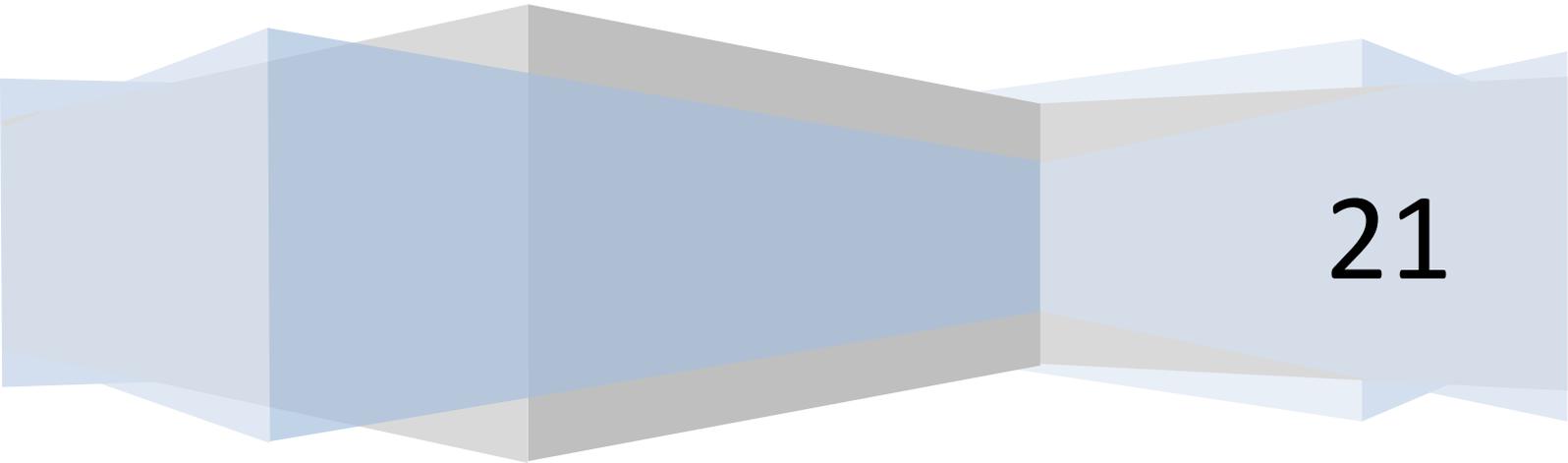


Version: R0 Draft
TOM
Datum: 13.08.2021 11:41:00
TBE

Funktionale Dokumentation aus DeltaV TBE

Technisches Büro Eichstetter
www.t-eichstetter.de
0170 383 4705

Funktionale Dokumentation aus DeltaV TBE



21

Funktionale Dokumentation aus DeltaV TBE

Index

1. Revisionsverzeichnis.....	3
2. Referenzierte Dokumente	3
3. Kurzbeschreibung	3
4. Short Glance DeltaV	4
4.1. Basis-Blöcke - Grundausstattung.....	4
5. MS VISIO als Engineering-Tool	5
6. MS ACCESS und DeltaV Parser	5
7. DeltaV Analyse und Sonderlogik	6
8. VISIO Templates	7
9. VISIO und EXCEL im Zusammenspiel	8
10. Höhere Automatisierungsfunktionen.....	9
11. Cause & Effect Matrix.....	10
12. Prozess-Grafikbilder	10
13. Fazit	10

Funktionale Dokumentation aus DeltaV TBE

1. Revisionsverzeichnis

Datum	Name	Beschreibung	Revisionskennung
23.05.2021	Eichstetter	Erstfassung der Dokumentation	RO
-	-	-	-

2. Referenzierte Dokumente

Nr	Dokumentenname	Typ	Beschreibung	Revisionskennung

3. Kurzbeschreibung

Die generelle Idee der funktionalen Rückdokumentation besteht darin, die spezifische DeltaV Programmierung von der eigentlich verfahrenstechnischen Funktion zu trennen.

