☺

☺ TMS9900/99105 Cross-assembler

☺ Rev. 1.0

☺ Derived from the original William C. Colley, III.
☺ M6800 Cross Assembler

☺ by

☺ Alexander Cameron

☺ Written and modified during June 1984.

☺ The Manual Such As It Is.

☺ 1.1 Format of Cross-assembler Commands
☺ 1.1.1 Command Strings

☺ Tᑎá ruε thσ 990⁝⁝ cross-assembler¼á typσá thσ ì

☺ A>a99 filename options

☺ filename‖
☺ááááááááááááááááááThσáá namσáá oμá thσá sourcσá inpu⌡áá filσáá i≤ ì

☺ options: See next section.

☺ 1.1.2 Options

☺ Thσá sourcσá filσá come≤á froφá thσá currentl· ì

☺ a, b, c, d Disk drives.
☺ - The currently logged in disk drive.

☺ Line≤á oμ inpu⌡ containinτ error≤ wil∞á alway≤ ì

☺ ì
☺ááááááááááááááááá° Consolσ dev
☺ y List device.

☺ Thσá he° filσ wil∞ no⌡ bσ outpu⌡ unles≤ calleΣ ì

☺ Option≤á mus⌡á bσ ruε togethe≥ intᑎá ßá singlσ ì

☺ 1.1.3 Examples

☺ A>a99 barf source -- a:barf.a99
☺ list -- none.
☺ hex -- none.

```
☺                A>a99 barf sblxha source -- b:barf.a99
☺                                    list   -- con:
☺                                    hex    -- a:barf.h99

☺                A>a99 barf ly    source -- a:barf.a99
☺                                    list   -- lst:
☺                                    hex    -- none.


☺                A>b:a99 barf sbh-  source -- b:barf.a99
è ☺                                  list   -- none.
```



```
☺                                    hex    -- a:barf.h99
```

☺ 1.2 Format of TMS9900 Cross-assembler Source Files

☺ Line≤ oµ sourcσ inpu⌠ arσ terminateΣ witΦ CR/L╞ ì

☺ Lowe≥á casσá letter≤ arσ converteΣá t∩á uppe≥ ì

☺ 1) In opcodes,

☺ 2) In checking for keywords such as NOT, and

☺ 3) In command strings.

☺ Thi≤ mean≤ tha⌠ "not"¼á "NOT"¼á "Not"¼ etc« ì

☺ 1.2.1 Statements

☺ Sourcσ file≤ inpu⌠ t∩ thσ 990⁞á Cross-assemble≥ ì

☺ [label] [opcode] [arguments] [;comments.]

☺ Label≤á arσá recognizeΣ b· thei≥ beginninτá iε ì

☺ Opcode≤áá ma·á bσá eithe≥á 990⠿áá instructioε ì

☺ Thσá argument≤ followinτ thσ opcodσ wil∞á var· ì

☺ Line≤á terminatσ witΦ eithe≥ ß CR/L╞ pai≥ o≥á ß ì

☺ 1.2.2 Symbols

è

☺ Symbol≤ ma· bσ oμ an· lengthΒ¼á bu⌠ onl·á thσ ì

☺ A-Z a-z ! & . : ? [\ ] ^ _ `
☺ { | } ~ 0-9

☺ Notσá tha⌠á symbol≤ ma· no⌠ begiε witΦ 0-╣á a≤ ì

☺ ⊥á specia∞á symbo∞ ñ i≤ alway≤ equa∞á t∩á thσ ì

☺ 1.2.3 Numeric Constants

☺ Number≤á begiεá witΦá 0-9,%«á Ñá leadinτá basσ ì

☺ 0ff80h evaluate to ff80 hex.
☺ 128 and 128d evaluate to 80 hex.
☺ 35o, and 35q evaluate to 1d hex.
☺ %0111000 and 0111000b evaluate to 39 hex.

☺ ┴á characte≥á constan⌠ i≤ ß strinτ compose Σá oμ ì

☺ "ab" evaluates to 6162 hex.
☺ "" evaluates to 0000 hex.
☺ "'" evaluates to 0027 hex.
☺ 'A' evaluates to 0041 hex.

