



Fakultät Informatik

**Entwicklung eines VHDL-basierten Applikationsbeispiels
zur Ablösung des FM350-2 Moduls durch das TM FAST
Modul inklusive Integration in das TIA Portal und
Aspekte des Software Engineerings**

Bachelorarbeit im Studiengang Informatik

vorgelegt von

Jan-Eric Gedicke

Matrikelnummer 3643446

Erstgutachter: Prof. Dr. Trommler

Betreuer: Conrad, Maximilian

© 2025

Inhaltsverzeichnis

1	Einleitung	1
1.1	Problemstellung und Motivation	1
1.2	Zielsetzung der Arbeit	1
1.3	Methodisches Vorgehen und Struktur der Arbeit	1
2	Grundlagen	2
2.1	SPS-Technologie und Module im industriellen Umfeld	2
2.2	Überblick über das FM350-2 Modul	2
2.3	Vorstellung des TM FAST Moduls	2
2.4	Einführung in VHDL und FPGA-Technologie	2
2.5	Grundlagen des TIA Portals und dessen Bedeutung für die Integration	2
3	Analyse der Ablösung des FM350-2 durch das TM FAST Modul	3
3.1	Funktionsweise und Limitierungen des FM350-2	3
3.1.1	Grundlegende Funktionen der FM 350-2	3
3.1.2	Einschränkungen der FM 350-2	4
3.2	Einsatzgebiete der FM 350-2	5
3.2.1	Haupteinsatzgebiete	5
3.2.2	Typische Anwendungen	5
3.2.3	Anwendungsbeispiel: Kartonabfüllung	5
3.3	Vorteile und neue Möglichkeiten durch das TM FAST Modul	6
3.4	Anforderungen an die Neuentwicklung	6
4	Entwicklung des VHDL-basierten Applikationsbeispiels	8
4.1	Konzept und Architektur des neuen Applikationsdesigns	8
4.2	Umsetzung der Funktionalität in VHDL	8
4.3	Test und Validierung der VHDL-Implementierung	8
4.4	Vergleich mit der bisherigen Lösung	8
5	Integration in das TIA Portal	9
5.1	Herausforderungen der Einbindung des TM FAST Moduls	9

5.2	Anpassungen an die Projektumgebung	9
5.3	Implementierung und Tests im TIA Portal	9
5.4	Evaluation der neuen Lösung im industriellen Kontext	9
6	Aspekte des Software Engineerings	10
6.1	Best Practices in der VHDL-Entwicklung	10
6.2	Modularisierung und Wiederverwendbarkeit	10
6.3	Teststrategien und Debugging-Techniken	10
6.4	Dokumentation und Wartung der Software	10
7	Fazit und Ausblick	11
7.1	Zusammenfassung der Ergebnisse	11
7.2	Kritische Reflexion und Herausforderungen	11
7.3	Potenzielle Weiterentwicklungen und zukünftige Anwendungen	11
8	Plan	13
9	Fragen	16
10	Tests	18
11	Passwörter	19

1 Einleitung

1.1 Problemstellung und Motivation

1.2 Zielsetzung der Arbeit

1.3 Methodisches Vorgehen und Struktur der Arbeit

2 Grundlagen

2.1 SPS-Technologie und Module im industriellen Umfeld

2.2 Überblick über das FM350-2 Modul

2.3 Vorstellung des TM FAST Moduls

2.4 Einführung in VHDL und FPGA-Technologie

2.5 Grundlagen des TIA Portals und dessen Bedeutung für die Integration

3 Analyse der Ablösung des FM350-2 durch das TM FAST Modul

3.1 Funktionsweise und Limitierungen des FM350-2

Die FM 350-2 ist ein Zähl- und Messmodul, das speziell für schnelle Zählprozesse und Frequenzmessungen entwickelt wurde. In diesem Abschnitt werden die wesentlichen Funktionen, deren Anwendung sowie die Einschränkungen des Moduls beschrieben.

