

Eine neue Hardware-Komponente zur Fuzzy-Pattern-Klassifikation

Michael Schlegel, Göran Herrmann, Dietmar Müller
TU Chemnitz, Fakultät für Elektrotechnik und Informationstechnik,
Professur Schaltungs- und Systementwurf

Tel.: 0371 531 3159, Fax: 0371 531 3186,
e-mail: michael.schlegel@infotech.tu-chemnitz.de

Kurzfassung

Dieser Beitrag beschreibt die Entwicklung einer neuen Hardware-Implementierung zur Klassifizierung unscharfer Meßdaten aus der Verschleißüberwachung hoch beanspruchter Maschinenteile nach der Methode der Fuzzy-Pattern-Klassifikation. Eine bereits an der Professur existierende Hardwarelösung verfügt nicht über die Leistungsfähigkeit und Flexibilität, die für einen universellen Einsatz notwendig ist. Der Prototyp des neuen Fuzzy-Pattern-Klassifikators wurde auf der Basis eines VirtexE 400 FPGAs implementiert und verfügt über eine leistungsfähige Integer- und Gleitkommaarithmetik. Das entstandene System wurde erfolgreich unter Labor- und Feldbedingungen getestet.

1 Einleitung

Die Verschleißerkennung an stark beanspruchten Maschinenteilen spielt bei der Sicherung der Produktqualität und des Produktionsprozesses eine wichtige Rolle. Dazu werden u. a. Vibrationen der Maschinenteile gemessen und ausgewertet. Für die Auswertung kann das in Abschnitt 2 beschriebene Verfahren der Fuzzy-Pattern-Klassifikation Anwendung finden. Die Umsetzung dieses Verfahrens auf einem PC ist z. B. in MATLAB oder C/C++ einfach möglich. Allerdings ist es sehr aufwendig und unpraktisch, neben den zu überwachenden Maschinenteilen einen zusätzlichen PC zu betreiben, insbesondere dann, wenn auch Massen Anwendungen (z. B. Lagerüberwachung im PKW oder Achsüberwachung in Zügen) in Betracht gezogen werden. Deshalb entstand im Rahmen von [1] eine Hardware-Lösung auf der Basis von sechs Xilinx-FPGAs der 4000er Serie.

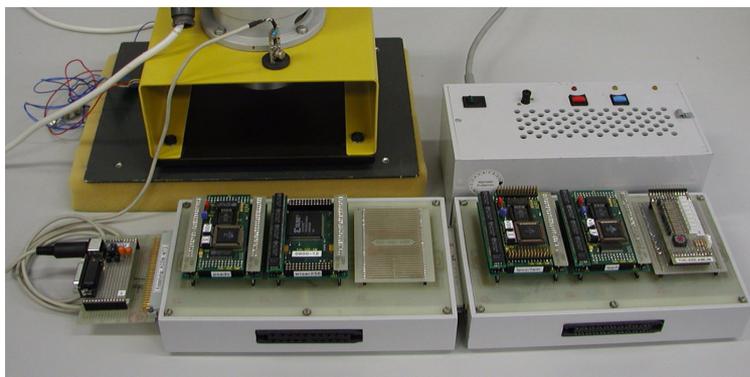


Bild 1 Bestehende Hardwarelösung nach [1]

Aufgrund der eingeschränkten Leistungsfähigkeit der eingesetzten FPGA-Bausteine ergibt sich auch eine eingeschränkte Leistungsfähigkeit des Klassifikationssystems hinsichtlich Flexibilität der Eingangsdaten und Klassifikatorparameter. So ist z. B. die Verwendung

von gedrehten Klassifikatoren (s. Abschnitt 2) nicht möglich, da die Koordinatentransformation nicht realisiert werden konnte.

Für ein universell einsetzbares Klassifikationssystem im Rahmen des Verbundprojektes „Mikrosystemgestützte Verzahnungsdiagnose“ war deshalb ein neues System zu entwickeln.

