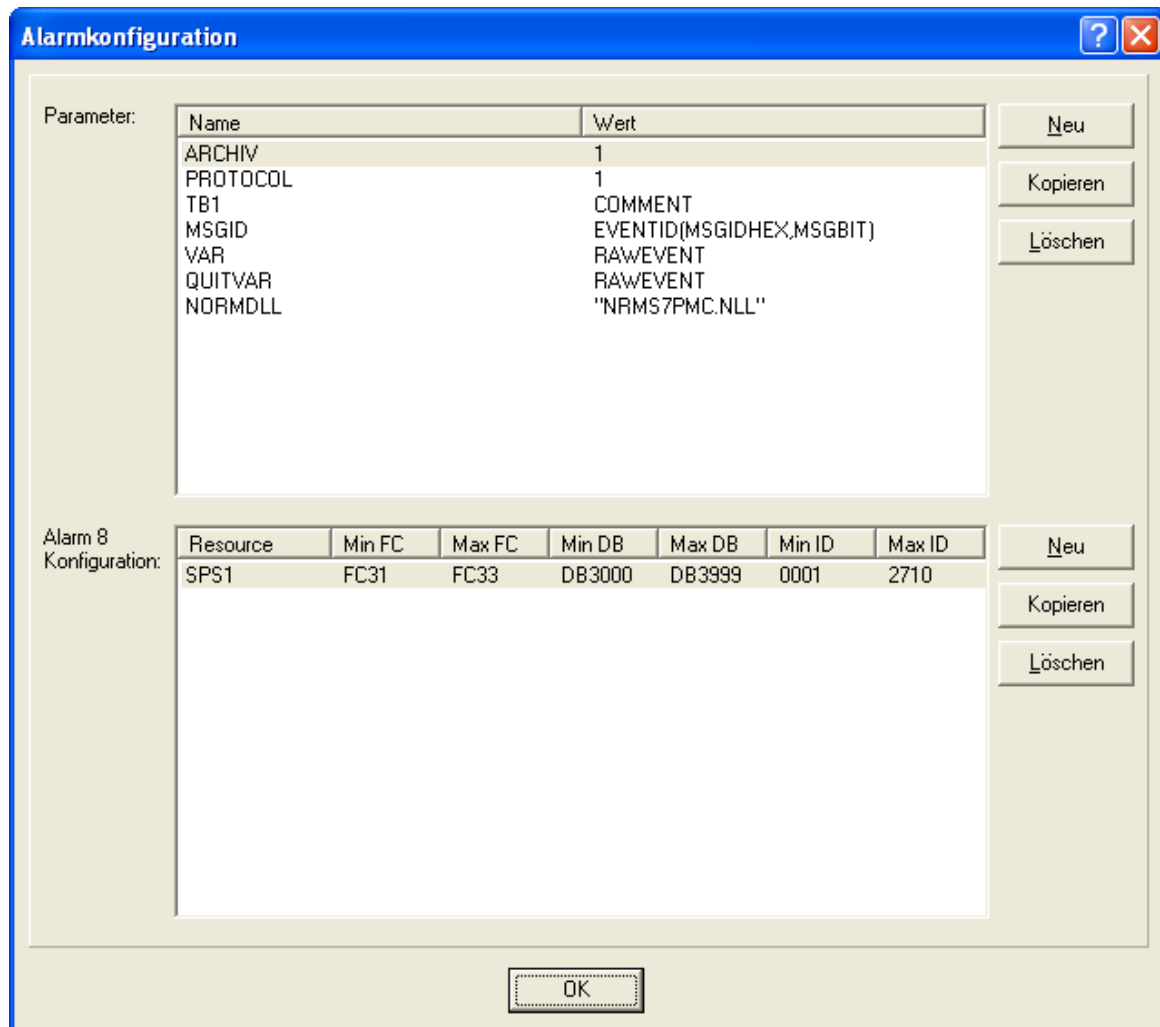


Abbildung 2.26. Alarm 8 Konfiguration

Alarm 8-Konfiguration

Resource: SPS1

Min. Alarm FC: FC31

Max. Alarm FC: FC33

Min. Alarm DB: DB3000

Max. Alarm DB: DB3999

Min. MSGIDHEX: 0001

Max. MSGIDHEX: 2710

OK Abbrechen

Resource

Resource der aktuellen Konfiguration

Min. Alarm FC

Nummer der ersten für die Alarmkommunikation reservierten Funktion, jede dieser Funktionen beinhaltet den Code für die Auslösung von jeweils 200 Alarmen.