3.1.1 Grundlegende Funktionen der FM 350-2

Die FM 350-2 bietet eine Vielzahl von Funktionen zur Steuerung und Überwachung von Zählprozessen. Hierzu gehören unter anderem das Initialisieren der Zähler, das Steuern von Digitalausgängen sowie das Laden und Auslesen von Zählwerten.

3.1.1.1 Steuerung der Baugruppe mit extttFC CNT2_CTR (FC2)

Die Funktion extttFC CNT2_CTR ermöglicht die Steuerung der Digitalausgänge der FM 350-2 sowie die Verwaltung der Software-Tore. Zudem können Rückmeldesignale ausgelesen werden.

Hauptfunktionen:

- Initialisierung des Zähler-DBs
- Auslesen der Rückmeldesignale und Speicherung in der Struktur CHECKBACK_SIGNALS
- Übertragen der Steuersignale aus der Struktur CONTROL_SIGNALS zur FM 350-2

Die Funktion muss zyklisch (z. B. im OB1 oder im Weckalarm OB35 der S7-300) aufgerufen werden, um eine kontinuierliche Überwachung und Steuerung zu gewährleisten.

3.1.1.2 Laden von Zählerständen, Grenz- und Vergleichswerten (extttFC3 / extttFB3)

Mit der Funktion extttFC CNT2_WR oder dem Funktionsbaustein extttFB CNT2WRPN können Zählerstände und Vergleichswerte neu geladen werden. Diese Funktion sollte nur verwendet werden, wenn im laufenden Betrieb neue Werte in das Modul eingespielt werden müssen.

3.1.1.3 Auslesen von Zähl- und Messwerten (extttFC4 / extttFB4)

Die extttFC CNT2_RD / extttFB CNT2RDPN Funktion dient zum zyklischen Auslesen der aktuellen Zählwerte. Falls keine Leseaufträge benötigt werden, kann auf diese Funktion verzichtet werden, um die Prozesslast zu minimieren.

3.1.1.4 Diagnosedaten lesen (extttFC5)

Im Falle eines Diagnosealarms ermöglicht die Funktion extttFC DIAG_RD das Laden der Diagnosealarmdaten in den Zähler-DB, um Fehlerursachen zu analysieren und entsprechende Maßnahmen einzuleiten.

3.1.2 Einschränkungen der FM 350-2

Trotz ihrer vielseitigen Einsatzmöglichkeiten weist die FM 350-2 einige Limitierungen auf:

- Begrenzte Flexibilität in der Anpassung an komplexe Steuerungsaufgaben
- Kein FPGA-basierter Aufbau, wodurch individuelle Anpassungen nicht möglich sind
- Begrenzte Echtzeitfähigkeit im Vergleich zu modernen Zählmodulen

3.2 Einsatzgebiete der FM 350-2

3.2.1 Haupteinsatzgebiete

Die FM 350-2 wird hauptsächlich dort eingesetzt, wo Signale gezählt und schnelle Reaktionen auf vorgegebene Zählerstände erforderlich sind. Besonders relevant ist sie für Anwendungen in der Fertigungs- und Automatisierungstechnik.

3.2.2 Typische Anwendungen

Typische Einsatzbereiche sind:

- Verpackungsanlagen
- Sortieranlagen
- Dosieranlagen
- Drehzahlregelungen und Überwachung von Gasturbinen

3.2.3 Anwendungsbeispiel: Kartonabfüllung

Ein typisches Beispiel für die Nutzung der FM 350-2 ist die Abfüllung von Teilen in Kartons:

- Kanal 0 zählt die Teile und steuert das Abfüllventil.
- Kanal 1 steuert den Transport der Kartons und erfasst die Anzahl der gefüllten Kartons.
- Das System stellt sicher, dass exakt die vorgegebene Anzahl an Teilen pro Karton abgefüllt wird.