2 Das Verfahren der Fuzzy-Pattern-Klassifikation

Der Klassifikator hat die Aufgabe, gemessene Daten, die sog. Merkmale (z. B. Vibrationen), bestimmten Klassen von Zuständen (z. B. scharfes/verschlissenes Werkzeug, gutes/defektes Getriebe oder Lager) zuzuordnen. Er beruht auf dem Fuzzy-Pattern-Verfahren nach [1], [2] und bestimmt mit Hilfe folgender unimodaler Zugehörigkeitsfunktion die Sympathie eines Satzes von Merkmalen zu einer bestimmten angelernten Klasse:

$$y(i) = \frac{1}{1 + \frac{1}{m} \cdot \sum_{l=1}^m \left(\frac{1}{B_{i,l}} - 1 \right) \cdot \left(\frac{|x_l - S_{i,l}|}{C_{i,l}} \right)^{D_{i,l}}} \quad (\text{GL 1})$$

Dabei ist $S_{i,l}$ der Schwerpunkt eines Merkmals l in der Klasse i und x_l das zu bewertende Merkmal. $B_{i,l}$ und $C_{i,l}$ beschreiben die Zugehörigkeit des Merkmals am Rand der Klasse und $D_{i,l}$ bestimmt den Formfaktor für den Rand der Klasse. Das Resultat ist der Sympathievektor $y(i)$, der die Sympathie eines Satzes von m Merkmalen zur Klasse i beschreibt. Der Klassifikator arbeitet dabei mit unsymmetrischen Zugehörigkeitsfunktionen, d. h. B , C und D können unterschiedliche Werte annehmen, je nachdem ob x größer oder kleiner als der Klassenschwerpunkt ist. Weiterhin enthält das Verfahren eine Koordinatentransformation der Merkmale bezüglich der Achsen des Merkmalsraumes. Dies ist dann erforderlich, wenn eine Klasse im Merkmalsraum nicht parallel zu einer Merkmalsachse liegt und die Größe des Klassengebietes für eine bestmögliche Trennung zu benachbarten Klassen minimiert werden soll.

Für eine zweidimensionale Zugehörigkeitsfunktion mit den Parametern $S=(-1; 0)$, $B_=(0,5; 0,3)$, $B_+=(0,1; 0,3)$, $C_=(1; 2)$, $C_+=(3; 2)$, $D_=(2; 4)$, $D_+=(20; 4)$ ergibt sich dann die in Bild 2 dargestellte Zugehörigkeitsfunktion.