è

☺ Notσáá tha⌠á iεá thσá two-characte≥áá characte≥ ì

☺ 1.2.4 Strings

☺ String≤á arσá formeΣ iε thσ samσ wa·á a≤ ì

☺ 1.3 Expression Evaluation

☺ Thσá followinтá operator≤á arσá alloweΣáá iε ì

☺ Unary Plus, Unary Minus

☺ *, /, MOD, SHL, SHR

☺ +, -

☺ >, >=, <, <=, <>, =
☺ (These can also be written as
☺ GT, GE, LT, LE, NE, EQ.)

☺ NOT (1's complement)

☺ AND (Bitwise logical AND)

☺ OR, XOR (Bitwise logical OR and Exclusive
☺áááááááááááááááá OR)

☺ HIGH, LOW
☺ááááááááááááááááááParentheseΣá arσá useΣ t∩ changσ thσá ordeΣá oμ ì

☺ Al∞ operatorΣ excep⌠ +¼á -¼ *¼ /¼ =¼ <>¼ >=¼ ì

☺ Thσ "HIG╚ mumbleó iΣ equivalen⌠ t∩ "mumblσ SH╥ ì

☺ááááááááááááááááááAN— 0ffH".

☺ Thσ relationa∞ operatorΣ (=¼á >¼ etc.⌐ evaluatσ ì

☺ ExpressionΣáááá arσáá terminateΣáá b·áá commas¼ ì

☺ 1.4 990 Family Instructions

☺ Thσ instruction≤ oμ thσ 9900/9910╡ ì

☺ The basic syntax is as follows:

```
☺ [label]   LI  R1,expression   ;The expression
☺                               ;is evaluated
☺                               ;from left to
☺                               ;right.
```

☺ Thσ typσ oμ addressinτ eacΦ instructioε i≤ ì

☺1.4.1Addressing Modes

☺Thσ 9900/9910╡ instructioε seΓ allow≤ ╡ basiπ ì

Mode	Example
Workspace register	MOV R1,R3
Workspace register indirect	A *R1,R2
Symbolic	CLR @expr
Indexe∑	A @expr(R1),R2
Workspacσ registe≥ indirecΓ	DEC *R7+
(autΩ increment)	

☺ Thesσ mode≤ wil∞ iε genera∞ bσ writteε iε thσ ì
è

☺ R, *R, *R+, @X, @X(R)

☺ááááááááááááááááááwhersá Rá refer≤á t∩á ß Registe≥ an Σá ╪á t∩á aε ì

☺ 1.4.2 Instruction Formats

☺ááááááááááááááááááThσá assemble≥ wil∞ properl· evaluatσá ╖á basiπ ì

☺ááááááááááááááááááá Format 0 - No operands at all (N)
☺ááááááááááááááááááá Format 1 - R,*R,*R+,@X,@X(R)
☺ááááááááááááááááááá Format 2 - Displacement (Disp)
☺ááááááááááááááááááá Forma⌠á│ - Destinatioε Field only (D)
☺ááááááááááááááááááá Format 4 - Signed Displacement (SD)
☺ááááááááááááááááááá Format 5 - Workspace Register (W or C)
☺ááááááááááááááááááá Format 6 - Immediate Operands (IOP)
☺ Forma⌠á╖ - Bi⌠ Manipulatioε Instrucions
☺ (TMS99105 only)

☺ááááááááááááááááááDependinτá oε thσ instructioε thesσ format≤ ma· ì

☺ 1.4.3 Group 1 Instructions

☺ This group contains the following opcodes:
☺
☺ NOP
☺ááááááááááááááááááá RSET IDLE CKOF CKON LREX
☺ááááááááááááááááááá RTWP

☺ Thesσá opcode≤ permi⌠ n∩ arguments¼á an Σá hencσ ì

☺ 1.4.4 Group 2 Instructions
☺ááááááááááááááááááá
☺ááááááááááááááááááThi≤ grou≡ oμ instruction≤ arσ oμ Forma⌠ typσ ▒ ì

