STart269 Evaluation Kit

Evaluation Kit for

ST10F269 Microcontroller

FORTH-SYSTEME GmbH

P.O. Box 11 03 ● D-79200 Breisach, Germany
Kueferstrasse 8 ● D-79206 Breisach, Germany
☎+49 (7667) 908-0 ● Fax +49 (7667) 908-200 ● e-mail: sales@fsforth.de

FS FORTH-SYSTEME GmbH
Postfach 1103, D-79200 Breisach a. Rh., Germany

Release of Document: May 10, 2001
Filename: STRT269a.doc
Author: Joachim Jaeger

Table of Contents

1. Introduction

The product line of FS FORTH-SYSTEME got a brand new member of user-friendly, easy-to-use and powerful evaluation kits which is offered as ”Start269”. It is shipped with a ST10F269 controller which provides a 256 kByte on-chip Flash memory, a 10 kByte on-chip extension RAM (XRAM), a 2 kByte internal SRAM (IRAM) two CAN controller which are working according to ISO11898 (profile 2.0B) and with many additional I/O features. The board supports up to 4MByte external Flash memory and up to 1MByte Fast-SRAM which allows to build a large number of sophisticated applications.

The ST10F269 is a derivative component based on an Infineon C167 architecture but with many additional features. Please refer to the ST10F269 datasheet on the CD-ROM shipped with this package. Check the web page of STMicroelectronics (***www.st.com***) for updated versions of the device datasheet. An intelligent battery management supports the assembly of optional, up to 1MByte, low power SRAM working as non-volatile memory. Furthermore a set of user selectable switches (DIP) offers a very flexible handling of the CPU specific start-up configuration. Therefore the STart269 Evaluation Kit promises a successful start-up development for beginners and experienced users. The EVA269 Evaluation Board comes with a Flash programming tool and a well prepared demo software which helps you to become an experienced specialist for ST10 controllers.

The engineers of FS FORTH-SYSTEME created this evaluation board to fulfill a lot of actual and future user requirements. Therefore the board comes with two pre-assembled CAN transceivers. Both can easily be used without the need of additional hardware. Furthermore the controller can run up to his maximum CPU-clock of 40 MHz (25MHz in TQFP package). For fast prototyping the EVA kit includes a small wire wrap field, a potentiometer for an analog input signal, a serial EEPROM, a set of LEDs and a push button to make development easy and simple. Please refer to the following chapters to get more details about the ST10F269, the board’s capabilities and the tools which are coming with the STart269 Evaluation Kit.

2. Features of STart269 Evaluation Kit

- Evaluation Kit based on the EVA269 Evaluation Board
- 256 kByte high speed SRAM, expandable up to 1MByte
- 512 kByte external Flash, expandable up to 4MByte
- up to 1MByte optional low powered SRAM as battery buffered memory
- Optional battery back-up circuit for low powered SRAM (see BQ2201SNN)
- Two CAN transceiver (PCA82C250T, ±15V common mode offset voltage)
- Two CAN plugs (DSUB-9)
- Selectable bus configuration via DIP switch
- Selectable bootstrap mode via DIP switch
- 128 pin connector providing all CPU signals for user purposes
- Additional wire-wrap field for individual hardwired applications
- External, serial EEPROM (24LC02, I2C-Interface) for user purposes
- Including a RTC-72423 realtime clock
- The EVA269 Evaluation Board is designed for running a ST10F269 at 40 MHz (25MHz in TQFP package)
- Power supply 90..264VAC/5VDC enclosed. Input voltage to the EVA269 Evaluation Board (X2) connector is 5VDC only
- RS232-Interface cable enclosed
- Flash programming utility for easy downloading user specific code to external Flash memory.
- Hex-to-binary converter tool provided on CD-ROM as freeware
- Stand alone Flash programming utility to program the embedded on-chip Flash of the ST10F269, provided by STMicroelectronics
- C compiler (demo version only) provided by TASKING and KEIL
- Demo version of the PLS-debugger
- Including a socket to carry a four pin crystal oscillator device
- Reserved pads to hold an optional ”Quad-Connector” for Lauterbach emulator systems.

3. Getting started with EVA269 Evaluation Board

As already mentioned above, this paragraph intents to explain how you can run the board without writing own software. You will learn how the controller's internal bootstrap mode can be used to initiate a simple download of a binary program file to the embedded on-chip and external Flash memories. If you would like to use the environments of KEIL or TASKING you should refer to chapter 10.

First we recommend to create a project path to copy the demonstration software FLASH166, from CD-ROM. This software includes a bootstrap and a monitor program. The bootstrap module is a tiny piece of FS FORTH-SYSTEME firmware which is downloaded to the internal 2 kByte SRAM (IRAM) memory of the controller. It makes sure that other programs, like the monitor can be loaded to any memory location. For more details please refer to chapter 5.1 "Memory-Configuration". By the way: the user may select both, the external FLASH or the external SRAM memory to download his individual application software.

To setup your board successfully, please take care of the following steps:

- Setup the board's DIP switches as shown in Figure 1 (the board is pre-configured by FS FORTH-SYSTEME according to Figure 1). Refer to chapter 6 "System Start-up Configuration" for detailed information about these settings.

- Check the connections of the cables at the X1-and X10-connector:
  B11(X1) to C20(X1),C16(X1) to 8(X10),D16(X1) to 7(X10),C17(X1) to 6(X10) , D17(X1) to 5(X10).
  These connections are necessary to use the interrupt-mode for the RTC and to toggle the LED's LE4 to LE7.

- Make sure that the BSL switch is set to ON (S3-5=ON / P0L.4=low). This will force the CPU to bootstrap mode right after reset or power on. Bootstrap mode is always entered by the CPU whenever pin P0L.4 has been grounded prior to a reset.

- Connect the board's serial-interface (X3) to COM1 or COM2 of your Personal Computer. Caution: CTS and RTS signals are not supervised by the board's serial interface and the bootstrap loader software.

- Copy the sample project from the attached CD-ROM into your project path: “\Start269\Keil\C166\Examples\Boards\FORTH_EVA269\RTC72423\Release\ RTC72423.BIN”., or
  “\Start269\Keil\C166\Examples\Boards\FORTH_EVA269\RTC72423\Release\ 25MHZ\RTC72423.BIN”, if you are using the ST10F269 in TQFP-package. This will show you how the realtime clock works on the EVA269 evaluation board.

- Copy ”FLASH166.EXE” and ”FLASH166.OVL” onto your hard disk. Both files allow you to run the bootstrap loader and monitor in an opened DOS box.

- Make sure that the path of the ”FLASH166”-directory is included in your path environment, if you choose another directory than for your project before.

- Now start the download:
  Type ”FLASH166 /P RTC72423.BIN” at the DOS prompted line in your project directory and press the <RETURN> key. The bootstrap loader tries to connect to COM1 by default. If this does not work properly try to connect to COM2.

- Type ”FLASH166 /P RTC72423.BIN /COM2” and see what happens. If you still have problems to start downloading, please revise the settings of S3 and S4 once more. Check if you have chosen the RS232 cable delivered with the STart269 Evaluation Kit.

- Now move the BSL switch to OFF and press the reset button S1. Run a terminal program and select the line parameters: 9600Baud,No parity,8 bit data,1stopbit.

If everything is working properly the realtime of the RTC-72423 device will appear on your PC screen as shown in Figure 2. Also the LED’s LE4 – LE7 will represent a flashing light upwards from LE5 to LE6.

