NAME

SRecord − manipulate EPROM load files

DESCRIPTION

The *SRecord* package is a collection of powerful tools for manipulating EPROM load files.

I wrote SRecord because when I was looking for programs to manipulate EPROM load files, I could not find very many. The ones that I could find only did a few of the things I needed. SRecord is written in C++ and polymorphism is used to provide the file format flexibility and arbitrary filter chaining. Adding more file formats and filters is relatively simple.

The File Formats

The SRecord package understands a number of file formats:

Ascii-Hex
: The ascii-hex format is understood for both reading and writing. (Also known as the ascii-space-hex format.)

ASM
: It is possible, for output only, to produce a serices of DB statements containing the data. This can be useful for embedding data into assembler programs. This format cannot be read.

Atmel Generic
: This format is produced by the Atmel AVR assembler. It is understood for both reading and writing.

BASIC
: It is possible, for output only, to produce a serices of DATA statements containing the data. This can be useful for embedding data into BASIC programs. This format cannot be read.

Binary
: Binary files can both be read and written.

B-Record
: Files in Freescale Dragonball bootstrap b-record format can be read and written.

C
: It is also possible to write a C array declaration which contains the data. This can be useful when you want to embed download data into C programs. This format cannot be read.

COE
: The Xilinx Coefficient File Format (.coe) is understood for output only.

Cosmac
: The RCA Cosmac Elf format is understood for both reading and writing.

DEC Binary
: The DEC Binary (XXDP) format is understood for both reading and writing.

Elektor Monitor (EMON52)
: The EMON52 format is understood for both reading and writing.

Fairchild Fairbug
: The Fairchild Fairbug format is understood for both reading and writing.

Formatted Binary
: The Formatted Binary format is understood for both reading and writing.

Four Packed Code (FPC)
: The FPC format is understood for both reading and writing.

Hexdump
: It is possible to get a simple hexdump as output.

IDT/sim
: The IDT/sim binary file format is understood for both reading and writing.

Intel
: The Intel hexadecimal format is understood for both reading and writing. (Also known as the Intel MCS-86 Object format.)

Intel AOMF
: The Intel Absolute Object Module Format (AOMF) is understood for both reading and writing.

Intel 16
: The Intel hexadecimal 16 format is understood for both reading and writing. (Also known as the INHX16 file format.)

LSI Logic Fast Load
: The LSI Logic Fast Load format is understood for both reading and writing. Logisim The Logisim format is understood for both reading and writing. See *srec_logisim*(5) for more information.

Memory Initialization Format
: The Memory Initialization Format (.mem) by Lattice Semiconductor is understood for writing only.

MIF
: The Memory Initialization File format by Altera is supported for both reading and writing.

MOS Technology
: The MOS Technology hexadecimal format is understood for both reading and writing.

MIPS-Flash
: The MIPS Flash file format is supported for both reading and writing.

Motorola S-Record
: The Motorola hexadecimal S-Record format is understood for both reading and writing. (Also known as the Exorciser, Exormacs or Exormax format.)

MsBin
: The Windows CE Binary Image Data Format is supported both for reading and writing.

Needham
: The Needham Electronics ASCII file format is understood for both reading and writing.

OS65V
: The Ohio Scientific hexadecimal format is understood for both reading and writing.

PPB
: The Stag Prom Programmer binary format is understood for both reading and writing.

PPX
: The Stag Prom Programmer hexadecimal format is understood for both reading and writing.

Signetics
: The Signetics format is understood for both reading and writing.

SPASM
: The SPASM format is used by a variety of PIC programmers; it is understood for both reading and writing.

Spectrum
: The Spectrum format is understood for both reading and writing.

Tektronix (Extended)
: The Tektronix hexadecimal format and the Tektronix Extended hexadecimal format are both understood for both reading and writing.

Texas Instruments Tagged
: The Texas Instruments Tagged format is understood for both reading and writing (both 8 and 16 bit). Also known as the TI-tagged or TI-SDSMAC format.

Texas Instruments ti-txt
: The TI-TXT format is understood for reading and writing. This format is used with the bootstrap loader of the Texas Instruments MSP430 family of processors.

TRS-80
: The Radio Shack TRS-80 object file format is understood for reading and writing.

VHDL
: It is possible to write VHDL file. This is only supported for output.

Verilog VMEM
: It is possible to write a Verilog VMEM file suitable for loading with `$readmemh()`. This format is supported for reading and writing.

Wilson
: The Wilson format is understood for both reading and writing. This mystery format was added for a mysterious type of EPROM writer.

