

ee9 RELEASE HISTORY

RELEASES

A. V0.1, 04-NOV-2010: FIRST ALPHA RELEASE.

B. V0.2, 24-NOV-2010: SECOND ALPHA RELEASE.

- a. I advanced the version number to 0.2.
- b. I fixed an inconsistency, between fast mode and other diagnostic modes, in the measurement of the KDF9 virtual CPU time consumed when running a program.
- c. I improved the declarations of some of the fundamental types in `KDF9.ads`.
- d. I provided a fail-safe way of dealing with programs that corrupt their EOU, so avoiding difficulties for postmortem diagnostic printing in U format.
- e. I fixed a bug in the reconstruction of normalised floating-point results from separate fraction and exponent parts.
- f. I added a facility to supply multiple 'rolls of tape' to a tape reader.

C. V0.3B, 20-JAN-2011: THIRD ALPHA RELEASE.

- a. I advanced the version number to 0.3, and the copyright date to 2011.
- b. I added a facility to provide a pre-prepared set of replies to console Flexowriter prompts, with the facility to make conditional responses. (This nods at the edge-punched card reader of the console Flexowriter.) An audible warning is given (i.e., BEL characters are output) when manual Flexowriter input from the user is needed.
- c. I enhanced the setting-file feature to provide separate settings for a first program (e.g., the Whetstone Translator) and overlay (e.g., the Controller).
- d. I much reduced **ee9**'s verbosity by omitting most confirmatory messages, and added the option to suppress even the output of the final KDF9 state.
- e. I fixed the timing of I/O transfers that are on-going at the end of a run, and of I/O transfers that are held up by store or buffer lockouts.
- f. I fixed a major bug that was detected by Brian Wichmann in the 'xD' order.
- g. I fixed the implementation of '-D', which was adding the top 48 bits of its operands instead of subtracting them. This had gone quite unnoticed until I called it as a subroutine in the new implementation of 'xD'!
- h. I fixed the usage-error message from **ee9**, given in response to incorrect flag parameters.
- i. I decoupled the generation of digital signatures from the running trace mode, thus making signature generation a more practical means of verifying correct operation by an **ee9** port or new version.
- j. I unified the conditions under which retrospective tracing, digital signature generation, and running trace output, are all activated.
- k. I improved the shell commands in `Testing`, to make running **ee9** easier, less error prone, and better diagnosed in the case of an error.
- l. I provided a HOWTO file that gives more complete instruction on the operating procedures to be followed for compiling and running KDF9 programs.
- m. I implemented a first cut at an automatic self-testing process to check the correct operation of an installation or new port of **ee9**.
- n. I simplified the handling of the diagnostic mode options, both internally and *vis-a-vis* the user-interface semantics; the latter now make some sense.
- o. For revision 0.3b I improved the installation and self-testing code, in line with feedback from David Holdsworth.

D. V0.4, 15-FEB-2011: FOURTH ALPHA RELEASE

- a. I advanced the version number to 0.4, and added a message to identify the version of **ee9** that is being used.
- b. I improved the encapsulation of the **IO** package, by making **IO.stream** a private type.
- c. I made **IO.stream** a tagged type, allowing prefix notation, so that component access could be re-interpreted as a call on a 'getter' function.
- d. I tidied up the diagnostic-mode confirmation message.
- e. I forced runtime checks in all modules, except some in the **kdf9** hierarchy; this greatly improves diagnosability, yet has no perceptible impact on the performance of an optimised build.
- f. I tidied up the contents of the **Testing** directory by introducing the **Binary** subdirectory for KDF9 machine code programs, and by renaming **Usercode** as **Assembly**. This puts all the subdirectories together in a listing.
- g. I created a command, **tsd**, to simplify running the **Time Sharing Director**.
- h. I arranged for the **tsd**, **nine**, and **whet** commands to set the **FW0** file suitably for their own type of program run, restoring it on termination to its former contents.
- i. I tidied up **IOC** a little.
- j. I moved digital signature generation into **kdf9**, where it properly belongs.
- k. I factored mode testing, and logging, out of the **check_*_points** procedures, gaining an 8% speedup in fast mode and no slowdown in other modes.
- l. I restricted the signature hashing to the two possibly-relevant **Q** stores, instead of all 16. This speeds up a run of **ee9_self_test** by a factor of 2.5!
- m. I added a facility to include Line Shift and Page Change characters in Flexowriter prompts, using the **Latin_1** characters ® and © respectively.
- n. I implemented the (EE model 1081) **MT** orders, with a (hopefully temporary) restriction that blocks are limited to a maximum of 512 KDF9 words in length.
- o. I corrected the timing of Flexowriter writes that include fillers and/or inject canned responses to prompts.
- p. I implemented **OUTs** 4 and 10, to let problem programs such as Whetstone Algol claim magnetic tapes as they would when running under Director.
- q. I made a version of the Whetstone Benchmark, called **MTW**, that outputs to a **MT** labelled **WHETLIST**, instead of **OUT 8 stream 30**.
- r. I included four **MT**-handling programs in the self-testing procedure. They are **RLT** (Rewind and Label Tape); **TRB** (Test Read Backwards); **OUT4** (tests the new **OUT**); and **OUTX** (tests **OUT 10**). I also included **MTW**.

