



Anwenderhandbuch

für den

ladbaren **iFIX-Block**

DAX

zur

digitalen Alarmierung mit formatierter Ausgabe von Zusatzwerten

Autor: Jürgen Stähler

Version: 1.01

Stand: 02.04.2014

Inhaltsverzeichnis

1.	Einführung.....	3
1.1	Zweck dieses Dokuments.....	3
1.2	Zielgruppe.....	3
1.3	Definitionen und Akronyme.....	3
1.3.1	Akronyme.....	3
1.4	Typografische Konventionen.....	3
2.	Installation.....	4
2.1	Slotnummer.....	4
2.2	Standard-Installationspfad von iFIX.....	4
3.	DAX Block Übersicht.....	5
3.1	Features.....	5
4.	Block-Konfiguration.....	5
4.1	Allgemeine Hinweise zu den Dialogen.....	5
4.2	Block-spezifische Felder.....	6
4.2.1	Description.....	6
4.2.2	Driver.....	6
4.2.3	Alarm Bit I/O Addr.....	6
4.2.4	1st Param I/O Addr.....	7
4.2.5	2nd Param I/O Addr.....	7
4.2.6	Use Text Table.....	7
4.2.7	Text Table File Path and Name.....	7
4.2.8	Alarm Type.....	8
4.2.9	Alarm Areas.....	8
4.2.10	Alarm Priority.....	9
4.2.11	Alarm Suspend Tag.....	9
4.2.12	Alarm Delay Time.....	9
4.2.13	Rearming Cycle Time.....	9
4.2.14	Alarm Extension Fields.....	9
4.3	Platzhalter für Zusatzwerte im Alarmtext.....	10
5.	Der DAX-Block zur Laufzeit.....	12
5.1	Einlesen der Textdateien.....	12
5.2	Abfrage der Zusatzwerte.....	13
5.3	Alarmtexte mit aufgelösten Platzhaltern.....	13
6.	Lizenzierung.....	14

Tabellen

Tabelle 1:	Beispiele für relative und absolute Dateipfade für die Textdatei.....	8
Tabelle 2:	Alarmtypen und deren Auslöseverhalten.....	8
Tabelle 3:	Formatstrings für Ganzzahlen.....	11
Tabelle 4:	Formatstrings für Fließkommawerte.....	11
Tabelle 5:	Formatstrings für Texte.....	12
Tabelle 6:	Fehlernummern am Feld F_FLIE_ERROR beim Einlesen der Textdateien.....	13
Tabelle 7:	Konfigurierte und aufgelöste Alarmtextfelder.....	13

Abbildungen

Abbildung 1:	Dialogmasken Basic und Alarms des DAX-Blocks.....	6
--------------	---	---

1. Einführung

1.1 Zweck dieses Dokuments

Dieses Dokument beschreibt die Installation, Konfiguration und das Laufzeitverhalten des ladbaren DAX-Blockes für die iFIX-Versionen 5.1 und 5.5

1.2 Zielgruppe

Diese Dokumentation richtet sich an iFIX-Applikationsentwickler, die diesen ladbaren Block in Ihre Anwendung integrieren möchten.

1.3 Definitionen und Akronyme

1.3.1 Akronyme

DAX - Digital Alarm eXtended
SAC - Scan Alarm Control

1.4 Typografische Konventionen

Kursivschrift wird für Dialognamen und Steuerelementnamen (Button, Eingabefelder usw.) verwendet.

Fette Schrift wird für iFIX-Feldnamen (z.B. **F_CV**) verwendet.

Fette Kursivschrift wird für Dateipfade, Dateinamen und Kapitelnamen verwendet.

2. Installation

Für den Block steht eine Installations-Batch-Datei zur Verfügung. Diese Batch-Datei muss aus einem Windows Befehlszeilenfenster (DOS-Box) heraus aufgerufen werden. Die Batchdatei kopiert die Dateien des ladbaren Blocks in die entsprechenden iFIX-Verzeichnisse und fügt in der Datei FIX.INI einen Eintrag für das Laden des Blockes hinzu.

Der Batch-Datei kann als Parameter der iFIX-Installationspfad übergeben werden. Wird die Batch-Datei ohne diesen Parameter aufgerufen, wird der interne Standard-Installationspfad **C:\Programme\Proficy\Proficy iFIX** verwendet.

Ist iFIX nicht in diesem Standard-Installationspfad installiert, muss dieser als Parameter übergeben werden!

Beispiel für den Aufruf der Batch-Datei des DAX-Blockes:

```
DAX_INST.BAT<Enter>  
DAX_INST.BAT "D:\Proficy\Proficy iFIX"<Enter>
```

Der Pfad ist stets in Anführungszeichen zu setzen!