The Tools

The primary tools of the package are *srec_cat* and *srec_cmp*. All of the tools understand all of the file formats, and all of the filters.

srec_cat The *srec_cat* program may be used to catenate (join) EPROM load files, or portions of EPROM load files, together. Because it understands all of the input and output formats, it can also be used to convert files from one format to another.

srec_cmp
The *srec_cmp* program may be use to compare EPROM load files, or portions of EPROM load files, for equality.

srec_info
The *srec_info* program may be used to print summary information about EPROM load files.

The Filters

The *SRecord* package is made more powerful by the concept of *input filters*. Wherever an input file may be specified, filters may also be applied to that input file. The following filters are available:

bit reverse
The *bit-reverse* filter may be used to reverse the order of bits in each data byte.

byte swap
The *byte swap* filter may be used to swap pairs of add and even bytes.

CRC The various *crc* filters may be used to insert a CRC into the data.

checksum
The *checksum* filters may be used to insert a checksum into the data. Positive, negative and bit-not checksums are available, as well as big-endian and little-endian byte orders.

crop The *crop* filter may be used to isolate an input address range, or ranges, and discard the rest.

exclude The *exclude* filter may be used to exclude an input address range, or ranges, and keep the rest.

fill The *fill* filter may be used to fill any holes in the data with a nominated value.

length The *length* filter may be used to insert the data length into the data.

maximum
The *maximum* filter may be used to insert the maximum data address into the data.

minimum
The *minimum* filter may be used to insert the minimum data address into the data.

offset The *offset* filter may be used to offset the address of data records, both forwards and backwards.

random fill
The *random fill* filter may be used to fill holes in the data with random byte values.

split The *split* filter may be used to split EPROM images for wide data buses or other memory striping schemes.

unfill The *unfill* filter may be used to make holes in the data at bytes with a nominated value.

unsplit The *unsplit* filter may be reverse the effects of the split filter.

More than one filter may be applied to each input file. Different filters may be applied to each input file. All filters may be applied to all file formats.

ARCHIVE SITE

The latest version of *SRecord* is available on the Web from:

URL:	http://srecord.sourceforge.net/	
File:	index.html	# the SRecord page
File:	srecord−1.64.README	# Description, from the tar file
File:	srecord−1.64.lsm	# Description, LSM format
File:	srecord−1.64.spec	# RedHat package specification
File:	srecord−1.64.tar.gz	# the complete source
File:	srecord−1.64.pdf	# Reference Manual

BUILDING SRECORD

Full instructions for building *SRecord* may be found in the *BUILDING* file included in this distribution.

It is also possible to build *SRecord* on Windows using the Cygwin (www.cygwin.com) or DJGPP (www.delorie.com/djgpp) environments. Instructions are in the *BUILDING* file, including how to get native Windows binaries.

COPYRIGHT

srecord version 1.64
Copyright © Scott Finneran <scottfinneran@yahoo.com.au>
Copyright © 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014 Peter Miller <pmiller@opensource.org.au>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

It should be in the *LICENSE* file included with this distribution.

AUTHOR

Peter Miller E-Mail: pmiller@opensource.org.au
/\/* WWW: http://miller.emu.id.au/pmiller/

RELEASE NOTES

This section details the various features and bug fixes of the various releases. For excruciating and complete detail, and also credits for those of you who have generously sent me suggestions and bug reports, see the *etc/CHANGES.** files.

Version 1.64 (2014-Jun-22)

• Cleaned up a few references to the maintainer's name.

• Fixed some warnings in test 38

• Fixed bugs discovered by Mike <russiane39@gmail.com> regarding the formatting of some of the examples where lines were wrapping the wrong way.

• Added the ability to compile without libgcrypt if the user specifies the option --without-gcrypt to configure. If gcrypt is missing, the user is prompted to either install it or explicitly compile with it disabled.

• Cleaned up a few recent compiler and doxygen warnings.

• Added Sourceforge Patch #4 contributed by Stas Sergeev <stsp@sourceforge.net> which contributed get_upper_bound() and makes it and the lower equivalent public (for use from libsrecord). Also fixed a bug in srec_memory::find_next_chunk() where it would fail to reset the find_next_chunk_index once it became equal to nchunks. After that, find_next_chunk() would always fail.

• Added prefix and postfix strings to the C Array format which are applied at each end of the data array declaration. This allows the user to add various compiler directives such as location flags or non-standard load address specifiers.