E. V0.5, 15-MAR-2011: FIRST BETA RELEASE

- a. I advanced the version number to 0.5; this is the first beta release, the change in status being justified by the lack of bug reports from heavy users.
- b. I fixed previously undetected bugs in '-DF' and 'x+±n'.
- c. I provided a lax nest-depth checking option (option **N**), to better approximate the behaviour of the KDF9 hardware when the nest is under-filled.
- d. I fixed bugs in the nest-depth checking of some 1-syllable and 2-syllable orders.
- e. I improved the promptness of time-limit expiry in fast mode: it is now put off to the next successful jump, rather than to the end of the time slice.
- f. I improved the discrimination of the digital signature by hashing **ICR** into it.
- g. I speeded **ee9** up by ~20% over V0.4: by judicious in-lining of store access primitives, by refactoring the instruction decoding, by removing some redundancy from several **Q**-store orders, and by implementing a better method of ensuring that **Q0** is always 0; all this with no adverse effect on maintainability or modularity.
- h. I replaced the **make** command as a build tool with a shell command file called **Build/mk9**, which takes the same parameter options as the **makefile** formerly did.

- i. I added a section on building **ee9** to the HOWTO file; and included URLs for EE Director documentation and the EE KDF9 Algol Manual.
- j. I made pop operations zeroize the affected nest cell(s), as the hardware did.
- k. I changed the V option flags to a more mnemonically consistent set: **DHIJPS**.

F. V1.0v, 21-APR-2011: FIRST PUBLIC RELEASE

- a. I advanced the version number to 1.0: **ee9** goes forth into the world!
- b. I fixed previously undetected bugs in 'NEGD' and '-D'. Thanks to the latter, and to David Holdsworth's discovery of a mis-transcription in the Whetstone Controller's Usercode, Walgol numeric input now works properly.
- c. I completely revamped the tracing feature, replacing the jump and skip traces with a single, consolidated retrospective trace of all instructions executed while tracing is enabled. Each traced instruction is accompanied by its most relevant operand (see the User Guide for details).
- d. In conformity with (c), I changed the trace mode specifiers from **JUMP_MODE** and **SKIP_MODE** to **RETRO_MODE**; and the verbosity (option V) flags from D to S, and from J/S to R; again, with further improvement in mnemonic quality.
- e. I speeded **ee9** up by ~10% over V0.5 by removing all tracing overhead from the fast mode code path. Tracing now invokes the fast mode code as a subroutine, wrapping it in calls that determine the operand to be traced. As well as being faster, this is more modular, since the details of tracing can now be changed without touching the fundamental emulation code.
- f. I wrote a program, **Algol/input.a60** to test Algol numerical input; and included it in the self-testing procedure.
- g. I added a verbosity option to the command-line parameters of **ee9** (see the Users' Guide), **whet**, **nine**, **dow**, and **tsd**. This allows the tracing verbosity to be set without needing a V line to be put into a settings file.
- h. I changed the self-test procedure to use the new verbosity option, instead of setting up a V line in the settings files.
- i. I reverted change (e) of V0.5, as its adverse effect on performance became more onerous as a result of V1.0's increases in overall efficiency.
- j. I corrected the absence of a final **OUT 0** or **OUT 1** order from the tracing.
- k. I simplified the self-test logs by omitting the 'last output line' feature.
- l. I improved the layout of tracing output, reducing the screen-width needed to view it without wrapping lines.
- m. I removed the facility to monitor locations changed by peripheral input.
- n. I allowed for separate post-run dumps for a program and its overlay (e.g., the Whetstone Translator and Controller).
- o. I added a warning to **ee9_self_test**, re: overwriting of I/O device files.
- p. I rewrote the circular and double-length shift orders in conformity with EE engineering document K/GD.y.80, "KDF 9: SHIFTING AND SHIFT CONTROL".
- q. I rewrote **KDF9.as_Q()** and **KDF9.as_word()** as normal functions, avoiding a portability problem revealed by the MacOS X PPC Leopard port (with thanks to Mike Hore).
- r. I rewrote the way **ee9** ensures that Q0 is never set to a non-zero value. In **LAX_MODE** an assignment to Q0 is now suppressed; in **STRICT_MODE** it is now treated as an execution error.