☺ááááááááááááááááááThis group contains the following opcodes:

☺ááááááááááááááááááá ┴á A┬ ├ C┬ ╙ S┬ SO├ SOC┬
☺ááááááááááááááááááá SZ├á SZC┬á MO╓ MOVB
☺ááááááááááááááááááá
☺ááááááááááááááááááIstruction format: [opcode│format1│format1]

☺ááááááááááááááááááExample: A *R1+,R2

☺ 1.4.5 Group 3 Instructions

☺ Thi≤á grou≡á oμ instruction≤ arσá Dua∞á Operan Σ ì

è

☺ááááááááááááááááááThis group contains the following opcodes:
☺ááááááááááááááááá
☺ááááááááááááááááá COC CZC XOR MPY DIV XOP LDCR STCR

☺ááááááááááááááááááInstruction format: [opcode|format3|format1]

☺ Example: XOP @LABEL,3

☺ááááááááá1.4.6ááááá Group 4 Instructions

☺ááááááááááááááááááThiᵴáá grouᵯá oµá instructionᵴá alloᵡá Multiplσ ì

☺ááááááááááááááááááThiᵴá followinтá instructionᵴá belonтá tΩá thiᵴ ì

☺ááááááááááááááááá ┬ B╠ BLW╨ CL╥ SET╧ IN╓ NE╟ AB╙ SWPB
☺ááááááááááááááááá á INC INCT DEC DECT X
☺ááááááááááááááááá BIND(a TMS99105 instruction)

☺ááááááááááááááááááInstruction format: [opcode|format1]

☺ Example: INC @LABEL(R3)

☺ááááááááá1.4.7 Group 5 Instructions

☺ááááááááááááááááááThiᵴá grouᵯ oµ instructionᵴ arσ CR╒á Single-Bi⌠ ì

☺ááááááááááááááááááThσá followinтá instructionᵴá belonтá tΩáá thiᵴ ì

☺ááááááááááááááááá SBO SBZ TB

☺ááááááááááááááááááInstruction format: [opcode|format4]

☺ Example: SBO 25

☺ 1.4.8ááááá Group 6 Instructions

☺ Thiᵴáá grouᵯáá coverᵴá thσáá Conditiona∞áá Jumᵯ ì

☺ááááááááááááááááááThσáá followinтá instructionᵴá belonтá tΩá thiᵴ ì

☺ááááááááááááááááá
☺ááááááááááááááááá JEQ JGT JH JHE JL JLE JLT JMP JNC JNE
☺ááááááááááááááááá JNO JOC JOP

☺ááááááááááááááááááInstruction format: [opcode|format2]

☺ Example: JEQ LABEL

☺ááááááááá1.4.9áááááGroup 7 Instructions
è

☺ááááááááááááááááááThi≤á grou≡á havσ aε Immediatσá Sourcσá Operanτ ì

☺ááááááááááááááááááThσáá followinτá instruction≤á belonτá t∩á thi≤ ì

☺áááááááááááááááááá AI ANDI CI LI ORI
☺áááááááááááááááááá BLSK (a TMS99105 instruction)

☺ááááááááááááááááááInstruction format: [opcode|format5]
☺áááááááááááááááááá [format6]

☺ááááááááááááááááááExample: AI R3,128

☺ááááááááá1.4.10ááááGroup 8 Instructions

☺ááááááááááááááááááThi≤á grou≡á cover≤ thσ Interna∞ Registe≥á Loaτ ì

☺ááááááááááááááááááThσá followinτá instruction≤á belonτá t∩áá thi≤ ì

☺áááááááááááááááááá
☺áááááááááááááááááá LWPI LIMI

☺ááááááááááááááááááInstruction format: [opcode]
☺áááááááááááááááááá [format6]

☺ááááááááááááááááááExample: LIMI 3

☺ááááááááá1.4.11ááááGroup 9 Instructions

☺ááááááááááááááááááThi≤á grou≡ cover≤ thσ Interna∞ Registe≥á Storσ ì

☺ááááááááááááááááááThσáá followinτá instruction≤á belonτá t∩á thi≤ ì

☺áááááááááááááááááá STST STWP
☺áááááááááááááááááá LSⁿá LW╨ (TMS9910╡ instructions)

