

Synapse Assembler 'SynAssembler' by Steve Hales. Copyright (C) 1982 SYNAPSE Software and Steve Hales#

Table of Contents

- [Synapse Assembler 'SynAssembler' by Steve Hales. Copyright \(C\) 1982 SYNAPSE Software and Steve Hales](#)
- [Introduction](#)
- [Review](#)
- [Origin](#)
- [Author](#)
- [Manual](#)
- [CAR-Images](#)
- [ROM-Images](#)
- [ATR-Image](#)
- [XEX-File](#)
- [SynAssembler on Atari XL/XE machines](#)
- [Source Code](#)
- [Atari 8bit Equates files](#)
- [Pictures and Themes](#)
- [Help File](#)

Introduction#

SYNAPSE Software

SYNASSEMBLER is a convenient and powerful tool for software development on the Atari computer system. The assembler uses standard 6502 mnemonics and syntax, and includes many useful features for creating, editing, assembling and testing your assembly language programs. Now assembly language programming is almost as easy as programming in BASIC.

Here is a summary of the most exciting features:

- Full use of the standard Atari Screen Editor
- Tab stops for opcode, operand, and comment fields
- Fast parameterized renumber and delete command
- Uses BASIC like commands for files (eg. LOAD, SAVE, BLOAD etc.)
- Labels up to 32 characters long (lowercase letters and . accepted)
- English ERROR messages
- Ability to append source programs from tape or disk
- Display of memory usage
- Multiple source files, using .IN directive
- Store object code directly to disk file, using the .TF directive
- Listing to screen option using .LI ON and .LI OFF
- Assembles 6500 lines/minute
- Local labels

- Offending line is listed after an ERROR occurs
- Value of .EQ and address of .BS are printed on listing
- ASSEMBLER, DOS, HARDWARE, and OS locations protected during assembly
- ATASCII literals in address expressions
- Symbol table printed in alphabetical order

Synassembler requires 48K of RAM and one disk drive to operate. Very large programs can now be developed, using the "INCLUDE" and "TARGET FILE" capabilities. These allow the assembly of multiple source files, and direct storage of object code on binary files.

Review#

Synassembler
 Synapse Software
 5327 Jacuzzi St., Suite 1
 Richmond, CA 94804
 (415) 527-7751
 \$49.95 Diskette 48K
 \$89.95 Cartridge

Reviewed by Adrian Dery

Synapse has come up with a really powerful Assembler, Editor and machine-language Monitor. All these are in a single program which is available on disk, or by special order on a ROM cartridge. This Editor does for Assembly programs what the BASIC cartridge does for BASIC programs, and it works much the same way. Additional editing commands include: Auto-Line Numbering; Renumber (all or part of a program); Delete Lines; Move and Copy (blocks of lines from one part of a program to another); and Search/ Replace (character strings).

The Editor has a unique HIDE feature that will protect a source program in memory. Load or type in a new program, then edit and assemble it completely apart from the program you are hiding. You can then save it, or delete it, or append it to the hidden program.

The Assembler part of Synassembler is incredibly fast! I have assembled programs as large as 1500 statements and it's average speed is about 100 statements per second, with the source file in memory and the listing turned off.

An Include feature assembles multiple source files in a single pass. This is useful for picking up "canned" subroutines or things like a list of Operating System equates. It also can assemble very large programs and it is quite possible, and sometimes practical, to have a main program that has only Include statements in it.

The Monitor is a full-featured : machine-language debugger. Memory can be displayed, changed or moved around. Registers can also be displayed and changed. Program execution can be traced, or you can singlestep through the instructions. There are also some special read/write commands that allow you to directly read and write any disk sectors without opening a file.

Synassembler is a professional development tool for the experienced programmer as well as the beginner. It has an excellent Editor, a very fast Assembler capable of assembling programs of virtually unlimited size, and a Monitor that should serve well in finding the trickiest of bugs. It's a step above the Atari cartridge because of its speed and ability to include multiple source files. Synassembler does require 48K and you need a disk drive to take advantage of all its features. If you have the memory and the disk, it is a good value for the money.

Origin#

SynAssembler is a port of the S-C Assembler II Version 4.0 from the Apple II. Information from [Apple Assembly Line V2N12](#)

SYNASSEMBLER: Synapse Software has just started marketing a conversion of the S-C Assembler II Version 4.0 for the Atari 800 or 400. You need 48K RAM and at least one disk drive. The

conversion was done by Steve Hales, of Livermore, California. He added global replace and copy commands, so this version falls somewhere between the Apple version 4.0 and the new Macro version. It assembles at about 6500 lines per minute, which is from 50 to over 100 times faster than the Atari ASM/ED program.