Figure 1: Factory setting

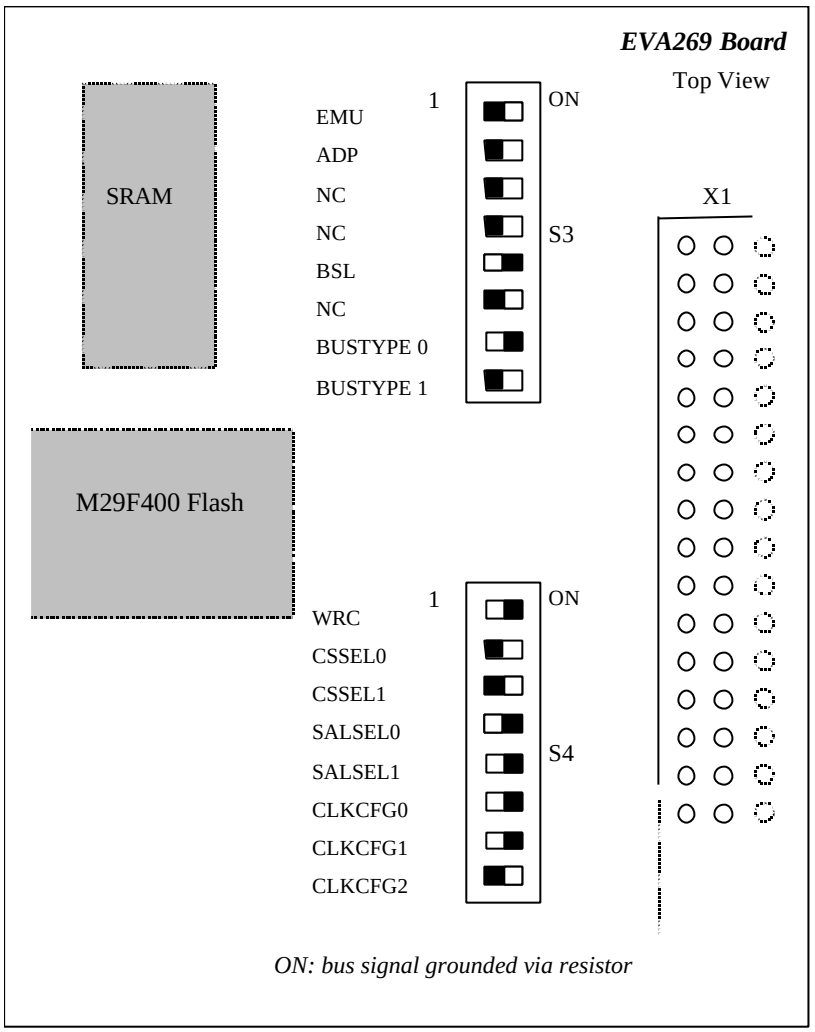
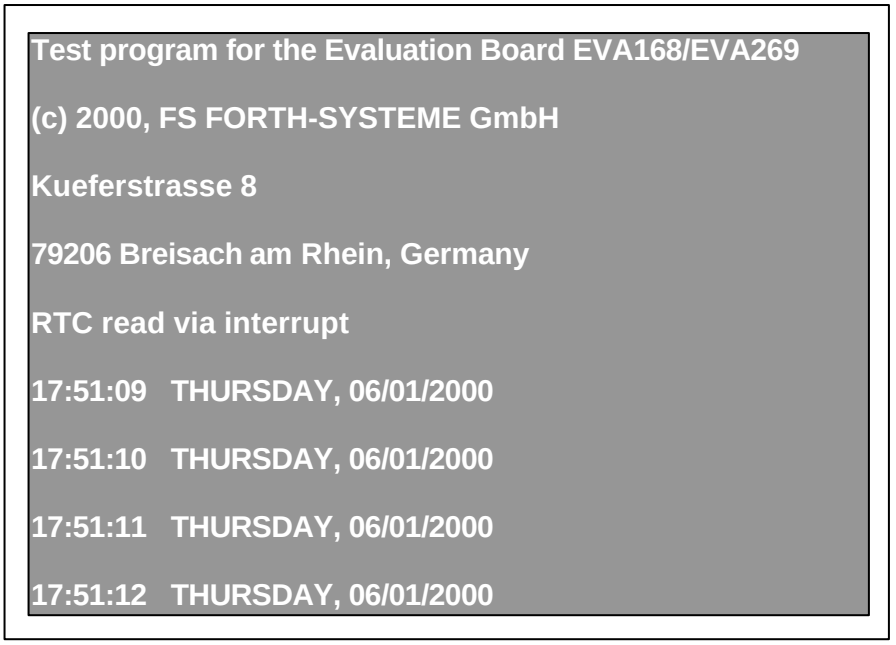


Figure 2: Output of the example RTC72423



4. Software Tools for EVA269 Evaluation Board

The *STart269* Evaluation Kit always includes the bootstrap loader and monitor program (FLASH166) from FS FORTH-SYSTEME. That's everything you need to run your individual software on a ST10F269 controller. This software is a part of the *STart269* Evaluation Kit.

FLASH166 package allows you

- to download your program to an external Flash or SRAM
- to request a memory dump from the target
- to test serial interface, CPU-I/Os and a lot of more things

If you would like to use a sophisticated programming tool which provides you the advantages and performance of "C", we would appreciate to make you an interesting offer at any time. FS FORTH-SYSTEME is distributor for KEIL and TASKING. Both companies are manufacturer of well known "C/C++" compilers and debugging tools. For the very first beginning you may use a free demo of the KEIL and TASKING "C" compilers which allows you to create individual programs with a restricted size of program code.

A detailed and well documented description of writing software with TASKING and KEIL environments is attached to this manual. If you want to run the ST10F269 as a standalone controller you may use the Flash utility which is designed and provided by STMicroelectronics. It allows a fast download of your program into the internal memory section of the CPU.

5. EVA269 Memory-Mapping

The EVA269 evaluation board supports a number of memory architectures according to the capability of the ST10F269 controller.

The board is provided by FS FORTH-SYSTEME with the following components:

Table 1: Memory

	Memory	Type	Chip-Select	Size	Comment
1	**FLASH, Boot**	**M29F400BB90N1** *or* *M29F800BB90N1* *or* *M29F160D*	**CSFLSHA**	**256k * 16** *or* *512k * 16* *or* *1024k * 16*	**assembled** *or* *optional* *or* *optional*
2	*FLASH, Extension*	*M29F400BB90N1* *or* *M29F800BB90N1* *or* *M29F160D*	*CSFLSHB*	*256k * 16* *or* *512k * 16* *or* *1024k * 16*	*optional*
3	**SRAM, High Speed**	**2 * µPD431008LE** *or* *2 *µPD434008LE*	**CSRAM**	**2 * 128k * 8** *or* *2 * 512k * 8*	**assembled** *or* *optional*
4	*SRAM, Low Power*	*2 * M5M51008AFP* *or* *2 * M5M5408LE*	*CSRAM_B*	*2 * 128k * 8* *or* *2 * 512k * 8*	*optional*

All components marked ”optional” are not mounted when the board is leaving FS FORTH-SYSTEME. You may expand the board’s addressable and non-volatile program memory up to a maximum of 2 MByte (both Flash parts must be a 29F800), or 4 MByte (both Flash parts must be a 29F160D).

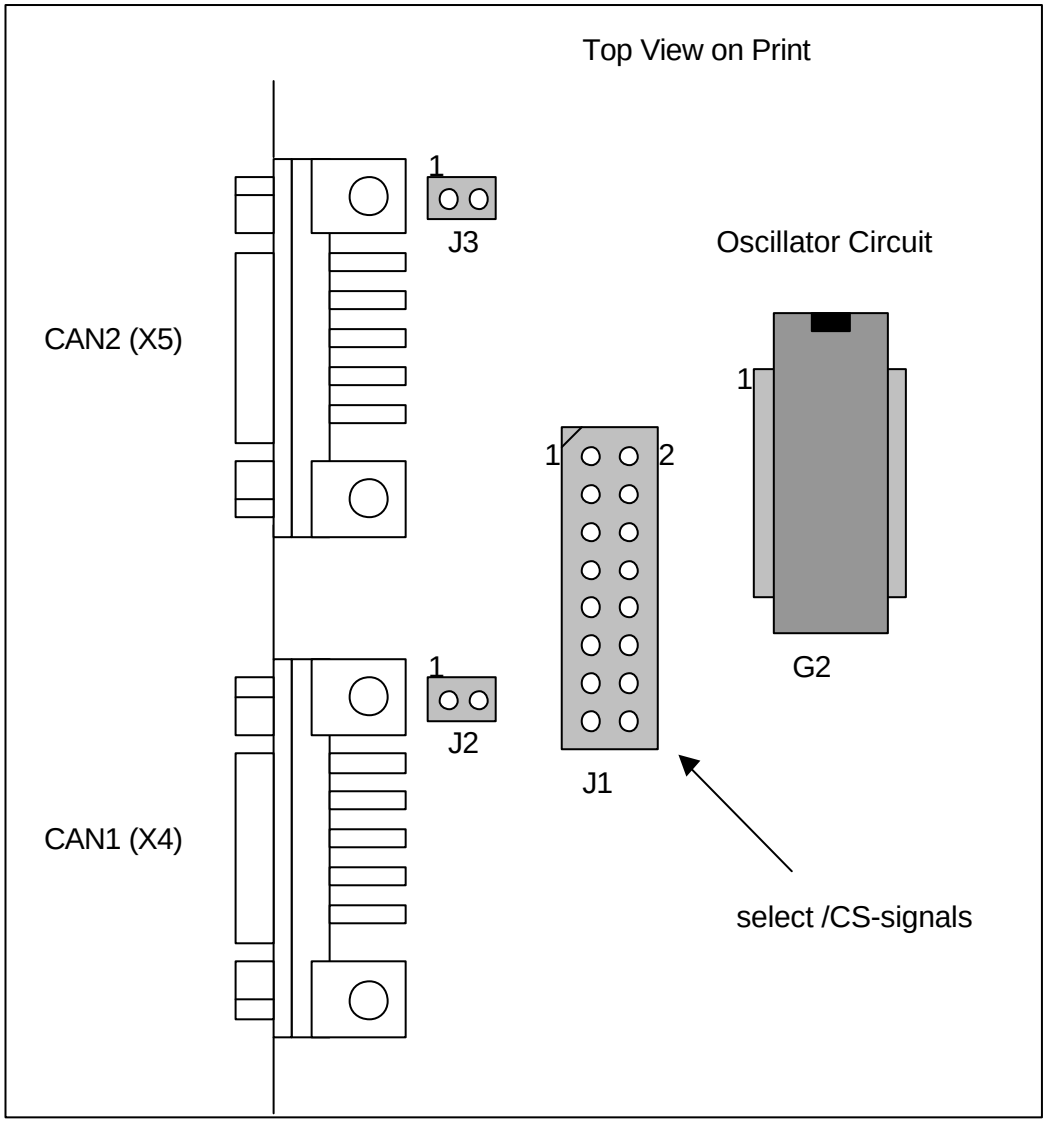
The optional static RAM (line 4 in table 1) can be assembled as an alternative memory block which provides lower power characteristics. This memory is interfaced to an external battery back-up and supervisory circuit which prevents loss of data. The circuit with the part number U6 (BQ2201) has to be added to the printed board, if you want to use the low power RAM block with a maximum size of 1 Mbyte.