Diese Batch-Datei kann mit einem einfachen Editor (z.B. Notepad.exe) ggf. individuell angepasst werden. Die Anpassungen beschränken sich auf die erste und zweite Zeile der Batch-Datei.

```
Zeile 3: SET SLOT=84  
Zeile 4: IF '%1'=='' SET TARGET="C:\Programme\Proficy\Proficy iFIX"
```

Die folgenden Anpassungen sind ggf. erforderlich:

2.1 Slotnummer

Die Slotnummer ist eine eindeutige Nummer, über die iFIX den ladbaren Block intern (im WSACTASK und in Datenbasismanager) referenziert. Jeder Blocktyp muss eine eigene Slotnummer erhalten, die nicht von einem anderen Blocktyp bereits verwendet wird.

Für ladbare Blöcke ist der Nummernbereich 51 bis 150 reserviert. Dem hier beschriebenen ladbarem DAX-Block wurde die Standard-Slotnummer 93 zugeteilt.

Vor der Installation des ladbaren Blocks ist unbedingt zu prüfen, ob dessen Standard-Slotnummer nicht bereits durch einen anderen ladbaren Block belegt ist.

Die Information über bereits installierte ladbare Blöcke und deren Slotnummern findet man in der Datei FIX.INI, die im Unterverzeichnis \LOCAL des iFIX-Installationsverzeichnisses angelegt ist. Die FIX.INI Datei kann mit einem Texteditor (z.B. Notepad.exe) bearbeitet werden. In der FIX.INI Datei gibt es eine sog. Section "[SCADA]" innerhalb der alle ladbaren Blöcke mit je zwei Zeilen eingetragen sind. Die folgenden zwei Zeilen zeigen beispielhaft den Eintrag für den ladbaren "ETR" Block:

```
DLL=BTK_ETR.DLL  
RUN=BTKINIT.EXE BTK_ETR 40
```

Die Zahl 40 am Ende der zweiten Zeile ist die Slotnummer unter der dieser ETR-Block im iFIX referenziert wird.

Prüfen Sie stets alle Blockeinträge in der Section "[SCADA]" auf duplizierte Slotnummern! Sollte die Standard-Slotnummer eines zu installierenden Blockes schon durch einen anderen Block belegt sein, muss der neue Block eine unbenutzte Blocknummer erhalten. Diese muss vor Aufruf der Installation-Batch-Datei in der ersten Zeile (SET SLOT=xx) der Batch-Datei eingetragen bzw. geändert werden.

2.2 Standard-Installationspfad von iFIX

In der zweiten Zeile des Batch-Datei wird das Macro TARGET mit dem Standard-iFIX-Installationspfad

C:\Programme\Proficy\Proficy iFIX gesetzt falls kein Parameter (%1 == ") übergeben wurde. Der Standard-iFIX-Installationspfad kann hier individuell angepasst werden.

3. DAX Block Übersicht

DAX steht für **D**igital **A**larm **eX**tended, also für einen funktional erweiterten Alarmblock für digitale Signale.

3.1 Features

Als Vorlage für den DAX-Block diente der Standard-DA-Block. Der DAX-Block bietet bis auf wenige Einschränkungen die gleichen Funktionen wie der Standard-DA Block. Außerdem wurden folgende Zusatzfunktionen realisiert:

1. In den Alarmtext können bis zu zwei Zusatzwerte dynamisch eingesetzt werden. Diese Zusatzwerte werden über zwei analoge E/A-Adressen aus der Prozeßhardware eingelesen.
2. Der zweite Zusatzwert kann als Index für einen Texttable verwendet werden. Anstatt des numerischen Wertes des Zusatzwertes wird dann der dem Index zugeordnete Text in den Alarmtext eingesetzt. Die Texttable ist als einfache CSV-Datei ausgelegt. Für jede Blockinstanz kann eine individuelle Textdatei konfiguriert werden.
Das Einlesen der Textdateien erfolgt in einen separatem Thread. Das Laufzeitverhalten des iFIX-SAC-Prozesses wird somit nicht negativ beeinflusst.
3. Die dynamische Formatierung der Alarmtexte mit den Zusatzwerten (bzw. Zusatztext) kann auch für die Alarmerweiterungstexte verwendet werden. Es können somit zwei Alarmzusatztexte mit maximal je 80 Zeichen dynamisch aufgebaut werden.
4. Für die Formatierung der numerischen Zusatzwerte stehen vielfältige Formatieroptionen wie z.B. feste Anzahl Vor- und Nachkommastellen, mit Vornulln usw. zur Verfügung.

4. Block-Konfiguration

Im folgenden werden sämtliche Block-spezifischen Eingabe- und Auswahlfelder beschrieben. Felder, die beim Standard-DA-Block ebenfalls vorhanden sind und die gleiche funktionale Bedeutung besitzen, werden hier nicht explizit beschrieben.

4.1 Allgemeine Hinweise zu den Dialogen

Aufgrund von Einschränkungen des verwendeten Block-Toolkits können für bestimmte Dialogfelder (z.B. Alarm Priorität) nicht die gleichen Steuerelemente verwendet werden, wie beim Standard-DA-Block. Deshalb werden beim DAX-Block anstatt Auswahllisten (Combo-Boxen) Radio-Buttons für eine 1-aus-n Auswahl verwendet.