☺ááááááááááááááááááInstruction format: [opcode│format5]
☺ááááááááááááááááááá
☺ááááááááááááááááááExample: STST R3

☺ 1.4.12 Group 10 (TMS99105 only)
☺
☺ This grou≡ cover≤ thσ Bit-Manipulatioε instruc-ì

☺ááááááááááááááááááá TMB TCMB TSMC
è

☺ááááááááááááááááááInstruction format: [opcode]
☺ááááááááááááááááááá [format3│format1]
☺ááááááááááááááááááá
☺ááááááááááááááááááExample: TMB @BITMAP(R3),8

☺ 1.5 Pseudo-operations
☺ 1.5.1 END

☺ Wheεá thi≤ statemen⌠ i≤á encountered¼á thσ ì

☺ [label] END

☺ Iμ an· I╞ statemen⌠ i≤ no⌠ closeΣ witΦ aε END╔ ì

☺ This statement permits no arguments.

☺ 1.5.2 EQU

☺ááááááááááááááááááThi≤á statemen⌠ i≤ useΣ tΩ assigεá ßá permenan⌠ ì

☺ label EQU expression

☺ ┴á phasinτ (P⌐ erro≥ wil∞ resul⌠ iµá an· ì

☺ 1.5.3 BYTE

☺ Thi≤á statemen⌠á i≤á use Σá t∩á placσá byte≤á iε ì

☺ [label] BYTE [expr1][,epxr2].....

☺ 1.5.4 TEXT

☺ Thi≤á statemen⌠á i≤ use Σ t∩á placσá string≤á iε ì

è

☺ [label] TEXT [string][,string].....

☺ 1.5.5 WORD

☺ Thi≤á statemen⌠á i≤á use Σá t∩á placσá word≤á iε ì

☺ [label] WORD [expr1][,expr2].....

☺ 1.5.6 BSS (Block Starting with Symbol)

☺ Thi≤á statemen⌠ i≤ use Σ t∩ reservσ ßá blocδá oµ ì

☺ [label] BSS expression

☺ááááááááá1.5.7áááááEVEN

☺ááááááááááááááááááThiςá statemenƒ iς useΣ tΩ forcσ thσ assembleΣ ì

☺
☺ááááááááááááááááá[label]   EVEN
☺
☺   1.5.8   DXOP

☺   Thiς statemenƒ allowς thσ programmeΣ tΩá definσ ì

☺ááááááááááááááááá   DXOP PUSH,3   ;assign the name PUSH to
☺ááááááááááááááááá   ;XOP number 3
☺ááááááááááááááááá
☺ááááááááááááááááááFoΣá examplσ iµ PUS╚ iς ß previousl· codeΣá XO╩ ì

è   ☺ááááááááááááááááá

☺ááááááááááááááááá
☺   1.5.9   Conditional Assembly

☺ááááááááááááááááááBlockςáá oµá codσá caεá bσá assembleΣá oΣáá noƒ ì

☺   IF   expression

☺   (lines of code)

☺   ENDI

☺   Iµá thσá expressioε evaluateς tΩ 0¼á thσá codσ ì

☺ Iεáá addition¼áá thσá ELS┼á directivσá i≤ ì

☺ IF expression

☺ (lines of code)

☺ ELSE

☺ (more lines of code)

☺ ENDI

☺ This is equivalent to:

☺ IF expression

☺ (lines of code)

☺ ENDI

☺ IF NOT expression

☺ (more lines of code)

☺ ENDI

☺ Notσá tha⌠á label≤ arσá no⌠á permitteΣá oε ì

☺ ⊥ phasinτ (P⌐ erro≥ wil∞ bσ flaggeΣ iµá therσ ì

☺ 1.5.10 AORG

☺ Thi≤á statemen⌠á i≤ useΣ t∩ loaΣ ßá valuσ ìè

☺áááááááááááááááááint∩ thσ assembl· prograφ counter« Thσ valuσ ì

☺ [label] AORG expression

☺ 1.5.11 SET

☺ Thi≤á statemen⌠ i≤ useΣ t∩ assigεá ßá temporar· ì

☺ The syntax is:

☺ label SET expression

☺ 1.6 Error Messages

☺ Erro≥á message≤á arσ flaggeΣ witΦá ßá singlσ ì

☺ A Presently not implemented.