Dazu gehört natürlich die gründliche Analyse und Aufbereitung der Gesamt-Applikation.

Das Problem einer Rückdokumentation, aus welchem Leitsystem auch immer, ist, dass das Dokument nur jemand versteht, der auch Kenntnis vom Ursprungssystem hat.

Soll nun eine funktionale Dokumentation entstehen, so müssen systemspezifische Programmteile von den rein verfahrenstechnisch notwendigen Teilen getrennt werden.

Hierfür muss zunächst typisiert werden, daraus entstehen Vorlagen, sog. Templates. Diese Templates werden pro Control Modul parametrisiert und referenziert.

Auf den Templates werden nur die verfahrenstechnisch notwendigen Funktionsblöcke dargestellt.

Auch hier werden Grundkenntnisse über die Funktionalität der benutzten Blöcke benötigt, aber die Anzahl und Funktionalität ist überschaubar.

4. Short Glance DeltaV

Die DeltaV Applikation besteht aus Modulen.

Man kann Module durchaus Funktionsplänen gleichsetzen.

Ein Modul ist die kleinste in einen Controller ladbar Funktionalität.

Ein Modul besteht aus Funktionsblöcken und Parametern, man spricht in DeltaV Sprache von Attributen.

Alarmer werden durch Attribute getriggert, wenn sich ihr Zustand ändert.

Es gibt verschiedene Arten von Modultypen, je nach Funktion, die dieses Modul innerhalb der Applikation einnehmen soll.

Ein Modul ist i.d.R. typisiert. D.h., der Aufbau eines bestimmten Typicals ist immer identisch, zumindest sollte es so sein. Es ist immer so, wenn die Module aus Klassen abgeleitet (instanziiert) werden.

Beispiel: Eine Messung, z. Bsp. Temperatur.

Das Typical könnte im einfachsten Fall aus einem Block, dem sog. AI-Block einem Ausgangsattribute und 5 Alarmen bestehen. Die 5 Alarme sind Messwert gestört jeweils 2 Hoch und 2 Tief Grenzwerte.

Der AI-Block wird auf den Messbereich des Messwert-Sensors eingestellt, fertig.

Kurzum, es gibt in einer DeltaV Applikation Basis-Module, Sonderlogik-Module und Module mit höheren Automatisierungsaufgaben wie z. Bsp. Schrittketten.

4.1. Basis-Blöcke - Grundausstattung.

Index	Name	Beschreibung
1	AI	Analogmessung
2	DI	Digitale Messung (MAX-MIN Sonde o. Statusrückmeldungen)
3	DC (Valve)	Device Control für ein Ventil o. Klappe AUF/ZU
4	DC (Motor)	Device Control für einen Antrieb Motor o. Pumpe
5	PID	Analoge-Regelung
6	AO	Analoger Ausgang für z. Bsp. eine Control Valve (Regelventil)
7	DO	Digitaler Ausgang zur Ansteuerung eines Ventils o. Klappe
8	TOT	Mengenerfassung analoger Größen. Messwert über Zeit. (Integrator)

5. MS VISIO als Engineering-Tool

Warum eignet sich MS VISIO als Engineering Tool?

VISIO in Verbindung mit EXCEL und/oder ACCESS ist durchaus flexibel als Engineering- und Ausgabe-Tool geeignet, wenn man es versteht VBA-Scripte zu programmieren.

Der Ansatz ist ein Excel-Frontend zu programmieren, welches als Regiezentrum eingesetzt wird. Die Datenspeicherung übernimmt Access, das Zeichnen und die Ausgabe VISIO.

In VISIO müssen Shapes, Schablonen und Templates erstellt werden.

In Excel werden Zuordnungen und Aktionen erstellt.

Eine Aktion ist z. Bsp. das Erstellen von Visio-Zeichnungen auf Basis eines Visio-Typicals.

Der Datenzugriff (Parameterdaten) erfolgt auf eine Access-Tabelle.