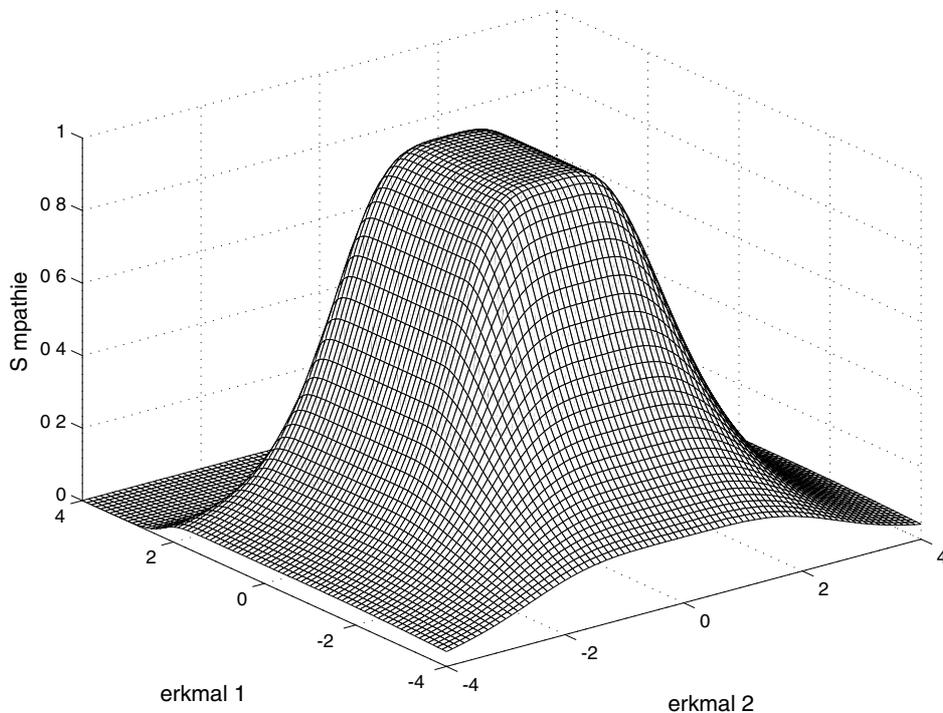


Bild 2 Beispiel einer zweidimensionalen Zugehörigkeitsfunktion

Der Vorteil der Fuzzy-Pattern-Klassifikation besteht darin, daß die Parameter B , C , D und S sowohl angelernt als auch aus Expertenwissen bestimmt werden können.

3 Implementierung der neuen Hardware-Komponente

Voruntersuchungen

Zu Beginn der Arbeiten wurden Untersuchungen mit dem Ziel durchgeführt, die notwendigen Bitbreiten der Eingangsdaten zu bestimmen. Bei der bestehenden Hardware ist diese auf 5 Bit begrenzt. Im Gegensatz dazu ergaben Berechnungen zur Merkmalsextraktion im Rahmen des Verbundprojektes „Mikrosystemgestützte Verzahnungsdiagnose“ [3] notwendige Bitbreiten von 36 Bit, wobei diese Zahl primär auf den zu überdeckenden Wertebereich der einzelnen Merkmale und weniger auf deren Genauigkeit zurückzuführen ist. Da eine Arithmetik für 36 Bit breite Eingangsdaten sehr viele Ressourcen benötigt und eine serielle Arithmetik langsam ist, wurde beschlossen, für die Klassifikation Gleitkommazahlen zu verwenden.

Als nächstes war eine geeignete Zielarchitektur für den Klassifikator zu bestimmen. Mikrocontroller sind zwar preiswert, für Gleitkommaberechnungen aber in der Regel zu langsam. DSP sind für die geplanten Anwendungen zu teuer und überdimensioniert. Da für mögliche spätere Anwendungen auch der Massenmarkt mit über 10.000 Exemplaren in Betracht gezogen wurde, ergab sich als günstigste Lösung die Umsetzung des Systems in Hardware, wobei zum Prototyping Xilinx-FPGAs der VirtexE-Reihe zum Einsatz kommen sollten. Um das System später in unterschiedlichen Zieltechnologien realisieren zu können, durfte der VHDL-Code keine technologiespezifischen Elemente, insbesondere Block-RAM oder Elemente der Xilinx-CoreLib, enthalten.

Realisierung

In einer ersten Version sollte der Klassifikator fest in Hardware implementiert werden. Da aber insbesondere die Koordinatentransformation eine sequentielle Berechnung erfordert und damit ein Pipelining nicht sinnvoll ist, wäre ein umfangreicher Automat zu Steuerung des Systems notwendig. Modifikationen an diesem Automaten erfordern aber eine neue, zeitaufwendige Hardware-Synthese des Chips. Um eine größere Flexibilität zu erreichen, erfolgte deshalb die Entwicklung eines speziellen Prozessors, dessen Befehlssatz, Ein- und Ausgänge auf die Klassifizierung von Meßdaten nach dem Algorithmus der Fuzzy-Pattern-Klassifikation abgestimmt sind. Einen Überblick über die Systemstruktur gibt Bild 3. Der Prozessor enthält neben einer 16 Bit Integer- auch eine 24 Bit Gleitkommaarithmetik, für die je eine Registerbank mit 16 universell nutzbaren Registern vorhanden ist.