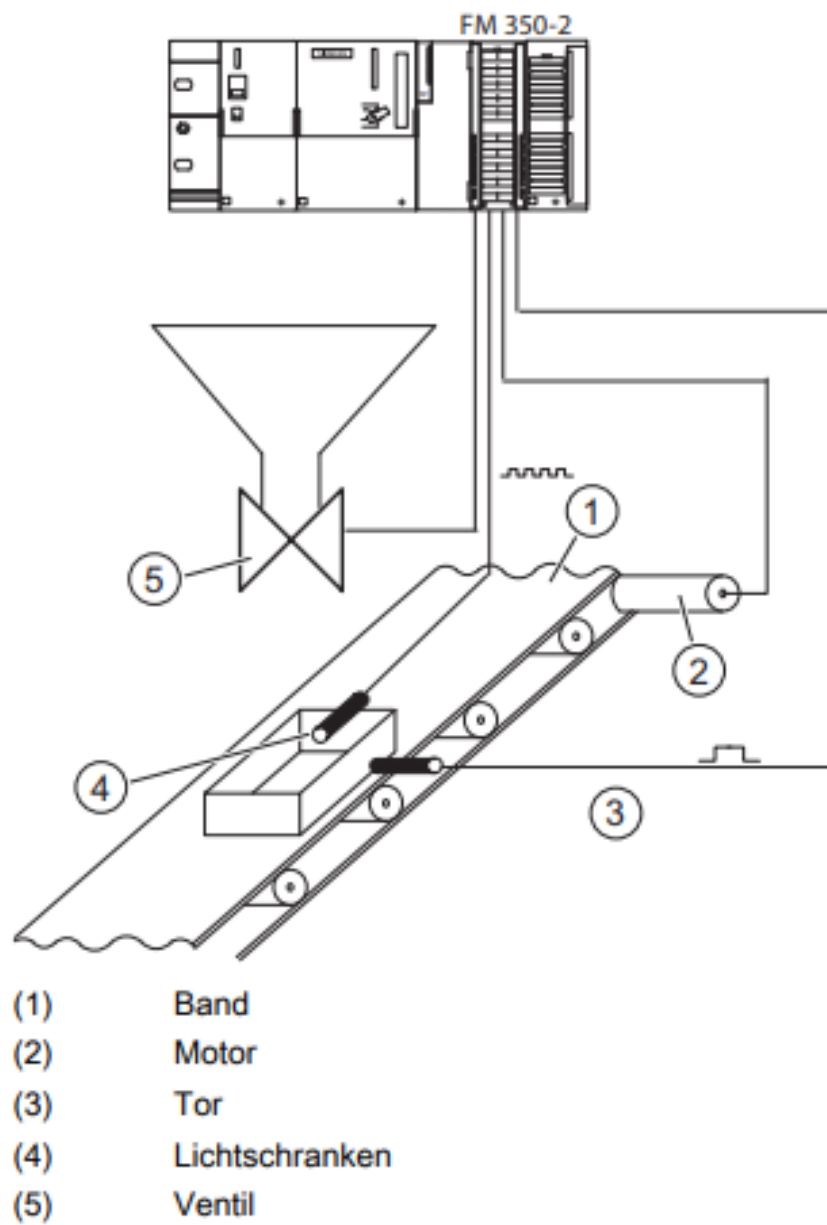


Abbildung 3.1: Beispiel für den Einsatz einer FM 350-2 in der S7-300 [7]

3.3 Vorteile und neue Möglichkeiten durch das TM FAST Modul

3.4 Anforderungen an die Neuentwicklung

Kriterium	FM350-2
Unterschiede	
Architektur	<p>Reines Zähler- und oh-pro-grammierbare Hard-ware-plattform</p>
Funktion	<p>Spezialisiert kom-plexe Prozess- und Überwachungs-gaben über-nehmen</p>
Prozess	<p>Klassische Zähler- und Mess-Produkte, die von der über-einge-ordnet und</p>

