ST10R163 KIT
USER'S GUIDE

Version 1.20

November 1997

Table Of Contents

1. OVERVIEW

The R-FLIC, is an industrial board designed to accept multiple processors in the SGS Thomson 166 family. The board has two footprints for the processors, a 144 pin PQFP, and a 100 pin TQFP (inside the 144 pin footprint). The board works with the following processors: ST10R(F)163, ST10R165, and the ST10F167 which has both CAN and 128K FLASH capabilities. There are separate user's guides for each of the different processors used on the R-FLIC This user's manual is for the R-FLIC board populated with the ST10R(F)163 processor.

1.1 Basic Features

The R-FLIC board is a 6 layer industrial board with separate VCC and GND planes. The microcontroller is run with a 16-bit nonmultiplexed data bus and a nonmultiplexed address bus. The various configuration options are set by jumpers which determine the reset options of the 163. Most notably, the width of the address bus, and the number of chip select lines are determined by these jumpers. The board has a 12 Volt regulator to program the FLASH versions of processor (ST10F163) built in.

There are two banks of memory, one for RAM and one for FLASH EEPROM. Each bank may hold up to 1 megabyte of memory for a total of up to 2 megabytes of memory. Memory banks are controlled by the chip select signals CS0# and CS1# of the 163. Jumpers determine which bank is controlled by which chip select line. The default configuration is the 256K RAM and no ROM mode. The evaluation board with the ST10R163 processor uses the external EEPROMs to store its monitor program.

The R-FLIC has the same standard operating modes and hardware features as the RMB167-CRI. The R-FLIC is designed to run each processor at its fastest speed with no wait states. A set of option headers, make the R-FLIC a flexible hardware platform. The R-FLIC is compatible with all of our other 16-bit boards. The board size, the location and function of all headers are kept the same.

1. **Industrial strength shielding:** The PCB is an six-layered board with separate VCC and GND planes designed to operate in noisy industrial environments.
2. **FLASH Memory Capability:** The F163 has 128K internal FLASH which may be programmed on-board. The board can also accept up to 1M of external FLASH memory.
3. **External bus system on a header.** The R-FLIC has a 2 x 16 pin header for selection of the operating modes, which are determined during reset by the state of Port 0 bits.
4. **Chip Select lines used for memory mapping.** Uses CS0# and CS1# to interface to external memory
5. **The HiTEX / HiTOOLS ICE connector** designed into the board for those needing emulation capabilities.
6. **On Board 5 Volt and 12 Volt Power Regulation.**

1.2 Hardware Features

- ST10R163 processor
  Two serial channels, one synchronous/asynchronous and one high-speed-Synchronous Serial Port (SSP)
  Up to 25 MHz internal system clock
  8 Peripheral Event Controller (PEC) channels
  Up to 16 MB linear address space
  Five programmable CS lines
  77 bits of general-purpose inputs/outputs
- Serial port uses a RS232 driver, terminates at DB-9 connector and a 3-post connector.
- SSP is terminated at a screw-type terminal block to facilitate connections to external devices.
- Chip Select lines are used for memory mapping.
- Accommodates 64KB - 1MB of SRAM (256K installed)
- Accommodates 256KB - 1MB of external EEPROM FLASH memory (256K installed with the monitor program).
- Support circuitry and software to program the internal flash of the ST10F163 processor.
- Push buttons for RESET# and NMI# (non-maskable interrupt).
- The HiTEX / HiTOOLS ICE connector designed into the board.
- Machine screw sockets under all other IC's.

- Power on LED.
- Flexible and embeddable 6 layer industrial board.
- Board size 4"x6 1/2".
- Mounting holes in corners.

1.3 READS166 Evaluation Software Overview

READS166, version 3.00, is Rigel Corporation's Integrated Development Environment for the SGS Thomson 16-bit processors. READS166 includes an editor, a host-to-board communications system, an assembler, and a C compiler. READS166 is completely rewritten in native 32-bit code to run on Windows95 and WindowsNT. READS166 includes a sophisticated project management system to simplify code reusability and version control. The C compiler is rewritten to support a full debugger. The debugger allows you to step through your code with breakpoints and variable watches as the compiled code runs on the target board, similar to the operation of an in-circuit emulator.

RMON163 - The READS166 monitor program

RMON163 is placed into ROM and supports basic memory and port functions. RMON163 allows downloading and running applications programs. The complete source code for user modifications or upgrades is included on disk.

Ra66 - The READS166 Assembler

Ra66 is an assembler for the C166 family of controllers. It is a multi-pass absolute assembler which generates HEX code directly from assembly source code. The assembler in the demo version of READS166 limits the size of code to about 2K.

Rc66 - The READS166 C Compiler

Rc66 is a C Compiler for the C166 family of processors. It compiles code for the tiny memory model which fully resides in the first segment of memory. Rc66 is a designed as a low-cost C compiler which provides a quick development cycle for simpler applications which do not need more than 64K of code, or the use of standard C libraries. Rc66 implements a subset of ANSI C. Rc66 works in conjunction with Ra66: first an assembly language program is generated from the C source, then a HEX file is generated. Currently, structures, unions, enumerated types, and the typedef directive are not implemented. The C-compiler in the demo version of READS166 limits the size of code to about 2K.

1.4 Parts List

Your R-FLIC package includes the following:

Hardware

1. R-FLIC board populated with the ST10R163, 256K of static RAM, and 256K FLASH ROM.
2. Serial modem cable with adapter

Software

1. RMON163 monitor program with source code loaded in FLASH ROM.
2. Evaluation version of READS166 for Microsoft Windows.

Documentation

1. User's Guide with circuit diagrams
2. Sample programs.

2.0 QUICK START

2.1 System Requirements

READS166 is designed to work with an IBM PC or compatible, 486 or better, running Windows 95 or Windows NT. READS166 supports the bootstrap loader feature and downloads a minimal monitor during bootstrapping. Since the ST10x163 processors do not have the bootstrap feature, the monitor program is placed in ROM. Segment 0 of the memory [0..FFFF] is equally partitioned into ROM [0..7FFF] and RAM [8000..FFFF]. With the monitor program in RAM, there is no need to bootstrap the board.