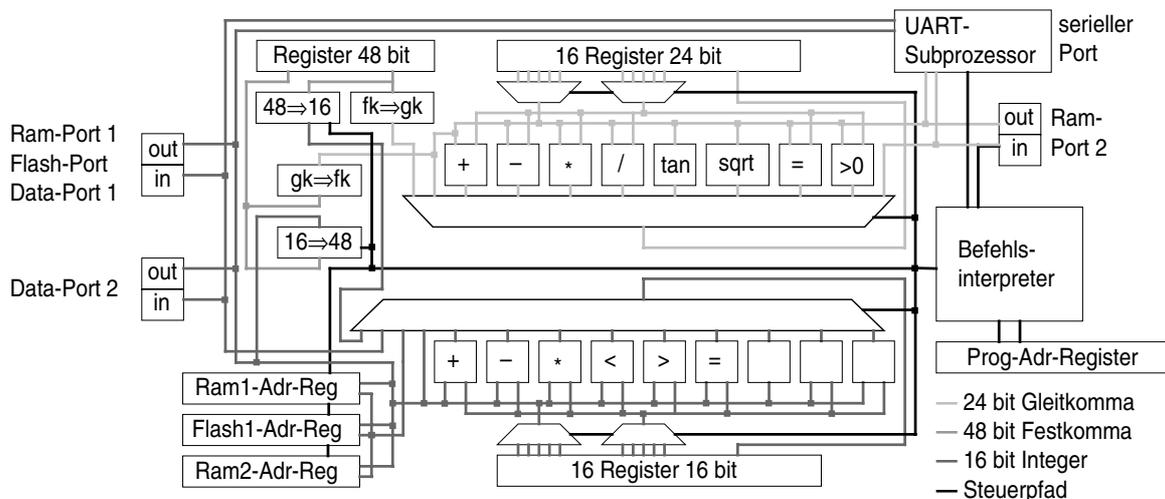


Bild 3 Systemstruktur des Fuzzy-Pattern-Prozessors

Das verwendete Gleitkommaformat hat folgende Struktur:

Vorzeichenbit 0: pos, 1:neg	Mantisse 16 Bit, unsigned MSB repräsentiert 2^{-1}	Exponent 7 Bit, signed
--------------------------------	---	------------------------

Daraus ergibt sich ein Wertebereich von -10^{18} bis $+10^{18}$ bei einer Genauigkeit von 4...5 Nachkommastellen. Im Gegensatz zum IEEE-Gleitkommaformat werden keine Sonderzustände abgebildet. Die mathematischen Operationen sind dabei so ausgelegt, daß ein Werteüberlauf abgefangen wird. Statt dessen liefert die Operation als Ergebnis den kleinsten/größten darstellbaren Wert zurück. Die mit dieser FPU erreichbare Genauigkeit ist für die Auswertung realer, gemessener technischer Größen in der Regel ausreichend und läßt sich mit vertretbarem Aufwand in Hardware implementieren.

Eine Besonderheit der FPU stellen die Operatoren $*$, $/$ und sqrt dar. Ihre Berechnung erfolgt auf der Basis einer Multiplikation und sukzessiven Approximation (Bild 4). Vorzeichen und Exponent lassen sich bei diesen drei Operatoren direkt ausrechnen. Bei der Multiplikation werden die Mantissen beider Operanden direkt multipliziert und anschließend wieder auf das Gleitkommaformat unter Anpassung des Exponenten normiert. Bei Division und Radix sucht der Approximierer diejenige Zahl X , die mit B oder mit sich selbst multipliziert A ergibt. Dazu beginnt der Approximierer mit dem Wert 1100000000000000 . Ist das Ergebnis kleiner als A , fährt er mit 1110000000000000 fort. Bei einem Ergebnis größer als A , wird als nächster Wert 1010000000000000 angesetzt usw. Ist das Ergebnis gleich A oder das 16. Bit der Mantisse geprüft, wird die Berechnung beendet. Besitzt der Exponent von A bei der Berechnung der Wurzel einen ungeraden Wert, so muß das Ergebnis noch mit Wurzel 2 multipliziert werden. Für die Multiplikation benötigt diese Struktur 1 Takt, für Division und Radix maximal 17 Takte.