4 Entwicklung des VHDL-basierten Applikationsbeispiels

4.1 Konzept und Architektur des neuen Applikationsdesigns

4.2 Umsetzung der Funktionalität in VHDL

4.3 Test und Validierung der VHDL-Implementierung

4.4 Vergleich mit der bisherigen Lösung

5 Integration in das TIA Portal

5.1 Herausforderungen der Einbindung des TM FAST Moduls

5.2 Anpassungen an die Projektumgebung

5.3 Implementierung und Tests im TIA Portal

5.4 Evaluation der neuen Lösung im industriellen Kontext

6 Aspekte des Software Engineerings

6.1 Best Practices in der VHDL-Entwicklung

6.2 Modularisierung und Wiederverwendbarkeit

6.3 Teststrategien und Debugging-Techniken

6.4 Dokumentation und Wartung der Software

7 Fazit und Ausblick

7.1 Zusammenfassung der Ergebnisse

7.2 Kritische Reflexion und Herausforderungen

7.3 Potenzielle Weiterentwicklungen und zukünftige Anwendungen

Literaturverzeichnis

- [1] Ernst Klett Verlag (2004): *Infoblatt Siemens AG*. Verfügbar unter: <https://www.klett.de/alias/1036900>. Zugriff: 5. Januar 2025.
- [2] Merkur (2022): *Siemens – Geschichte, Aktie und Tätigkeitsfelder*. Verfügbar unter: <https://www.merkur.de/wirtschaft/siemens-geschichte-aktie-und-taetigkeitsfelder-91268844.html>. Zugriff: 19. Januar 2024.
- [3] Siemens, 2023, zitiert nach de.statista.com: *Umsatz von Siemens SE seit 2005*. Verfügbar unter: <https://de-statista-com.thn.idm.oclc.org/statistik/daten/studie/73827/umfrage/umsatz-von-siemens-seit-2005/>. Zugriff: 5. Januar 2025.
- [4] ResearchGate: Verfügbar unter: https://www.researchgate.net/figure/SA-88s-physical-model-and-an-example-production-allocation_fig2_308384237/. Zugriff: 11. Januar 2025.
- [5] Siemens (2022): *Pflichtenheft-SmartFactory*. Internes Dokument.
- [6] Liam Bee (2022): *PLC and HMI development with Siemens TIA Portal: develop PLC and HMI programs using standard methods and structured approaches with TIA Portal V17*. Packt Publishing, Limited.
- [7] Siemens AG (2011): *Zählerbaugruppe FM 350-2 Gerätehandbuch*. Verfügbar unter: https://cache.industry.siemens.com/dl/files/178/1105178/att_20226/v1/s7300_fm350_2_operating_instructions_de_de-DE.pdf. Zugriff 11. März 2025.
- [8] Siemens AG (2023): *TMFast Programmeier Handbuch*. Verfügbar unter: https://cache.industry.siemens.com/dl/files/088/109816088/att_1144842/v2/s71500_tm_fast_programming_manual_de-DE_de-DE.pdf. Zugriff 12. März 2025.

8 Plan

Hier sind einige Schritte, die Ihnen dabei helfen können, eine VHDL-Applikation für das TM FAST-Modul zu entwickeln, um das alte FM350-2-Modul zu ersetzen:

Anforderungsanalyse

- Analysieren Sie sorgfältig die Funktionalität und Schnittstellen des FM350-2-Moduls.
- Identifizieren Sie die Kernfunktionen, die das neue TM FAST-Modul erfüllen muss.
- Definieren Sie klar die Eingangs- und Ausgangssignale, die Kommunikationsprotokolle und andere relevante Spezifikationen.

Architekturentwurf

- Entwerfen Sie eine modulare Architektur für Ihre VHDL-Applikation.
- Unterteilen Sie die Funktionalität in logische Blöcke wie Steuerungslogik, Kommunikationsschnittstellen, Datenpfade usw.
- Legen Sie die Schnittstellen zwischen den einzelnen Modulen fest.

VHDL-Entwicklung

- Beginnen Sie mit der Implementierung der einzelnen VHDL-Module basierend auf Ihrem Architekturentwurf.
- Achten Sie auf eine strukturierte und modulare Codeerstellung, um die Wartbarkeit und Erweiterbarkeit zu erleichtern.
- Verwenden Sie VHDL-Sprachkonstrukte wie Prozesse, Signale, Typen und Pakete, um Ihre Applikation übersichtlich und effizient zu gestalten.