2.2 Software Installation

Place the first READS166 disk in your floppy disk drive.

To install from Windows 95 /NT select **START | Settings | Control Panel**. Choose the **Add/Remove Programs**, and follow the standard install directions answering the questions with the appropriate answers.

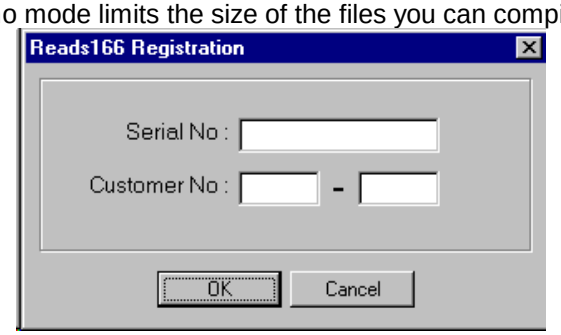
2.3 Start up

1. Connect your RIGEL board to the PC host via a serial cable.
2. Connect the board to a well-regulated 5 volt power supply.
3. Check to make sure the correct jumpers are in place.
4. Run the READS166 host driver by selecting **Start | Programs | READS166**. You may also start READS166 by double clicking on the READS166 short cut icon if installed.

2.4 Serial Number

2.4.1 Demo Version

When you run the READS software a pop-up registration box will open asking for your serial and customer numbers. If you are using the demo version of the software you may press **CANCEL** and the box will disappear and the About Box will appear. Press the **OK** button in the About Box and the software will run in the Demo mode. The demo mode limits the size of the files you can compile to about 4K.

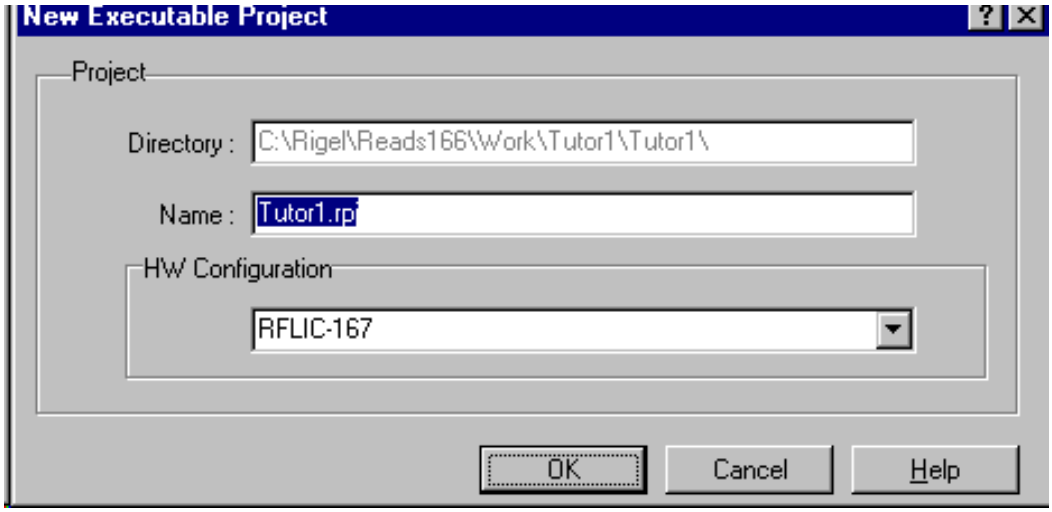


2.4.2 Full Version

When you run the READS software for the first time a pop-up registration box will open asking for your serial and customer numbers. Both numbers will be on a registration sheet and included with the printed documentation. Insert these numbers in the correct boxes, press **OK** and then press **OK** in the About Box. Your software is registered and ready to use.

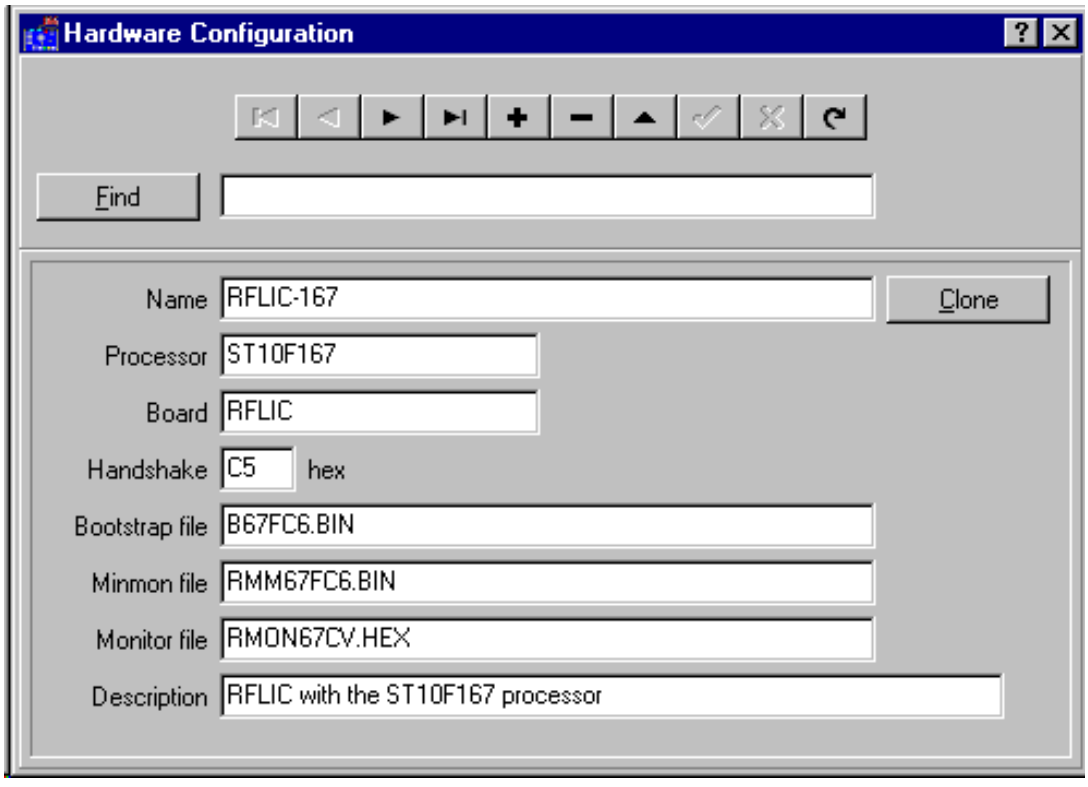
2.5 Configuring READS166 and Initiating Host-to-Board Communications