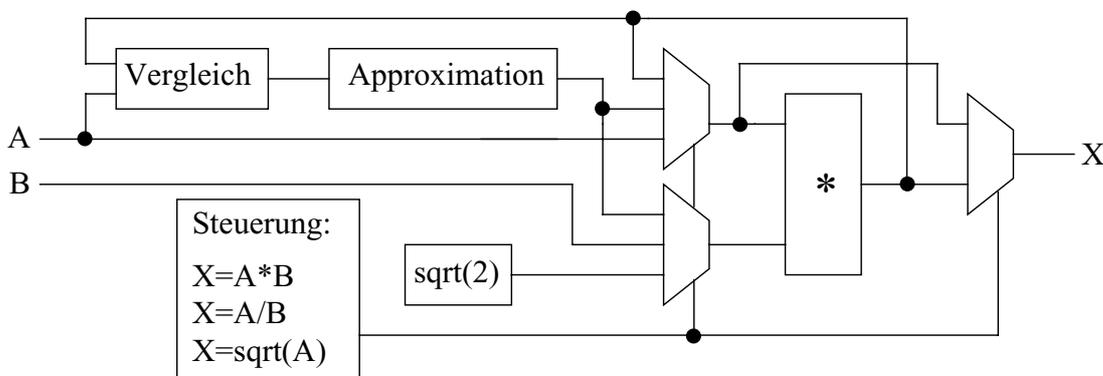


Bild 4 Funktionseinheit zur Multiplikation, Division und Radix

Neben den Integer- und Gleitkommaregistern verfügt der Prozessor über ein 48 Bit breites Festkomma-Eingangsregister. Dieses dient der flexiblen Übernahme von Merkmalen mit unterschiedlichen Bitbreiten. Die Daten in diesem Register können dann ins Gleitkommaformat gewandelt oder ausschnittsweise in die Integer-Register übertragen werden.

Der Prozessor verfügt über drei externe Bussysteme. Das erste ist 16 Bit breit und dient zum Anschluß von RAM und FLASH-Bausteinen mit 16 Bit Adressraum sowie zur Dateneingabe. Die FLASH-Bausteine können mittels des eingebauten FLASH-Controllers direkt angesprochen werden. Der zweite Bus dient zur Verbindung mit einem 24 Bit breiten RAM bei 16 Bit Adressraum, wobei die untersten 2 KBit als Programmspeicher dienen. Der dritte Bus ist wieder 16 Bit breit und dient der Datenausgabe. Weiterhin verfügt der Prozessor über ein I^2C -Interface, das mit einem seriellen EEPROM verbunden ist, aus dem der Programmcode beim Booten in den 24 Bit breiten RAM geladen wird. Ein UART-Interface dient zur Datenein- und Ausgabe

sowie zum Debuggen. Bei aktivierter UART hat diese vollen Zugriff auf alle Speicherbereiche, alle Datenregister sowie lesenden Zugriff auf Programm-Counter und Speicheradressregister.

Softwareunterstützung

Zur Programmierung des Prozessors wurde ein Assembler geschrieben, der im Gegensatz zu herkömmlichen Assemblern längere, selbsterklärende Ausdrücke für die Befehle verwendet. Für Register können Alias-Namen vergeben werden, die sich jederzeit im Programm ändern lassen. Der Prozessor verfügt über 64 Befehle, zur Befehlsverarbeitung sind 2 Takte oder bei einigen Befehlen entsprechend mehr erforderlich.

Für die Simulation und das Debuggen der Klassifikationsprogramme entstand ein Simulator (Bild 5) und ein Hardwaredebugger, wobei die Funktionalität des Hardwaredebuggers aufgrund der begrenzten Datenübertragungskapazität der seriellen Schnittstelle eingeschränkt ist.

