



Starter-Kit für SGS-Thomson- Mikrocontroller der ST62-Familie

Original Starter-Kits von SGS-Thomson mit fertig aufgebauten und getesteten Programmiergeräten und Original-Datenbüchern ermöglichen den schnellen Einstieg in die Welt der ST62-Mikrocontroller.

Allgemeines

Noch vor wenigen Jahren waren Mikrocontroller-Entwicklungssysteme sehr teuer und somit nur den industriellen Anwendern zugänglich. Es gab kaum Hersteller, die Entwicklungsumgebungen anboten, die auch für den interessierten Hobbyanwender erschwinglich waren.

Heute nun bieten viele namhafte Mikrocontroller-Hersteller entsprechende Entwicklungssysteme an. Einer der Vorreiter auf diesem Gebiet ist SGS-Thomson mit Mikrocontroller-Entwicklungssystemen für die umfangreiche ST62-Familie.

Ausgestattet mit Assembler, Simulator, Programmiergerät, Dokumentation und einigen Mikrocontroller-Mustern für die ersten Programmierversuche kann bei SGS-Thomson sofort mit der Entwicklung begonnen werden.

Starter-Kit für ST620X, ST621X und ST622X

Dieses Entwicklungssystem beinhaltet alle Hardware- und Softwarekomponenten zum Programmieren der neuen ST620X, ST621X und ST622X-Mikrocontroller von SGS-Thomson. Mikrocontroller-Prototypen sind somit ohne weitere Hilfsmittel schnell und einfach zu erstellen.

Die Hardware des Starter-Kits erlaubt sowohl die „In socket“- als auch die „In circuit“-Programmierung. Zur „In socket“-Programmierung ist das Board mit einem 28poligen und einem 16poligen Programmiersockel (Textool) ausgestattet, wobei gleichermaßen OTP und EPROM-Versionen programmierbar sind.

Eine besonders komfortable Art der Programmierung ist die „In circuit“-Programmierung von Mikrocontrollern. Hierbei wird der Controller direkt auf dem Applikationsboard des Anwenders, d. h. in der Schaltung, programmiert. Die Verbindung zwischen dem Starter-Kit und der Anwenderschaltung wird dabei über ein Programmierkabel hergestellt. Das Starter-Kit ist in der Lage, alle OTP- und EPROM-Versionen der ST620X, ST621X und ST622X-Mikrocontroller, unabhängig von der Gehäuseform, innerhalb der Anwenderschaltung zu programmieren.

Das Programmiergerät ist vielseitig nutzbar. So dient es gleichermaßen als Evaluations-, Emulations- und Simulationsboard.

Umfangreiche Softwaretools mit sinnvoller Befehlsstruktur und übersichtlicher Architektur erlauben die schnelle Entwicklung von eigenen Anwendungen. Die Software ist ausgestattet mit Assembler, Linker und Debugger. Des Weiteren erlaubt die Software auch eine Simulation, inklusive IO-Read/Write.

Neben der Programmierung von Prozessoren dient das Starterkit-Board auch als Hardware-Interface für den Software-Simulator. Die Verbindung zum PC erfolgt über den Parallel-Port. Zur Simulation ist ein Monitor-Programm in einem der beiden Prozessoren des Typs ST62E25 geladen. Die Applikation wird über den Simulator gesteuert, und ein Interface-Programm verarbeitet dann die Ein- und Ausgangswerte der Applikation (analog oder digital).

Wird das Programm des vorprogrammierten Mikrocontrollers versehentlich gelöscht, so besteht jederzeit mit der beiliegenden Programmdiskette die Möglichkeit der Neuprogrammierung.

In einem Demonstrations-Mode führt der vorprogrammierte Mikroprozessor des Typs ST62E25 verschiedene Routinen in Verbindung mit dem Starter-Kit aus. Hier stehen als Beispielroutinen eine Tastaturabfrage, eine LED-Ansteuerung, ein AD-Wandler, eine Thermistor-Abfrage und ein serielles RS232-Interface zur Verfügung. Änderungen werden über das Simulationsprogramm gesteuert und angezeigt.

Zum Hardware-Lieferumfang gehören neben dem Programmierboard auch zwei Mikrocontroller des Typs ST62E25, zwei Prozessoren ST62E20 und ein Prozessor ST62E01. Ebenfalls sind das PC-Anschlusskabel und das Steckernetzteil enthalten.