1. Press the **Projects | Options | Project Properties | HW Configuration** to select the board you are using from the list.



Note that since the bootstrap loader is not used, the existing hardware configuration, RFLIC-167 is fully compatible with the R163 processor.

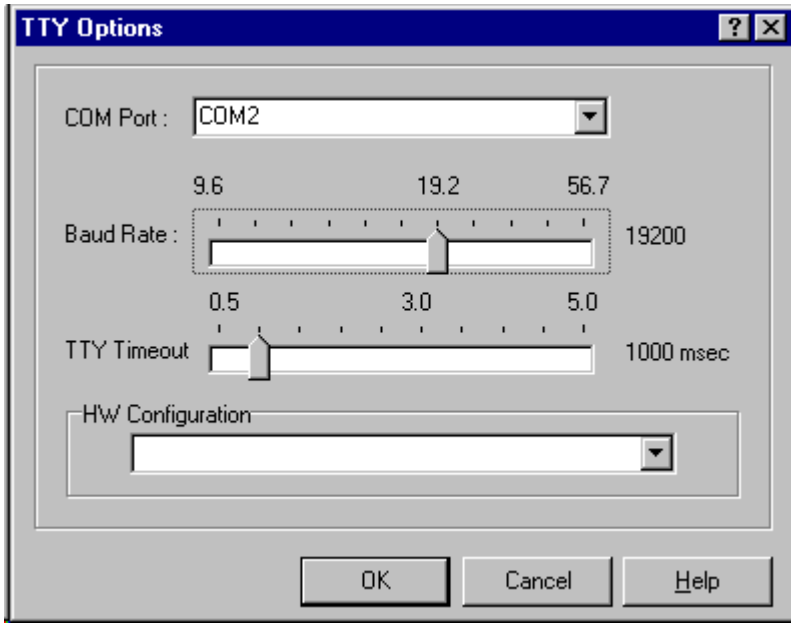
2. Or using the **Tools | Hardware Configuration** menu command. Choose the name of the board you are using from the board list which appears.



Again, note that since the bootstrap loader is not used, the R-FLIC board populated with the ST10x163 processor is fully compatible with the ST10F167 processor. Several fields of the hardware configuration, such as the handshake byte are ignored by Reads166 when the monitor is already loaded into ROM.

3. Open the TTY window using the **Tools | TTY** menu command. Select the communication port parameters using the **Tools | TTY | TTY Options** menu command. You will need to select the

COM port you are using. Set the baud rate to 19.2K.



2.6 Starting The Board

Apply power to the board is sufficient to initialize the board. You may press the RESET button to reset the processor and the monitor program at any time.

2.7 Verifying that the Monitor is Running

Make sure the TTY window is active, clicking the mouse inside the TTY window to activate it if necessary. Then type the letter '***H***' (case insensitive) to verify that the monitor program is responding. The 'H' command displays the available single-letter commands the monitor will recognize.

The READS monitors use single-letter commands to execute basic functions. Port configurations and data, as well as memory inspection and modifications may be accomplished by the monitor. Most of the single-letter commands are followed by 4 hexadecimal digit addresses or 2 hexadecimal digit data bytes. The following is a list of the commands.

R E A D S C O M M A N D S

C nn	read port nn Configuration (DPnn)
C nn=mmmm	set port nn Configuration (DPnn=mmmm)
D	Download HEX file
G XXXX	Go, execute code at XXXX
H	Help, display this list
M XXXX	Memory, contents of XXXX
M XXXX=nn	Memory, change contents of XXXX to nn
M XXXX-YYYY=nn	Memory, change block XXXX-YYYY to nn
P nn	read Port nn (Pnn)
P nn=mmmm	write to Port nn (Pnn=mmmm)
W XXXX	Word memory, contents of XXXX
W XXXX=mmmm	Word memory, change contents of XXXX to mmmm
W XXXX-YYYY=mmmm	Word memory, change block XXXX-YYYY to mmmm

2.6 Using Help

You can get more information about READS166 from the help system. To access the help system select **Help | Contents** from the main menu. Once in the Help System, select the topic you are interested in for more information.

3. OPERATING NOTES

The R-FLIC needs two connections: to a power supply and to the serial port of a host via a modem cable.

3.1 Power

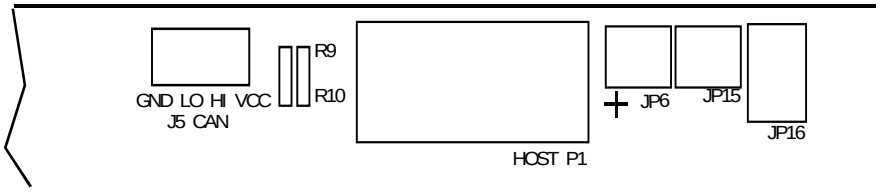
Power can be applied to the board by three different methods. There is a power jack JP16, and 2 two-position screw-type terminal blocks, JP6 or JP15.