• when converting from AOMF Fixed Sourceforge bug 11 raised by patryks. In the analysis, it was discovered that a start address (of zero) was being generated when converting from AOMF. AOMF does not support the concept of an execution start address.

• The Windows build instructions have been updated by Jens Heilig <jens@familie-heilig.net>.

• Fixed the length field description in extended tektronix documentation. The implementation was already correct.

• Added a regression test for calculated address for CRC (Sourceforge bug 19).

Version 1.63 (2014-Apr-07)

• The srec_tools now understand how to read and write Logisim formt.

• Daniel Anselmi <danselmi@gmx.ch> fixed a bug with generating Lattice Memory init files.

• This change set generalises the code that handles redundant byte settings and contradictory byte settings. The defcon_t type describes what to do: ignore, warn or error. The −−multiple option is no more.

• Daniel Anselmi <danselmi@gmx.ch> discovered that the "mem" output format was malformed. The lines now break every "width" bits.

• Hanspeter Niederstrasser disoverd some false negatives in the test suite, when used on OSX. Thank you for the bug report. This has been fixed.

• Simplification of selection code to choose which CRC16 implmentation is used.

• Juliano MourÃÂ£o Vieira <juliano@utfpr.edu.br> discovered a problem with the srec_mif.5 description. This mistake is not present in the code.

• Liju Prasanth Nivas (RBEI/ECA1) <Liju.PrasanthNivas@in.bosch.com> suggsted another use case for the examples. In the case of "joining" files that are meant to be “stacked in layers”. Contributions for the examples are always appreciated. Thank you.

• Added more comments to the code, so that the use of URL_deode for the command line is better explained. And comment to explain why not to do turl_encode when building header records.

• Added URL quoting to the command line. There are times you need to be able to insert unprintable characters on the command line. The immediate use case prompting this was a user wanting to put a NUL in the header string. So now you can, as "%00". The other choice was quoted printable encoding, but that

was a bit obscure.

• Fixed new warnings when building with g++ 4.8.1

• Added more links to the windows files on SourceForge, maybe it will boost download numbers.

Version 1.62 (2013-Jun-05)

- Luc Steynen <LucSteynen@edna.be> discovered that the −hecksum-big-endian opion was a counter-intuitive alias for the the −checksum-bitnot-big-endian option. The −checksum-big-endian option is now deprecated, in favor of the −checksum-bitnot-big-endian option; the code will warn uers of the old option they will need to change. Ditto little-endian variants
- Alin Pilkington <apilkington@moog.com> found that the Tektronics Extended format was calculating the record length incorrectory. Thanks you for the bug report. This has been fixed for both reading and writing.
- Dr. Benedikt Schmitt <Benedikt.Schmitt@safeintrain.de> suggested being able to inject arbitrary data into the file header (such as NUL termination characters). This change set adds URL-style escapes (e.g. %25) to the string on he command line. For example: −header or −generate −string

Version 1.61 (2013-Jan-04)

- Izzet Ozcelik <izzetozcelik@cscope.co.uk> discovered a bug in the Tektronix-Extenden format line checksum calculations. The comparison should have been in 8 bits, not int.
- Daniel Anselmi <danselmi@gmx.ch> contributed a Memory Initialization Format by Lattice Semiconductor, for output only.
- Daniel Anselmi <danselmi@gmx.ch> contributed a Xilinx Coefficient File Format (.coe) output class.

Version 1.60 (2012-May-19)

• There are now several additional CRC-16 polynomials, plus the ability to select a polynomial by name, rather than by value. See http://en.wikipedia.org/wiki/Cyclic_redundancy_check for a table of names and values.

Version 1.59 (2012-Feb-10)

- A number of additional CRC-16 polynomials have been added, as well as the ability to select a polynomial by name, rather than by value. See *srec_input*(1) for more information.

Version 1.58 (2011-Dec-18)

- The **−guess** command line option, for guessing the file format, now also tells you the command line option you could have used instead of **−guess** for the exact format.
- The Intergated Device Technology (IDT) system integration manager (IDT/sim) binary format is now understood for both reading and writing.
- The Stag Prom Programmer binary format is now supported for both reading and writing.
- The Stag Prom Programer hexadecimal format is now understood for both reading and writing.
- The MIPS-Flash fiel format is now supported for both reading and writing.
- Bernhard Weirich <Bernhard.Weirich@riedel.net> discovered that a backward compatible option had been omitted when the **−INtel_16** option was renamed **−INtel_HeX_16** to more closely match the usual abbreviation (INHX16) for this format. The backwards compatible option name has been reintroduced.
- The windows build instructions have been greatly imptoved, based on the experiences of Jens Heilig <jens@familie-heilig.net> which he has generously shared.
- The documentation in the manual about sequence warnings has been improved. The **−disable-sequence-warnings** option must come before the input file on the command line. My thanks to Emil Gracic <emil_kruki@yahoo.com> for reporting this problem.

