

SOFTWARE DEVELOPMENT TOOLS FOR ST6 MCU FAMILY

- Includes:
 - Macro assembler
 - Linker
 - Software simulator
- Runs on MS-DOS systems
- Window based graphic interface
- Extensive symbol manipulation

GENERAL DESCRIPTION

Full software development tools is achieved using the ST6 Software Development Tools consisting of a powerful macro assembler, a linker and a software simulator.

The ST6 Macro assembler accepts a source file written in ST6 assembly language using any text editor package and transforms it in an ST6 executable file.

To enable good testability and fast debugging, many application software are made up of several modules, each of them performing an elementary

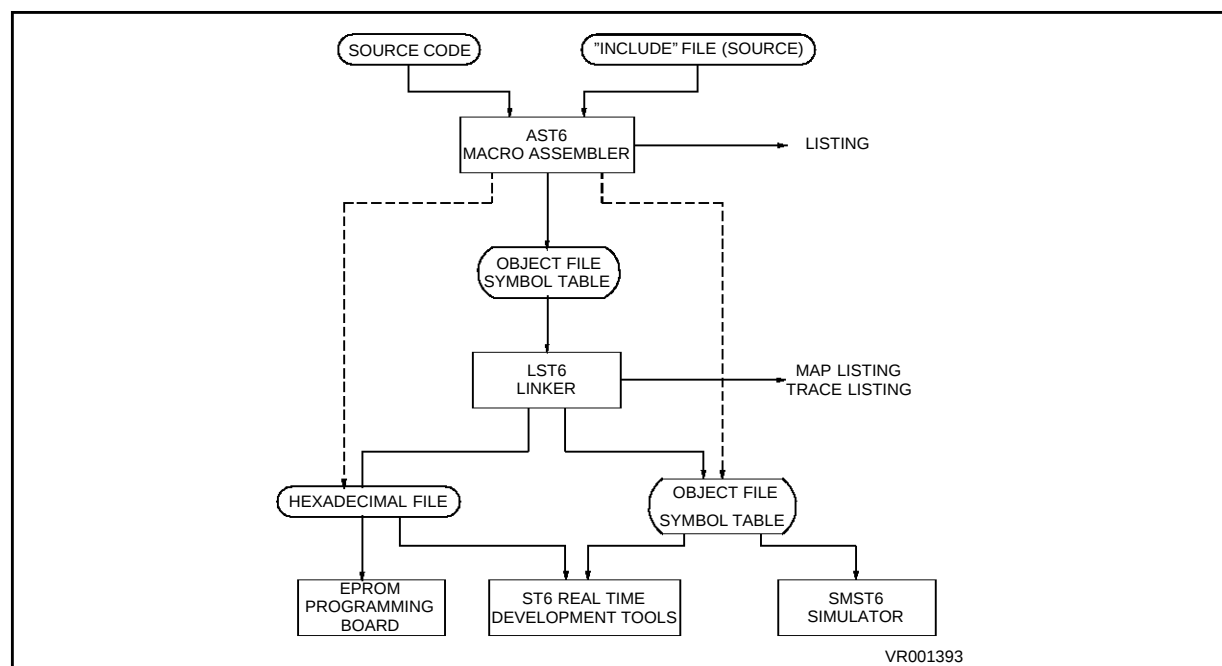
task. Each module is assembled independently of the others, thus producing a number of object files. The ST6 Linker combines these object files into a single executable program. Both object format and hexadecimal format are produced. The hexadecimal file is used to program an EPROM while the object file is used to run the simulator or the debugger.

The ST6 Software Simulator allows the user to debug and execute any executable program written for any member of the ST62/ST63 family of microcontrollers without the aid of additional hardware.

Once debugged with the simulator, the program can be programmed into an EPROM device by using the hexadecimal file and the ST6 programming board. By plugging the EPROM device into the application hardware, simple applications can be debugged without the need of an emulator.

The ST6 Hardware Development Tools are required where high performance debugging is needed.

Figure 1. Development Flow Chart



ST6 ASSEMBLER

- Macro call and conditional assembly
- Extensive symbol manipulation
- Error diagnostics

General Description

The ST6 Macro assembler accepts a source file written in ST6 assembly language and transforms

it into an executable file in relocatable object code format. When the whole program is in one file only, the assembler also generates an hexadecimal file (INTEL hex format) ready to be programmed into an EPROM device.