6. MS ACCESS und DeltaV Parser

Woher kommen die DeltaV Daten?

DeltaV kennt den Export in eine sog. fhx-Datei. Das ist eine Textdatei in einem bestimmten standardisierten Ausgabeformat.

Diese Datei wird durch einen Parser analysiert und in einer Access-Datenbank serialisiert.

In der Access-Datenbank sind alle Parameter und Referenzdaten enthalten.

Der Zugriff erfolgt über den Modulnamen.

Jedes Modul hat Top-Level Properties wie Name, Beschreibung, Type, Sub-Type etc.

Diese Modul-Eigenschaften werden in das Excel-Frontend eingelesen. In der Modulliste erfolgt die Zuweisung eines Typicals. Hierfür können in Excel hilfreiche Filter gesetzt werden.

7. DeltaV Analyse und Sonderlogik

Soweit so einfach.

Leider ist die Welt selten so einfach.

Abhängig von der Historie des Systems und den Fähigkeiten des DeltaV Applikations-Engineers sehen die meisten DeltaV Applikationen meist wesentlich „chaotischer“, besser ausgedrückt, individualisierter aus.

Deshalb müssen vor der Zuweisung und Ausgabe in Excel und Visio die Module analysiert werden.

Wenn keine klassenbasierte Lösung programmiert wurde kann es letztlich sein, dass sich in jedem Modul eine Sonderlogik befindet.

Deshalb muss man sich zunächst die Fragen stellen:

1. Was ist die Hauptaufgabe des Modules?
2. Welches sind die verfahrenstechnischen Sonderlogiken?
3. Was ist DeltaV-Spezifischer Code?
4. Ist das Kunst oder kann das weg?

Sonderlogiken können oftmals auch keinem spezifischen Templates zugeordnet werden, weil darin u.U. mathematische Berechnungen und spezielle Logiken abgebildet werden. Diese Blöcke werden auf einer neutralen VISIO-Zeichnung mit der spezifischen Verschaltung und den spezifischen Parametern dargestellt.

8. VISIO Templates

In VISIO-Templates werden die verfahrenstechnisch relevanten Blöcke und Parameter dargestellt.

Die Module werden den Templates zugeordnet.

Während des Doku-Laufs werden die individuellen Parameter und die externen Referenzen auf die VISIO Zeichnung übertragen.

Alle Zeichnungen sind miteinander verlinkt.