Zur Dokumentation des Starter-Kits gehören das User-Manual, das ST62 General Purpose Data Book, das ST62 General Purpose Application Manual und das ST62/63 Software Development Tools User Manual.

Das in Abbildung 1 dargestellte Blockdiagramm zeigt die Ein- und Ausgangsbuchsen des Boards sowie die vorhandenen Komponenten zur Testunterstützung.

Zum Test stehen neben vier Leuchtdioden und einem Einstelltrimmer die in Abbildung 2 dargestellte Temperatur-Überwachungsschaltung und das in Abbildung 3 dargestellte serielle Interface (RS232) zur Verfügung.

Beim einfachen Software-Demonstrationsbeispiel zur Temperaturüberwachung (Abbildung 2) wird ein Heizwiderstand über dem Timerausgang des Mikrocontrollers mit einem PWM-Signal gesteuert. Die Temperaturabfrage erfolgt dann mit einem einfachen Thermistor (NTC-Widerstand), dessen temperaturproportionaler Spannungsabfall über die Kodierbrücke W5-PA4 auf den Port PA4 des Prozessors geführt wird. Dieser Port arbeitet dann als ADC-Input, wobei der analoge Spannungswert mit steigender Temperatur fällt.

In unserem auf dem Starter-Kit auch hardwaremäßig vorhandenen Beispiel ist der Temperatursensor physikalisch direkt am Heizwiderstand montiert.

Das in Abbildung 3 dargestellte RS232-Interface ist ebenfalls auf dem Starter-Kit hardwaremäßig vorhanden und zur Kommunikation mit einem Mikrocontroller des Typs ST62E25 vorgesehen. Die Schnittstelle steht an einer 9poligen Sub-D-Buchse zur Verfügung und ist kompatibel zu Standard-PC-Schnittstellen.

Über vier Kodierstecker (W7) sind die TX-, RX-, CTS- und RTS-Signalleitungen mit Port PC 4 bis PC 7 eines ST62E25-Mikrocontrollers zu verbinden. Die TX-, RX-, CTS- und RTS-Leitungen des Boards sind als Slave definiert. Soll das Board des Starterkits als Master fungieren, sind die RX- und TX-Leitungen sowie die CTS- und RTS-Leitungen im Verbindungskabel zu kreuzen.

Zur Auswahl der unterschiedlichen Demonstrationsroutinen im vorprogrammierten ST62E25 dient eine doppelreihige 6-polige Stiftleiste, wo mit Hilfe einer Kodierbrücke unterschiedliche Spannungsteiler-Einstellungen möglich sind.