Version 1.57 (2011-Jun-09)

- The byte order of the fletcher16 output has been reversed.

- The meaning of the **–address-length** option has been change for the Intel output format. Previously, 2 meant using i16hex 20-bit segmented addressing, and >2 meant using i32hex extended addressing. This has been changed: a value of 2 requests i8hex 16-bit addressing, a value of 3 requests i16hex 20-bit segment addressing, and a value >=4 requests i32hex 32-bit addressing. My thanks to Stephen R. Phillips <srp@CSECorporation.com> for reporting the absence of i8hex support.
- The **–generate –repeat-string** option is now able to take a string that looks like a number as the text to be repeated. My thanks to Stephen R. Phillips <srp@CSECorporation.com> for reporting this problem.
- Luca Giancristofaro <luca.giancristofaro@prosa.com> discovered a WinAVR linker that is a sandwich short of a picnic: it generated non-conforming Intel hex end-of-file records. This is no longer an error, but only a warning.
- There were some problems with the RPM spec file, these have been improved. My thanks to Galen Seitz <galens@seitzassoc.com> for reporting this problem.

Version 1.56 (2010-Sep-15)

- A bug has been fixed in the MsBin output, it now concatenates records correctly, and calaulate checksums appropriately.
- It is now possible to ask the Fletcher 16 filter to give you a specific answer, and adjusting the checksum to achieve that result. It is also possible to specify different seed values for the sums.
- There is a new srec_cat **–enable=optional-address** option to cause output formats capable of omitting addresses, to omit a leading zero address, as those formats usually default the address to zero, if no address information is seen before the first data record. Defaults to false (disabled).
- There is a new *srec_cat*(1) **–output-block-packing** option, that may be used to pack output records even when they cross internal SRecord boundaries.
- There is a new *srec_cat*(1) **–output-block-size** so that you can specify the block size exactly, rather than implying it with the line length option.

Version 1.55 (2010-Feb-10)

- The Makefile.in has been improved, it now copes with non-standard `--prefix` options.
- The rpm.spec file has been improved, it now separates the commands, shared libraries and development files.

Version 1.54 (2010-Jan-29)

- There is now a shared library installed, including the necessary header files so that you can use all of the file formats and filters in your own projects.
- The license on the shared library code is GNU Lesser General Public License, version 3.0 or later.
- The code can cope with older versions of GNU Libgcrypt. In the case of very old versions, by ignoring it.
- A number of build problems have been fixed.

Version 1.53 (2009-Nov-10)

- There is a new MsBin (Windows CE Binary Image Data) file format, supported for both reading and writing.
- The lintian(1) warning about hyphen in the manual pages has been silenced, by careful use of –, - and – as appropriate. Sure makes some of the sources ugly, tho. The lintian(1) warning about the undefined .XX macro has been silenced, by making it conditional.
- The code will build without libgcrypt.

Version 1.52 (2009-Sep-17)

- There is a new *srec_cat −generator −l-e-constant* data generator (and also *−b-e-const*) that may be used to insert multi-byte constants into your data. See *srec_input*(1) for more information.

Version 1.51 (2009-Sep-13)

- A number of gcc 4.4 build problems have been fixed.
- A bugs has been fixed in the Intel output format. When using the segemented format (address-length=2) records that span the end of segment boundary are tricky. The code now carefully splits such output records, to ensure the two parts are explicitly placed into separate segments.

Version 1.50 (2009-Jul-09)

- The CRC16 code has been enhanced to provide low-to-high bit order, in addition to the previous high-to-low bit order. It is also possible to specify the polynomial, with the default the CCITT standard polynomial, as was in the previous code. See *srec_input*(1) for more information.
- The MD5, RipeMD-160, SHA1, SHA224, SHA256, SHA384, SHA 512 and Whirlpool message digests are now supported. See *srec_input*(1) for more information.
- There is a new *srec_cat −bit-reverse* filter, that may be used to reverse the bits in each data byte. See *srec_input*(1) for more information.

Version 1.49 (2009-May-17)

- A typo in the srec_input(1) man page has been fixed.

Version 1.48 (2009-Apr-19)

- There are new Fletcher Checksum filters, both 32-bits and 16-bits, both little-endian and big-endian.
- There are new Adler Checksum filters, both 32-bits and 16-bits, both little-endian and big-endian.