G. V1.5w, 01-AUG-2011: SECOND PUBLIC RELEASE

- a. I advanced the version number from 1.0v to 1.5w, because adding support for Microsoft Windows seems like quite a big step. I am mindful that LISP never got past its version 1.5, but hope that there will be V2 of **ee9**!
- b. I corrected the instruction timings for `=R]{QCIM}q` type orders; as a result the **ee9** timing for the Whetstone benchmark is more accurate: 420.7s, down from 422.7s, the real KDF9's measured time being about 417s.
- c. I added a test, in trace mode cycles only, to verify that `Q0 = Q0/0/0` at the end of each instruction execution.
- d. I discovered that, with GNAT GPL 2011, optimised builds with all runtime checking suppressed run little (~6%) faster than with most checks enabled. Now the default **ee9** build is optimised, with runtime checking.
- e. I added new verbosity flag options: **A** to suppress the output of messages when **ee9** services an OUT API successfully; **E** to suppress the output of mere progress messages from **ee9**; **F** to suppress the output of the KDF9's **F**inal state; and **Z** to suppress **all** output from **ee9** itself, rather than KDF9.
- f. I amended the Flexowriter emulation package to use separate input and output file handles. This was necessitated by Microsoft Windows, but has been done in a way that is completely source-code compatible with UNIX systems.
- g. I provided alternative bodies for `open_ui ()` that opens the terminal user interface: one for Windows and another for UNIX systems.
- h. I amended `IO.open ()` to ensure that I/O transfers are transparent to all byte values. This is necessary on Windows to allow KDF9 binary machine code to be read uncorrupted, but has no effect on UNIX systems.
- i. I recast the **mk9** shell command to better support builds of **ee9** for Windows.
- j. I amended the shell files in **Testing** to eschew the use of file system links, which do not work well under Windows (to put it mildly).
- k. I amended the way **ee9** ensures `Q0` is never set to a non-zero value so that, in Director state, an assignment to `Q0` is suppressed even in **STRICT_MODE**.
- l. I provided separate subroutines for Windows and for others, to isolate the change of font colour in Flexowriter I/O. In this first version, the colour is **not** changed on Windows.
- m. I amended `actual_prompt ()` to cope with the presence and/or the absence of ANSI-terminal font colour escape sequences.
- n. I consolidated the Windows amendments for Flexowriter output into a single package, **terminal**. Alternative bodies for **terminal** are selected by the OS (second) parameter in a build call to **mk9**.
- o. I enhanced **mk9** to take a parameter indicating whether a Windows or a non-Windows OS build is wanted. Several other options have been changed, in name and effect (see the HOWTO file, 'Building Your Own Version of **ee9**').
- p. I changed **ee9_self_test**, and its ancillary routines, to make them compatible with Windows/Cygwin.
- q. I updated all the documentation to take account of Windows deployment, and I added a new Appendix to the **ee9** Users' Guide, giving various examples of the use of **ucc**, **nine**, **whet** and **tsd**.
- r. I converted all the documentation that was formerly in simple text files into PDFs, as the easiest way to circumvent the—eccentric—handling of plain data in Windows.

