

ELV-Fuzzy-Controller Teil 3

Den Datenaustausch zwischen dem Fuzzy-Controller und einem externen Mikrocontroller beschreibt dieser Artikel nebst einer kurzen Einführung in die zugehörige Windows-Software.

Mikrocontroller-Interface

Im vorherigen Artikel („ELVjournal“ 4/96) hatten wir 2 Möglichkeiten kennengelernt, den ELV-Fuzzy-Controller mit einem externen Mikrocontroller über das Mikrocontroller-Interface zu verbinden. Im folgenden erläutern wir den Austausch einzelner Bytes zwischen beiden Systemen sowie das Protokoll zur Abwicklung des gesamten Datenaustausches.

Kommunikationsaufbau

Das Interface besteht intern aus einer Steuerlogik sowie aus zwei 8Bit-Speicherzellen (Latches), wobei von einem externen Mikrocontroller die eine Zelle nur gelesen und die andere nur beschrieben werden kann.

Zum Aktivieren und Deaktivieren der Schnittstelle dient die \overline{CS} -Leitung. Führt diese Leitung High-Pegel, befindet sich der Fuzzy-Controller im Stand-by-Betrieb. In diesem Zustand ist der Datenbus-Ausgang hochohmig, und die Pegel der restlichen Steuerleitungen (\overline{RD} , \overline{WD}) haben keinen Einfluß auf das Interface.

Soll nun eine Kommunikation über die Schnittstelle aufgebaut werden, müssen zuvor alle Steuerleitungen High-Pegel führen. Wird z. B. ein Byte aus dem Fuzzy-Controller gelesen, so ist dieser mit einem Low-Pegel auf der \overline{CS} -Leitung zu aktivieren. Anschließend wird die \overline{RD} -Leitung ebenfalls auf „low“ gesetzt, und das Byte erscheint auf dem Datenbus, um vom externen Mikrocontroller eingelesen zu werden. Zum Abschluß ist zuerst die \overline{RD} -Leitung und nachfolgend die \overline{CS} -Leitung auf High-Pegel zu setzen.

Bei der Übermittlung eines Bytes an den Fuzzy-Controller, wird wie beim Lesen zuerst die \overline{CS} -Leitung auf Low-Potential gelegt. Im nächsten Schritt muß vom externen Mikrocontroller das zu übertragende Byte auf den Datenbus gegeben werden, und anschließend ist die \overline{WD} -Leitung auf „low“ zu setzen. Nach einer Mindest-Wartezeit von 20 ns kann die \overline{WD} -Leitung und nach weiteren 20 ns auch die \overline{CS} -Leitung wieder auf High-Pegel gesetzt werden, wodurch die Übertragung abgeschlossen ist.

Zur Vermeidung von Übertragungsfeh-

lern dürfen die Zeiten, die im Timing-Diagramm des Mikrocontroller-Interface (Abbildung 1) angegeben sind, nicht unterschritten werden. In der Praxis tritt diese Problematik jedoch äußerst selten auf, da nach dem heutigen Stand der Technik Mikrocontroller mit Zugriffszeiten von 50 ns zu den schnellsten gehören.

Regler B. Bei einer Konfiguration des Fuzzy-Controllers mit 3 Eingängen für Regler A und nicht aktiviertem Regler B sind im Kommunikationsbyte die Bits 0, 1 und 7 gesetzt, sofern keine Datenübertragung vom PC stattfindet. Ist der Fuzzy-Controller bereit zum Datenaustausch, sendet der externe Mikrocontroller eine Null, wenn

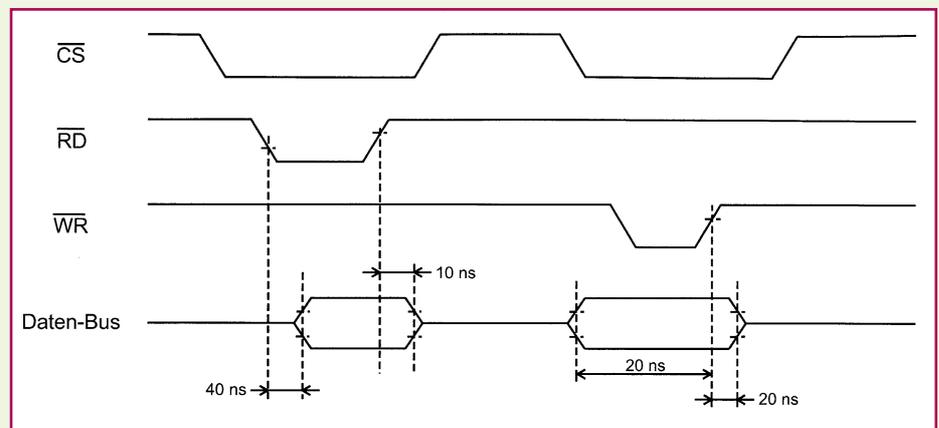


Bild 1: Timing-Diagramm des Mikrocontroller-Interfaces

Protokoll des Mikrocontroller-Interfaces

Mit dem Protokoll wird festgelegt, auf welche Weise dem Fuzzy-Controller die Eingabedaten übermittelt werden und in welcher Form dieser die Ausgabedaten zurückgibt. Um eine Datenübertragung aufzubauen, liest der externe Mikrocontroller ein sogenanntes Kommunikationsbyte (siehe Tabelle 1) vom Fuzzy-Controller über das Interface ein.