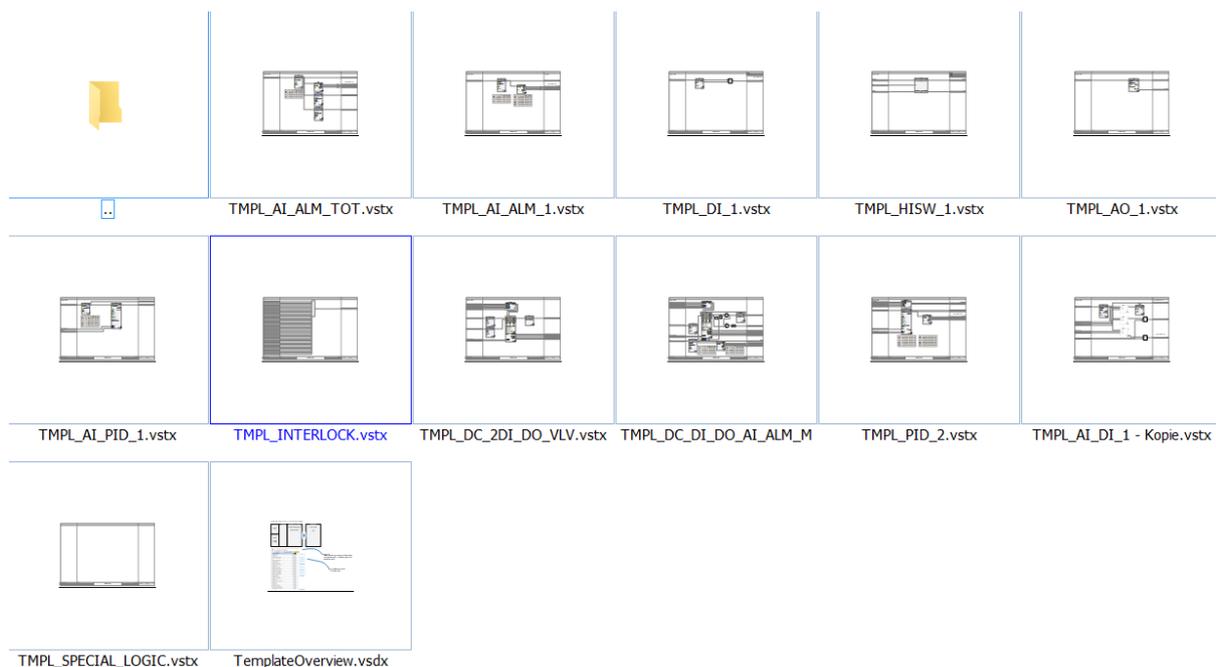


Abbildung 1 VISIO-Template Ablage

In einem projektspezifischen Dokument werden die Templates beschrieben.

9. VISIO und EXCEL im Zusammenspiel

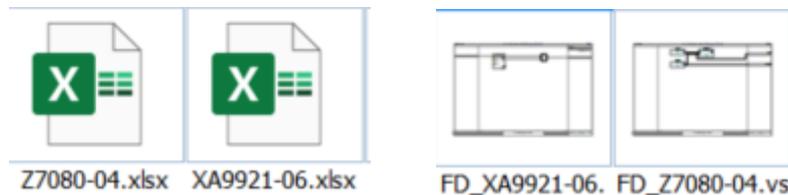
VISIO eignet sich exzellent um logische Zusammenhänge grafisch darzustellen.

So können Module in DeltaV, programmiert in der FBD oder SFC Sprache, sehr gut dokumentiert werden.

Die Grafik, d.h. die Shapes in VISIO sind jedoch nur die halbe Wahrheit.

Blöcke werden parametrisiert, es gibt Blöcke wie CALC, ACTION oder Condition Blöcke, die sog. „Expressions“, d.h. Programm-Code enthalten. Diese Information ist i.d.R. sehr gut in Listenform dokumentierbar. Um diesem Punkt Rechnung zu tragen, werden pro Modul 2 Dokumente erzeugt, eine VISIO-Zeichnung und ein EXCEL-Dokument. Beide Dokumente sind horizontal miteinander verlinkt.

Die VISIO und EXCEL Dokumente sind wiederum vertikal miteinander verlinkt.



10. Höhere Automatisierungsfunktionen

In DeltaV Applikationen werden gewöhnlich nicht nur Module in Funktions-Block-Diagramm Sprache verwendet, kurz FBD, sondern eben auch automatische Abläufe in der sog. SFC Sequence-Function-Chart Sprache programmiert.

Der SFC Dialekt kann ein Modul sein, ein darin befindlicher Composite, also ein anwenderspezifischer Baustein, ein sog. EQM Equipment-Modul, ein PLM oder Phase Logic Modul, oder auch Batches.

Die SFC Struktur mit ihren Steps, Aktionen und Transitionen werden ebenfalls in VISIO dargestellt.