Max. Alarm FC

Nummer der letzten für die Alarmkommunikation reservierten Funktion

Min. Alarm DB

Nummer des ersten für die Alarmkommunikation reservierten Datenbausteins

Max. Alarm DB

Nummer des letzten für die Alarmkommunikation reservierten Datenbausteins

Min. MSGIDHEX

Hexadezimale Nummer der ersten automatisch vergebenen MSGIDHEX, diese Nummer wird in der automatisch angelegten Spalte 'MSGIDHEX' gespeichert. Jeweils 8 Alarmen wird die selbe MSGIDHEX zugeordnet, wobei jeder dieser 8 Alarme eine Bitnummer (0-7) in der automatisch angelegten Spalte 'MSGBIT' zugeordnet wird. Der an dieser Stelle angegebene Wert muß mit dem in Step7 projektierten Wert für den Start des Meldenummernbereiches übereinstimmen.

Max. MSGIDHEX

Hexadezimale Nummer der letzten automatisch vergebenen MSGIDHEX. Der an dieser Stelle angegebene Wert muß mit dem in Step7 projektierten Wert für das Ende des Meldenummernbereiches übereinstimmen.

2.2.3.1. Bitmeldeverfahren

In diesem Abschnitt werden die unterstützten Alternativen bei der Verwendung des Bitmeldeverfahrens beschrieben.

Bitmeldeverfahren mit V-Variablen

Bei diesem Verfahren wird die jeder Meldung zugeordnete Meldevariable als externe Variable im WinCC-Variablenhaushalt angelegt. Die zu exportierenden Meldungen müssen mit 'VA' in der Exportspalte gekennzeichnet werden. Die folgenden beiden Abbildungen zeigen die Konfiguration in der Projektdatenbank und die in WinCC angelegten externen Variablen.

Abbildung 2.27. Projektdatenbank Bitmeldeverfahren mit V-Variablen

ID	RESOURCE	NAME	TYP	DP	COMMENT	WINCC_VISU
13	SPS1	06H102_STOER	BOOL	DB111.DBX0.0	06H102 Sammelstörung	VA(CLASS:1,TYPE:1)
14	SPS1	06H103_STOER	BOOL	DB111.DBX0.1	06H103 Sammelstörung	VA(CLASS:1,TYPE:1)
15	SPS1	06H104_STOER	BOOL	DB111.DBX0.2	06H104 Sammelstörung	VA(CLASS:1,TYPE:2)
16	SPS1	06H105_STOER	BOOL	DB111.DBX0.3	06H105 Sammelstörung	VA(CLASS:1,TYPE:2)

Abbildung 2.28. Variablenhaushalt Bitmeldungen mit V-Variablen

The screenshot shows the WinCC Explorer interface for the project 'G:\PROJEKTE\WISU\WISU.MCP'. The 'Variable Declaration' table is visible, listing the variables defined in the project:

Name	Typ	Parameter	Letzte Änderung
06H102_STOER	Binäre Variable	DB111,D0.0	08.06.2007 14:35:48
06H103_STOER	Binäre Variable	DB111,D0.1	08.06.2007 14:35:48
06H104_STOER	Binäre Variable	DB111,D0.2	08.06.2007 14:35:48
06H105_STOER	Binäre Variable	DB111,D0.3	08.06.2007 14:35:48