This circuit generates the /CSRAM_B signal which is derived from the origin /CSRAM signal controlled by the CPU's internal address logic. If power fails this integrated circuit takes care about the power level at /CSRAM_B. To prevent damage of the SRAM block, both the supply voltage and /CSRAM_B must be stable (3V) in case of a power fail situation.

Since the selection signal /CSRAM_B is derived from the /CSRAM signal, both memory blocks (3 or 4) may never work at the same time. If you want to have both memory blocks working together you may connect the last chip select signal (/CS4) from the CPU's address logic. This signal is not used yet and can be interfaced to the BQ2201 through a short wire.

5.1. Memory-Configuration

Figure 3: Selection of Chip Select



5.2. Settings of Jumper J 1

Table 2: Chip Select

PIN	A	B	C	Chip Select	Pre-configured
1	▮			/CS0 -> CSFLSHA	Yes
2	▮				
3	▮			/CS1 -> CSRAM	Yes
4	▮				
5		▮		/CS0 -> CSRAM	No
6		▮			
7		▮		/CS1 -> CSFLSHA	No
8		▮			
9			▮	/CS2 -> CSRTC	Yes
10			▮		
11			▮	/CS3 -> CSFLSHB	No
12			▮		

▮ = *connected pins*

To make sure that your system works properly you may combine the setting shown in column (A) with column (C) provided that the system boots from an external Flash device. Otherwise if the program resides within an external SRAM you should select column (B) in combination with column (C). The jumper setting (C) works independent of the other selections. Open jumper J1(9)-J1(10) (see table 2) for disabling the chip-select signal on the RTC unit. Since the board comes without the second Flash device, J1(11)-J1(12) are removed. All jumpers which are marked as pre-configured are set by FS FORTH-SYSTEME prior to delivery.

6. System Start-up Configuration

Although most of the programmable features of the ST10F269 are either selected during the initialization phase (software) or repeatedly done during the program execution (software), there are some configurations that MUST be selected earlier, because they are used for the very first access of the CPU (hardware). For example, the CPU has to know how to deal with the external bus interface since the external memory may be organized as an 8 bit multiplexed/non multiplexed or a 16 bit multiplexed/non multiplexed bus. These selections are made during reset at the pins of PORT0, which are read at the end of the internal reset sequence. During reset, CPU-internal pull-up devices at the PORT0 lines are active, so a high level will be read, if the respective pin is left open. To get a low level, the pin must be pulled down by an external resistor (4.7k..8.2k). These resistors are already provided on the board.

The following paragraphs explain how to set all of these start-up configuration pins. Please refer to Figure 1 to find the corresponding location at the S3 and S4 switches. If a signal has to be configured as “0” (low level), move the switch to the ON-position. A “1” (high level) appears when the switch is left open (OFF-position).

6.1. EMULATION Mode

PIN EMU, S3-1

Pin P0L.0 selects the Emulation Mode, when low during reset. This mode allows the access to the integrated XBUS peripherals via the external bus interface. This mode is used for special emulator purposes and, in general, is of no use in basic applications.

Default: Emulation Mode off (P0L.0=1 / S3-1=OFF)

6.2. Adapt Mode

PIN ADP, S3-2

Pin P0L.1 selects the ADAPT Mode when low during reset. In this mode, the ST10F269 goes into a passive, floating state, which is similar to the state during reset. This mode allows, to switch the ST10F269 virtually off, so that an emulator may control the circuitry, since the pins are floating. This mode is used for special emulator purposes only.

Default: Adapt Mode off (P0L.1=1 / S3-2=OFF)

6.3. Bootstrap Mode

PIN BSL, S3-5

Pin P0L.4 activates the on-chip bootstrap loader when switch S3-5 is on during reset. The bootstrap loader allows moving a tiny start code (about 32 Byte) into the internal RAM of the ST10F269 via the serial interface ASC0 (asynchronous, serial mode). The controller remains in BSL mode until a hardware reset with P0L.4 set high occurs. Bootstrap loader mode may also be exited through a software reset instruction. In this case P0L.4 is ignored and program execution starts, fetching code from 0x00000 (internal or external). Refer to the user manual for more details.

Default: Bootstrap Mode on (P0L.4=0 / S3-5=ON)

6.4. Bus-Access-Types

PIN BUSTYP0, S3-7
PIN BUSTYP1, S3-8

With the pins P0L.6 (BUSTYP0) and P0L.7 (BUSTYP1) the bus mode can be selected for /CS0.

Table 3: Boot Bus Mode Selection

S3-8	S3-7	External Data Bus Width	External Address Bus Mode
ON	ON	8-bit Data	Demultiplexed Addresses
ON	OFF	8-bit Data	Multiplexed Addresses
OFF	ON	16-bit Data	Demultiplexed Addresses
OFF	OFF	16-bit Data	Multiplexed Addresses

In multiplexed bus modes PORT0 drives both, the 16-bit intra-segment address and the 8 (16) bit output data, while PORT1 remains in high impedance state. In demultiplexed bus modes PORT1 drives the 16-bit intra-segment address, while PORT0 (P0H, P0L) drives the 16-bit Data output. If a 8-bit data width has been selected, only P0L is driving the data output.

Default: Data-Bus-Width = 16-bit; Bus Mode = Demultiplexed

6.5. Write Configuration

Pin WRC, S4-1

Pin P0H.0 selects the initial operation of the control pins /WR and /BHE. When set high (P0H.0=1 / S4-1=OFF), this pin selects the standard mode, i.e. /WR and /BHE. When set low (P0H.0=0 / S4-1=ON) the alternate function is selected: /WR = /WRL and /BHE = /WRH

Default: /WRL and /WRH selected (P0H.0=0 / S4-1=ON)

6.6. Chip-Select-Lines

PIN CSSEL0, S4-2
PIN CSSEL1, S4-3

These signals choose the number of chip select lines, which are active after reset. Unused chip select lines may be used as general purpose I/O.

Table 4: Active Chip Select

S4-3	S4-2	active chip select	Note
OFF	OFF	/CS0.../CS4	Default
OFF	ON	none	5 Port 6 Pins free for I/O
ON	OFF	/CS0.../CS1	3 Port 6 Pins free for I/O
ON	ON	/CS0.../CS2	2 Port 6 Pins free for I/O

6.7. Segment Address Lines

PIN SALSEL0, S4-4
PIN SALSEL1, S4-5

During reset this control lines defines the number of active segment address lines. This allows to select pins of port 4 to work as general I/O lines as well as to become a part of the controller address logic. Depending on the selection of SALSELx, the required address space is chosen right after the system start-up and the program may address all locations without prior programming. Even if not all segment address lines are enabled on port 4, the ST10 uses it's complete 24-bit addressing mechanism. This allows to generate all CS signals over the entire address space even if the external bus width is less in size.

CAUTION!
The selected number of segment address lines ***cannot*** be changed by software after reset.