Sind in diesem Byte die Bits 6 und 7 gesetzt (Bit 7 steht hier und in der weiteren Beschreibung für das höherwertigste Bit im Byte), bedeutet dies, daß vom PC eine Datenübertragung stattfindet und die Ein- und Ausgabe von Werten zur Zeit nicht möglich ist. Wird hingegen ein Byte eingelesen, indem das Bit 6 Low-Pegel und das Bit 7 High-Pegel führt, so signalisiert der Fuzzy-Controller seine Bereitschaft zur Datenaufnahme. Die Bits 0 bis 2 des gleichen Bytes geben in binärer Form die Anzahl der Eingänge von Regler A an und die Bits 3 bis 5 die Anzahl der Eingänge von

Regler A selektiert werden soll, während ein Wert ungleich Null für die Selektion von Regler B steht. Anschließend liest der Mikrocontroller die Schnittstelle solange aus, bis er ein Byte empfängt, in dem alle Bits Null sind. Damit zeigt der Fuzzy-Controller an, daß er das übertragene Byte verarbeitet hat und fordert den externen Mikrocontroller auf, den ersten Eingabewert zu übersenden.

Im Anschluß an die Übermittlung liest der externe Mikrocontroller die Schnittstelle erneut aus, bis er ein Byte empfängt, in dem nur Bit 0 gesetzt ist. Der Fuzzy-Controller zeigt hiermit wieder die Verarbeitung an und fordert den externen Mikrocontroller zur Übertragung des zweiten Eingangswertes auf. Dieser Einlesezyklus wiederholt sich solange, bis alle Eingangswerte des gewählten Reglers übermittelt sind, wobei mit den Bits 0 bis 2 der jeweils geforderte Wert angegeben wird (unabhängig, ob Regler A oder B selektiert ist). Die Einlesezykluszeit beträgt ca. 2,6 µs, da der Fuzzy-Controller diese Zeit benötigt, um einen übermittelten Wert zu verarbeiten und einen neuen anzufordern.

Nach Abschluß der Eingabe beginnt der Fuzzy-Controller mit der Berechnung des

Tabelle 1: Bedeutung der Bits im Kommunikationsbyte des Mikrocontroller-Interface

Bit 7	Bit 0 bis 6	Bedeutung
„high“	Bit 0 bis 2	Anzahl der Eingänge von Regler A in binärer Form
„high“	Bit 3 bis 5	Anzahl der Eingänge von Regler B in binärer Form
„high“	Bit 6	„high“, wenn eine Datenübertragung vom PC stattfindet, sonst „low“
„low“	Bit 0 bis 2	Nummer des jeweils vom Fuzzy-Controller geforderten Eingabewertes für Regler A oder B in binärer Form
„low“	Bit 3	keine Bedeutung („low“)
„low“	Bit 4	keine Bedeutung („low“)
„low“	Bit 5	„high“, wenn der Reglerausgabewert berechnet ist und ausgelesen werden kann, sonst „low“
„low“	Bit 6	„high“, wenn der Reglerausgabewert berechnet wird, sonst „low“

Reglerausgabewertes. Liest in dieser Zeit der externe Mikrocontroller ein Byte über das Interface ein, so sind in diesem Byte sämtliche Bits gelöscht bis auf das Bit 6. Hat der Fuzzy-Controller das Ergebnis ermittelt, gibt er ein Byte aus, indem bis auf Bit 5 alle Bits gelöscht sind und signalisiert somit das Ende der Berechnung.

Liest der externe Mikrocontroller dieses Byte ein, kann er durch die Übertragung eines beliebigen Bytes an den Fuzzy-Controller diesen dazu veranlassen, den Regler-Ausgabewert bei dem nächsten Lesezugriff über das Interface bereitzustellen. Voraussetzung ist jedoch, daß zwischen der Anforderung (beliebiges Byte) und dem Einlesen des Ausgabewertes seitens des externen Mikrocontrollers eine Zeit von mindestens 3 µs eingehalten wird. Sobald der Wert gelesen wurde, kehrt der Fuzzy-Controller automatisch zur Eingabebereitschaft zurück und steht für die Berechnung neuer Ausgabewerte bereit.

Zusätzlich besteht die Möglichkeit, für Informationen direkt über die Stiftleiste (BU 2) das Interface abzufragen. Findet z. B. eine Datenübertragung vom PC statt, wechselt die Steuerleitung ST 0 (Pin 12, BU 2) von Low- auf High-Pegel.

Über die Steuerleitung ST 1 (Pin 13, BU 2) kann die Zeitspanne ermittelt werden, die der Fuzzy-Controller für die Berechnung eines Ausgangswertes benötigt. Für den Betrieb ist jedoch die Auswertung dieser Leitungen nicht erforderlich, da diese Informationen über die Bus-Schnittstelle auch direkt abfragbar sind.