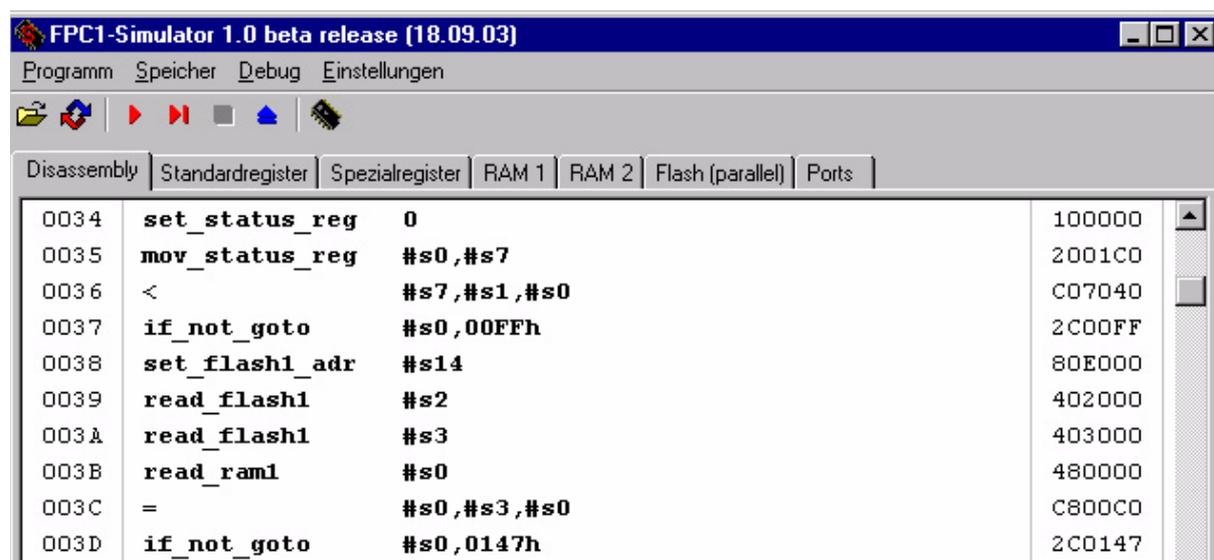


Bild 5 Softwaresimulator

4 Ergebnisse

Mit dem im Rahmen dieser Arbeit entwickelten Prozessor steht eine leistungsfähige Hardware-Basis für die Fuzzy-Pattern-Klassifikation zur Verfügung. Dabei besitzt der Prozessor durch seine Gleitkommaarithmetik eine große Flexibilität und Geschwindigkeit. Die maximale Anzahl von Merkmalen und Klassen bei der Klassifikation wird nur durch den zur Verfügung stehenden Speicher begrenzt. Der Prozessor belegt ca. 80 % eines VirtexE 400, arbeitet mit 20 MHz und kann bis zu 10 Millionen Gleit- oder Festkommaoperationen pro Sekunde ausführen. Eine Synthese für einen 0,15 µm CMOS-Prozess ergab eine Chipfläche von unter 10 mm² bei maximal 200 MHz Taktfrequenz. Über Europractice könnte der Fuzzy-Klassifikationsprozessor für unter 2 Euro pro Stück (ohne Test und Packaging) hergestellt werden.

Für den Klassifikator wurde zudem eine eigene Leiterplatte entwickelt, die neben dem FPGA diverse RAM-, FLASH- und I/O-Bausteine trägt. Bild 6 zeigt die Anwendung des Klassifikators zur Getriebediagnose im Rahmen des Verbundprojekts „Mikrosystemgestützte Verzahnungsdiagnose“. Für die Anwendung des Fuzzy-Pattern-Klassifikators in diesem Projekt wurde zusätzlich eine A/D-Wandlerkarte und eine weitere FPGA Baugruppe für eine 8K-FFT entwickelt. Das System wurde im Feldversuch angelernet und ist in der Lage, unter realen Bedingungen Schäden an Getriebezahnrädern frühzeitig zu erkennen.