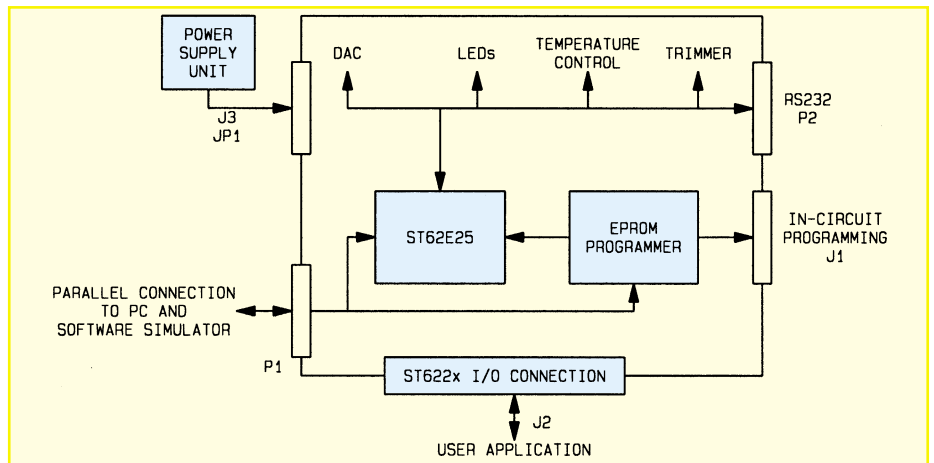


Bild 1: Blockschaltbild des Starterkit-Boards

982156101A

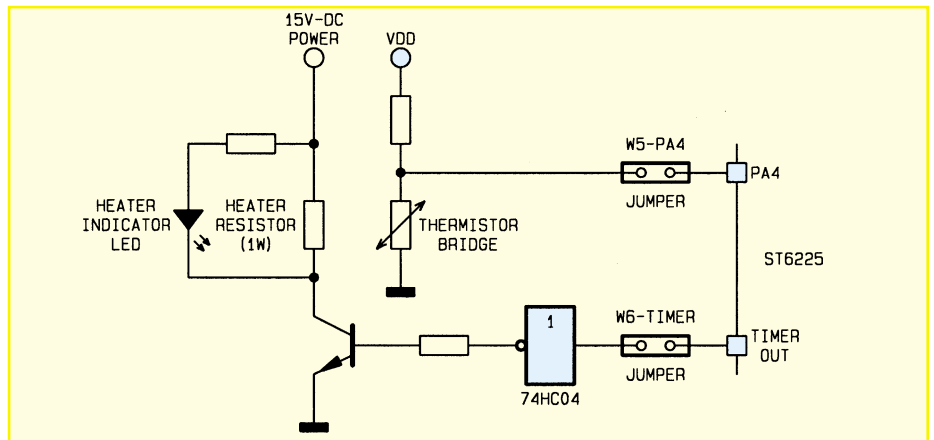


Bild 2: Temperatur-Überwachungsschaltung

982156102A

Der IO-Port (PA7) des Demoprozessors ist als AD-Wandler programmiert, so daß abhängig vom Gleichspannungspegel das Programm nach einem Reset in unterschiedliche Demonstrationsroutinen springt.

Abbildung 4 zeigt den auf der Leiterplatte vorhandenen, veränderbaren Spannungsteiler und Tabelle 1 die theoretischen Gleichspannungswerte am Prozessorport.

Im vorprogrammierten Mikrocontroller stehen dabei folgende, mit D1 bis D6 selektierbare Demonstrationsprogramme zur Verfügung:

- D1: Tastaturabfrage und Zähler
- D2: AD-Wandlung und Pegelanzeige
- D3: Temperatursteuerung
- D4: RS232-Kommunikation
- D5: Fuzzy-Logik Entwicklungsbeispiel
- D6: selbsterstelltes Programmbeispiel

Das ST622X-Starterkitboard verfügt über drei unterschiedliche Betriebsmodi.

1. User-Mode

In diesem Betriebsmode arbeitet das Starterkit als Stand-Alone-Application-Board, um im Prozessor abgespeicherte Demonstrationsroutinen oder Anwenderprogramme abzuarbeiten.

2. Simulator-Mode

Im Simulator-Linked-Mode arbeitet das Starterkit mit der auf einem PC laufenden SIMST6-Simulatorsoftware. Die Verbindung zwischen Starterkit und PC erfolgt über den Parallelport.

3. Programmier-Mode

In Verbindung mit einem PC und der ST622X-PG-Programmiersoftware dient das Board in diesem Betriebsmode zum Programmieren von EPROM- und OTP-Prozessoren der Typen ST620X, ST621X und ST622X.

Die beiden Betriebsmodi Simulator und Stand-Alone stehen ausschließlich mit dem zum Lieferumfang gehörenden, vorprogrammierten Mikrocontroller des Typs ST6225 zur Verfügung. Das Datenübertragungsprotokoll nutzt im Simulatormode die I/O-Pins PB0, PB1, PB2 und NMI des Mikrocontrollers und im Programmier-

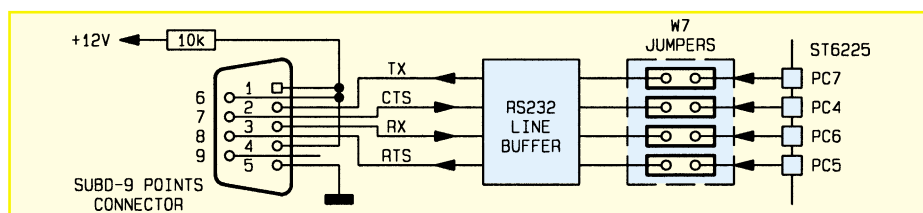


Bild 3: RS 232 Interface des Starterkits

982156103A

Tabelle 1

Kodierbrücke gesetzt	Spannung
keine	5 V
D 1	0 V
D 2	318 mV
D 3	626 mV
D 4	918 mV
D 5	1,226 V
D 6	1,54 V

mode die I/O-Pins PB5, PB6, PB7, OSCIN und Reset.

Die Spannungsversorgung des Boards erfolgt über ein zum Lieferumfang gehörendes Steckernetzteil.

Im Stand-Alone-Application-Mode können mit dem Starterkit auch eigene, selbstgeschriebene Programme gestartet werden. Dabei stehen zum Test die gleichen Ressourcen wie bei den Demoprogrammen zur Verfügung.