Eine weitere Besonderheit der Dialoge eines ladbaren Blockes ist die Navigation zwischen den zwei Dialog-Seiten. Der *Alarms* Dialog besitzt keinen *Save* Button. Sie müssen also immer erst auf den *Basic* Dialog zurück wechseln, um die Einstellungen (beider Dialogseiten) über den *Save* Button sichern zu können.

4.2 Block-spezifische Felder

In diesem Kapitel werden sämtliche Block-spezifischen Felder des DAX-Blockes beschrieben. Die folgenden zwei Bildschirm-Abzüge stellen die beiden Dialogmasken (*Basic* und *Alarms*) des DAX-Blockes dar. Die in die Dialoge eingefügten nummerierten roten Kreise referenzieren auf die folgenden Unterkapitelnummern des aktuellen Kapitels **4.2 Block-spezifische Felder**. Die Beschreibung des Feldes **1st Param I/O Addr.** mit dem roten Kreis Nr. 4 finden Sie also im Unterkapitel 4.2.4.

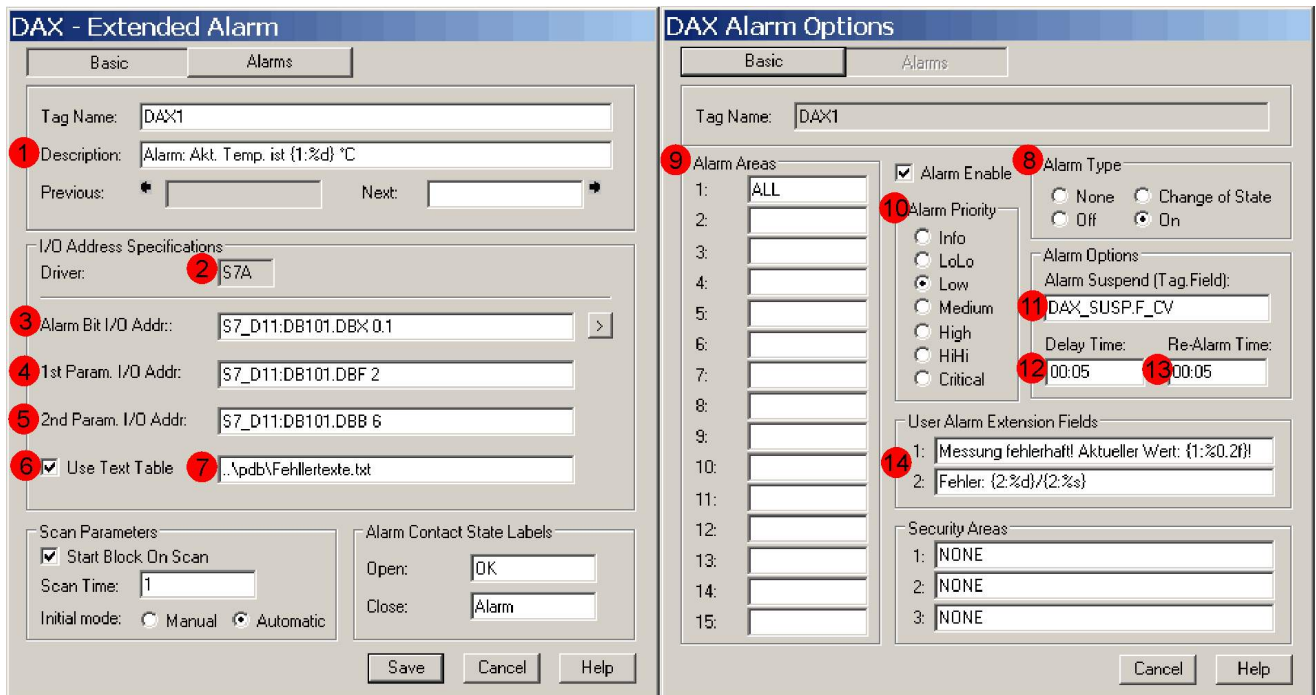


Abbildung 1: Dialogmasken **Basic** und **Alarms** des DAX-Blockes.

4.2.1 Description

Feldname: **A_DESC**

Dieser Text wird als Alarmtext an das iFIX-Alarmsystem übergeben. Die maximale Länge des Textes ist 40 Zeichen. Der DAX-Block erlaubt es in diesen Text Zusatzwerte dynamisch einzufügen. Die Position und das Format dieser Zusatzwerte wird in dem Alarmtext über Platzhalter definiert. Der Syntax der Platzhalter und die Formatierung der Zusatzwerte wird im Kapitel **4.3 Platzhalter für Zusatzwerte im Alarmtext** ausführlich beschrieben.

4.2.2 Driver

Feldname: **A_IODV**

Diese Version des DAX-Blockes wird nur vom S7A-Treiber unterstützt. Deshalb ist das Feld für die Eingabe des Treiber-Akronyms nicht editierbar sondern der feste Text **S7A** eingetragen. Zukünftige Versionen dieses Blockes werden auch andere Treiber unterstützen.

