Bibliography and Appendices

Bibliography

The main bibliography is arranged alphabetically by author, and all references in the preceding discussion are to this list. However, since so many macro processors are known by their name rather than by their authors, a subsidiary list of references is provided. The subsidiary list cross-references names of macro processors with the entries on the main reference list for the papers describing them. It also gives a very brief description of each.

Main List of References

1. Arden, B.W., Galler, B.A and Graham, R.M., *Michigan Algorithm Decoder*, Univ. of Michigan Press, Ann Arbor, Michigan, 1966.
2. Bennett, R.K. and Neumann, H.D., Extension of existing compilers by the sophisticated use of macros, *Comm. ACM* **7**, 9 (Sept. 1964), 541–542.
3. Brooker, R.A. and Morris, D., A general translation program for phrase structure languages, *J. ACM* **9**, 1 (Jan. 1962), 1–10.
4. Brown, P.J., The ML/I Macro Processor, *Comm. ACM* **10**, 10 (Oct. 1967) 618–623.
5. Brown, P.J., *The ML/I User's Manual*, 3rd edn., University Mathematical Laboratory, Cambridge, June 1967.
6. Cheatham, T.E., The introduction of definitional facilities into higher level programming languages, *Proc. AFIPS 1966 Fall Joint Computer Conference*, Vol. 29, 623–637.
7. Conway, M.E., Design of a separable transition-diagram compiler, *Comm. ACM* **6**, 7 (July 1963), 396–408.
8. Dellert, G.T., Jr., A use of macros in translation of symbolic assembly language of one computer to another", *Comm. ACM* **8**, 12 (Dec.1965), 742–748.
9. Elcock, E.W. and Wilson, D.M., *803 Macro-generator*, University of Aberdeen, Nov. 1965.
10. Ershov, A.P. and Rar, A.F., SYGMA — A symbolic generator and macro-assembler, *Proc. IFIP Working Conference on Symbol Manipulation Languages*, Pisa, Sept. 1966.
11. Ferguson, D.E., The evolution of the meta-assembly program, *Comm. ACM* **9**, 3 (March 1966), 190–193.
12. Fletcher, J.G., A program to solve the pentomino problem by the recursive use of macros, *Comm. ACM* **8**, 10 (Oct. 1965), 621–623.
13. Freeman, D.N., Macro language design for System/360, *IBM Systems J.* **5**, 2 (1966), 63–77.
14. Galler, B.A. and Perlis, A.J., A proposal for definitions in ALGOL, *Comm. ACM* **10**, 4 (Apr. 1967), 204–219.
15. Garwick, J.V., *A General Purpose Language (GPL)*, Intern Rapport S-32, Norwegian Defence Research Establishment, Kjeller, Norway, June 1967.
16. Graham, M.L. and Ingerman, P.Z., An assembly language for reprogramming, *Comm. ACM* **8**, 12 (Dec. 1965), 769-773.
17. Halpern, M.I., XPOP: a meta-language without metaphysics, *Proc. AFIPS 1964 Fall Joint Computer Conference*, vol. 26, 57–68.

18. Halpern, M.I., *A Manual of the XPOP Programming System*, Lockheed Missiles and Space Company, Palo Alto, California, California, March 1967.

19. Halpern, M.I., Towards a general processor for programming languages, *Comm. ACM* **11**, 1, (Jan. 1968), 15–25.

20. Halpern, M.I., Foundations of the case for natural-language programming, *IEEE Spectrum* **4**, 3, (March 1967), 140–149.

21. Halstead, M.H., *Machine-Independent Computer Programming*, Spectrum Books, Washington, D.C., February 1962.

22. Hopewell, B., *Extensions to autocode using ML/I*, diploma dissertation, University Mathematical Laboratory, Cambridge, 1967.

23. *IBM Operating System/360: Assembler Language*, Form C28-6514, IBM Corporation, Poughkeepsie, New York, 1966.

24. *IBM Operating System/360: PL/I Language Specifications*, Form C28-6571, IBM Corporation, Poughkeepsie, New York, 1966.

25. *IBM 709/7090 Programming Systems: FORTRAN Assembly Program (FAP)*, Form C28-6235, IBM Corporation, Poughkeepsie, New York, 1962.

26. *IFIP-ICC Vocabulary of Information Processing*, North Holland Publishing Company, Amsterdam, 1966.