H. V1.9e, 10-DEC-2011: THIRD PUBLIC RELEASE

- a. I advanced the version number from 1.5w to 1.9e, in view of the large number of improvements.
- b. I added a feature, `AUTHENTIC_TIME_MODE`, whereby the real elapsed time is made to approximate the elapsed time that would have been taken by a real KDF9 (by inserting timed pauses at suitable points during the run).
- c. I changed the N option designator to A, for ‘authenticity’, and added `AUTHENTIC_TIME_MODE` to its options.
- d. I reconceptualised ‘verbosity’ mode as ‘visibility’ mode, and added two new visibility flag options: D to request that optional debugging messages be output, and T to request that `AUTHENTIC_TIME_MODE` be set.
- e. I fully implemented the TSD’s device (de)allocation OUTs, using CPDAR properly to prevent access, in problem program mode, to unallocated devices.
- f. I improved the way the FW package deals with prompts, with a resulting simplification of the two `OS_specifics` packages.
- g. I amended the FW package to support authentic timing of typewriter output.
- h. I provided implementations of the `P[IO][ABCD]` and `PM[ADF]` orders for the fixed disc drive, based on information from David Holdsworth, and on hypotheses developed by analysing P198 in the Eldon 2 Director and further by extrapolating from the Data Products 5022 disc drive manual and the I.C.T. 1956 disc drive flyer.
- i. I enhanced the IO package to support fixed disc emulation and authentic timing of slow device output.
- j. I improved the traced operand value for instructions that update a Qk, as opposed to a Qq, operand.
- k. I tidied up the procedures that deal with LIVs.
- l. I rephrased the error message produced by a parity error, to make it clearer that it is probably only due to reading past the end of data.
- m. I added support for ‘character’ (in reality, *one character per word*) transfers to the TR, TP and FW devices, so that PIC, PID, POC and POD are now fully implemented by the two-shift devices.
- n. I corrected the timing of transfers on two-shift devices, allowing properly for shift characters, for filler characters, and for the fact that FW0 has separate input and output streams.
- o. I improved the realism with which short loops and short-loop jumps are timed, removing some ugliness in the process. The CPU time is now monotonically non-decreasing.
- p. I implemented POF for the TP, on the basis of the only reasonable hypothesis I can make of its action.
- q. I implemented all documented orders for the CR and CP devices.
- r. I added a card-code dumping format, code C; a lineprinter-code dumping format, code L; separate Case Normal and Case Shift formats, codes N and S; and a Latin-1/ASCII dumping format, code A, intended to facilitate the use of ‘character’ transfers with Latin-1 data.
- s. I somewhat tidied the `state_display` package, which ‘just grew’. Ditto, the various IOC packages.
- t. I changed the Qp terminology to Qk, for consistency with the Manual and KAL4.
- u. I radically reworked the instruction-fetch logic in order to remove a kluge in the implementation of the `=EeQq` instruction. (It was needed to prevent Director from overwriting the executing instructions in a `JrCqNZS` loop that zeroizes the whole of core.) Instead, I implemented a much closer approximation to the logic and the instruction buffer registers (IWB0 and IWB1) of Main Control. Partly in consequence, the virtual CPU time for the Whetstone Benchmark now much more closely approximates the actual measured time.
- v. I added a new **test program mode** of execution, which operates like problem program mode, but runs the code in Director state. This is not authentic, but is very useful for executing ‘hardware’ test programs. I added a **nine_test** command to the `Testing` directory, to make running in test program mode more convenient, and changed the `ee9_self_test` command to use **nine_test** in place of **nine** for the MT test programs.
- w. To better exploit the new device allocation feature (point e, above), tape reader 0 is detached from the binary program file after the (initial or overlay) program load, and re-attached to the TR0 text file. This makes it possible to run a program that has two separate paper tape input streams.
- x. I corrected various bugs in the SHAD and divide orders, revealed by David Holdsworth when running KAL4. I took this opportunity to tidy up the code for shifts, multiplications, and divisions.
- y. I now set E7 of test-programs and problem-programs to a string, in the format DD/MM/YY, the date exactly 28 years ago; so that 1962 < 19YY < 2000, and 19YY has the same date/day-of-the-week mapping as this year.
- z. I corrected OUT 9 to deliver the time-of-day (instead of the elapsed time), and implemented OUT 17 to return both the CPU time and the ‘notional’ elapsed time.
- aa. I corrected a bug in `INTQq`, which was limiting the hold-up to 1 virtual elapsed second in non-boot modes.