4.2.3 Alarm Bit I/O Addr.

Feldname: **A_IOAD**

Die I/O Adresse des digitalen Signals, welches auf eine Alarmbedingung überwacht werden soll bzw. den

Alarm auslösen soll. Schauen Sie sich die Online-Hilfe des S7A-Treibers an, um detaillierte Information zum Syntax der I/O-Adresse zu erhalten.

4.2.4 1st Param I/O Addr.

Feldname: **A_IOAD2**

Die I/O Adresse des ersten Zusatzwertes, der im Alarmtext dargestellt werden soll. Es sind nur analoge Adressen zulässig. Beim S7A-Treiber können alle analogen Datentypen (DBB, DBW, DBD oder DBD) verwendet werden. Liefert die I/O-Adresse einen Fließkommawert zurück, können im Formatstring des Platzhalters im Alarmtext die Umwandlungszeichen **"d"** und **"f"** verwendet werden. Bei Umwandlungszeichen **"d"** gehen dann allerdings die Nachkommastellen verloren. Liefert die I/O-Adresse einen Ganzzahlwert zurück kann auch das Umwandlungszeichen **"f"** im Formatstring verwendet werden, allerdings werden dann die Dezimalstellen stets mit Nullen angezeigt.

4.2.5 2nd Param I/O Addr.

Feldname: **A_IOAD3**

Die I/O Adresse des zweiten Zusatzwertes, der im Alarmtext dargestellt werden soll. Es sind nur analoge Adressen zulässig. Beim S7A-Treiber können alle analogen Datentypen (DBB, DBW, DBD oder DBD) verwendet werden. Liefert die I/O-Adresse einen Fließkommawert zurück, können im Formatstring des Platzhalters im Alarmtext die Umwandlungszeichen **"d"** und **"f"** verwendet werden. Bei Umwandlungszeichen **"d"** gehen dann allerdings die Nachkommastellen verloren. Liefert die I/O-Adresse einen Ganzzahlwert zurück kann auch das Umwandlungszeichen **"f"** im Formatstring verwendet werden, allerdings werden dann die Dezimalstellen stets mit Nullen angezeigt.

4.2.6 Use Text Table

Feldname: **B_TXTTABLE**

Ist diese Checkbox markiert, wird der Wert des zweiten Zusatzparameters als Textindex für die Textdatei verwendet, um einen Text zu adressieren. Dieser Text kann mit dem Umwandlungszeichen **"s"** im Alarmtext ausgegeben werden. Es ist auch eine Kombination aus numerischem Wert und Text aus Datei erlaubt. Der folgende Alarmtext zeigt eine solche Kombination:

"Messstellenfehler {2:%d} : {2:%s} !"

Bei einer Texttabelle mit den folgenden Zeilen

1;Drahtbruch
2;Kurzschluß
3;Nullabgleichfehler
4;Bereichsunterschreitung
5;Bereichsüberschreitung

würde bei einem eingelesenen Wert 3 des Zusatzparameters 2 die Alarmtextzeile wie folgt formatiert:

"Messstellenfehler 3 : Nullabgleichfehler !"

4.2.7 Text Table File Path and Name

Feldname: **A_TXTFN**

Ist die Checkbox **Use Text Table** markiert, muss hier der Pfad und Dateiname der Textdatei angegeben werden, aus der die Texte gelesen werden sollen. Das Bezugsverzeichnis ist das iFIX-Installationsverzeichnis.

Alle Pfadangaben sind relativ zu diesem Bezugsverzeichnis. Wird also lediglich ein Dateiname angegeben, versucht der Block diesen im iFIX-Installationsverzeichnis zu finden und zu öffnen.

Unter der Bedingung, dass iFIX im Verzeichnis **C:\Programme\Proficy\Proficy iFIX** installiert ist, ergeben sich für die in diesem Feld eingegebenen relativen Pfad- und Dateinamen folgende absolute Dateipfade:

Eingabe in Feld Text Table File Path and Name	Absoluter Dateipfad
Fehlertexte.txt	C:\Programme\Proficy\Proficy iFIX\Fehlertexte.txt
PDB\Fehlertexte.txt	C:\Programme\Proficy\Proficy iFIX\PDB\Fehlertexte.txt
..\Fehlertexte.txt	C:\Programme\Proficy\Fehlertexte.txt
..\..\..\InCoSol\Fehlertexte.txt	C:\InCoSol\Fehlertexte.txt

Tabelle 1: Beispiele für relative und absolute Dateipfade für die Textdatei.

Die Textdatei ist eine einfache CSV-Datei mit lediglich zwei Spalten. In der linken Spalte steht der Textindex, also ein numerischer Wert. Der Wertebereich erstreckt sich von 0 bis 65535. Die von der linken Spalte durch ein Semikolon getrennte rechte Spalte enthält den zugehörigen Text. Die Länge der Texte ist begrenzt durch die maximale Länge der Alarmtexte (40 beim Standard-Alarmtext des Feldes **A_DESC** und 80 Zeichen bei den erweiterten Alarmtextfeldern **A_ALMEXT1** und **A_ALMEXT2**).