Since the Atari does not have nice monitor commands built-in, like the Apple does, Steve added a complete set of monitor commands to SYNASSEMBLER. They look exactly like the Apple monitor commands, except that he added some new ones to allow reading and writing a range of disk sectors, delete the tape I/O commands, and included the old Step and Trace commands which were in Apples before the Autostart ROM.

The price is only \$49.95 on disk. A ROM version is available by special order for \$89.95. I will carry these, if you want to order from me.

The original sourcecode of S-C Assembler by Bob Sander-Cederlof can be found at <http://www.txbobsc.com/scsc/scdisassembler/index.html>

The Apple Assembly line (also by Bob Sander-Cederlof) contains additional information for the S-C Assembler: <http://www.txbobsc.com/aal/index.html>

Author#

Synapse Assembler has been written by Steve Hales. Steve Hales was and still is an Atari hero and living legend. Describing Steve is quite easy. If you compare Atari to the whole world, then Steve would be a global player. He has done so much for Atari and the community. Steve is already a member in the [hall of fame](#) and carrier of the sigma, the highest award possible on the AtariWiki. We owe him so much! In 2015 he has given us the source code for Fort Apocalypse, further he has given us the source code listing from Star Raiders! Without his help they would be lost. Steve, tera-thanks to you, may you live long and prosper.

- [Homepage of Steve Hales](#)
- [Interview with Steve Hales](#) ; thanks to Mr. Bacardi
- [Interview with Ihor Wolosenko](#) ; thanks to Mr. Bacardi

Manual#

- [Synapse Assembler Manual](#) ; online manual
- [Synapse Assembler Manual](#) ; size: 172 KB; pdf-file
- [Synapse Assembler Manual](#) ; size: 197 KB; doc-file
- [Synapse Assembler Manual](#) ; size: 225 KB; pdf-file with OCR and bookmarks
- [Synassembler Original Manual Synapse.pdf](#) ; size: 6.8 MB ; 62 pages, thank you so much Allan Bushman for preserving this original manual! :-)))

CAR-Images#

- [Synassembler.car](#) ; cartridge image for Atari 400-800 with OS B
- [Synassembler_XL.car](#) ; cartridge image for Atari XL-XE

ROM-Images#

- [Synassembler.rom](#) ; rom image for Atari 400-800 with OS B; md5: e5dd9f57ff9b807d52a99869dfbfcc17
- [Synassembler_XL.rom](#) ; rom image for Atari XL-XE; md5: 7375df1158ad81f52160f417a49ff78e
- [Synassembler\(classic\).zip](#) ; rom images for Atari 400-800 with OS B with different themes

- [Synassembler\(XL-XE\).zip](#) ; rom images for Atari XL-XE with different themes

ATR-Image#

- [SynAssembler.atr](#) ; atr image for all Atari computers ; with special DOS for direct jump back to the program
- [SynAssembler_XL.atr](#) : atr image for XL/XE Atari computers ; with special DOS for direct jump back to the program
- [Synapse page 6 utilities](#) ; Synapse page 6 utilities

XEX-File#

- [SynAssembler_XL.xex](#) ; xex file for XL-XE Atari computers

SynAssembler on Atari XL/XE machines#

The original SynAssembler does not work on XL/XE machines, due to the use of direct jumps into the 400/800 OS.

There is a basic program in [ANTIC VOL. 3, NO. 2 / JUNE 1984 / PAGE 38](#) that patches the SynAssembler binary to run on XL/XE type machines.

- [SynAssembler_patch_from_classic_to_XL-XE.atr](#) ; patch in BASIC

Source Code#

- [SynAssembler_Source_Code.atr](#) ; source code of the SynAssembler in MAC/65 format with DOS 2.5 color; Thank you so much Steve Hales, we really appreciate your help for the community. :-)))

Atari 8bit Equates files#

- [Synapse Assembler Atari 800 OS Equates](#)

Pictures and Themes#

SynAssembler diskette

SynAssembler startscreen-original after starting with \$CA in \$02C5 and \$C2 in \$02C6