27. Keese, W.M., Jr., *A Note on Automatic Generation of Documentation by Macro-Assemblers*, Memorandum TM-64-1031-1, Bellcom, Inc., Washington, D.C., Sept. 1964.

28. Lampson, B.W., Interactive machine programming, *Proc. AFIPS 1965 Fall Joint Computer Conference*, vol. 27, 473–481.

29. Leavenworth, B.M., Syntax macros and extended translation, *Comm. ACM* **9**, 11 (Nov. 1966), 790–793.

30. Leroy, H., A macro-generator for ALGOL, *Proc. AFIPS 1967 Spring Joint Computer Conference*, vol. 30, 663–669.

31. Magnuson, R.A., *Extended Use of Macro Assemblers*, Technical Paper RAC-TP-175, Research Analysis Corporation, McLean, Virginia, July 1965.

32. McIlroy, M.D., Macro instruction extensions of compiler languages, *Comm. ACM* **3**, 4 (April 1960), 214–220.

33. Mooers, C.N., TRAC, procedure-describing language for the reactive typewriter, *Comm. ACM* **9**, 3 (March 1966), 215–219.

34. Mooers, C.N., How some fundamental design problems are treated in the design of the TRAC language, *Proc. IFIP Working Conference on Symbol Manipulation Languages*, Pisa, Sept. 1966.

35. Mooers, C.N. and Deutsch, L.P., TRAC, a text-handling language, *Proc. 20th ACM National Conference*, Aug. 1965, 229–246.

36. Shaw, C.J., *On Halpern's XPOP*, unpublished memorandum, Systems Development Corporation, Santa Monica, California, March 1964.

37. Sippl, C.J., *Computer Dictionary and Handbook*, Howard W. Sams and Co., Indianapolis, 1966.

38. Strachey, C., A general-purpose macrogenerator, *Computer J.* **8**, 3 (Oct. 1965), 225–241.

39. Waite, W.M., A language-independent macro processor, *Comm. ACM* **10**, 7 (July 1967), 433–440.

40. Wilkes, M.V., An experiment with a self-compiling compiler for a simple list programming language, *Annual Review in Automatic Programming*, vol. 4, Pergamon Press, Oxford, 1964, 1–48.

41. Wilkes, M.V., The Outer and Inner Syntax of a Programming Language, *Technical Memorandum 67/1*, University Mathematical Laboratory, Cambridge, July 1967.

Cross-References

Numbers refer to the items in the Bibliography.

Compiler Compiler	See 3.
Elliott 803 Macro-generator	See 9. Similar to GPM.
GPL	See 15. General-purpose language.
GPM	See 37. Simple general-purpose macro processor requiring fixed notation.
IMP	See 27. Macro-assembler with unusual facilities.
LIMP	See 38. General-purpose, notation-independent macro processor based on SNOBOL and WISP.
MACRO	See 6. Syntactic macro facility.
Macro-ALGOL	See 29. Macro facility for high-level languages.
Macro FAP	See 24. Macro-assembler.
MAD	See 1. Computation macro facility.
Meta-Assembler	See 11, 16. Generalised macro assembler.
ML/I	See 4, 5 (Appendix A). General-purpose, notation-independent macro processor based on GPM.
PL/I	See 23. Macro facility for high-level language.
SMACRO	See 6. Companion to MACRO.
SYGMA	See 10. Something of a cross between GPM and WISP.
System/360 Macro-Assembler	See 13, 22.
TRAC	See 32, 33, 34. Similar in concept to GPM but oriented for on-line use.
WISP	See 39. Very simple, notation-independent, macro processor.
XPOP	See 17, 18. Elaborate, notation-independent, macro-assembler.

Appendix A ML/I User's Manual

This Appendix (originally named Appendix 1) was published as a separate document.

It has been updated several times; the latest edition is available from the official ML/I web site (`http://www.ml1.org.uk`).

Because of the updates, references to the Appendix may not be completely accurate when it comes to precise section numbers.

Appendix B L-Map Implementor's Manual

This Appendix (originally named Appendix 2) was published as a separate document.

It has been updated several times; the latest edition is available from the official ML/I web site (`http://www.ml1.org.uk`).

The title of this document has also changed over time: the different titles are:

1. *L-map Implementor's Manual.*
2. *Technical Memorandum No. 68/1: The Use of ML/I in Implementing a Machine-Independent Language in order to Bootstrap Itself from Machine to Machine.*
3. *Implementing software using the L language.*

Because of the various updates, references to the Appendix may not be completely accurate when it comes to precise section numbers.