Beispiel für den Inhalt einer Textdatei:

```
1;Drahtbruch
2;Kurzschluß
3;Nullabgleichfehler
4;Bereichsunterschreitung
5;Bereichsüberschreitung
```

Es können Vornullen für die Textindex-Werte der linken Spalte verwendet werden. Die Reihenfolge der Einträge ist beliebig, sie müssen also nicht numerisch aufsteigend sortiert sein.

4.2.8 Alarm Type

Feldname: **A_ALMCK**

Die möglichen Alarmtypen entsprechen den Alarmtypen des Standard-DA-Blockes. Die folgende Tabelle listet die möglichen Alarmtypen und die zugehörige Alarmbedingung auf:

Alarm Type	Alarm wird ausgelöst wenn...
None	Es wird kein Alarm ausgelöst.
Change of State	...das digitale Alarmsignal sich von 0 nach 1 oder von 1 nach 0 ändert.
Open	...sich das digitale Alarmsignal von 1 nach 0 ändert.
Close	...sich das digitale Alarmsignal von 0 nach 1 ändert.

Tabelle 2: Alarmtypen und deren Auslöseverhalten.

4.2.9 Alarm Areas

Feldnamen: **A_AREA1...A_AREA15**

Es können bis zu 15 Alarmbereichsnamen für den Alarm angegeben werden. Der einzelne Name darf maximal

28 Zeichen lang sein.

4.2.10 Alarm Priority

Feldname: **A_PRI**

Die Alarmpriorität des Blockes.

4.2.11 Alarm Suspend Tag

Feldname: **A_ASUSP**

Der Tag-Name und dessen Feld, über den die Alarmierung des DAX-Blockes ein- bzw. ausgeschaltet werden kann. Es muss stets der Datenblockname und - durch einen Punkt getrennt- das Datenblock-Feld angegeben werden.

Beispiel: **DAX_SUSP.F_CV**

HINWEIS: Dieses Feld hat Vorrang vor dem **Alarm Enable** Feld (**A_ENAB** und **F_ENAB**). Dies bedeutet, dass über die Felder **A_ENAB** und **F_ENAB** die Alarmbearbeitung nicht beeinflusst werden kann, solange die Alarmierung über den **Alarm Suspend Tag** deaktiviert ist.

4.2.12 Alarm Delay Time

Feldname: **A_DELAY**

Gibt eine Zeitspanne an, die das Alarmsignal kontinuierlich anstehen muss, bevor der Alarm an die iFIX-Alarmverarbeitung gemeldet wird. Das Anzeige- und Eingabeformat ist <MM>:<SS> wobei <MM> ein Minutenwert zwischen 00 und 60 und <SS> ein Sekundenwert zwischen 00 und 59 sein kann. Alternativ kann auch ein Sekundenwert zwischen 0 und 3600 eingegeben werden, der dann jedoch als Stunden- und Minutenwert umgerechnet und dargestellt wird.

4.2.13 Realarmsing Cycle Time

Feldname: **A_REALM**

Gibt eine Zykluszeit an, mit welcher ein anstehender Alarm erneut an das iFIX-Alarmsystem gemeldet wird. Das Anzeige- und Eingabeformat ist <MM>:<SS> wobei <MM> ein Minutenwert zwischen 00 und 60 und <SS> ein Sekundenwert zwischen 00 und 59 sein kann. Alternativ kann auch ein Sekundenwert zwischen 0 und 3600 eingegeben werden, der dann jedoch als Stunden- und Minutenwert umgerechnet und dargestellt wird.

4.2.14 Alarm Extension Fields

Feldnamen: **A_ALMEXT1**, **A_ALMEXT2**

Über die beiden Alarm Extension Felder können zusätzliche Alarmtexte definiert werden. Auch in diese Alarmtexte können über Platzhalter Zusatzwerte eingeblendet werden.

Hinweis: Bei der Verwendung von Platzhaltern ist darauf zu achten, dass die Länge des Alarmzusatztextes nach Auflösung der Platzhalter die maximale Länge von 80 Zeichen nicht überschreitet. Der Syntax der Platzhalter und die Formatierung der Zusatzwerte wird im Kapitel **4.3 Platzhalter für Zusatzwerte im Alarmtext** ausführlich beschrieben.

4.3 Platzhalter für Zusatzwerte im Alarmtext

Ein Platzhalter ist durch geschweifte Klammern gekennzeichnet, wobei die öffnende (nach links) Klammer("{") den Anfang und die schließende (nach rechts) Klammer("}") das Ende eines Platzhalters markiert. Als erstes Zeichen innerhalb der geschweiften Klammern muss die Nummer (1 oder 2) des Zusatzwertes stehen, der zur Laufzeit anstatt des Platzhalters in den Alarmtext eingesetzt werden soll. Nach dieser Nummer folgt ein Doppelpunkt (":") und anschließend ein sogenannte Formatstring, der das Format des Zusatzwertes steuert. Der Syntax dieses Formatstrings ist angelehnt an den Syntax der formatierten Ausgabe über die Funktion *printf()* im ANSI-Standard-C. Der Formatstring steuert

- die Datentyp-Umwandlung des Zusatzwertes, also ob der Wert als Ganzzahl, Fließkommazahl oder als Text dargestellt werden soll.
- Die minimale Anzahl der darzustellenden Stellen (Feldbreite) einer Ganz- oder Fließkommazahl.
- Ob mit Nullen oder mit Leerzeichen bis auf die angegebene Feldbreite aufgefüllt werden soll.
- Ob bei positiven Werten explizit ein +-Zeichen dargestellt werden soll.
- Ob ein Text innerhalb der Feldbreite links- oder rechtsbündig formatiert werden soll.