H. V2.0R, 30-JUN-2015: BETA RELEASE OF V2

- a. I advanced the version number from 1.9e to 2.0r, to indicate the first release of a major new mark that includes graph plotting; updated the copyright date to 2015; and updated the acceptable system name parameters of **mk9**.
- b. I changed the way that OUTs are logged in the retrospective traces, to make OUT 8 more conspicuous and its parameters more perspicuous. Small integers (OUT numbers < 64) are logged in decimal; large integers (parameters > 2⁴⁷) are logged in octal; and other values are logged in Q store format.
- c. I amended the behaviour of OUT 8 when addressing a Flexowriter, to conform more closely to the behaviour of the TSD as notified to me by David Holdsworth. Each message is now truncated to 8 words, if need be; embedded Line Shift and Horizontal Tab characters are disallowed; “;” characters must be in bits 42-47 and must not be in the last word of the transfer; but anything after an End Message can safely be ignored. Failures are logged as error “730”, in the manner of the TSD. A correct message is always typed on a fresh line, a new line being taken if the typing position is not there already.
- d. I corrected some minor portability problems in the **mk9** script and added support for the Raspberry Pi platform under Raspbian Linux.
- e. I added a new **non-interactive mode**, set with the new option **N**, to adapt **ee9** to runs invoked by a command script. In this mode it is not possible to supply responses to prompts, whether from the KDF9 program or from **ee9** itself; so if an interactive input is requested in non-interactive mode, **ee9** terminates with a suitable diagnostic message. Non-interactive mode can also be set with the command line miscellany parameter.
- f. I provided an implementation of the model 564 Calcomp graph plotter, as described in Appendix 6, §5, p.302 of the Manual. There was provision on the KDF9 to switch a buffer manually between a tape punch and a graph plotter; in **ee9** this is done with a new settings file option, **G**. If G is given, GP0 replaces TP1. This can also be done with a miscellany parameter including **g** on the **ee9** command line.

The drawing is output to the GP0 file in the form of Encapsulated PostScript. I expect any competent modern OS to have a utility that will open such a file and display the plot on screen.

- g. It was possible to fit pens with a variety of ink colours and ball-point tip sizes to the plotter. To simulate this, option G can set the pen colour from the list: Black (the default), Blue, Brown, Cyan, Dark_Blue, Dark_Cyan, Dark_Green, Dark_Grey, Dark_Magenta, Dark_Red, Green, Grey, Magenta, Red, White, and Yellow; and, additionally, can set the pen tip size from the list: Extra_Extra_Fine (the default, one plotting step wide), Extra_Fine, Fine, Medium, Medium_Broad, Broad, Extra_Broad.
- h. I implemented OUT 5 for GP0, taking the device type code to be #20, as stated in “Order Code Notes 18- Further OUTs” and consequently changed the output format of device type codes from decimal to octal.
- i. The Usercode program GPT.k3 in the Assembly directory runs a test on the plotter. The Usercode program TR2GP.k3 in the Assembly directory copies a file of plotting commands from a tape reader to the plotter.
- j. I added a new mode, **D**, to pseudo-Usercode core printing, such that operand and jump target word addresses are shown in decimal instead of octal, for ease of use with other software that works with addresses in this format.
- k. I provided decimal interpretations of the internal registers, SJNS and Q store in the final state display, for use with the previous facility.
- l. I added a decimal Q store interpretation of whole words in the NEST and core prints, and some whitespace to make for easier reading of their listings.
- m. I modified the syntax of options with address parameters in the settings files to allow both octal and decimal addresses to be specified, octal addresses beginning with ‘#’, as before.
- n. I removed the largely ineffectual checking of the miscellany parameter from **nine**, **nine_test**, **tsd** and **whet**; the checking is now done by **ee9** itself.
- o. I made a number of other minor improvements, e.g. to logging and error messages; and renamed the external tracing file as **trace.txt** instead of **full_trace.txt**.
- p. I implemented the PHU store.
- q. I modified the command line syntax for better mnemonic significance.
- r. I refactored FD support packages to avoid a cyclical dependency.
- s. I added a clearer interpretation of the parameter of an FD order in instruction traces.
- t. I implemented CRLF line terminators for text-file output under MS Windows.

TO DO:

Make it possible to choose between Latin_1 and KDF9 code for two-shift devices, on a file-by-file basis.