3.1.1 JP16, Power Jack

JP16 is the default power connector for use with the standard wall transformer. A 13.5 VAC power supply should be used with the power jack. The board has a built-in regulator which will provide the circuitry with the +5 volts needed for standard operation. The 13.5 VAC transformer also is used to provide the programming voltage for the on-chip FLASH of the F163.

3.1.2 JP15, 9VAC Terminal Block

The terminal block JP15, is in parallel with the power jack, providing an alternative connection for the 13.5 VAC power supply. Use JP15 if your power supply does not have the correct connector on it.



3.1 Power Supply and Serial Port

3.1.3 JP6, 5 V Terminal Block

JP6 may be used for power if a well-regulated +5V DC (+/- 5%) power source is available. Using JP6 for power, bypasses the on-board power regulator. The (+) terminal of JP6 is marked on the board.

Note

When using JP6 for power, a +5 volt regulated power supply must be used. A lower voltage will not operate the board. A higher voltage will irreversibly damage the active components on the board. An unregulated power supply may cause unpredictable failure conditions. Always check that the power supply is plugged into the board correctly. A diode is placed across the input in reverse. Thus if the power is applied to the R-FLIC board in reverse polarity, the diode will short the power supply attempting to prevent damage to the board.

3.2 Serial Port 0

Serial port 0 of the microcontroller is connected to the DB-9 port P1 through an RS-232 level converter. The microcontroller supports the transmit and receive signals. A minimal serial port may be constructed with just 3 lines: transmit, receive, and ground, disregarding all hardware handshake signals. Port P1 (HOST) of the R-FLIC is a DB-9 female connector used to connect the board to an IBM compatible PC. A straight-through modem cable may be used. That is a cable connecting pin 2 of the R-FLIC to pin 2 of the host, and similarly pin 3 to pin 3, and pin 5 to pin 5. This cable and a DB9-DB25 adapter is supplied when the board is purchased directly from Rigel Corporation. Note that J3 is a 3-pin header, which carries the same signals as P1. This header is convenient for embedded applications where a modem cable is not needed.

3.3 The High-Speed Synchronous Serial Port (SSP)

The SSP is implemented as an X-Peripheral in the address range [EF00..EFFF]. It uses 4 bits of Port 4, namely P4.4 to P4.7. Two of the four SSP bits are the chip select bits, SSPCE0 and SSPCE1. The remaining two are the data and clock signals SSPDAT and SSPCLK, respectively. The four SSP bits are terminated at a screw-type terminal block, J8. The terminal block J8 also presents access to the power lines VCC and GND. Note that port 4 bits P.20 to P4.23 are also used for the most significant 4 bits of the address, that is A20 to A23. A row of four jumpers, JP17 to JP20, is provided to connect the port bits P4.20 to P4.23 to the terminal block when the SSP is used. The jumpers JP17 to JP20 are placed adjacent to the SSP terminal block J8.

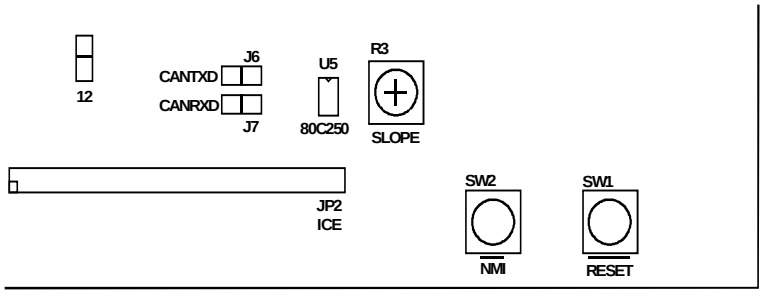
Care must be taken in software and in setting up the reset options to limit the memory used to 1 M Byte, so as not to interfere with the SSP operation. If the SSP is not used, simply remove the 4 shunt jumpers, JP17 to JP20. With these jumpers removed, the ST10x163 may address the entire 16 M byte space.

The width of the address bus is set by the reset options jumpers at reset. Place a jumper in location P0.12 to limit the width of the address bus.

3.4 Push Buttons

3.4.1 Reset (SW1)

The reset button is connected to the reset pin of the processor and resets the board. You may reset the board at any time by simply pressing and release the reset button. Upon reset, the board will run the monitor program in ROM and initialize the board. The initialization is terminated with an EINIT instruction which also places the RO# line to a logic high. This, in turn is used to turn on the green LED (D4), marked RSTOUT. Please allow a second or two for the board to initialize.



3.2 Push Buttons

3.4.2 NMI (SW2)

The NMI button (non-maskable interrupt) is connected to the NMI pin of the processor. When pressed, it generates a non-maskable interrupt. RMON163 places a jump instruction at the NMI vector (address 8). Pressing the NMI, while the RMON is present, invokes the monitor program. This scheme works as long as the interrupt vector table placed in RAM is not altered. Pressing the NMI button is usually sufficient to interrupt a user program which is downloaded and run under RMON. Application programs placed in ROM may use a similar scheme to initialize the system when the NMI button is pressed.

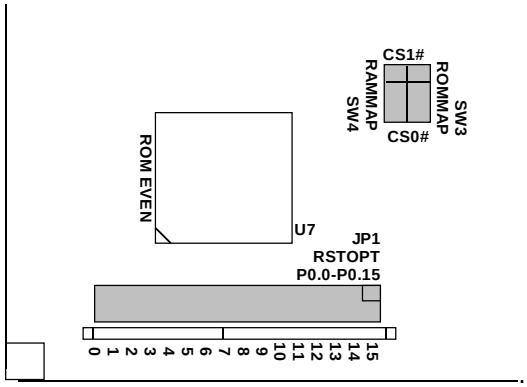
3.5 LEDs

The R-FLIC has three LEDs. The power LED, D2 is lit whenever power is applied to the board. The programming LED, D3 lights up when the programming voltage for the internal FLASH is activated. The RSTOUT LED D4 lights up when the chip is initialized, that is, when RO# becomes high as a result of an EINIT instruction.