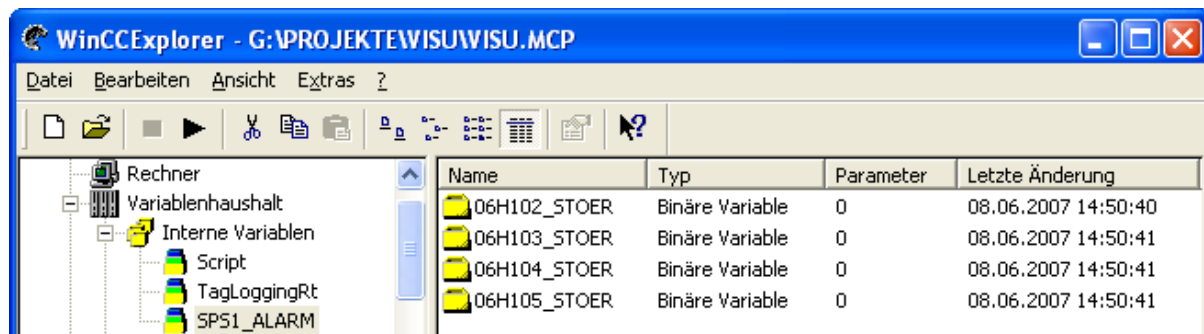
Bitmeldeverfahren mit P-Variablen

Bei diesem Verfahren werden die den Meldungen zugeordneten Meldevariablen durch eine Rohdatenkommunikation mit P-Variablen auf interne Variablen abgebildet. Das Bitmeldeverfahren mit **P-Variablen** kann z.B. dann angewendet werden, wenn die eingesetzte SPS den für eine Rohdatenkommunikation mit R-Variablen benötigten BSEND-Funktionsbaustein nicht unterstützt. Die folgenden beiden Abbildungen zeigen die Konfiguration in der Projektdatenbank und die in WinCC angelegten internen Variablen. Zusätzlich wird für die Rohdatenkommunikation ein WinCC-Skript sowie eine zugehörige Rohdatenvariable angelegt.

Abbildung 2.29. Projektdatenbank Bitmeldeverfahren mit P-Variablen

ID	RESOURCE	NAME	TYP	OP	COMMENT	WINCC_VISU	+ RAWDATA
13	SPS1	06H102_STOER	BOOL	DB111.DBX0.0	06H102 Sammelstörung	PA(CLASS:1,TYPE:1)	3
14	SPS1	06H103_STOER	BOOL	DB111.DBX0.1	06H103 Sammelstörung	PA(CLASS:1,TYPE:1)	3
15	SPS1	06H104_STOER	BOOL	DB111.DBX0.2	06H104 Sammelstörung	PA(CLASS:1,TYPE:2)	3
16	SPS1	06H105_STOER	BOOL	DB111.DBX0.3	06H105 Sammelstörung	PA(CLASS:1,TYPE:2)	3

Abbildung 2.30. Variablenhaushalt Bitmeldeverfahren mit P-Variablen



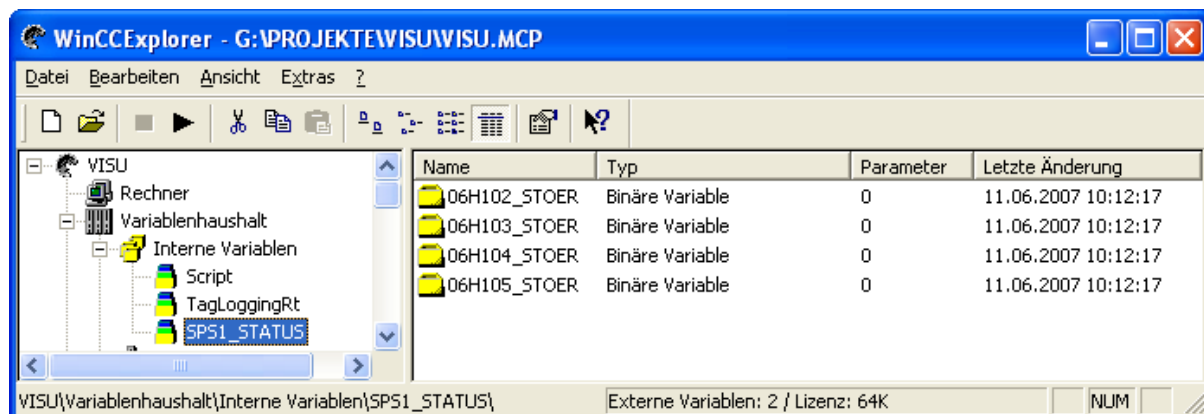
Bitmeldeverfahren mit R-Variablen

Bei diesem Verfahren werden die den Meldungen zugeordneten Meldevariablen durch eine [Rohdatenkommunikation mit R-Variablen](#) auf interne Variablen abgebildet. Für die Verwendung dieses Meldeverfahrens muß die eingesetzte SPS den Funktionsbaustein BSEND unterstützen. Die folgenden beiden Abbildungen zeigen die Konfiguration in der Projektdatenbank und die in WinCC angelegten internen Variablen. Zusätzlich wird eine entsprechende Rohdatenvariable, ein Funktionsbaustein und ein Datebaustein für die Rohdatenkommunikation zwischen SPS und WinCC angelegt.