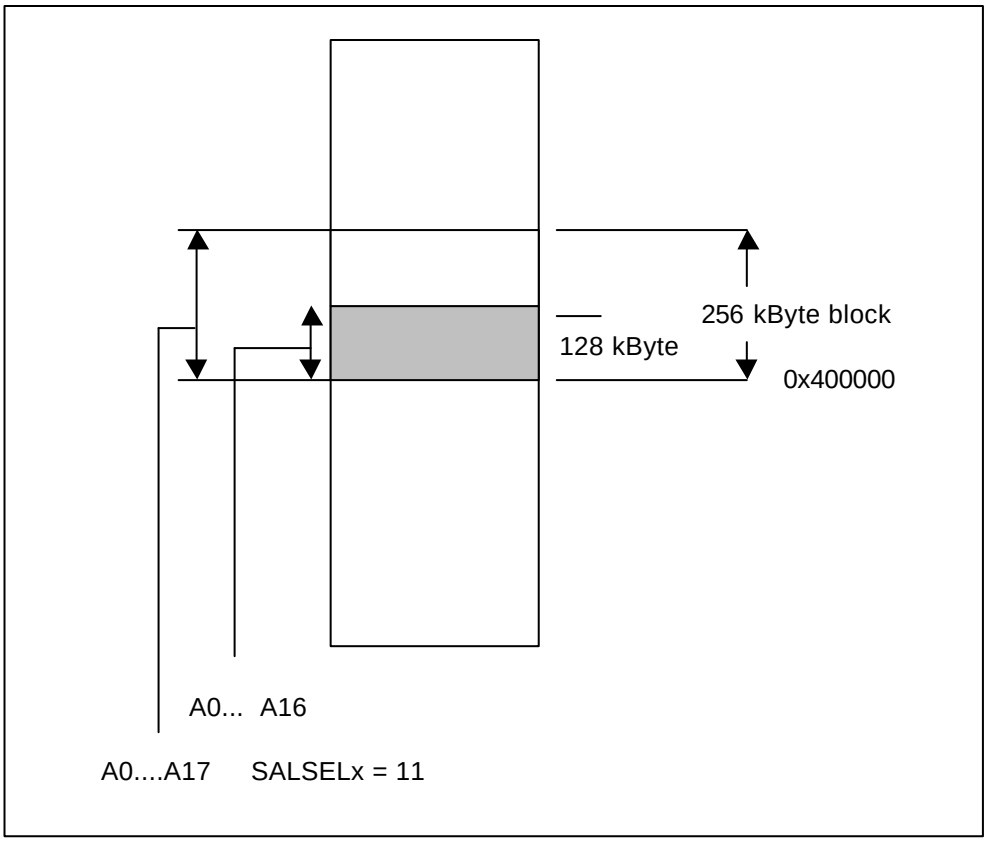
Table 5: Segment Address Lines

S4-5	S4-4	Segment Address Lines	Directly addressable space
OFF	OFF	A17.....A16	256 kByte
OFF	ON	A23.....A16	16 Mbyte, CAN modules disabled !
ON	OFF	None	64 kByte
ON	ON	A19.....A16	1 Mbyte; Default

Example:

Assume that your application requires the 128 kByte SRAM block at the absolute address 0x400000. Furthermore the /CS1 signal should control both, reading from and writing to the SRAM. The controller must at least supervise the address lines A0 to A16 in the direct addressing mode (SALSEL0 = OFF, SALSEL1 = OFF).

Figure 4: Example



Define BUSCON1 Register:

/CSWEN = /CSREN = 1, RDYPOL = 0; RDYEN = 0; BUSACT = 1;
ALECTL = 0; BTYP = 10; MTTC = 1; RWDC = 1; MCTC = 1111;

Define ADDRSEL1 Register

RGSZ (Range-Size-Selection) = 0101 (128 kByte block)
RGSAD (Range-Start-Address) = 0100000

6.8. RANGE START ADDRESS

Depending on the chosen block size (see previous example) the range start address ”RRRRRRRxxxxx” specifies the upper bits (A23....A12) of the respective address area. An ”R” stands for a bit which has to be set within the RGSAD register if this address line is not supervised by the controller bus logic. All bits expressed as an ”x” are exclusively maintained by the bus address signals.

Referring to the example, the register configuration ”RRRRRRRxxxxx” lets A12 to A16 being controlled by the controller address bus. The address lines from A17 to A23 are configured within the RGSAD bits which will be taken by the CPU to compute the internal, absolute 24-bit address.

All memory areas which are not handled by BUSCON(x) and ADRSEL(x) (x = 1...4) are maintained by BUSCON0 and ADRSEL0.

Now specify ADRSEL1 register:

0x400000 = 0100.0000.0000.0000 0000.0000
*

RGSAD = 0100.000*0.0000* **

ADRSEL1 = 0100.0000.0000.0101

* controlled by bus address lines A0..A16
**cursive bits are discarded since they are already controlled by address bus

6.9. EVA269 Memory Layout

The following table shows some address selections which are only suggestions by FS FORTH-SYSTEME. The memory areas may be changed to fulfill different user requirements. Use BUSCONx, ADRSELx registers and the configuration registers as already described in a previous chapter to setup your individual memory layout.

Table 6: Typical Memory Layout

Address Range	Chip-Select	Bus Width	Function
0x4F.FFFF 0x40.0000	/CS4	?	Not connected yet
0x3F.FFFF 0x30.0000	/CS3	16 bit	Flash Extension, up to1 MByte
0x20.0FFF 0x20.0000	/CS2	8 bit	RTC; 4 kByte
0x1F.FFFF 0x10.0000	/CS1	16 bit	SRAM, up to 1 MByte
0x0F.FFFF 0x00.0000	/CS0	16 bit	Boot-Flash, up to 1 MByte

6.10. Clock Generation

PIN CLKCFG0, S4-6
PIN CLKCFG1, S4-7
PIN CLKCFG2, S4-8

Since the ST10F269 includes an internal PLL circuit, the external oscillator device does not need to be driven by a high frequency crystal (Quartz). The external oscillator output either directly feeds the CPU and peripherals or it is connected to the on-chip PLL circuit which then provides the CPU clock.

Table 7: Selection of CPU clock

S4-8	S4-7	S4-6	CPU-Clock Fcpu = fxtal * F	External Clock Input Range in MHz (1)	PLL mode
OFF	OFF	OFF	fxtal * 4	2.5 to 10	Active
OFF	OFF	ON	fxtal * 3	3.33 to 13.33	Active
OFF	ON	OFF	fxtal * 2	5 to 20	Active
OFF	ON	ON	fxtal * 5	2 to 8	Active (2)
ON	OFF	OFF	fxtal * 1	1 to 40	Inactive
ON	OFF	ON	fxtal * 1.5	6.66 to 26.66	Active
ON	ON	OFF	fxtal * 0.5	2 to 80	Active (3)
ON	ON	ON	fxtal * 2.5	4 to 16	Active

(1) This range is not valid for the ST10F269 in TQFP-package, refer to the datasheet.
(2) The EVA269 Evaluation Board comes with an external 8 MHz oscillator circuit. The CLKCFG pins are set by default to a factor of 5 (CPU internal = 40 MHz !). For the ST10F269 in TQFP-package, a 5 MHz oscillator circuit is placed on board, so the internal CPU clock is 25 MHz!
(3) Check the additional information in the datasheet of the ST10F269.

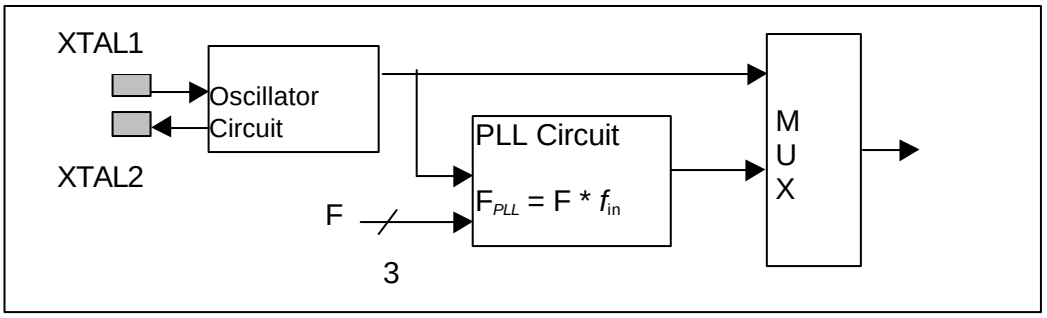


Figure 5: Clock generation

6.11. Single-Chip-Mode

In single chip mode the program is fetched from the on-chip Flash memory. Therefore the embedded Flash memory must contain a valid program to execute correctly. A utility for this can be found on the CD shipped with your EVA269 Evaluation Board.

Single chip mode is entered when pin /EA is set high during reset. Switch S2-3 to ON position before restarting the evaluation board. If the switch is set to OFF, a pull-down resistor configures the CPU to run in external bus mode. Refer to Figure 6.

6.12. User Switches and analog Components

The EVA269 Evaluation Board includes a wire-wrap field, four LEDs and analog components for user specific purposes like self-training and trials. Most of them are directly wired with connector X10 and X1 whereby this components can be accessed by the CPU's port lines. A small serial EEPROM (24LC01) allows to store and restore user data through a standard I2C-Bus interface. To create a simple application the evaluation board comes with a set of junction cords which are provided to interface the CPU-connector X1 with X10.