4. JUMPER CONFIGURATIONS

4.1 JP1 (Reset Options)

JP1 is a 16 x 2 header. Each pair of posts corresponds to a bit of Port 0. The silk-screen is labeled from 0 to 15 (Port0.0 to Port0.15) indicating bit or jumper positions. Some of the 163 operating modes are determined during reset by the state of Port 0 bits. (Refer to the SGS data book for further information.) Inserting a jumper in JP1 connects the corresponding bit of Port 0 to ground via a 6.8 to 8.2K resistor. This, in turn, sets the operating mode. The R-FLIC uses five of these bits for configuring the board at 6,8,9,10, and 11. It is important not to populate the remaining jumpers, since these are either used by the system (for example in testing modes or emulation modes) or are reserved.



4.1 Options Header

4.1.1 P0.7 and P0.6

Bits 7 and 6 of Port 0 determine the bus type of the ST10R163. In the default configuration P0.7 is not grounded and P0.6 is grounded, resulting in a startup bus type BUSTYP=10. This is the 16-bit nonmultiplexed bus mode. The R-FLIC's default configuration is with jumper P0.6, installed

4.1.2 P0.8

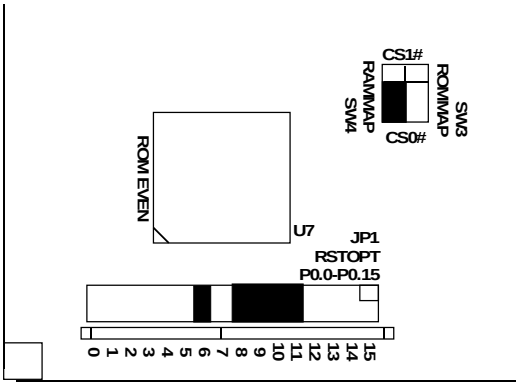
Bit 8 of Port 0 determines the configuration of the WR# and BHE# signals. When P0.8 is grounded, WR# acts as WRL# and BHE# as WRH#. The latter, WRL# and WRH# are then used to select the low and high halves of the memory banks. P0.8 is grounded in the default configuration, by inserting a jumper in P0.8.

4.1.3 P0.9 and P0.10

Bits 9 and 10 are copied to the CSSEL field of the RP0H register. They determine the number of chip select lines the R163 will generate. The default configuration is three chip select lines, which is obtained by grounding both P0.9 and P0.10, with jumpers inserted. The two chip select lines are used by the RAM and ROM memory banks. The third chip select line is used in experiments.

4.1.4 P0.11 and P0.12

Bits 11 and 12 are copied to the SALSEL field of the RP0H register. They determine the width of the address bus the R163 will use. The default configuration is the 24-bit wide address bus. This is achieved by grounding P0.11 but not P0.12. Place a jumper in P0.12 if the SSP is to be used.



4.2 Default Option Jumpers

4.2 SW3 and SW4 (Chip Select Assignment)

SW3 and SW4 are the RAM and ROM jumpers located next to U7. These jumpers determine which R163 chip select signal controls the RAM, and which controls the ROM. Care should be taken not to map both memory banks to the same chip select signal. Use one chip select line for RAM and the other for ROM.

4.2.1 SW3

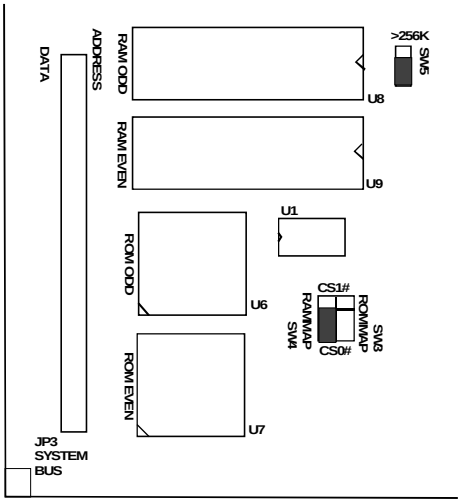
Note that CS0# is active upon reset. Since ROM holds the monitor program, ROMMAP must be in the CS0# position.

4.2.2 SW4

This jumper determines which 163 chip select signal controls the RAM. Place RAMMAP to the CS1# position. The monitor program activates CS1# in the range of [8000..FFFF].

4.3 SW5 (RAM Size)

SW5 selects the size of the RAM chips. This jumper should be placed in the lower position for 32K and 128K devices, for a total RAM of 64K or 256K respectively. For 512K devices, yielding a total memory of 1M Byte, the jumper needs to be populated in the upper position marked ">256K".



4.3 Default Memory Jumpers

4.4 CANTXD, CANRXD Jumpers

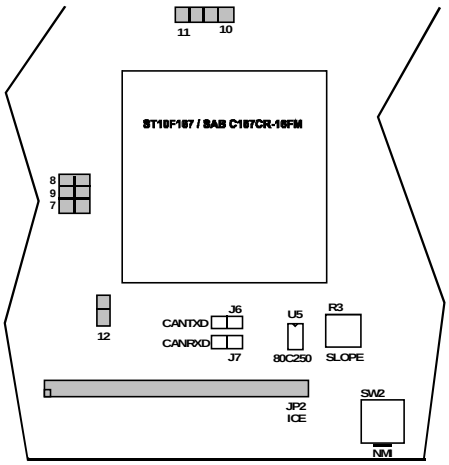
The jumpers J6 and J7 are not used with the 163 processor.

4.5 Analog-to-Digital Converter Reference Jumpers

The 163 processor does not have an analog-to-digital converter, thus these jumpers need not be populated.

4.6 The ICE Connector and the ICE Options Jumpers

The header JP2 accommodates the HiTEX / HiTOOLS In-Circuit Emulator (ICE). The remaining jumpers, (JP)8, (JP)9, (JP)10, (JP)11, (JP)12 are used in conjunction with this ICE system. The JP designation is not marked on the board but is used in the schematics. Please call Rigel Corporation or HiTOOLS for detailed instructions on how to use these jumpers and the ICE connector.