SynAssembler after starting with \$CA in \$02C5 and \$94 in \$02C6

SynAssembler after starting with \$CA in \$02C5 and \$90 in \$02C6

SynAssembler after starting with \$0F in \$02C5 and \$00 in \$02C6

SynAssembler after starting with \$00 in \$02C5 and \$0F in \$02C6

SynAssembler after starting with \$CA in \$02C5 and \$34 in \$02C6

SynAssembler after starting with \$CA in \$02C5 and \$46 in \$02C6

SynAssembler after starting with \$CA in \$02C5 and \$FF in \$02C6

Help File#

```
00010 *
00020 * SYNASSEMBLER DIRECTIVES
00030 *
00040 .AS ASCII STRING
00050 .AS "TEST"
00060 .AT ATASCII STRING
00070 .AT "SCORE:"
00080 .BS BLOCK STORAGE
00090 .BS 5 SKIPS 5 BYTES
00100 .DA WORD
00110 .DA HELLO
00120 .EN END OF SOURCE (OPTIONAL)
00130 .EN
00140 .EQ EQUATE
00150 TEST .EQ 5
00160 .HS HEX STRING
00170 .HS 0001AA5500339944344
00180 .IN INCLUDE FILE WITH ASM
00190 .IN "D:FILE NAME"
00200 .LI LIST ON OR OFF
00210 .LI OFF
00220 .OR SET ORG
00230 .OR $6000
00240 .TA SET TARGET FOR OBJECT
00250 .TA $9000
00260 .TF SEND OBJECT TO DISK
00270 .TF "D:FILE NAME"
00280 *
00290 * SYNASSEMBLER COMMAND
00300 *
00310 LOAD "D:FILE NAME"
00320 SAVE "D:FILE NAME"
00330 BLOAD "D:FILE NAME", <START>
00340 BSAVE "D:FILE NAME", <START>, <END>
00350 ENTER "D:FILE NAME"
00360 DIR DIRECTORY DRIVE 1
00370 DIR "D4:*.*" DIRECTORY DRIVE 4
00380 DOS
00390 MEM MEMORY STATUS
00400 LIST L1,L2
00410 NEW
00420 REN L1,L2
00430 RENUMBER L1 USING INC OF L2
00440 REN L1,L2,L3
00450 RENUMBER L1 USING INC OF L2
00460 STARTING AT L3
00470 COPY L1,L2
00480 COPY L1 AT L2
00490 COPY L1,L2,L3
00500 COPY L1 TO L2 AT L3
00510 MOVE L1,L2
00520 MOVE L1 TO L2
00530 MOVE L1,L2,L3
00540 MOVE L1 TO L2 AT L3
00550 FIND STRING
00560 ASM
00570 DEL L1,L2
```

```

00580 HID
00590  HID CURRENT SOURCE PROGRAM
00600 MER
00610  COMBINE HIDDEN SOURCE WITH
00620  CURRENT SOURCE
00630  RUN EXPRESSION
00640  RESTORE
00650  RETURN FROM AN INCLUDE FILE
00660  OR TO UN-HIDE A HIDDEN FILE
00670  SYM
00680  PRINT SYMBOL TABLE
00690  VAL EXPRESSION
00700  REP /.BYTE/.HS/,P
00710  REPLACE WITH PROMPT
00720  PRESS 'Y' TO CHANGE
00730  PRESS 'N' FOR NO CHANGE
00740  PRESS 'X' TO CANCEL REPLACE
00750  REP /.BYTE/.HS/
00760  REPLACE ALL
00770  OUT "P:"
00780  PRINT OUTPUT TO DEVICE
00790  TYPE "P:"
00800  LIST TO DEVICE
00810  LOMEM $4000  OR  LOM $4000
00820  LOMEM OF SYMBOL TABLE
00830  HIMEM $8000  OR  HIM $8000
00840  HIMEM OF SOURCE CAUTION
00850          DESTROYS SOURCE
00860  MON
00870  JUMP TO MONITOR
00880  Q
00890  JUMP BACK TO ASSEMBLER
00900  *****
00910  * MONITOR INSTRUCTIONS *
00920  *****
00930  EXAMINE MEMORY
00940  *
00950  adrs          C0F2
00960  adrs1.adrs2  1024.1048
00970  (RETURN)    DISPLAY NEXT 8 LOC
00980  .adrs        .4096
00990  *
01000  CHANGE MEMORY
01010  *
01020  adrs;data data      A256;EF 20 43
01030  ;data data data    ;F0 A2 12
01040  *
01050  MOVE MEMORY
01060  *
01070  adrs1<adrs2.adrs3M 100<B010.B410M
01080          FROM adrs2 TO adrs3 STARTING          AT adrs1
01090  *
01100  VERIFY MEMORY
01110  *
01120  adrs1<adrs2.adrs3V 100<B010.B410V
01130  *
01140  * READ SECTOR(S)
01150  *
01160  adrs<sec1.sec2r    2000<1.AFr

```

```
01170 READ SECTORS 1 TO AF
01180 AND PUT THEM INTO BUFFER
01190 AT 2000
01200 *
01210 * WRITE SECTOR(S)
01220 adrs<sec1.sec2w 2000<1.AFw
01230 WRITE SECTORS 1 TO AF
01240 AND GET THEM FROM BUFFER
01250 AT 2000
01260 *
01270 *
01280 adrsL C080L
01290 L NEXT 20 INSTRUCTIONS
01300 *
01310 DEBUGGING
01320 *
01330 adrsG 300G RUN
01340 adrsT 800T TRACE
01350 adrsS C050S SINGLE STEP
01360 R DISPLAY REGISTERS
01370 TO CHANGE REGISTERS
01380 TYPE ; THEN DATA
01390 FOR EACH ONE
```