Abbildung 2.31. Projektdatenbank Bitmeldeverfahren mit R-Variablen

ID	RESOURCE	NAME	TYP	OP	COMMENT	WINCC_VISU	RAWDATA
13	SPS1	06H102_STOER	BOOL	DB111.DBX0.0	06H102 Sammelstörung	RA(CLASS:1,TYPE:1)	1
14	SPS1	06H103_STOER	BOOL	DB111.DBX0.1	06H103 Sammelstörung	RA(CLASS:1,TYPE:1)	1
15	SPS1	06H104_STOER	BOOL	DB111.DBX0.2	06H104 Sammelstörung	RA(CLASS:1,TYPE:2)	1
16	SPS1	06H105_STOER	BOOL	DB111.DBX0.3	06H105 Sammelstörung	RA(CLASS:1,TYPE:2)	1

Abbildung 2.32. Variablenhaushalt Bitmeldeverfahren mit R-Variablen



Bitmeldeverfahren mit übergeordneter Meldevariablen

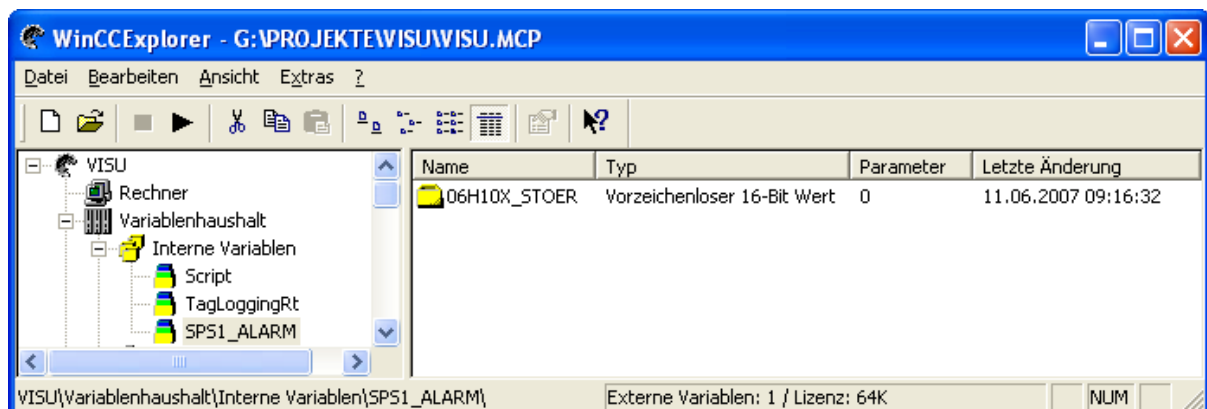
Bei den beschriebenen Bitmeldeverfahren mit V-, P-, und R-Variablen können durch die Angabe einer übergeordneten Meldevariablen mehrere binäre Einzelmeldungen zu einer Meldevariable zusammengefaßt werden. Die folgende Abbildung zeigt eine solche Konfiguration. Hier wurde für die dargestellten Meldungen die übergeordnete WORD-Variable '06H10X_STOER' angelegt, welche den Speicherbereich (DBW0) der einzelnen Meldungen (DBX0.0-DBX0.3) umfaßt. Dabei wurden die Meldungen nur mit 'A' und die eigentliche Meldevariable mit 'P' gekennzeichnet. Die Kennzeichnung der Meldevariable als R- oder V-Variable wäre ebenfalls möglich. Wegen der Speicherbereichsüberdeckung der binären Meldungen mit der WORD-Meldevariablen sind diese Variable in der Spalte TEMP mit '1' zu kennzeichnen. Bei Verwendung von Strukturen können z.B. verschiedene Strukturelemente als Meldungen ('A') gekennzeichnet werden und die Strukturvariable als Meldevariable ('P', 'R' oder 'V'), in diesem Fall entfällt der Eintrag in der TEMP-Spalte.