Version 1.47 (2009-Feb-19)

- Memory Initialzation File (MIF) format by Altera is now supported for reading and writing.

Version 1.46 (2009-Jan-13)

- There is a new option for the *−−x*-e-length filters, they can now accept a width, and this is divided into the byte lenght, so that you can insert the length in units of words (2) or longs (4).
- Some small corrections have been made to the documentation.
- The −minimum and −maximum options have been renamed −minimum-address and −maximum-address, to avoid a command line grammar syntax problem.

Version 1.45 (2008-Sep-30)

- A bug has been fixed in the *srec_cat*(1) command. You are now able to specify several inputs within parentheses, instead of just one. This allows filters to be applied to the concatenation of several inputs.
- The *srec_cat*(1) command is now able to write FORTH output.

Version 1.44 (2008-Aug-29)

- Some compilers issue a warning when const appears before extern. "warning: storage class is not first". The C output has been updated to conform to this expectation.
- The manual page for *srec_cat*(1) has been enhanced to describe the in-memory data model, and the resulting output data order.
- The **−motorola** optional width argument now produces a better error message when it is out of range.
- The **−fill** filter now checks the size, and fails for absurdly large fills, with a **−big** override if they really want >1GB fills.
- A bug in the .spec file for rpmbuild has been fixed, it now takes notice of $RPM_BUILD_ROOT
- There is a new **−line-termination** option, which may be used to select the desired line termination of output text files.

Version 1.43 (2008-Jul-06)

- The *srec-cat −data-only* option has been broken down into four separate controls. It is now possible to **−enable** and **−disable** individual features, such as “header”, “data-count”, “execution-start-address” and “footer”. See *srec_cat*(1) for more information.

- The *srec_cat −start-address* option has been renamed **−execution-start-address** to remove any confusion with the **−offset** filter. The documentation now explicitly explains the difference between the two.
- Examples of converting to and from binary files have been added to the *srec_examples*(1) man page.
- A bug has been fixed in the MOS Tech format, it now emits an end record even when there is no execution start address passed in.

Version 1.42 (2008-Jun-01)

- The MOS Technology format was not reading and writing end records correctly, this has been fixed. The name of the company has been corrected.
- Some examples of how to insert constant or scripted data into your EPROM load files have been added to the *srec_examples*(1) man page.

Version 1.41 (2008-May-12)

- False negative being reported by tests on Cygwin have been fixed.
- There are six new filters (−be-exclusive-length, −le-exclusive-length, −be-exclusive-maximum, −le-exclusive-maximum, −be-exclusive-minimum and −le-exclusive-minimum) which are very similar to their non-exclusive equivalents, except that they do not include the adress range covered by their output in their output.
- A bug has been fixed in the C word-array output. It was getting offsets and lengths wrong in some cases.
- A bug has been fixed in the generated C array header file, it no longer omits the section descriptor arrays.
- A problem with building RPM packages with the names of the executables in the .spec file has been fixed, and the BuildRequires has been updated.

Version 1.40 (2008-Mar-13)

- An RPM build problem has been fixed.
- The dependency on the Boost library is now documented in the BUILDING file.
- Some build problems with g++ 4.3 have been fixed
- A bug has been fixed in the calculation of ranges on the command line, it no longer goes into an infinite loop for "−fill 0xFF −over { foo.hex −exclude −within foo.hex }" construct, which should have been calculating an empty fill set, but was instead calculating a 4GB fill set.
- The CRC32 filters now take an −xmodem option, to use an xmodem-like (all bit zero) initial state, rather than the default CCITT (all bits on) initial state.

Version 1.39 (2008-Feb-04)

- A bug has been fixed in the use of parentheses to group filters and override the default precedences.

Version 1.38 (2008-Jan-14)

- The CRC16 filters now support a −Broken option, to perform a common-but-broken CRC16 calculation, in addition to the CCITT and XMODEM calculations.
- A link has been added to the CRC16 man page section to the www.joegeluso.com/software/articles/ccitt.htm web page, to explain the difficulties in seeding CRC16 calculations.
- A buglet has been fixed in the *srec_motorola*(5) man page, it now includes `S6` in the list of things that can appear in the type field.
- The ability to negate expressions is now mentioned in the *srec_examples*(1) man page.

Version 1.37 (2007-Oct-29)

- It is now possible to have negative expressions on the command line, to facilitate "−−offset − −minimum foo" usages.
- The *srec_cat*(1) command now has a simple hexadecimal dump output format.