To access these user components, you must connect your selected component from connector X10 to the CPU connector X1 with external wire, e.g. like the wires shipped with your EVA269 Evaluation Board.

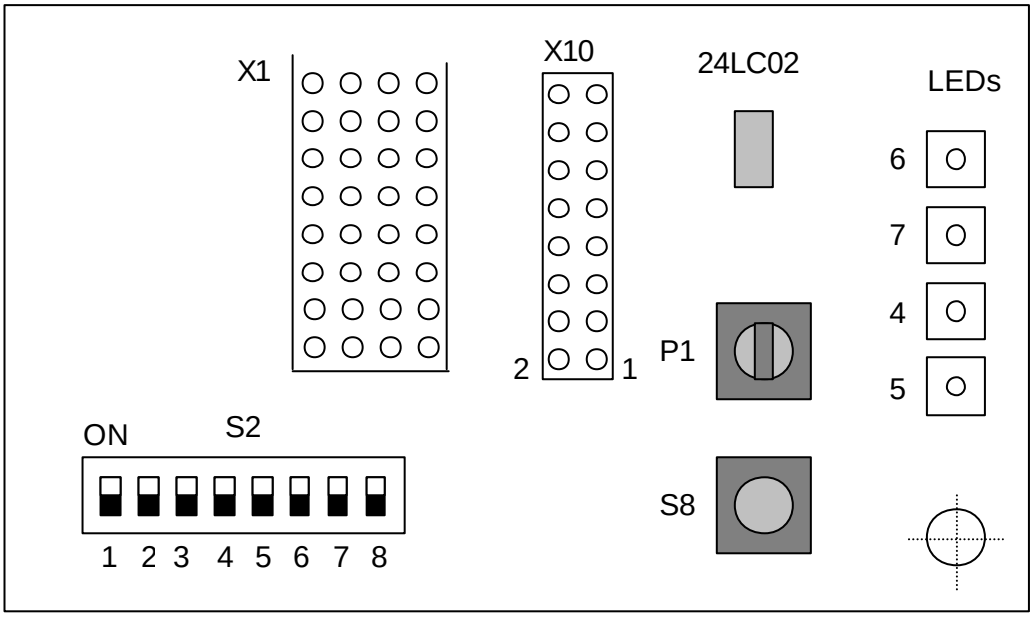


Figure 6: User switch and analog components

Refer also to the attached top view schematics in the appendix.

6.13. Switch S2

PIN	Function description	X1	X10
1	The VPP programming voltage is not used by the ST10F269, so switch S2-1 always **OFF**	16c	---
2	The VPP programming voltage is not used by the ST10F269, so switch S2-2 always **OFF**	11a	---
3	If **ON** the /EA pin of the CPU is set to +5V starting the device in single chip mode. A pull down resistor terminates the /EA pin at the CPU part if switch is OFF !	12d	
4	If **ON** the Watchdog oscillator is disabled	13d	
5	USER-Switch 1: If **ON** pulls down signal at X10.1 otherwise the pin becomes high active (5V)		X10.1
6	USER-Switch 2: If **ON** pulls down signal at X10.2 otherwise the pin becomes high active (5V)		X10.2
7	USER-Switch 3: If **ON** pulls down signal at X10.3 otherwise the pin becomes high active (5V)		X10.3
8	USER-Switch 4: If **ON** pulls down signal at X10.4 otherwise the pin becomes high active (5V)		X10.4

Table 8: User Switch S2

6.14. LED 4, 5, 6, 7

All user LEDs are pulled into a high active level (5V) through a series resistor (1K5). Switching one of these LEDs **ON,** the CPU signal, interfaced to X10, must be pulled down by software.

LED	Pin of X10
5	6
4	5
7	8
6	7

Table 9: LEDs (LE4..LE7)

6.15. LED 1, 2, 3

Please refer to the top view schematic attached in the manual's appendix!

LED	Name	Function
1	POWER	Power On/Off signal
2	VPP	The VPP programming voltage is not used by the ST10F269
3	EINIT	Off after Reset, On after execution of EINIT

Table 10: LEDs (LE1..LE3)

6.16. Push button S8

Gets into high level state when key is being pressed. Signal is interfaced to X10.11 connector.

6.17. Potentiometer P1

This variable resistor allows the adjustment of the reference voltage provided by the CPU (connector X1.30c). The voltage can be adapted to the user specific application via X10.12.

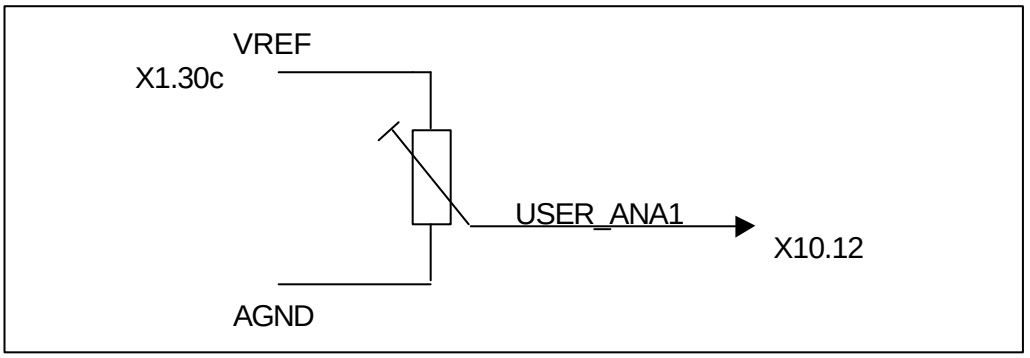


Figure 7: Potentiometer

6.18. CAN Bus-Termination

A CAN bus system has to be terminated at both ends of the network with two resistors. The resistors should have a value of nominal 124 Ω (typically 120 Ω). The terminating resistor can be activated through jumpers J2 (CAN1) and J3 (CAN2).

Caution: Never use more than two termination resistors for a network. This might result into any unexpected bus transmission errors or may force the transceiver circuits to drop down their operation.

6.19. RS232 Interface

The RS232 is interfaced to the board through connector X3 (see Figure 8). It allows the download of the monitor and bootstrap loader as well as the user application file. If the monitor is not in use, this interface can be taken as a standard RS232 for individual purposes. The RS232 hardware does not support any handshake signals, like RTS, CTS or DTR.

6.20. Battery Back-up

A Lithium Battery (3V) is also included on the EVA269 Evaluation Board when shipped. It supplies both low-powered SRAMs, provided they exist on board (not mounted on EVA269). According to the position of jumper J4 the battery can be connected or disconnected (see Table 11).