Der Formatstring beginnt mit dem Zeichen % und endet mit einem Umwandlungszeichen. Zwischen dem % Zeichen und dem Umwandlungszeichen kann, der Reihe nach, folgendes angegeben werden:

- Ein Minus-Zeichen, damit der umgewandelte Zusatzwert nach links ausgerichtet wird.
- Eine Zahl, die eine minimale Feldbreite definiert. Der umgewandelte Zusatzwert wird in einen String formatiert, der mindestens so breit ist und bei Bedarf auch breiter sein kann als die angegebene minimale Feldbreite. Hat der umgewandelte Zusatzwert weniger Zeichen als die minimale Feldbreite verlangt, wird dieser links (oder rechts, wenn Ausrichtung nach links angegeben ist) auf die minimale Feldbreite aufgefüllt.
- Ein Punkt, der die minimale Feldbreite von der Angabe der Nachkommastellen (bei Umwandlungszeichen *f*) oder der Angabe der maximalen Anzahl Zeichen (bei Umwandlungszeichen *s*) trennt.
- Eine Zahl, welche die Anzahl Ziffern die nach dem Dezimalpunkt bei einer Fließkommazahl (Umwandlungszeichen *f*) ausgegeben werden bestimmt. Bei Texten (Umwandlungszeichen *s*) bestimmt diese Zahl die maximale Anzahl Zeichen, die ausgegeben werden.

Umwandlungszeichen

Der DAX-Block erlaubt die folgenden drei Datentyp-Umwandlungen mit den Umwandlungszeichen:

- d** für die Umwandlung bzw. Darstellung des Zusatzwertes als Ganzzahl.
- f** für die Umwandlung bzw. Darstellung des Zusatzwertes als Fließkommawert.
- s** für die Darstellung eines Textes aus der Textdatei.

Alle weiteren im ANSI Standard-C spezifizierten Umwandlungszeichen sind nicht erlaubt!

Die folgenden Tabellen enthalten Beispiele von erlaubten Formatstrings für die verschiedenen Datentypen. Zur besseren Darstellung der Leerzeichen innerhalb des formatierten Wertes, werden diese in der Tabelle durch Unterstriche dargestellt. Der komplette formatierte Wert (die Zeichenkette, die den Platzhalter ersetzt) wird in hellblauer Schriftfarbe dargestellt.

Formatstrings für Ganzzahlwerte (Beispielwert = 123)		
Alarmtext mit Platzhalter und Formatstring	Alarmtext mit aufgelöstem Platzhalter	Bem.
Alarm: Akt. Temp. ist {1:%d} C!	Alarm: Akt. Temp. ist 123 C!	1
Alarm: Akt. Temp. ist {1:%+d} C!	Alarm: Akt. Temp. ist +123 C!	2
Alarm: Akt. Temp. ist {1:%4d} C!	Alarm: Akt. Temp. ist _123 C!	3
Alarm: Akt. Temp. ist {1:%05d} C!	Alarm: Akt. Temp. ist 00123 C!	4
Alarm: Akt. Temp. ist {1:%+05d} C!	Alarm: Akt. Temp. ist +0123 C!	5
Alarm: Akt. Temp. ist {1:%-5d} C!	Alarm: Akt. Temp. ist 123__ C!	6

Tabelle 3: Formatstrings für Ganzzahlen.

- 1) Standardformatierung mit %d. Der Wert wird mit der dem aktuellen Wert erforderlichen Anzahl Stellen dargestellt.
- 2) Explizite Darstellung des Vorzeichens auch bei positiven Werten durch Einfügen eines + Zeichens zwischen dem % Zeichen und dem Umwandlungszeichen **d**.
- 3) Die minimale Feldbreite wird auf 4 Zeichen festgelegt. Hat der umgewandelte Wert weniger als 4 Ziffern, wird links mit Leerzeichen aufgefüllt.
- 4) Die minimale Feldbreite wird auf 5 Zeichen festgelegt. Hat der umgewandelte Wert weniger als 5 Ziffern, wird links mit (Vor-)Nullen aufgefüllt.
- 5) Wie 4) jedoch mit expliziter Ausgabe des Vorzeichens. Das Vorzeichen ersetzt die erste (linke) Vornull.
- 6) Hat der umgewandelte Wert weniger als 5 Ziffern, wird er linksbündig ausgerichtet und rechts bis auf die minimale Feldbreite von 5 Zeichen mit Leerzeichen aufgefüllt.