- The use of *uudecode*(1) in the tests has been removed, so *sharutils* is no longer a build dependency.

Version 1.36 (2007-Aug-07)

- A bug has been fixed in the CRC-16 CCITT calculation; the algorithm was correct but the start value was incorrect, leading to incorrect results.
- The CRC16 filters have a new −−no-augment option, to omit the 16 zero bits augmenting the message. This is not CCITT standard conforming, but some implementations do this.
- A problem has been fixed in the generated Makefile.in file found in the tarball.
- The license has been changed to GNU GPL version 3.

Version 1.35 (2007-Jun-23)

- A major build problem with the generated makefile has been fixed.

Version 1.34 (2007-Jun-22)

- The C and ASM output formats have been improved in the word mode.
- Several build problems have been fixed.

Version 1.33 (2007-May-18)

- More examples have been added to the documentation.
- It is now possible to perform set intersection and set difference on address ranges on the command line.
- There is a new category of data source: generators. You can generate constant data, random data and repeating data.
- The assembler and C-Array outputs now support additional options to facilitate MSP430 systems. They can also optionally write shorts rather than bytes.
- You can now round address ranges on the command line to be whole multiples of a number of bytes.

Version 1.32 (2007-Apr-24)

- The TI-TXT format output has been improved; it is less spec conforming but more reality conforming. It now allows odd alignment without padding. It also ends with a `q` instead of a `Q`.
- The warning for odd input addresses has been dropped. The spec didn't like them, but the MSP430 handles them without a hiccup.

Version 1.31 (2007-Apr-03)

- The Verilog format now suppresses comments when you specify the −−data-only option.
- The Texas Instruments ti-txt (MSP430) format is now understood for reading and writing.

Version 1.30 (2007-Mar-21)

- The ascii-hex output format has been improved.
- The ti-tagged 16-bit format is now understood for reading and writing.
- The Intel format no longer warns about missing optional records.
- A bug in the ti-tagged format has been fixed, it now understands the '0' tag.

Version 1.29 (2007-Mar-13)

- A serious bug has been fixed in the generated Makefile.

Version 1.28 (2007-Mar-08)

- It is now possible to read and write files in the Freescale MC68EZ328 Dragonball bootstrap b-record format

Version 1.27 (2006-Dec-21)

- [SourceForge Feature Request 1597637] There is a new warning issued when input data records are not in strictly ascending address order. There is a new command line option to silence the warning.
- [SourceForge Feature Request 1592348] The command line processing of all srecord commands now understands `@`*file* command line options, filled with additional space separated strings witch will be treated as of they were command line options. This gets around absurdly short command line length

limits in some operating systems.

Version 1.26 (2006-May-26)

- It is now possible to place parentheses on the command line in more places to clarify your intent.
- This change prepares SRecord for the next public release.

Version 1.25 (2006-May-18)

- The assembler output has been enhanced to produce ORG directives, if necessary, to change the data address.
- The *srec_cat*(1) command now only writes an execution start address into the output if there was an execution start address present in the input.

Version 1.24 (2006-Mar-08)

- Additional information has been added to the lseek error when they try to seek to addresses >= 2**31
- The CRC 16 filters have been enhanced to accept an argument to specify whether CCITT or XMODEM calculations are to be performed.

Version 1.23 (2005-Sep-23)

- A segfault has been fixed on x86_64 when running the regression test suite.
- A compile problem with the lib/srec/output/file/c.cc file has been fixed.

Version 1.22 (2005-Aug-12)

- The **−byte-swap** filter now has an optional *width* argument, to specify the address width to swap. The default is two bytes.
- The motorola file format now accepts an additional 'width' command line argument, so you can have 16-bit and 32-bit address multiples.
- A bug has been fixed in the VMEM output format. It was failing to correctly set the next address in some cases. This fixes SourceForge bug 1119786.
- The −C-Array output format now uses the `const` keyword by default, you can turn it off with the −no-const option. The −C-Array output format can now generate an additional include file if you use the −INClude option. This answers SourceForge feature request 942132.
- A fix for the "undefined symbols" problem when using g++ 3.x on Cygwin and MacOsX has been added to the ./configure script.
- There is a new −ignore-checksum command line option. The −ignore-checksums option may be used to disable checksum validation of input files, for those formats which have checksums at all. Note that the checksum values are still read in and parsed (so it is still an error if they are missing) but their values are not checked.