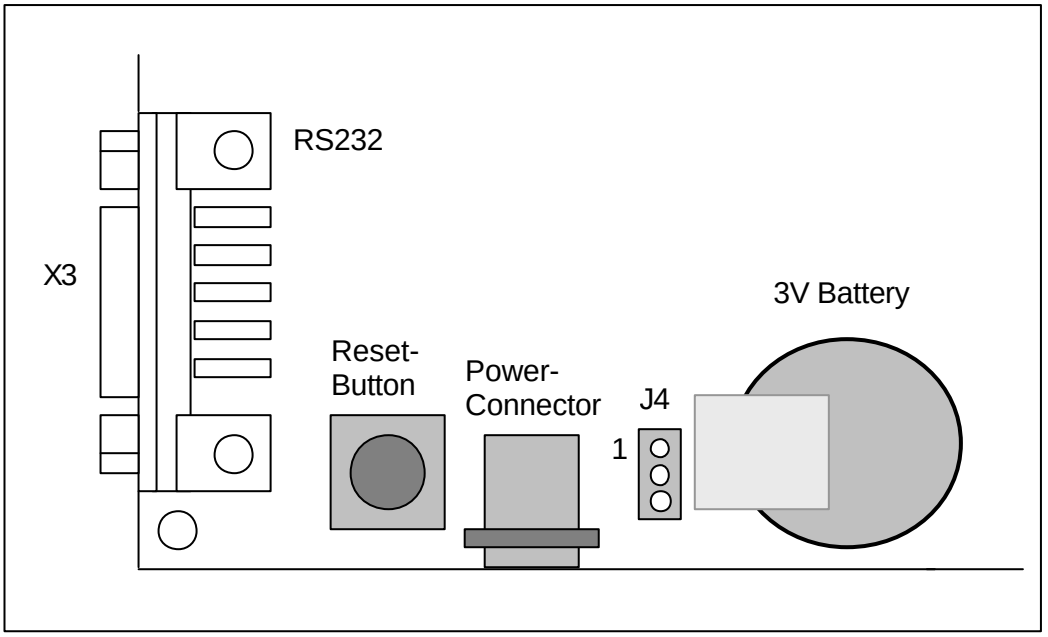


Figure 8: Power Supply

Connected Pins	Function
1-2	Battery connected
2-3	Battery disconnected

Table 11: Jumper J4

7. Connectors

7.1. CPU signal connector X1

Pin	Row A	Row B	Row C	Row D
1	Address A0	Address A1	Data D0	Data D1
2	Address A2	Address A3	Data D2	Data D3
3	Address A4	Address A5	Data D4	Data D5
4	Address A6	Address A7	Data D6	Data D7
5	Address A8	Address A9	Data D8	Data D9
6	Address A10	Address A11	Data D10	Data D11
7	Address A12	Address A13	Data D12	Data D13
8	Address A14	Address A15	Data D14	Data D15
9	Address A16	Address A17	NC	NC
10	Address A18	Address A19	TXD0, serial	RXD0, serial
11	/RESTIN	/INT_RTC	/RSTOUT	/READY
12	/CS0	/CS1	ALE	/EA
13	/CS2	/CS3	/NMI	/RD
14	/CS4	PORT 6.5	CAN_RXD1, P4.5	/WRH, P3.12
15	PORT 6.6	PORT 6.7	CAN_RXD2, P4.4	/WRL,
16	CAN_TXD1, P4.6	CAN_TXD2, P4.7	PORT 2.0	PORT 2.1
17	PORT 8.0	PORT 8.1	PORT 2.2	PORT 2.3
18	PORT 8.2	PORT 8.3	PORT 2.4	PORT 2.5
19	PORT 8.4	PORT 8.5	PORT 2.6	PORT 2.7
20	PORT 8.6	PORT 8.7	PORT 2.8	PORT 2.9
21	PORT 7.0	PORT 7.1	PORT 2.10	PORT 2.11
22	PORT 7.2	PORT 7.3	PORT 2.12	PORT 2.13
23	PORT 7.4	PORT 7.5	PORT 2.14	PORT 2.15
24	PORT 7.6	PORT 7.7	PORT 3.15	PORT 3.13
25	PORT 5.0	PORT 5.1	PORT 3.9	PORT 3.8
26	PORT 5.2	PORT 5.3	PORT 3.7	PORT 3.6
27	PORT 5.4	PORT 5.5	PORT 3.5	PORT 3.4
28	PORT 5.6	PORT 5.7	PORT 3.3	PORT 3.2
29	PORT 5.8	PORT 5.9	PORT 3.1	PORT 3.0
30	PORT 5.10	PORT 5.11	VAREF	VPP
31	PORT 5.12	PORT 5.13	GND	GND
32	PORT 5.14	PORT 5.15	VDD (+5V)	VDD (+5V)

Table 12: Connector X1

7.2. User hardware connector X10

PIN	FUNCTION
1	User Switch 1 (S2-5)
2	User Switch 2 (S2-6)
3	User Switch 3 (S2-7)
4	User Switch 4 (S2-8)
5	User LED 4
6	User LED 5
7	User LED 6
8	User LED 7
9	I2C Bus Clock Line
10	I2C Data Read/Write Line
11	User Pushbutton
12	Analog Voltage from potentiometer
13	NC
14	NC
15	NC
16	NC

Table 13: Connector X10

7.3. X4, CAN Connector 1

PIN	Signal
1	NC
2	CAN_L, differential Low signal
3	GND, optional
4	NC
5	NC
6	GND, optional
7	CAN_H, differential High signal
8	NC
9	VCC (5V), optional

Table 14: Connector X4

7.4. X5, CAN Connector 2

PIN	Signal
1	NC
2	CAN_L, differential Low signal
3	GND, optional
4	NC
5	NC
6	GND, optional
7	CAN_H, differential High signal
8	NC
9	VCC (5V), optional

Table 15: Connector X5

Pins 2 and 7 may be connected via a terminating resistor (120 Ω) via jumpers J2 and J3. Refer to chapter 6.18 ”CAN Bus-Termination”

7.5. X3, RS232 Interface

PIN	Signal
1	NC
2	TXD
3	RXD
4	NC
5	GND
6	DTR0, static 10V, no handshake
7	NC
8	NC
9	NC

Table 16: Connector X3

The housings of all DSUB connectors are grounded.

8. Power-Supply

The EVA269 Evaluation Board expects a stabilized 5 VDC (±0.25V) voltage at the X2 power connector. An external power mains unit is also part of the *STart269* Evaluation Kit.

CAUTION:
DON'T USE ANY OTHER POWER SUPPLY THAN THAT ONE COMING WITH THE START269 EVALUATION KIT. OTHERWISE YOU TOLERATE THE DAMAGE OF THE ENTIRE BORD !

IN THIS CASE WE CANNOT GRANT ANY WARRANTY!

9. User Oscillator

The oscillator unit (8MHz, or 5MHz for ST10F269 in TQFP-package) is already plugged into a DIP8 socket module (refer to the G2 part on top view schematics) when shipped. It is pretty easy to replace this part by your own, individual circuitry. Just remove the oscillator from the DIP socket and replace it by your discrete solution. This could be a small, 3-pin printed circuit board, which keeps all parts, to run the CPU through an external crystal.

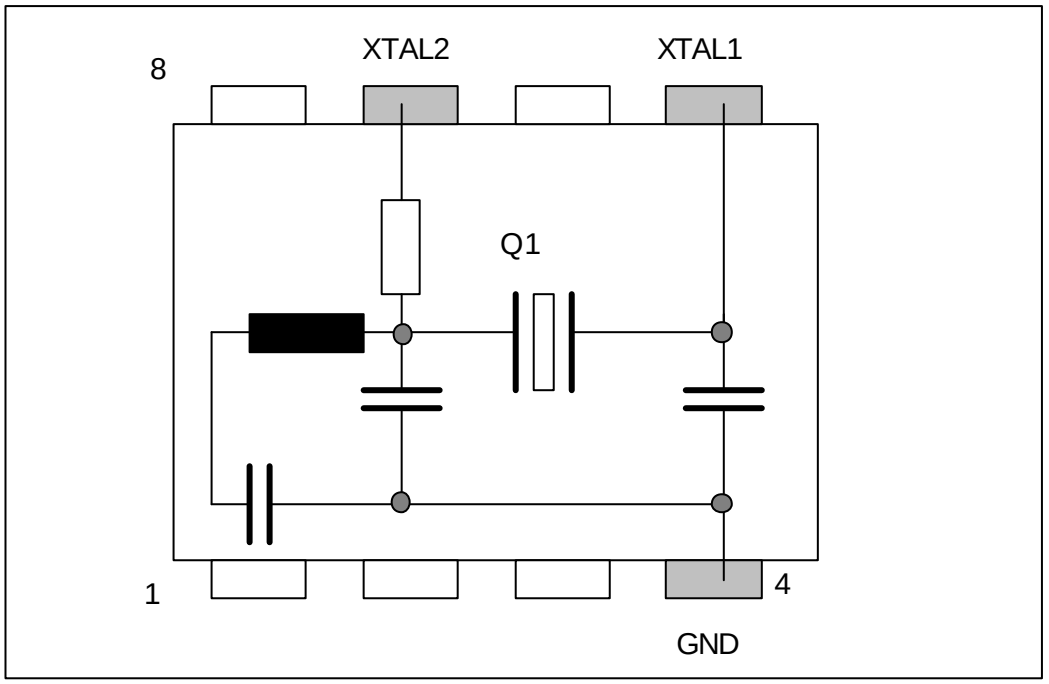


Figure 9: External crystal

10. CD-ROM

Within the STart269 Evaluation Kit you will find everything you need for a fast start-up development. Two C compilers (demo versions) coming from KEIL and TASKING allow you to write small sample programs or to run the examples provided on the CD-ROM. The use of the compiler is reduced in memory size. You will find datasheets of the devices on the CD-ROM as well as a utility to program and erase the embedded Flash memory.

10.1. Content of CD-ROM

On your CD, you can find four main directories. The content is described below. Please refer to the TXT files in the sub-folders for detailed information.

DC166	Tasking demo, board configuration for the monitor and examples.
Keil	Keil demo, monitor, examples.
Manual	Manual of this Evaluation Kit and datasheets for the ST10F269.
Tools	This directory contains tools for programming the external/internal Flash of the ST10F269 and the PLS demo debugger.

10.2. DEMO for KEIL

The example RTC72423 can be built, using a KEIL C compiler (full or demo version). Within the project options you can choose between a Debug-version, a version running from the internal Flash and a version, running from the external Flash. For this the following switches have been used:

_KEIL, _DEBUG:	Shows how the realtime clock works and how to make a project for a debug session.
_KEIL, _INTERN:	Shows how the realtime clock works and how to make a project for the internal Flash-memory of the ST10F269.
_KEIL:	Shows how the realtime clock works and how to make a project for the external Flash-memory.
_ST10_25MHZ:	Use this switch additionally for the ST10F269 in TQFP-package

10.3. DEMO for TASKING

The example RTC72423 can be built, using a TASKING C compiler (full or demo version). Within the workspace you can choose between a Debug-version, a version running from the internal Flash and a version, running from the external Flash.