5. MEMORY BLOCK OPTIONS

There are two blocks of memory available on the R-FLIC board when populated with the ST10R163. There are three blocks of memory available on the R-FLIC board when populated with the ST10F163. There is a RAM block which may hold up to 1MB of memory, a ROM block which may hold up to 1MB of (external) FLASH memory, and there is the on-chip 128K (internal) FLASH block of memory on the F163 processor.

In its default configuration, memory region [0..7FFF] is allocated to the monitor program, and the region [8000..FFFF] to RAM. You may use different memory maps. For example, a user code of up to 1 M Byte may be placed in ROM. Similarly, you may place monitor ROMs available from Keil or Tasking to be used in conjunction with their development tools.

5.1 RAM Memory Options

The RAM block is designed to take static RAMs, either 32KB 62C256-type, 128KB 681000-type or 512KB 628512-type static RAM chips. Alternately 62C256-type battery-backed RAMs may be used in the RAM block. Two chips are needed, one for EVEN and the other for ODD addresses. These chips are placed in 32-pin sockets marked U8 and U9. Place 28-pin 32K RAM devices closer to the 2 X 25 header, away from the processor.

RAM size must be selected with jumper SW5. Place a jumper in the lower position for 32K and 128K devices, for a total of 64K or 256K memory. For 512K devices (for a total of 1 M Byte) place the jumper in SW5 in the upper position marked “>256K”.

The 163 may be programmed to insert wait cycles during external memory access. However, in order to run the 163 at its full potential, the RAMs should be rated at 70 nano seconds or faster.

5.2 ROM Memory Options

The ROM block of memory accepts a variety of FLASH devices in the SGS 29F0xx family. Two chips are needed, one for EVEN and the other for ODD addresses. These chips are placed in the 32 pin PLCC sockets marked U6 and U7.

The 163 may be programmed to insert wait cycles during external memory access. However, in order to run the 163 at its full potential, the FLASH should be rated at 70 nano seconds or faster.

5.3 On-Chip FLASH Memory

The ST10F163 has 128K of on-chip FLASH memory. The R-FLIC board is designed to provide the 12 volts needed to program the FLASH. The R-FLIC is sold with software, the R166 Utility Software, especially written to program the FLASH. The on-chip FLASH has 4 memory banks, two 48K, one 24K, and one 8K , which may be programmed separately. Please refer to the ST10F163 data book for details of the FLASH memory.

Rigel Corporation’s utility rFLI is a Windows95/WindowsNT program to program the internal and external FLASH memory on all Rigel evaluation boards. RFLI is available on the Rigel Corporation WEB page free of charge.

5.4 Default Memory Setting

The default memory map assumes 256K of ROM containing the monitor program and 256K of RAM. In this configuration, the board is bootstrapped and run. JP1 should have jumpers in locations 6, 8, 9,10 and 11. ROMMAP selection SW3 should have a jumper in the lower CS0# position. RAMMAP selection SW4 should have a jumper in the upper CS1# position. RAMSIZE selection SW5 should have a jumper in the lower position. This is the default configuration for the R-FLIC and the board will be populated this way from the manufacturer. Programs are downloaded into RAM and then run.

5.5 Alternate Memory Maps

The RAM and ROM blocks of memory can be mapped in a variety of ways depending on the chip select signals assigned to each bank and on the size of each memory bank programmed into the external bus control registers BUSCON0 and BUSCON1. Refer to the SGS data books for more information on system programming. Please contact Rigel Corporation if you would like to install the monitor programs from Keil or Tasking in to ROM.

6. HEADERS

The R-FLIC board has five headers: the system bus header JP3, the input/output header JP4, the extra input/output header JP5, and the serial port headers. The system bus header contains the address, data, and control busses, the input/output header contains Ports 2, 3, and 5, and the extra input/output header contains Ports 5, 6, 7, and 8. Individual signals of these jumpers are listed below. Pin 1 may be identified as the post with the square pad on the printed circuit board.

6.1 JP3 - System Bus Header

Signal			Pins		Signal		
167	**163**	**166**			**166**	**163**	**167**
Ground			1	2	VCC (+5V)		
Ground			3	4	VCC (+5V)		
D0			5	6	A0		
D1			7	8	A1		
D2			9	10	A2		
D3			11	12	A3		
D4			13	14	A4		
D5			15	16	A5		
D6			17	18	A6		
D7			19	20	A7		
D8			21	22	A8		
D9			23	24	A9		
D10			25	26	A10		
D11			27	28	A11		
D12			29	30	A12		
D13			31	32	A13		
D14			33	34	A14		
D15			35	36	A15		
RD#			37	38	A16		
ALE			39	40	A17		
RSTIN#			41	42	WR#	WR# or WRL#	
RSTOUT#			43	44	BHE#		
NMI#			45	46	NC		A18
A22		NC	47	48	NC		A19
A21		NC	49	50	NC		A20

6.2 JP4 - Input/Output Header

Signal			Pins		Signal		
167	**163**	**166**			**166**	**163**	**167**
Ground			1	2	VCC (+5V)		
Ground			3	4	VCC (+5V)		
P5.0	not used	P5.0	5	6	P5.1	not used	P5.1
P5.2	not used	P5.2	7	8	P5.3	not used	P5.3
P5.4	not used	P5.4	9	10	P5.5	not used	P5.5
P5.6	not used	P5.6	11	12	P5.7	not used	P5.7
P5.8	not used	P5.8	13	14	P5.9	not used	P5.9
VAGND	not used	VAGND	15	16	S1I	not used	
VAREF	not used	VAREF	17	18	S1O	not used	
P2.0	not used	P2.0	19	20	P3.0		
P2.1	not used	P2.1	21	22	P3.1		
P2.2	not used	P2.2	23	24	P3.2		
P2.3	not used	P2.3	25	26	P3.3		
P2.4	not used	P2.4	27	28	P3.4		
P2.5	not used	P2.5	29	30	P3.5		
P2.6	not used	P2.6	31	32	P3.6		
P2.7	not used	P2.7	33	34	P3.7		
P2.8			35	36	P3.8		
P2.9			37	38	P3.9		
P2.10			39	40	P3.10		
P2.11			41	42	P3.11		
P2.12			43	44	P3.12		
P2.13			45	46	P3.13		
P2.14			47	48	P3.14	not used	
P2.15			49	50	P3.15		