Formatstrings für Fließkommawerte (Beispielwert = 123,4)		
Alarmtext mit Platzhalter und Formatstring	Alarmtext mit aufgelöstem Platzhalter	Be.
Alarm: Akt. Temp. ist {1:%f} C!	Alarm: Akt. Temp. ist 123,40000 C!	1
Alarm: Akt. Temp. ist {1:%.2f} C!	Alarm: Akt. Temp. ist 123,40 C!	2
Alarm: Akt. Temp. ist {1:%+.2f} C!	Alarm: Akt. Temp. ist +123,40 C!	3
Alarm: Akt. Temp. ist {1:%7.2f} C!	Alarm: Akt. Temp. ist _123,40 C!	4
Alarm: Akt. Temp. ist {1:%07.2f} C!	Alarm: Akt. Temp. ist 0123,40 C!	5

Tabelle 4: Formatstrings für Fließkommawerte.

- 1) Standardformatierung eines Fließkommawertes mit %f. Hat den Nachteil, dass die Anzahl der Nachkommastellen nicht begrenzt ist und ggf. bis zu 5 Nachkommastellen angezeigt werden. Wir empfehlen für Fließkommawerte stets die Anzahl Nachkommastellen durch explizite Angabe im Formatstring zu begrenzen. Siehe auch 2).
- 2) Begrenzung der Nachkommastellen auf 2 (".2") durch Einfügen von ".2" zwischen dem % -Zeichen und

dem Umwandlungszeichen *f*.

- 3) Explizite Darstellung des Vorzeichens auch bei positiven Werten durch Einfügen eines **+** Zeichens nach dem **%** Zeichen .
- 4) Die minimale Feldbreite wird auf 7 Zeichen ("**7**") festgelegt. Hat der umgewandelte Wert weniger als 7 Ziffern (inkl. Dezimalzeichen und Nachkommastellen), wird links mit Leerzeichen aufgefüllt.
- 5) Die minimale Feldbreite wird auf 7 Zeichen ("**07**") festgelegt. Hat der umgewandelte Wert weniger als 7 Ziffern (inkl. Dezimalzeichen und Nachkommastellen), wird links mit Nullen aufgefüllt.

Formatstrings für Texte aus Textdatei (Beispieltext = "Kurzschluss")		
Alarmtext mit Platzhalter und Formatstring	Alarmtext mit aufgelöstem Platzhalter	Be.
Messstellenstörung: {2:%s}	Messstellenstörung: Kurzschluss	1
Messstellenstörung: {2:%15s}	Messstellenstörung: _____Kurzschluss	2
Messstellenstörung: {2:%-15s}	Messstellenstörung: Kurzschluss_____	3
Messstellenstörung: {2:%.8s}	Messstellenstörung: Kurzschl	4

Tabelle 5: Formatstrings für Texte.

- 1) Standardformatierung eines Textes mit %s. Es wird der Text wie er ist ausgegeben. Nachteil: Es gibt keine Längenbegrenzung, was dazu führen kann, dass der vollständige Alarmtext die maximale Länge (40 bei Description und 80 bei Alarm Extension) überschreiten kann.
- 2) Die minimale Feldbreite wird auf 15 Zeichen ("**15**") festgelegt. Hat der Text weniger als 15 Zeichen, wird links mit Leerzeichen aufgefüllt. Nachteil wie unter 1) beschrieben.
- 3) Die minimale Feldbreite wird auf 15 Zeichen ("**-15**") festgelegt. Hat der Text weniger als 15 Zeichen, wird rechts mit Leerzeichen aufgefüllt. Nachteil wie unter 1) beschrieben.
- 4) Die minimale Feldbreite wird auf 8 Zeichen ("**.8**") festgelegt. Ist der Text länger als 8 Zeichen, wird dieser nach dem 8. Zeichen (rechts) abgeschnitten. Hat den Vorteil, dass die maximal erlaubte Alarmtextlänge (statischer Text plus dynamische Zusatzwerte) eindeutig definiert ist und somit eingehalten wird.

5. Der DAX-Block zur Laufzeit

Das Kapitel beschreibt das Verhalten und die Funktion des DAX-Blockes zur Laufzeit, also wenn der Block vom SAC-Task zyklisch bearbeitet wird. Außerdem werden alle Block-spezifischen Felder beschrieben, die der DAX-Block zur Laufzeit mit Daten versorgt.

5.1 Einlesen der Textdateien

Das Einlesen der Textdateien erfolgt in einem eigenen Thread, der unabhängig vom Zyklus des SAC-Task arbeitet. Durch die Auslagerung der Datei-Lesezugriffe in einen separaten Thread wird das Laufzeitverhalten des SAC-Task nicht negativ beeinflusst. Es wird der Forderung von GE Rechnung getragen, in der zyklischen Bearbeitungsroutine eines ladbaren Blocks keine Ein-/Ausgabeoperationen durchzuführen, um die Ausführungszeit der zyklischen Bearbeitungsroutine minimal kurz zu halten. Das Laden der Textdateien erfolgt einmalig beim Starten des SAC-Tasks. Der Hintergrund-Thread lädt die Textdateien aller DAX-Blockinstanzen in eine interne Text-Liste. Wenn der DAX-Block in der zyklischen Bearbeitungsroutine einen Alarmtext mit einem eingebetteten Text aus einer Textdatei zusammenbaut, wird lediglich auf diese interne Liste (im RAM-Speicher) zugegriffen.