Simulation und Test

- Erstellen Sie umfangreiche Testbänke, um Ihre VHDL-Applikation gründlich zu überprüfen.
- Führen Sie Simulationen durch, um das korrekte Verhalten der einzelnen Module und der Gesamtapplikation zu verifizieren.
- Testen Sie die Applikation schrittweise, um mögliche Fehler frühzeitig zu erkennen und zu beheben.

Integration und Validierung

- Integrieren Sie Ihre VHDL-Applikation in die Gesamtarchitektur des TM FAST-Moduls.
- Testen Sie die Applikation in der realen Umgebung, um ihre Kompatibilität und Funktionalität zu validieren.
- Passen Sie die Applikation gegebenenfalls an, um Anforderungen und Spezifikationen des TM FAST-Moduls zu erfüllen.

Dokumentation und Versionskontrolle

- Erstellen Sie eine ausführliche Dokumentation, die die Architektur, Schnittstellen, Konfiguration und Verwendung Ihrer VHDL-Applikation beschreibt.
- Nutzen Sie ein geeignetes Versionskontrollsystem, um Änderungen an Ihrer Applikation nachzuverfolgen und zu verwalten.

Durch diese strukturierte Vorgehensweise können Sie sicherstellen, dass Ihre VHDL-Applikation für das TM FAST-Modul die Anforderungen des FM350-2-Moduls erfüllt und in die Gesamtarchitektur des TM FAST-Systems nahtlos integriert wird.

9 Fragen

Müssen Alarme umgesetzt werden? -> Müssen nicht berücksichtigt werden
Müssen Hardwareconfigs von der CPU oder von TMFast gehandhabt werden? -> hardwareconfig
der (z.B. Ventile) ist in VHDL (andere Kommunikationsprotokolle und co pro z.B.
Ventile) Wie verhalten sich die In- und Outputs?

- Sollen alle Eingänge Zähl- und Richtungseingang sein oder Aufteilung zwischen den beiden? Nur zähler
-

wie funktionieren ventile überhaupt?

bit an die cpu senden

nachlauf bei inkrement zum abfüllen

cpu macht - tm fast macht ventil zu

auf ist 1 zu ist low für ventil

abfüllende menge muss übergeben werden

vhdl auf hoher ebene darstellen (als baustein)

synchron oder asynchron (für zähler oder generell) -> Alles Synchron (TM Fast arbeitet (einstellbar mit 50mHz) schneller als ein ventil) -> dadurch zählt es schneller als inkremente des ventils

clk wird bei änderung aktiviert (High->Low / Low->High)

steuerung des ventils über PLC oder TMFAST? Erstmal Plc mit bool senden (überlauf für counter (justierung)) / TMFast erfordert Kommunikations-Protokolle
Filter?

Anwendung in mehreren oder einem Process in vhdl?

soll der timer nur zählen, wenn die flanke positiv wird? oder auch wenn sie negativ wird? -> nur bei positiv

Allgemein: Schriftgröße und Seitenränder? Zitieren von Handbüchern (Graue Literatur) und ChatGPT?

10 Tests

rückgabe über das (fb_if) feedback_interface

EncEmu encoder für impulse

schwierigkeit: di1 oder di0 zählen

RD_Rec sind azyklisch und nur selten benutzt

library test 8 counter -> name mit test cases Jürgen, ..., Kollege aus Prag

Muss ich nicht allein schreiben

Mathias fragen wenn funktionierendes Programm -> in den Nightly test muss in den nightly build

-systemlogic updaten auf v2

-zähler des sea teams verwenden

- 0000_0000_0000_0000_0000_0000_0000_0001 -> FB_IF(0)(0)<='1'; - /-> Q0.0 (FB_IF(0))

- FB_IF(1)(0) == QD4

11 Passwörter

PLC: Siemens123