6.3 JP5 - Extra Input/Output Header

Signal		Pins		Signal	
167	**163**			**163**	**167**
Ground		1	2	VCC (+5V)	
Ground		3	4	VCC (+5V)	
P6.0		5	6	P6.1	
P6.2		7	8	P6.3	
P6.4		9	10	P6.5	
P6.6		11	12	P6.7	
A23	not used	13	14	READY#	
VAGND	not used	15	16	not used	
VAREF	not used	17	18	not used	
P5.0	not used	19	20	not used	P7.0
P5.1	not used	21	22	not used	P7.1
P5.2	not used	23	24	not used	P7.2
P5.3	not used	25	26	not used	P7.3
P5.4	not used	27	28	not used	P7.4
P5.5	not used	29	30	not used	P7.5
P5.6	not used	31	32	not used	P7.6
P5.7	not used	33	34	not used	P7.7
P5.8	not used	35	36	not used	P8.0
P5.9	not used	37	38	not used	P8.1
P5.10	not used	39	40	not used	P8.2
P5.11	not used	41	42	not used	P8.3
P5.12	not used	43	44	not used	P8.4
P5.13	not used	45	46	not used	P8.5
P5.14	not used	47	48	not used	P8.6
P5.15	not used	49	50	not used	P8.7

6.4 Serial Port Headers

The ST10R163 has two serial ports. Serial port 0 is accessed through the RS-232 level converter. Serial port 1 is used as a high speed synchronous serial port (SSP) by the R-FLIC.

6.4.1 J3 (S0)

J3 is the secondary access port of serial port 0. J3 is a 3-pin header, which carries the same signals as P1. J3 is also denoted by S0 and its 3 lines by G (Ground), T (Transmit), and R (Receive), on the R-FLIC silk-screen. J3 is intended for embedded uses of the R-FLIC when P1 is not populated. J3 is located above the I/O headers.

6.4.2 J4 (SSC)

Header J4 is not used with the R163 processor.

7. SYSTEM PROGRAMMING

7.1 The SYSCON Register

The system control register (SYSCON) specifies many of the operating parameters of the 163 microcontroller. While many of these parameters may be changed depending on the application, the BYTDIS bit must be cleared and the Write Configuration Control bit (WRCFG) must be set. With BYTDIS=0, the BHE# signal is generated by the microcontroller. With WRCFG=1, the WR# signal of the microcontroller is used as WRL# and the BHE# as WRH#. These control signals, WRL# and WRH#, in turn are used to enable the odd and even memory devices of the RAM and ROM banks.

The following value is used in the RMON163C monitor program given in the distribution disk.

```
mov     SYSCON, #0080h  ; 0000:0000:1000:0000b
```

which selects the following parameters:

Field Name	Field Description	Value	Function	Remarks
STKSZ	stack size	000	256 words	default
ROMS1	map internal ROM to the first segment	0	do not map internal ROM in segment 1	default
SGTDIS	segmentation disabled	0	disable segmentation (64K memory)	only IP is used (CSP is not used) in calls and returns
ROMEN	ROM enable	0	disable internal ROM	
BYTDIS	Disable BHE#	0	do not disable BHE#	BHE# is used as WRH#
CLKEN	Clock enable	0	CLKOUT disabled	CLKOUT is the alternate function of P3.15
WRCFG	Write configuration	1	WR# is WRL#, BHE# is WRH#	
VISIBLE	Visible mode	0	XBUS accesses are done internally	default
XPER-SHARE	XBUS peripheral share mode	0	External accesses to XBUS peripherals are disabled	default

7.2 The BUSCON0 and BUSCON1 Registers

The width, type of multiplexing, and various wait states of the buses depend on the Bus Control (BUSCON) registers. Each chip select line (CSx#) has a corresponding BUSCONx register. The selection of the various wait states depend on the application and on the speed of the memory devices. Refer to the SGS Thomson Data Book for more information about the fields of BUSCON.

Since CS0# is the chip select line activated upon a reset, BUSCON0 has the special designation of determining the bus configuration upon reset. The bus type field (BTYP) of BUSCON0 is copied from Port 0 during reset. The values of Port 0 during reset are determined by the jumpers on the R-FLIC board as explained in Section 4.1. CS0# is used to activate the external ROM which holds the monitor program. External RAM is activated by CS1#. BUSCON1 is set to 48Fh to run the RAM with no wait states.

The three assignment statements in RMON163 pertinent to the memory map are given below.

```
mov    BUSCON0, #048Fh   ; 0 wait states
mov    ADDRSEL1, #0083h  ; RAM at 32K, size of 32K
mov    BUSCON1, #048Fh   ; 0 wait states
```

7.3 The ADDRSEL1 Register

Each CSx# line except CS0# has a corresponding ADDRSELx register which determines its offset (beginning address) and its range. CS0# does not have an ADDRSEL register. All addresses outside the ranges collectively defined by ADDRSELx registers are accessed by CS0#.

CS1# is used for external RAM, and thus ADDRSEL1 needs to be programmed. RMON163 sets ADDRSEL1 to 83h which sets up a window of 32K starting at 8000h. With this convention, the first 32K of the memory is ROM, and the second 32K, RAM.

8. THE MONITOR PROGRAM

The monitor program RMON163 is placed in the external ROM. By default the external FLASH occupies memory addresses [0..7FFF]. Memory region [8000..FFFF] is mapped to RAM. The source code for the monitor is given as a project named Rm163, under the Work subdirectory. RMON163 is written to communicate with Reads166. It supports downloading and running code written in assembly or C.