Abbildung 2.33. Projektdatenbank Bitmeldeverfahren mit übergeordneter Meldevariablen

ID	RESOURCE	NAME	TYP	OP	COMMENT	WINCC_VISU
13	SPS1	06H102_STOER	BOOL	DB111.DBX0.0	06H102 Sammelstörung	A(CLASS:1,TYPE:1)
14	SPS1	06H103_STOER	BOOL	DB111.DBX0.1	06H103 Sammelstörung	A(CLASS:1,TYPE:1)
15	SPS1	06H104_STOER	BOOL	DB111.DBX0.2	06H104 Sammelstörung	A(CLASS:1,TYPE:2)
16	SPS1	06H105_STOER	BOOL	DB111.DBX0.3	06H105 Sammelstörung	A(CLASS:1,TYPE:2)
18	SPS1	06H10X_STOER	WORD	DB111.DBW0		P

Die folgende Abbildung zeigt die in WinCC angelegte interne Meldevariable. Ihr Wert wird durch Rohdatenkommunikation mit P-Variablen von der SPS auf die interne Variable abgebildet.

Abbildung 2.34. Variablenhaushalt Bitmeldeverfahren mit übergeordneter Meldevariablen



2.2.3.2. Meldenummernverfahren

Meldenummernverfahren (Alarm-8)

Beim Meldenummernverfahren wird der Zeitstempel einer Meldung in der SPS gebildet und durch einen Alarm-8 Bausteinaufruf an WinCC übermittelt. In TeWinCC muß für jede Resource eine eigene Alarm-8-Konfiguration angelegt werden, dabei werden automatisch in der Projektdatenbank die Spal-

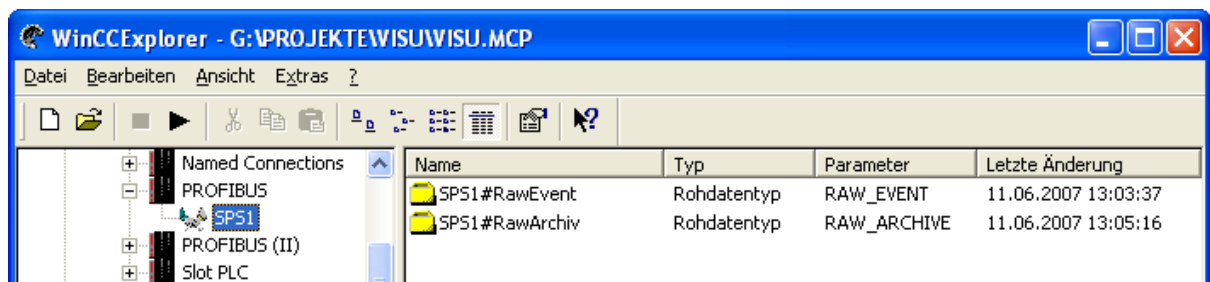
ten MSGIDHEX, MSGBIT, RAWEVENT und FMV (Fehlmeldeverhinderung) eingefügt. Meldungen können durch die Kennzeichnung mit 'A' in der Exportspalte und durch die Angabe der Rohdatenvariable (z.B. 'SPS1#RawEvent') in der Spalte RAWEVENT einer Alarm-8-Konfiguration zugeordnet werden. Die in der Spalte RAWEVENT angegebene Rohdatenvariable muß im WinCC-Variablenhaushalt manuell angelegt werden. Die Einträge in den Spalten MSGIDHEX und MSGBIT werden während des Alarmexports automatisch angelegt.

Abbildung 2.35. Projektdatenbank Meldenummervverfahren

+ TYP	OP	COMMENT	WINCC_VISU	MSGIDHEX	MSGBIT	RAWEVENT
▶ BOOL	DB111.DBX0.0	06H102 Sammelstörung	A(CLASS:1,TYPE:1)	0001	0	SPS1#RawEvent
▶ BOOL	DB111.DBX0.1	06H103 Sammelstörung	A(CLASS:1,TYPE:1)	0001	1	SPS1#RawEvent
▶ BOOL	DB111.DBX0.2	06H104 Sammelstörung	A(CLASS:1,TYPE:2)	0001	2	SPS1#RawEvent
▶ BOOL	DB111.DBX0.3	06H105 Sammelstörung	A(CLASS:1,TYPE:2)	0001	3	SPS1#RawEvent

Die in der folgenden Abbildung dargestellten Rohdatenvariablen 'SPS1#RawEvent' und 'SPS1#RawArchiv' müssen manuell angelegt werden.