Des weiteren besteht die Möglichkeit, das Board mit einer externen Hardware (eigene Applikation) zu verbinden. Dazu ist auf dem Starterkit-Board eine 28polige, zweireihige Stiftleiste nachzurüsten, an der

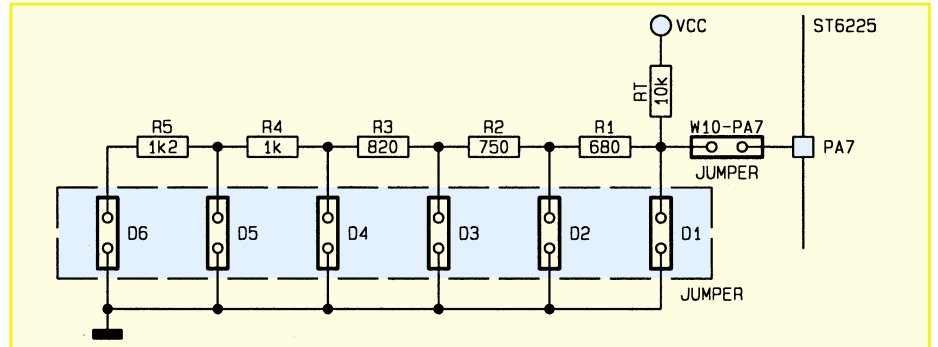


Bild 4: Spannungsteiler zur Auswahl von Demonstrationsroutinen

982156104A

sämtliche I/O-Pins des ST 6225 zur Verfügung stehen.

Tabelle 2 zeigt die Prozessor-Pin-Belegung mit den entsprechenden über Jumper selektierbaren On-Board-Ressourcen.

Zur Spannungsversorgung der externen Hardware kann dem Starterkit eine stabilisierte Betriebsspannung von 5V mit 100 mA Strom-Belastbarkeit entnommen werden.

Die zum Starterkit gehörende Dokumentation läßt kaum noch Wünsche offen. So ist neben dem Mikrocontroller-Datenbuch auch ein Applikations-Handbuch mit zahlreichen Beispielschaltungen, von der Motorsteuerung bis zum kompletten Ladegerät für NC- und NIMH-Zellen vor-

handen. Im Applikations-Datenbuch wird zu den jeweiligen Applikationen auch die zugehörige Software ausführlich vorgestellt.

Für Anwender, die über keine Assemblerkenntnisse verfügen, bietet SGS Thomsen den ST6 Realizer an. Beim Realizer handelt es sich um ein graphisches Programmierwerkzeug zum einfachen Programmieren der Mikrocontroller.

Für die größeren Prozessoren aus der ST6-Familie (ST6240 LCD, ST626X) stehen ebenfalls Starterkits direkt vom Hersteller zur Verfügung. Im „ELVjournal“ 3/98 stellen wir ein weiteres Entwicklungssystem aus diesem Bereich vor.

Tabelle 2: J2 User I/O Interface Connector

On-board resource	Jumper	ST6225 I/O	PIN	PIN	ST6225 I/O	Jumper	On board resource
+5V Supply		VDD	1	28	VSS		GND
Heater resis.						W3-PA0	LED
Control	W6-Timer	Timer	2	27	PA0		Indicator
8MHz OSC.	W2	OSCIN	3	26	PA1	W3-PA1	LED Indicator
none	-	OSCO	4	25	PA2	W3-PA2	LED Indicator
System tasks (simulator link)		NMI	5	24	PA3	W3-PA3	LED Indicator
RS 232 Driver TX Output	W7-PC7	PC7	6	23	PA4	W5-PA4	Thermistor
RS 232 Driver RX Input	W7-PC6	PC6	7	22	PA5	W4-PA5	Trimmer
RS 232 Driver RTS Output	W7-PC5	PC5	8	21	PA6	-	none
RS 232 Driver CTS Input	W7-PC4	PC4	9	20	PA7	W10-PA7	Routine selector
System tasks (Programming)	W2	VPP	10	19	PB0		System task (Simul. link)
Push-button Power on	-	RESET/	11	18	PB1		System task (Simul. link)
System tasks (Programming)	W2	PB7	12	17	PB2		System task (Simul. link)
System tasks (Programming)	W2	PB6	13	16	PB3	W9-PB3	Push-but.(-)
System tasks (Programming)	W2	PB5	14	15	PB4	W8-PB4	Push-but.(+)