RTC_deb:	Shows how the realtime clock works and how to make a project for a debug session.
RTC_int:	Shows how the realtime clock works and how to make a project for the internal Flash-memory of the ST10F269.
RTC_rel:	Shows how the realtime clock works and how to make a project for the external Flash-memory.
_ST10_25MHZ:	Use this switch additionally for the ST10F269 in TQFP-package

10.4. TOOLS\FLASH166

In this folder you will find the FLASH166 tool which has already been mentioned in this manual. Furthermore this directory keeps the technical documentation of the Flash tool, where you can find all needed information for debugging and downloading binaries into the Flash circuit. FLASH166 supports a lot of different Flash memory types.

Each sample folder for the release version includes a certain batch execution file (prog.bat) which makes it easy for you to create a binary file. FCONV is used to generate a binary output file, based on a hex file typically generated by all C-compilers. Since FLASH166 expects a binary formatted file you should use FCONV and FLASH166 to program the ST10F269 controller. You must not pay attention to these subjects if you are using any other programming tool.

We are permanently upgrading the FLASH166 tool. So we actually support a large number of different standard FLASH memories. If you are planning to use a Flash type which is not supported by FLASH166 yet, you should not hesitate to contact us.

10.5. Board configuration

For each demonstration program refer to the associated text file, like readme.txt or abstract.txt. Herein you will find all technical details about the board configuration, CPU speed and other parameters. Also have look at the PDF file, provided by STMicroelectronics, which represents the datasheet of the ST10F269.

10.6. Error conditions

Batch file does not run:	Make sure that you have adapted the batch file according to your individual path structure. You may be forced to edit the batch file slightly.
Wrong Flash number:	Please note that FLASH166 expects a Flash device at the address 0x00.0000 (/CS0). If you have changed the chip selection lines this error message occurs. The switch "/CSx" changes the chip select line.
Bootstrap mode timeout.	Make sure that you have connected a serial (RS232) cable to COM1 (COM2) and X3 of your board. Then press the RESET push button again. Check if the BSL switch is into bootstrap mode (BSL=ON). If you are using a serial interface different than COM1 you must revise the "prog.bat" file according to: "flash166 /P <file.bin> /COM2".

10.7. ST10Flasher

This tool allows programming the internal Flash memory of the ST10F269 controller. The tool is provided within the tool directory on CD-ROM. Just unzip the file and run Setup.exe to install the tool.

Before running the ST10Flasher, establish the RS232 connection between your PC and the EVA269 board, switch the board into BSL-mode (S3-5 = ON), switch on VPP-MAN (S2-2 = ON) and switch on the power supply.

Now you can run the ST10Flasher and select the Hex-file to download into the internal Flash:

"\Start269\Keil\C166\Examples\Boards\FORTH_EVA269\RTC72423\Intern\obj\RTC72423.h86" or

"\Start269\Keil\C166\Examples\Boards\FORTH_EVA269\RTC72423\Intern\obj\25MHZ\RTC72423.h86".

After the programming was successful, switch S2-2 and S3-5 OFF and switch S2-3 ON. Then reset the controller by the Reset push button S1.

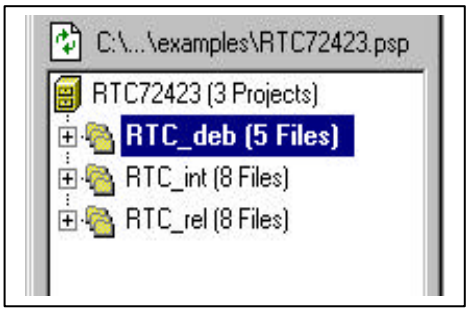
11. Building the sample project with the TASKING C-Compiler

First of all you have to install the Tasking C166 Compiler:
Copy the file dc166-70.zip to a free directory on your harddisk and unzip this file. Then run the setup.exe to install the Tasking Compiler. We recommend to use the default directories and to copy the files from the \TOOLS\FLASH166-directory of the CD-ROM into a directory on your Hard disk, which is known by your system.

Now copy the files from the structure \DC166\... from the CD-ROM to the corresponding directories on your harddisk, remove the “Read-only”-attribute from all of these files.

Then run the Tasking Compiler and select from the menu:
Project -> Project Space -> Open
From the next menu select the file “RTC72423.PSP” from the directory \DC166\EXAMPLES\.

Now you will see the project space from the RTC72423-sample project:



Go to the menu EDE and select the item “Scan all dependencies”

If you have the license for the full version of the EDE, you can remove the macro _TASKING_DEMO from the preprocessing project options of the C-compiler options.

If you have the EVA269-board with the ST10F269 in TQFP-package, you have to add the macro _ST10_25MHZ at the C-compiler options.

11.1. Project for the debugger

To build the project for the debugger, choose the project "RTC_deb" as current project and "Rebuild" the project. There will be 4 warnings if you are working with the demo-version. But this doesn't matter. The results can be found in the RTC72423-directory.

Connect the delivered cable for the RS232-interface between your PC and the connector X3 on the EVA269-board. Switch the BSL-switch S3-5 to ON and switch on the power supply of the board.

Now you can start the CrossView debugger by choosing the item Debug at the menu Project. After the correct download of the project you will see the environment of the debugger. Here you can start the application by choosing the item Run from the menu Run.

Now you will see the time of the real time clock in the virtual I/O-window.

11.2. Project for the external Flash

To build the project, running from the external Flash, select the project "RTC_rel" as current project and "Rebuild" the project. There will be 4 warnings if you are working with the demo-version. But this doesn't matter. The results can be found in the RTC72423-directory.

Connect the delivered cable for the RS232-interface between your PC and the connector X3 on the EVA269-board. Switch the BSL-switch S3-5 to ON and switch on the power supply of the board.

Choose the item Execute from the menu Project to start the batch-file "PROG.BAT", which converts the hex-file of the project into a binary file, which will be downloaded into the external Flash, by the FLASH166 program.

After the correct download, you have to switch OFF the BSL-switch S3-5, press the Reset push button S1 and start a terminal program on your PC with the line parameters 9600Baud, no parity, 8 bit data and 1 stop bit.

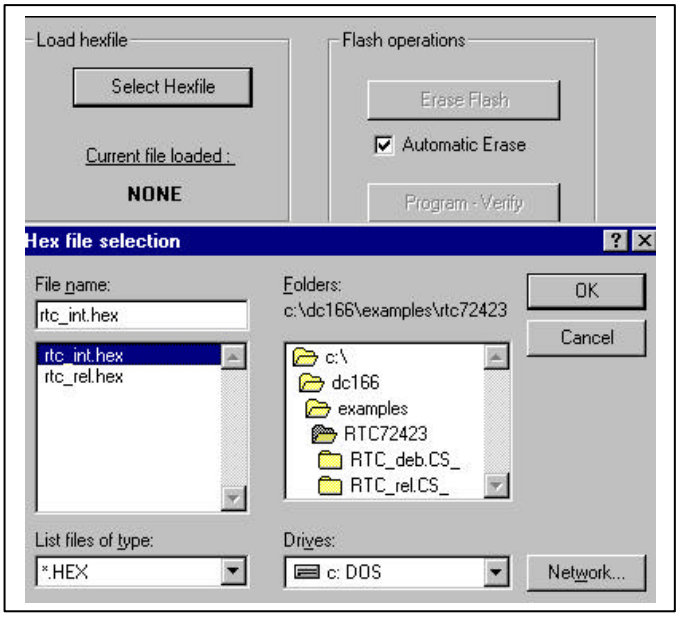
At the terminal program you will receive the time from the real time clock.

11.3. Project for the internal Flash

To build the project, running from the internal Flash, select the project "RTC_int" as current project and "Rebuild" the project. There will be 4 warnings if you are working with the demo-version. But this doesn't matter.

Connect the delivered cable for the RS232-interface between your PC and the connector X3 on the EVA269-board. Switch the BSL-switch S3-5 to ON and switch on the power supply of the board.

Run the software ST10Flasher from STMicroelectronics and load the hex-file "RTC_INT.HEX" from the RTC72423-directory:



Now select the Flash operation Program – Verify, to download the project into the internal Flash. After a successful download, you have to switch OFF the BSL-switch S3-5, switch ON the single chip-switch S2-3, press the Reset push button S1 and start a terminal program on your PC with the line parameters 9600Baud, no parity, 8 bit data and 1 stop bit.

12. Building the sample project with the KEIL C-Compiler

First of all you have to install the Keil C166 Compiler:
Copy the file ek166v410a.exe to a free directory on your harddisk and run it to install the Keil Compiler. We recommend to use the default directories and to copy the files from the \TOOLS\FLASH166-directory of the CD-ROM into a directory on your Hard disk, which is known by your system.