Abbildung 2.36. Variablenhaushalt Meldenummervverfahren

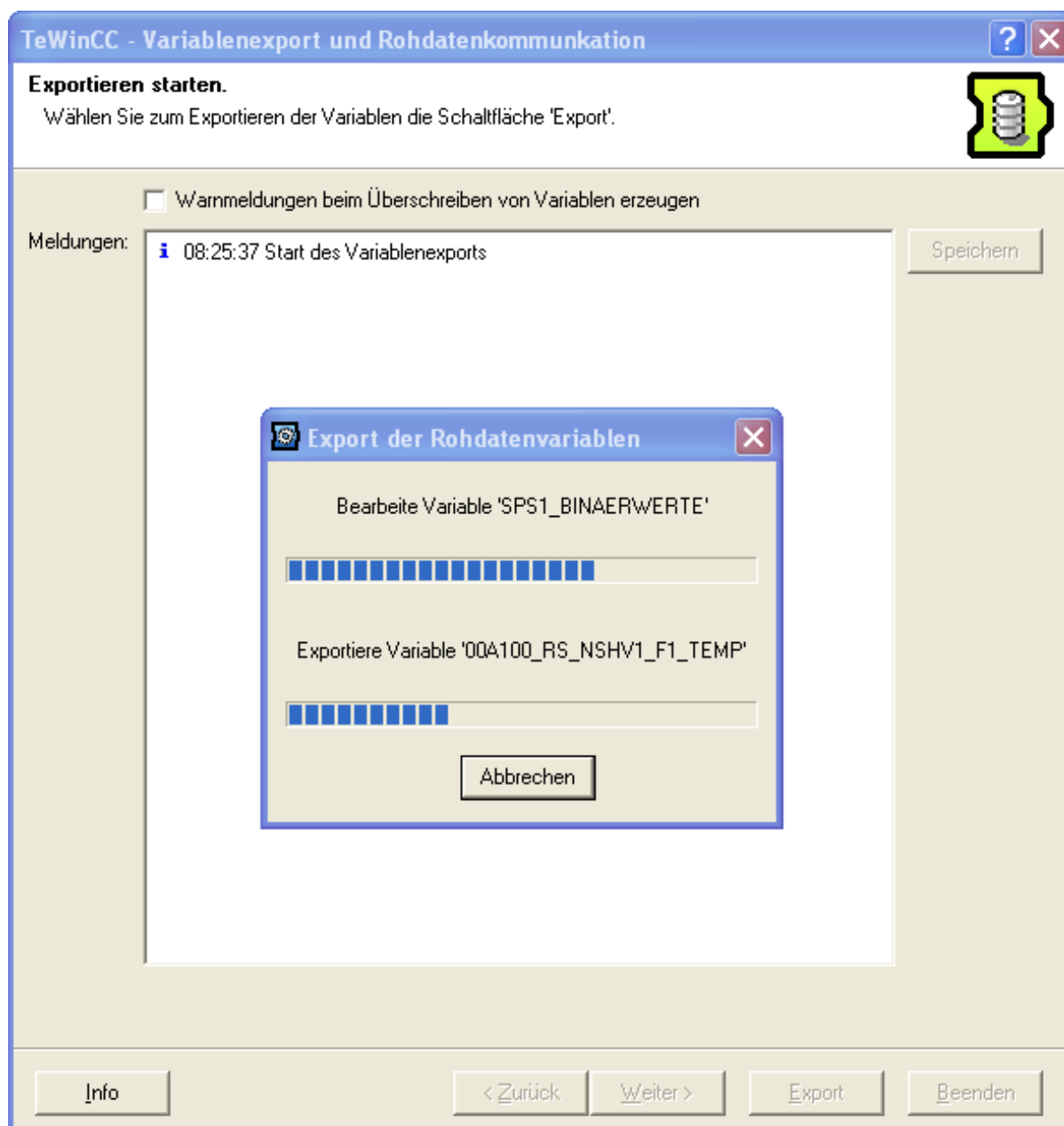


Beim Meldenummervverfahren wird die eindeutige MSGID einer Meldung durch die Funktion EVENTID aus den Einträgen der Spalten MSGIDHEX und MSGBIT gebildet. Dies erfolgt nach der folgenden Formel: $0x34000000 | ((MSGBIT * 2) \ll 20) | MSGIDHEX$