Alternatively, you may place monitor programs by Keil or Tasking into external ROM. This allows using Keil or Tasking development tools with the evaluation board.

8.1 RMON163 Monitor

The monitor program RMON163 allows inspecting and modifying the first 64K segment of R-FLIC memory, configuring the ports, inputting and outputting from the general purpose ports, downloading code in the Intel Hex format, and branching to user code. RMON163 features are invoked by single-letter commands. Serial port 0 is initialized to run at 19200 Baud with 8 bits of data, 1 stop bit and no parity bits.

RMON163 is intended to be downloaded after bootstrapping the R-FLIC board. RMON163 is placed starting at address 0. The monitor program reserves 512 bytes of external RAM to set up an interrupt vector table and to store some system variables. All processor interrupts except the reset vector are mapped to the beginning of RAM. That is, interrupt vector originally at address INTVEC is redirected to address INTVEC+8000h. All processor interrupts are redirected in memory back to the monitor. These assignments may be changed in RAM to allow user interrupt service routines to be called upon interrupts requests.

RMON163 is invoked upon reset. Since all unused interrupts are redirected to the monitor, RMON163 may subsequently be invoked by pressing the NMI push button on the R-FLIC. RMON163 initializes the stack and resets the interrupts. Thus, even after the NMI button is pressed, RMON163 clears the NMI interrupt by executing a dummy 'return from interrupt' instruction.

The single-letter commands of RMON163 are explained below.

D Download HEX file

The D command places RMON163 in a download mode. The monitor expects to receive code in the Intel Hex format through serial port 0. The download mode is terminated when the last line of Intel Hex code is received (when the byte count is 0).

C Port Configuration

The C command is used to configure the ports, i.e., the port direction registers DPnn. Cn displays the current setting of DPn. Cn=mmmm writes the word mmmm to register DPn.

G Go

The user code at address xxxx is branched to by the Gxxxx command. Note that the user program may return to RMON163 by a branching instruction to address 0C000h. RMON163 initializes the stack, thus, either a jump or a call instruction may be used to return to RMON163.

H Help

The H command displays a summary of available monitor commands.

M Memory

The first 64K segment of the R-FLIC memory may be inspected or modified by the M command. The M command is also useful to poke short programs into memory.

M XXXX displays the current contents of memory address XXXX.
M XXXX=nn inserts the byte nn into memory address XXXX. When this command is used, RMON163 displays the current contents as well as the new contents. The address XXXX is incremented and the current contents of (XXXX+1) are displayed. Consecutive bytes may be written starting at XXXX. The process is terminated if a carriage return or an illegal hexadecimal digit is keyed in.
M XXXX-YYYY displays the block of memory between addresses XXXX and YYYY.
M XXXX-YYYY=nn fills the memory block XXXX to YYYY with byte nn.

P Port Data

The P command is used to read from or write to the ports. Pn displays the current value of port n. If port n is an input port, then the value read is the current voltage levels applied to the ports. If port n is an output port, Pn returns the current output value to port n. Pn=mmmm sets the current value of output port n to mm.

Note that individual bits of the ports may be programmed as input or output. Thus, the word returned by Pn gives the external voltage levels applied to the input bits and the current values of the output bits.

W Word Memory

This command is identical to the M command, except that the memory contents are displayed and modified as words (2 bytes). Words start at even address.

9. Downloading and Running Code, ST10R163 Processor

Care must be taken in selecting a start address for any code written to run on the board. Since memory starts at 8200h (the first 512 bytes are reserved for the interrupt vector table and the system variables) code origins must be above this address. If you write C code and would like to debug the program, the pragma

```
#pragma Debug_R163
```

should be used. The program Factorial_R163 given on the distribution disk illustrates the necessary steps to debug C code with RMON163.

10. PARTS LIST

QUANTITY	PART	DESIGNATOR
	CAPICATORS	
1	1nF axial	C30
34	10nF axial	C1-C28, C41-47, C49
1	1uF MONO	C29
4	1.0 uF radial	C32-35
1	22uF radial	C31
5	47uF radial	C36-39, C48
1	100uF radial	C46
1	220uF radial	C44
1	330uF radial	C40
	DIODES	
1	1N4001	D1
2	RED LED	D2, D3
1	GREEN LED	D4
1	2N7000	Q1
1	BRIDGE DIODE	D5
	CONNECTORS	
3	25x2 HEADER	JP3, 4, 5
1	3.5MM DC JACK	JP16
1	DB 9 CONNECTOR	P1
2	6mm PUSH BUTTON	SW1, SW2
4	1X2 HEADER	J6, J7, JP13, JP14
7	3.5 TERM BLOCK	JP6, J5, J8, JP15
4	1X3 HEADER	J1, J2, J3, SW5
2	2x3 HEADER	SW3/SW4, JP21
1	2X4 HEADER	JP17-JP20
1	1X5 HEADER	JP4
1	2x16 HEADER	JP1
2	ICE CONN	JP2
1	1X4 HEADER	10/11
1	1X2 HEADER	12
1	2x3 HEADER	7/8/09
	RESISTORS	
1	470 OHM 1/2 W	R6
1	100 OHM 1/4W	R1
2	10K 1/4W	R9, 10
1	8.2K GANG (10)	R4
1	8.2K GANG (8)	R5
1	10K (6 GANG)	R2

2	1K	R7, R8
	VARISTOR	
2	ZNR	V1, V2
	SOCKETS	
1	8 PIN SOCKET	U1
1	16 PIN SOCKET	U4
2	32 PLCC SOCKET	U6, U7
2	32 PIN SOCKET	U8, U9
	ICS	
1	ST10R163	U3
1	LM7805-TO220	U10
1	LM7812-TO92	U11
1	RS232	U4
2	621000 SRAM	U8. U9
1	5 MHz CLOCK	U1

11. SCHEMATICS