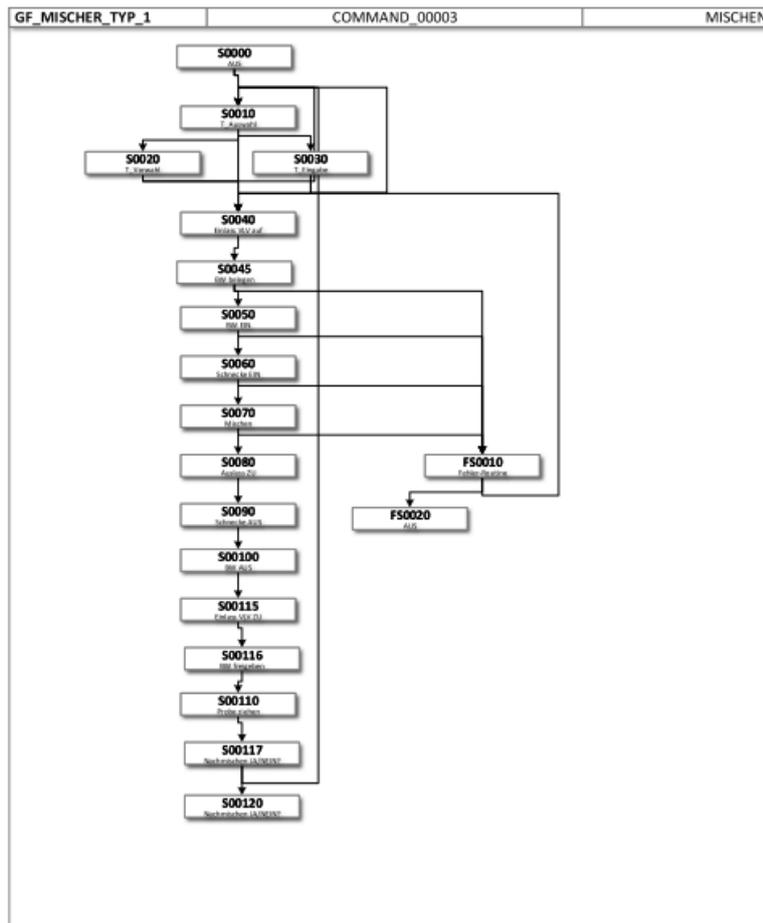


Abbildung 2 SFC Struktur in VISIO

11. Cause & Effect Matrix

Neben der FBD und SFC Darstellung der Automatisierung sind sog. C&E Matrizen hilfreich, die Gesamapplikation bzw. deren Einzelnen Area und Units besser zu verstehen.

Die Darstellung erfolgt in EXCEL.

Zu unterscheiden sind Abschalt-Matrizen (Cause&Effect-Matrix) und Schalt-Matrizen (SFC) für die höheren Automatisierungs-Funktionen.

12. Prozess-Grafikbilder

Die Prozess-Grafikbilder können ebenfalls in VISIO dokumentiert werden, indem die Dynamos mit ihrer XY-Position in VISIO dargestellt werden.

13. Einbindung in existierende Planungs-Tools

In aller Regel existieren in einer laufenden Anlage Planungs-Tools zum Erzeugen von Stromlaufplänen eventuell sogar Funktionsplänen oder R&I's. Zu nennen wären COMOS oder PRODOK. Es gibt natürlich noch unzählige viele andere Tools oder Methoden.

Was auf jeden Fall zu bedenken ist an dieser Stelle, wie bekomme ich vorhandene Stromlaufpläne oder allgemeiner Planungsdokumente mit den Software-Plänen verheiratet, d.h. wie startet ein Planer seine Suche und wie sind Dokumente miteinander verlinkt, welche Methoden und Ablagesysteme gibt es.

Die Funktionspläne befinden sich im VISIO oder pdf-Format in einem Unterverzeichnis. D.h. die Verlinkung der Funktionspläne untereinander ist sichergestellt.

14. Fazit

Mit Hilfe von EXCEL, ACCESS, VISIO und ein wenig VBA Scripting ist es möglich eine DeltaV Applikation so zu dokumentieren, dass ALLE mitdiskutieren können, Programmierer, Automatisierer, Verfahrenstechniker und Mitarbeiter des Betriebs.

Version: R0 Draft
TOM
Datum: 13.08.2021 11:41:00
TBE

Funktionale Dokumentation aus DeltaV TBE

Technisches Büro Eichstetter
www.t-eichstetter.de
0170 383 4705

***** END OF DOCUMENT *****