2.3. Export

Auf der zweiten Seite des Wizards kann über die Schaltfläche Export der Variablenexport gestartet werden. Mit der Schaltfläche < **Zurück** kann zur Konfiguration der Parameter gewechselt werden.

Abbildung 2.37. Exportieren der Variablen und Meldungen



Diese Wizard-Seite enthält die folgenden Einstellungen.

Warnmeldungen beim Überschreiben von Variablen erzeugen

Durch die Aktivierung dieser Option wird für jede überschriebene Variable ein Meldung erzeugt und in der Meldungsliste von TeWinCC ausgegeben.

Meldungen

Alle während des Exports aufgetretenen Hinweismeldungen, Warnungen und Fehlermeldungen werden in einer Meldungsliste ausgegeben. Mit der Schaltfläche Speichern können diese Meldungen als Textdatei gespeichert werden. Diese Funktion ist vor allem für die Fehlersuche nützlich. Der Export kann über die Schaltfläche Abbrechen in der Fortschrittsanzeige abgebrochen werden. Die bis zum Zeitpunkt des Abbruchs vorgenommen Änderungen im WinCC-Variablenhaushalt bleiben erhalten.

Kapitel 3. System Informationen

3.1. Technische Voraussetzungen

- Betriebssysteme: Windows 2000, Windows XP, Windows 2003

für TeList ist ein installiertes Microsoft-Excel ab Version 97 Voraussetzung, Microsoft-Word für die Dokumentation

(Prinzipiell ist die TeRANiS-Gruppe auf Systemen ab Windows 98 oder Windows NT lauffähig, ggf. müssen Treiber für die Datenbank (ADO-Jet) nachinstalliert werden. Es wird jedoch keine Garantie für die uneingeschränkte Benutzbarkeit übernommen.)
- PC, auf denen die angegebenen Betriebssysteme arbeiten, Bildschirm/Grafikkarte ab 800x600 Punkte
- Für die Online-Darstellung ist eventuell weitere Hard- und Software notwendig, die abhängig von der Zielplattform ist

Weitere Empfehlungen:

- möglichst hohe Bildschirmauflösung
- Microsoft -Excel und -Access können zur Erweiterung der Funktionalität von TeVar verwendet werden.
- Insbesondere für grössere Projekte sollte genügend Arbeitsspeicher vorhanden sein. (>128MB)

3.2. Installation und Deinstallation

Sofern Ihr System CDs automatisch abspielt, brauchen Sie nur die TeRANiS CD einzulegen.

Die Installation erfolgt durch Starten der Teranis.msi-Datei, die sich auf der TeRANiS-CD befindet, oder die Sie auf anderem Weg erhalten haben.

Um Platz auf der Festplatte zu sparen, können Sie bei der benutzerdefinierten Installation die Module auswählen die Sie benutzen möchten bzw. für die Sie eine Lizenz erworben haben. Funktionen für die keine Lizenz vorhanden ist, laufen im DEMO-Modus.

Die Deinstallation kann über den entsprechenden Eintrag im Startmenü oder in der Systemsteuerung unter der Rubrik Software erfolgen.

3.3. Hardwaremäßiger Softwareschutz

Das Produkt TeRANiS wird zusammen mit einem HASP-HL USB-Dongle ausgeliefert. Auf diesem Dongle sind die Funktionen entsprechend Ihrer erworbenen Lizenz freigeschaltet.

Besitzen Sie keinen Dongle oder ist eine Funktion nicht freigeschaltet, kann die Funktion nur eingeschränkt im Demomodus verwendet werden. Nicht freigeschaltete Funktionen dürfen nur zum Testen des Programmes und nicht für produktive Projekte verwendet werden.