Version 1.21 (2005-Feb-07)

- More Doxygen comments have been added to the class header files.
- There is a new *srec_cat −−crlf* option, which may be used for force CRLF output on operating systems which don't use that style of line termination.
- A number of problems with GCC, particularly with the early 3.*x* series.
- There is a new "Stewie" format, an undocumented format loosely based on the Motorola S-Record format, apparently used in mobile phones. More information would be most welcome.
- A number of build problems have been fixed.

Version 1.20 (2004-Feb-08)

- The AOMF format now accepts (and ignores) more record types.

Version 1.19 (2004-Jan-03)

- It is now possible to set the execution start address in the output using the *srec_cat −Execution_Start_Address* command line option.

- The Intel Absolute Object Module Format (AOMF) is now supported for reading and writing.
- There is a new *srec_cat −Random_Fill* filter, like the *srec_cat −Fill* filter except that it uses random values.

Version 1.18 (2004-Jan-01)

- The VMEM format is now able to output data for 64 and 128 bits wide memories.
- A bug in the SRecord reference manuals has been fixed; the CRCxx had a copy-and-paste glitch and always said big-endian where little endian was intended half the time.

Version 1.17 (2003-Oct-12)

- There is now support for Intel Extended Segment addressing output, via the −−address-length=2 option.
- There is now support for output of Verilog VMEM format. See *srec_vmem*(5) for more information.
- There is now support for reading and writing the INHX16 format, used in various PIC programmers. It looks just like the Intel Hex format, except that the bytes counts and the addresses refer to words (hi,lo) rather than bytes. See *srec_intel16*(5) for more information.

Version 1.16 (2003-Jul-28)

- Some updates have been made to cope with GCC 3.2

Version 1.15 (2003-Jun-16)

- The ASCII-Hex implementation is now slightly more complete. I still haven't found a definitive description.
- The Fairchild Fairbug format has been added for reading and writing. See *srec_fairchild*(5) for more information.
- The Spectrum format has been added for reading and writing. See *srec_spectrum*(5) for more information.
- The Formatted Binary format has been added for reading and writing. See *srec_formatted_binary*(5) for more information.
- The RCA Cosmac Elf format has been added for reading and writing. See *srec_cosmac*(5) for more information.
- The Needham EMP programmer format has been added for reading and writing. See *srec_needham*(5) for more information.

Version 1.14 (2003-Mar-11)

- Numerous fixes have been made to header handling. It is now possible to specify an empty header with the `-header` command line option.
- Some more GCC 3.2 build problems have been fixed.

Version 1.13 (2003-Feb-05)

- Bugs have been fixed in the Texas Instruments Tagged and VHDL formats, which produced inconsistent output.
- A couple of build problems have been fixed.
- There are two new output formats for ASM and BASIC.

Version 1.12 (2002-Dec-06)

- It is now possible to put **−minimum** *input.spec* (also **−maximum** and **−length**) almost anywhere on the command line that you can put a number. It allows, for example, the −offset value to be calculated from the maximum of the previous file. The values calculated by **−Minimum**, **−Maximum** and **−Length** may also be rounded to arbitrary boundaries, using **−Round_Down**, **−Round_Nearest** and **−Round_Up**.
- The malformed Motorola S5 records output by the Green Hills tool chain are now understood.

Version 1.11 (2002-Oct-21)

- The Ohio Scientific OS65V audio tape format has been added for reading and writing. See *srec_os65v*(5) for more information.

- Some build problems have been fixed.

Version 1.10 (2002-Jun-14)

- The Intel format now emits the redundant extended linear address record at the start of the file; some loaders couldn't cope without it.
- The Binary format now copes with writing to pipes.
- The Motorola format now understands the S6 (24-bit data record count) records for reading and writing.
- The DEC Binary format now works correctly on Windows machines.
- The LSI Logic Fast Load format is now understood for both reading and writing. See *srec_fastload*(5) for more information.

Version 1.9 (2001-Nov-27)

- The DEC Binary (XXDP) format is now understood for both reading and writing. See *srec_dec_binary*(5) for more information.
- The Elektor Monitor (EMON52) format is now understood for both reading and writing. See *srec_emon52*(5) for more information.
- The Signetics format is now understood for both reading and writing. See *srec_signetics*(5) for more information.
- The Four Packed Code (FPC) format is now understood for both reading and writing. See *srec_fpc*(5) for more information.
- Wherever possible, header data is now passed through by *srec_cat*(1). There is also a new *srec_cat −header* option, so that you can set the header comment from the command line.
- The Atmel Generic format for Atmel AVR programmers is now understood for both reading and writing. See *srec_atmel_generic*(5) for more information.
- The handling of termination records has been improved. It caused problems for a number of filters, including the −fill filter.
- A bug has been fixed in the checksum calculations for the Tektronix format.
- There is a new SPASM format for PIC programmers. See *srec_spasm*(5) for more information.