The assembler recognizes the use of section, symbols, macros and conditional assembly directives. In addition, the ST6 Assembler is able to produce detailed assembly listing and symbol cross reference file.

Figure 2. AST6 Directives

.ASCII	Stores in program space a string as a sequence of ASCII codes
.ASCIZ	Same as .ASCII followed by a null character
.BLOCK	Reserves a block of contiguous memory location
.BYTE	Stores successive bytes of data in program space
.DEF	Defines the characteristics of a data space location
.DISPLAY	Displays a string during assembly process
.DP_ON	Segments the data space
.EJECT	Starts a new listing page
.ELSE	Beginning of the alternative part in conditional assembly block
.END	End of source file
.ENDC	End of conditional assembly block
.ENDM	End of a macro definition
.EQU	Assigns the value of an expression to a label
.ERROR	User defined assembly error
.EXTERN	Defines a symbol as external
.IFC	Beginning of conditional assembly block
.INPUT	Includes an additional source file in the present one
.GLOBAL	Defines a symbol as global
.LABEL.W	Initializes Data ROM Window Register
.LABEL.D	Gains access to a label in a Data ROM Window
.LINESIZE	Set listing line length
.LIST	Enables the listing of specified fields of the source file
.MACRO	Beginning of a macro definition
.MEXIT	End of a macro expansion
.NOTRANSMIT	Inhibits symbol transmission to the linker
.ORG	Set current location counter
.PAGE_D	Specifies the page number in data space
.PL	Set listing page length
.PP_ON	Segments the program space in 2K pages
.ROMSIZE	Defines the available ROM size
.SECTION	Provides a logical partitioning of program space
.SET	Same as .EQU, but can be redefined in the source file
.TITLE	Assigns title to the document
.TRANSMIT	Transmits symbol definitions to the linker
.VERS	Defines the target ST6 device
.WARNING	User defined assembly warning
.WINDOW	Defines a continuous relocatable block of program code
.W_ON	Enables the use of the .WINDOW directive
.WORD	Stores successive words of data in program space

ST6 LINKER

- Links up to 32 modules
- Extensive symbol manipulation
- 33 sections (including interrupt vectors)
- Error diagnostics

The ST6 Linker is responsible for combining a number of object files into a single program, associating an absolute address to each section of code, and resolving any external references.

The ST6 Linker produces an hexadecimal file in INTEL format to be down loaded into an EPROM device and an object code file to be used with the simulator. The linker also produces a map file which gives information about the sections, pages, modules and labels. Finally, listing files are produced which update the assembler listings with real addresses of symbols and statements.

This software program allows the user to develop modular programs, which may then be combined and addressed as defined by the user. The flexibility of the ST6 Linker is greatly increased by the use of sections allowing the user to group pieces of software from different modules. The location and the size of each section is user selectable.

ST6 SIMULATOR

- Window based graphic interface
- On line assembler/disassembler
- Supports symbolic debugging
- 128 breakpoints and 128 software traps
- TRACE mode
- I/O and CLOCK simulation

SIMST6 allows the user to debug and execute any program written for any of the current and future members of the ST6 family of microcontrollers, without the aid of additional hardware.

The user specifies the target device, its mapping and the object code file to be used. The simulator functionally duplicates the operation of the ST6 and completely supports the instruction set. I/O channels may be opened, read, and written, in order to simulate the I/O functions of peripherals, while interrupts may be set, and then set pending, in order to simulate the handling of interrupts. The simulator uses the clock frequency assigned by the user, along with the number of clock cycles needed by each instruction to keep track of the real time execution speed.

The ST6 Simulator accepts command lines in both interactive and batch mode.

ORDERING INFORMATION

Sales Type	Description
ST6-SW	ST6 software development tools (includes assembler, linker and emulator)

Note : The ST6 software package is included in all ST6xxx-EMU real time development tools.

Information furnished is believed to be accurate and reliable. However, SGS-THOMSON Microelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of SGS-THOMSON Microelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or systems without the express written approval of SGS-THOMSON Microelectronics.

© 1994 SGS-THOMSON Microelectronics - All rights reserved.

Purchase of I²C Components by SGS-THOMSON Microelectronics conveys a license under the Philips I²C Patent. Rights to use these components in an I²C system is granted provided that the system conforms to the I²C Standard Specification as defined by Philips.

SGS-THOMSON Microelectronics Group of Companies

Australia - Brazil - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - Malta - Morocco - The Netherlands
Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.