Achtung ! Der Dongle muss die gesamte Zeit, während die Anwendung gestartet ist, eingesteckt bleiben, da er zu verschiedenen Zeitpunkten abgefragt wird.

3.3.1. Umfang der Lizenzen

Ihre erworbene Lizenz bezieht sich auf freigeschaltete Funktionen, die maximale TeRANiS-Version und die maximale Betriebssystem-Version auf der TeRANiS eingesetzt werden kann. Sie können mit dieser Lizenz also auch jede frühere TeRANiS-Version verwenden. Auch die Verwendung auf einer früheren Betriebssystem-Version ist möglich. Beachten Sie jedoch die Einsatzvoraussetzungen für TeRANiS.

3.3.2. Treiber

Die Treiber für den HASP-HL-Dongle werden bei der Installation von TeRANiS mit installiert, sofern Sie die Option nicht abgewählt haben. Die Treiber sind von Microsoft zertifiziert und sind Bestandteil von Windows-Update. Damit können die Treiber auch direkt aus dem Internet installiert werden.

3.3.3. Hilfsprogramme

Auf der Installations-CD finden Sie verschiedene Diagnose-Tools für den Donglebetrieb. Diese können auch von der Website des Dongle-Herstellers heruntergeladen werden: <http://www.alladin.de>

3.3.4. Updates

Um weitere Funktionen freizuschalten, eine höhere TeRANiS Version zu verwenden oder es auf einem neuen Betriebssystem einzusetzen müssen Sie ein Update erwerben. Dabei erhalten Sie normalerweise keinen neuen Dongle, sondern eine E-Mail, mit der Ihr Dongle umprogrammiert wird.

3.4. Einschränkungen der Demo-Version

Wenn keine Lizenz auf einem Dongle für eine bestimmte Funktion gefunden werden kann, schaltet das betreffende Modul in den Demomodus. Dieser Zustand wird beim Starten des Moduls in einer Meldung angezeigt. Um den Demomodus zu beenden muss das Modul neu gestartet werden. Ein nachträgliches Einstecken des Dongles hat keine Wirkung.

Im Demomodus haben die einzelnen Module unterschiedliche Einschränkungen. Alle Funktionen der Module sind zugänglich und können somit getestet werden.

Allgemeines

- Beim Starten bestimmter Funktionen erscheint eine Dialogbox, die auf den Demomodus hin-

weist. Diese muss von Hand quittiert werden.

- Beim Drucken erscheint ein Text der auf die Demoversion hinweist auf dem Ausdruck

TeCAD

- Die Variablenanzahl ist auf 30 beschränkt. Da Variablen teilweise automatisch angelegt werden ist damit auch der Planungsmodus begrenzt.
- In den übersetzten ST-Code ist ein Text eingefügt, der vor der Weiterverarbeitung manuell entfernt werden muss

TeList

- Die Anzahl von übersetzbaren Instanzen ist auf 2 beschränkt

TeVar

- Die Variablenanzahl ist auf 100 beschränkt. (Dies gilt nur für das Einfügen und Ändern von Variablen)
- Es wird nur ein frei deklariertes Datentyp übernommen.

TeCOM

- Die maximale Anzahl generierter Codezeilen ist auf 500 beschränkt

TeRANiS Studio

- Die Anzahl durchführbarer Transaktionen ist auf 10 beschränkt. Danach muss das Programm neu gestartet werden.

3.5. Bezugsquellen

TeRANiS wird von der ViDEC-GmbH vertrieben. Nähere Informationen erhalten Sie dort.

ViDEC Data Engineering GmbH Osterdeich 108 D-28205 Bremen Telefon: 0421 - 33 950 - 0 Telefax: 0421 - 33 795 - 61 E-Mail: Info@videc.de

<http://www.videc.de>

Kapitel 4. Referenz

Stichwortverzeichnis

A

Alarmvariablen, 3 , 24

B

B-Variablen, 3 , 16

Bezugsquellen, 36

I

Installation und Deinstallation, 34

P

P-Variablen, 20

Projektdatenbank, 6

R

R-Variablen, 3 , 12

Rohdatenvariablen, 3 , 10

S

Softwareschutz, 34

System Information, 34

T

Technische Voraussetzungen, 34

V

V-Variablen, 3 , 10