Software

Für den ELV-Fuzzy-Controller steht eine leistungsfähige, bedienerfreundliche PC-Software zur Verfügung. Der Anwender kann zwischen einer Windows 3.1x sowie einer reinen Windows95-Version wählen. Letztere hat den Vorteil, daß sie im 32Bit-Modus unter Windows95 läuft und somit eine höhere Verarbeitungsgeschwindigkeit als die 3.1x-Version besitzt.

Die Bedienung und die grafische Ober-

fläche ist bei beiden Windows-Versionen gleich, wobei die Windows 3.1x-Version auch unter Windows95 arbeitet. Zur Kommunikation mit dem ELV-Fuzzy-Controller-Board wird von der Software die serielle Schnittstelle benutzt. Hierüber erfolgt sowohl die Datenübertragung vom PC zum Fuzzy-Controller-Board als auch umgekehrt.

alle Funktionen der Software zu erreichen.

Zusätzlich befindet sich an der linken Fensterseite eine visuelle Funktionsleiste, mit der die wichtigsten Programmpunkte direkt aufgerufen werden können. Dies ist bei häufig benötigten Programmfunktionen von Vorteil, da der Aufruf über das Hauptmenü und weitere Untermenüs entfällt.

Mit den Graphiken, die sich auf jeder Schaltfläche der Leiste befinden, wird die jeweilige Funktion beschrieben. Des weiteren werden in der Statuszeile der gewählte Fuzzy-Regler (A oder B), die Anzahl der Eingänge, die Zahl der Regeln und das verwendete Interface angezeigt.

Mit der Systemansicht (siehe Abbildung 2) hat der Anwender die Möglichkeit, den gewählten Fuzzy-Regler in seiner Grundstruktur, den Ein- und Ausgängen mit deren Bezeichnungen sowie der Regeltabelle, graphisch darzustellen.

Darüber hinaus kann durch Anklicken

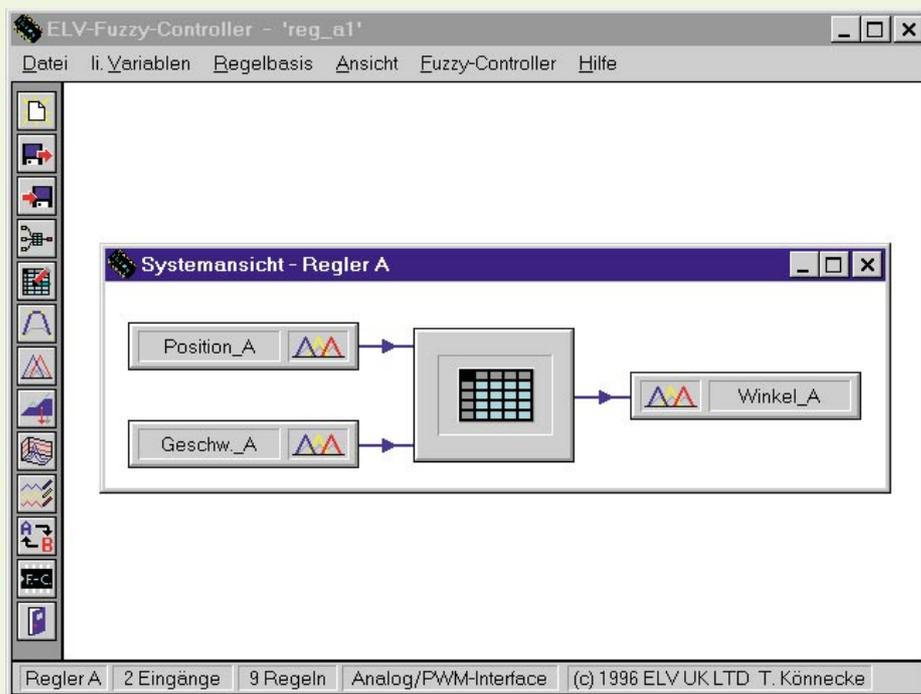


Bild 2: Oberfläche der Fuzzy-Controller-Software mit aktivierter Systemansicht.

In der Abbildung 2 ist das Hauptmenü der Fuzzy-Software mit aktivierter Systemansicht zu sehen. Der Programmname und nachfolgend die jeweilige Bezeichnung des geladenen Reglersystems wird in der Titelleiste dargestellt.

Die darunter liegende Leiste beinhaltet die Hauptmenüpunkte. Von ihnen aus kann der Anwender per Maus oder durch die Eingabe des unterstrichenen Buchstabens bei gedrückter Alt-Taste verschiedene Untermenüs aufrufen. Über diese sind

von Symbolen, wie zum Beispiel der Regeltabelle, diese editiert werden. Auch das Anklicken der Ein- und Ausgangsvariablen-Bezeichnungen ist möglich und führt zum Aufruf der jeweiligen Konfigurationsmenüs, in denen z. B. der Name und andere Parameter änderbar sind.

Im vierten, abschließenden Artikel dieser Serie wird die weitere Funktionsweise der Software beschrieben und es werden Möglichkeiten zur Konfiguration des Fuzzy-Controller-Boards aufgezeigt. **ELV**