Eventuell auftretende Fehler beim Lesen der Textdateien werden an dem numerischen Feld **F_FILE_ERROR**

angezeigt. Konnte die Textdatei ohne Fehler eingelesen werden, wird an diesem Feld der Wert 0 ausgegeben. Bei einem Fehler werden die in der folgenden Tabelle aufgelisteten Fehlernummer ausgegeben:

Fehlernummer an F_FILE_ERROR	Fehlerursache
0	Kein Fehler. Textdatei konnte eingelesen werden.
1	Die Datei konnte nicht gefunden werden. Evtl. ist der Dateipfad und/oder der Dateiname ungültig.
2	Ein Textindex(Zahl links vom Semikolon) hat eine ungültige Anzahl Ziffern. Der Textindex darf von 0 bis 65535 laufen, also max. 5 Ziffern lang sein.
3	Ein Textindex(Zahl links vom Semikolon) enthält ungültige Zeichen und ist deshalb nicht als numerischer Wert zu dekodieren.
4	Beim Einlesen der Textdatei ist eine Ausnahme (Exception) aufgetreten.

Tabelle 6: Fehlernummern am Feld F_FLIE_ERROR beim Einlesen der Textdateien.

5.2 Abfrage der Zusatzwerte

Das Alarmbit (Alarm Bit I/O Addr.) wird bei jedem Aufruf der Block-Bearbeitungsroutine vom Treiber eingelesen und an den Feldern F_CV und A_CV ausgegeben.

Die in einer Block-Instanz angegebenen E/A-Adressen für die Zusatzwerte (**1st Param. I/O Addr.** und **2nd Param. I/O Addr.**) werden nur dann durch den Block vom Treiber eingelesen, wenn die Alarmbedingung für das Alarmbit erfüllt ist. Der Block gibt die zuletzt eingelesenen Zusatzwerte an den Feldern **F_CV_P1** und **F_CV_P2** aus. Solange also die Alarmbedingung ansteht, werden an diesen beiden Feldern bei jedem Aufruf der Bearbeitungsroutine die aktuellen Zusatzwerte aktualisiert.

5.3 Alarmtexte mit aufgelösten Platzhaltern

Neben den Feldern für die konfigurierten Alarmtexte (**A_DESC**, **A_ALMEXT1** und **A_ALMEXT2**), welche ja die Platzhalter mit den Formatierinformationen für die Zusatzwerte enthalten, bietet der DAX-Block auch Felder, an denen die ausgelösten Alarmtexte abgegriffen werden können. Diese Felder sind durch die Namenserverweiterung **_RES** (für Resolved / aufgelöst) gekennzeichnet.

Die folgende Tabelle listet sämtliche Alarmtextfelder (die konfigurierten als auch die aufgelösten) mit Beispieltextrn auf:

Feldname	Feldtext
A_DESC	Fehler {2:%d}! Messwert = {1:%.2f}
A_DESC_RES	Fehler 3! Messwert = 123.40
A_ALMEXT1	Aktueller Messwert: {1:%.2f}
A_ALMEXT1_RES	Aktueller Messwert: 123.40
A_ALMEXT2	Messstellenfehler {2:%d} : {2:%s} !
A_ALMEXT1_RES	Messstellenfehler 3 : Nullabgleichfehler

Tabelle 7: Konfigurierte und aufgelöste Alarmtextfelder.

6. Lizenzierung

Die Lizenzierung des DAX-Blockes erfolgt über einen Hardware-Key (Dongle). Ein vorhandener S7A-Lizenzkey kann mit der DAX-Lizenz erweitert werden.

Die Lizenzkosten sind nach Anzahl der möglichen Blockinstanzen gestaffelt. Ein Upgrade von einer kleinen Lizenz (von z.B. 50 Blockinstanzen) auf eine größere (z.B. unbegrenzte Anzahl Blockinstanzen) ist jederzeit und mit geringem Aufwand möglich. In diesem Fall muss der Lizenzkey nicht ausgetauscht werden, sondern er wird lediglich mit der neuen Lizenzinformation umprogrammiert, was über ein Lizenz-Updateprogramm vom Anwender selbst vorgenommen werden kann.

Zu Evaluierungszwecken kann der DAX-Block auch ohne Lizenzkey betrieben werden. Zur Laufzeit von iFIX werde dann allerdings nur zwei Blockinstanzen aktiv im SAC-Zyklus bearbeitet. Es können beliebig viele DAX-Blöcke angelegt werden, die zur Laufzeit jedoch "Off scan" bleiben, also nicht zyklisch bearbeitet werden.