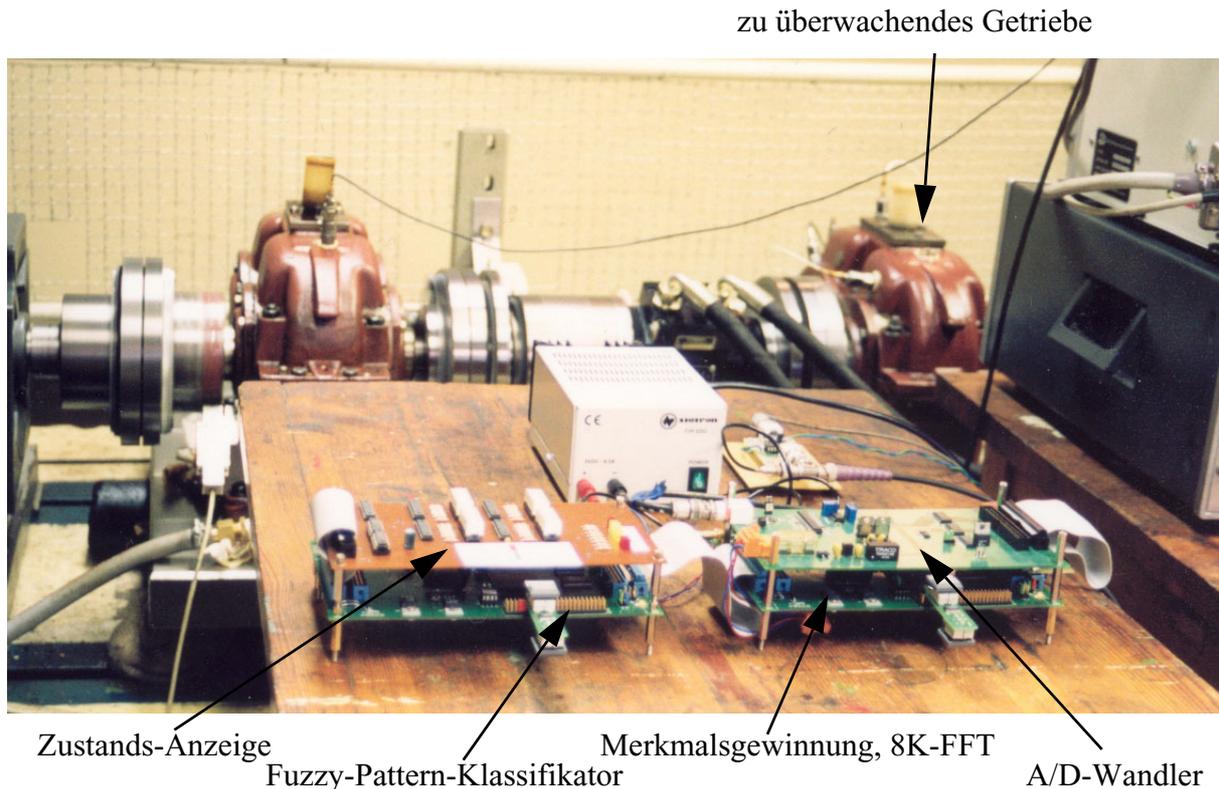


Bild 6 Einsatz des Demonstrators im Prüffeld der Fakultät für Maschinenbau der TU Dresden

5 Ausblick

Die gegenwärtig vorliegende Version ist hervorragend zur Klassifikation von unscharfen Mustern geeignet. Zum Anlernen des Klassifikators ist allerdings noch ein PC erforderlich. In einer erweiterten Version des Klassifikators soll dieser in der Lage sein, sich auf der Basis von Lern-daten selbst anzulernen, wodurch sich weitere Anwendungsfelder erschließen lassen. Dazu erfolgt gegenwärtig die Erweiterung der FPU um trigonometrische und logarithmische Funktionen sowie eine Überarbeitung des Befehlssatzes. Parallel dazu findet eine Weiterentwicklung der Softwarekomponenten statt. So wird zukünftig die Programmierung auch mit einem klassischen Assembler-Dialekt möglich sein.

Danksagung

Die hier vorgestellten Arbeiten entstanden im Rahmen des durch die FVA (Forschungsvereinigung Antriebstechnik e.V.) geförderten Verbundprojekts „Mikrosystemgestützte Verzahnungsdiagnose“ [3].

Literatur

- [1] Eichhorn, K.: *Entwurf und Anwendung von ASICs für musterbasierte Fuzzy-Klassifikationsverfahren*. Dissertation, Technische Universität Chemnitz, 21. März 2000
- [2] Bocklisch, S. F.: *Prozeßanalyse mit unscharfen Verfahren*. Verlag Technik, Berlin 1987
- [3] Abschlußbericht des Verbundprojekts „Mikrosystemgestützte Verzahnungsdiagnose“. LVA-Forschungsvorhaben 242, Forschungsheft 702 der Forschungsvereinigung Antriebstechnik e.V., Frankfurt a. M.