Version 1.8 (2001-Apr-20)

- There is a new "unfill" filter, which may be used to perform the reverse effect of the "fill" filter.
- There is a new bit-wise NOT filter, which may be used to invert the data.
- A couple of bugs have been fixed in the CRC filters.

Version 1.7 (2001-Mar-19)

- The documentation is now in PDF format. This was in order to make it more accessible to a wider range of people.
- There is a new *srec_cat −−address-length* option, so that you can set the length of the address fields in the output file. For example, if you always want S3 data records in a Motorola hex file, use the `--address-length=4` option. This helps when talking to brain-dead EPROM programmers which do not fully implement the format specification.
- There is a new *−−multiple* option to the commands, which permits an input file to contain multiple (contradictory) values for some memory locations. The last value in the file will be used.
- A problem has been fixed which stopped SRecord from building under Cygwin.
- A bug has been fixed in the C array output. It used to generate invalid output when the input had holes in the data.

Version 1.6 (2000-Dec-03)

- A bug has been fixed in the C array output. (Holes in the input caused an invalid C file to be produced.)

- There is are new CRC input filters, both 16-bit and 32-bit, both big and little endian. See *srec_cat*(1) for more information.
- There is a new VHDL output format.
- There are new checksum filters: in addition to the existing one's complement (bit not) checksum filter, there are now negative and positive checksum filters. See *srec_cat*(1) for more information.
- The checksum filters are now able to sum over 16-bit and 32-bit values, in addition to the existing byte sums.
- The *srec_cmp* program now has a **−−verbose** option, which gives more information about how the two inputs differ. See *srec_cmp*(1) for more information.

Version 1.5 (2000-Mar-06)

- There is now a command line option to guess the input file format; all of the tools understand this option.
- The "MOS Technologies" file format is now understood for reading and writing. See *srec_mos_tech*(5) for more information.
- The "Tektronix Extended" file format is now understood for reading and writing. See *srec_tektronix_extended*(5) for more information.
- The "Texas Instruments Tagged" file format is now understood for reading and writing. (Also known as the TI-Tagged or SDSMAC format.) See *srec_ti_tagged*(5) for more information.
- The "ascii-hex" file format is now understood for reading and writing. (Also known as the ascii-space-hex format.) See *srec_ascii_hex*(5) for more information.
- There is a new *byte swap* input filter, allowing pairs of odd and even input bytes to be swapped. See *srec_cat*(1) for more information.
- The "wilson" file format is now understood for reading and writing. This mystery format was added for a mysterious type of EPROM writer. See *srec_wilson*(5) for more information.
- The *srec_cat* program now has a **−data-only** option, which suppresses all output except for the data records. This helps when talking to brain-dead EPROM programmers which barf at anything but data. See *srec_cat*(1) for more information.
- There is a new *−Line-Length* option for the *srec_cat* program, allowing you to specify the maximum width of output lines. See *srec_cat*(1) for more information.

Version 1.4 (2000-Jan-13)

- SRecord can now cope with CRLF sequences in Unix files. This was unfortunately common where the file was generated on a PC, but SRecord was being used on Unix.

Version 1.3 (1999-May-12)

- A bug has been fixed which would cause the crop and exclude filters to dump core sometimes.
- A bug has been fixed where binary files were handled incorrectly on Windows NT (actually, any system in which text files aren't the same as binary files).
- There are three new data filters. The −−OR filter, which may be used to bit-wise OR a value to each data byte; the −−AND filter, which may be used to bit-wise AND a value to each data byte; and the −−eXclusive-OR filter, which may be used to bit-wise XOR a value to each data byte. See *srec_cat*(1) for more information.

Version 1.2 (1998-Nov-04)

- This release includes file format man pages. The web page also includes a PostScript reference manual, containing all of the man pages.
- The Intel hex format now has full 32-bit support. See *srec_intel*(5) for more information.
- The Tektronix hex format is now supported (only the 16-bit version, Extended Tektronix hex is not yet supported). See *srec_tektronix*(5) for more information.

- There is a new *split* filter, useful for wide data buses and memory striping, and a complementary *unsplit* filter to reverse it. See *srec_cat*(1) for more information.

Version 1.1 (1998-Mar-22)
First public release.