Now copy the files from the structure \Keil\... from the CD-ROM to the corresponding directories on your harddisk, remove the "Read-only"-attribute from all of these files.

Before starting to work with the debugger, you have to choose the correct monitor for the debug session:

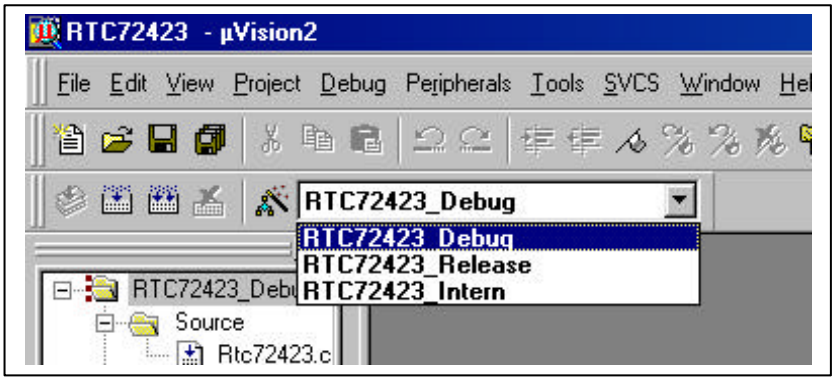
Run the Keil Compiler and open the project "MONITOR.UV2" from the directory:
"\Keil\C166\MONITOR\FORTH_EVA269\"

Choose the target "Bootstrap" and rebuild the project. Now you have built the Monitor for the EVA269-board and you can leave this project

Open the project "RTC72423.UV2" from the directory:
"\Keil\C166\EXAMPLES\BOARDS\FORTH_EVA269\RTC72423\"

Select RTC72423_Debug as target.

If you have the EVA269-board with the ST10F269 in TQFP-package, you have to add the switch _ST10_25MHZ at the target options.



12.1. Project for the debugger

To build the project for the debugger, select “RTC72423_Debug” as target and “Rebuild” the project. The results can be found in the directories:

\RTC72423\Debug\List and \RTC72423\Debug\Obj

Connect the delivered cable for the RS232-interface between your PC and the connector X3 on the EVA269-board. Switch the BSL-switch S3-5 to ON and switch on the power supply of the board.

To start the debugger, Select Debug -> Start/Stop Debug Session. After the correct download of the project you will see the environment of the debugger. Here you can start the application by choosing the item Go from the menu Debug.

After you have started the application, activate the "Serial #1"-window, to see the output from the real time clock.

12.2. Project for the external Flash

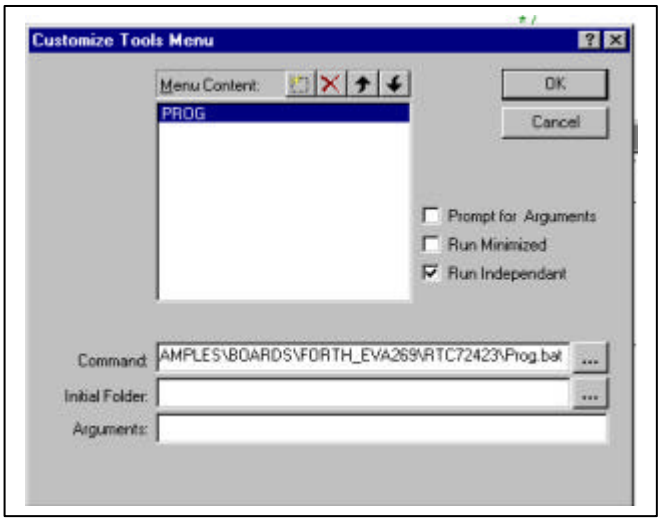
To build the project, running from the external Flash, select “RTC72423_Release” as target and “Rebuild” the project. The results can be found in the directories:

\RTC72423\Release\List and \RTC72423\Release\Obj

Connect the delivered cable for the RS232-interface between your PC and the connector X3 on the EVA269-board. Switch the BSL-switch S3-5 to ON and switch on the power supply of the board.

To make the project ready to download you can either open a DOS-box to start the batch-file “PROG.BAT” in your project-directory, or you can enter the call of the batch-file “PROG.BAT” at “Customize Tools Menu” from the menu Tools:

Don’t forget to select the option “Run independant” !



The batch-file “PROG.BAT” converts the hex-file of the project into a binary file, which will be downloaded into the external Flash, by the FLASH166 program.

After the correct download, you have to switch OFF the BSL-switch S3-5, press the Reset push button S1 and start a terminal program on your PC with the line parameters 9600Baud, no parity, 8 bit data and 1 stop bit.

At the terminal program you will receive the time from the real time clock.

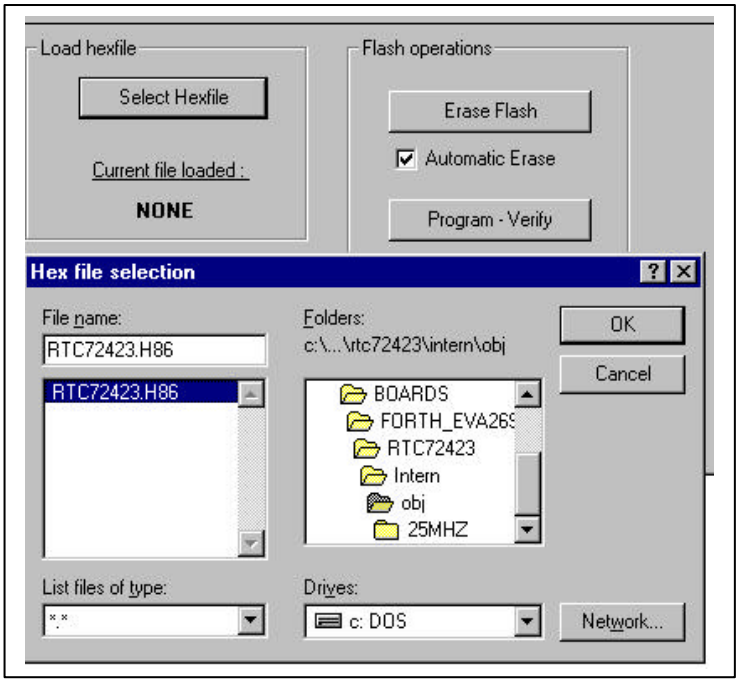
12.3. Project for the internal Flash

To build the project, running from the internal Flash, select “RTC72423_Intern” as target and “Rebuild” the project. The results can be found in the directories:

\RTC72423\Intern\List and \RTC72423\Intern\Obj

Connect the delivered cable for the RS232-interface between your PC and the connector X3 on the EVA269-board. Switch the BSL-switch S3-5 to ON and switch on the power supply of the board.

Run the software ST10Flasher from STMicroelectronics and load the hex-file "RTC72423.H86" from the Obj-directory:



Now select the Flash operation Program – Verify, to download the project into the internal Flash. After a successful download, you have to switch OFF the BSL-switch S3-5, switch ON the single chip-switch S2-3, press the Reset push button S1 and start a terminal program on your PC with the line parameters 9600Baud, no parity, 8 bit data and 1 stop bit.

13. Know-how for handling ST10F269 IDEs

There is no general way how to handle IDEs to get fast and good results but we can give some useful hints to avoid mistakes.

There are a few differences in handling projects for debugging the application and for making a release version for the ROM (Flash).

For a debug session, you need a RAM or ROM monitor. This is a small program which initializes registers and provides functions in order to make your hardware able to communicate with a debugger, usually via serial line. The most important point is, that the registers for RAM memory are set to a correct value by the monitor, because the remote debugger loads your application into RAM and has to modify it for setting breakpoints. The configuration for the Keil-Monitor can be found at the project MONITOR.UV2 and the configuration for the Tasking-Monitor can be found, when opening the file \DC166\ETC\EVA269.CFG

A RAM monitor is loaded via bootstrap loader, the ROM monitor has to be programmed into the ROM (Flash) and is automatically activated after RESET. This Evaluation Kit works mainly with RAM monitors because the download time is so short that a direct start from ROM yields no advantages.

Please make sure that your application doesn't overwrite the monitor in the RAM!

For a release session, you have to link a "start-up" to your application, which is automatically started after RESET. The start-up is usually an assembler program, linked to your application (Keil example **RTC72423_Release**: Start269.a66), or is automatically generated via the project settings (Tasking example **RTC_rel**). The memory settings for linker/locator has to match with the start-up settings! I.e., if you set the address selection register for RAM to 100 000h and locate your data to 10 000h your program won't run.

Generally, the start-up initializes your hardware. Please check the settings for memory access times for RAM and ROM (Flash) to avoid your hardware running slower than it could.

If you have any questions, please don't hesitate to contact the Technical Support of FS FORTH-SYSTEME!

14. Appendix

14.1. Top View of the EVA269 Evaluation Board