☺ B Distancσ oε ß brancΦá instructioε ì

☺ D Digi⌠á to∩á largσ fo≥á thσá basσá wa≤ ì

☺ E Expressioε ill-formed« Looδ fo≥ noε ì

☺ I Iμ stacδ imbalance«á Looδ fo≥ ELS┼ ì

☺ L InvaliΣ label«á Labe∞ ma· contaiε ì

☺ M Labe∞áá multipl·áá defined«áá Labe∞ ì

☺ááááááááááááááááááááááááááádefinitions are not SET statements.

☺ O InvaliΣáá opcode«ááá Looδáá foΣ ì

☺ P Phasinτ error« Looδ foΣ expressionΣ ì

☺ááááááááááááááááá R RegisteΣ valuσ toΩ large«á RegisterΣ ì

☺ááááááááááááááááááááS Syntaºá error«á Checδá youΣá syntaº ì

☺ T ToΩáá manᐧá argumentΣá oεáá thiΣ ì

☺ U UndefineΣá symboαáá encountereΣ ì

☺ V Valuσááá ou⌠áá oμáá bounds«ááá Iε ì

☺ * ThiΣá statemen⌠á generateΣá bᐧá thσ ì

☺ " Quotσá imbalancσá error«á BeaΣá iε ì

☺ (ParenthesiΣá imbalancσ error«á Coun⌠ ì

☺ 1.7 Assembler Abort Conditions

☺ UndeΣ certaiε circumstances¼á thiΣá assembleΣ ì

☺áááááááááááááááááassembly«á Iµ yo⌡ don'⌠ ge⌠ thσ erro≥á coun⌠ ì

☺ 1) Can't open source.

☺ Thσá sourcσ filσ doe≤ no⌠ exis⌠ ì

☺ 2) Can't open list.
☺ Can't open hex.

☺ N∩á director·á entrie≤ lef⌠á oεá thσ ì

☺ 3) Illegal command line.

☺ Bone up on command lines.

☺ 4) No file info supplied.

☺ Bone up on command lines.

☺ 5) If stack overflow.

☺ I╞á directive≤ ma· onl·á bσá nesteΣ ì

☺ 6) Disk read error.

☺ Sourcσá filσá ha≤ ßá baΣá CR╞á o≥ ì

☺ 7) Disk write error.

☺ Ou⌠ oµ disδ o≥ director· spacσ oε thσ ì

☺ 8) Error closing file.

☺ Probleφáá closinτá lis⌠á o≥áá he° ì

☺ 9) Symbol Table Overflow.

☺ You≥á sourcσ prograφ define≤ to∩ man· ì

☺ 1.8 Compiling the Assembler

è

☺ TMS9900/99105 Cross-assembler Rev. 1.0 Page

☺ T∩á compilσ thσ assemble≥ a≤ i⌠á stands¼á yo⌡ ì

☺ 1) 40K of RAM.

☺ 2) The BDS C Compiler Version 1.5
☺áááááááááááááááá (good box, Leor!!).

☺ 3) Digita∞áá Research'≤áá AS=áá o≥áá MA├ ì

☺ T∩ ge⌠ a99tbls.cr∞ u≡ froφ scratch¼á yo⌡ ì

☺ A>casm a99tbls ;source a99tbls.csm

☺ This should yield a file a99tbls.asm on drive A.

☺ A>maπá a99tbl≤ $p· -≤ ;ma· usσ asφ fn.aa· ì

☺ A>cloaΣá a99tbl≤ ;producσ CR╠ filσ ì

☺ You should now have a99tbls.crl on drive A.

☺ áNo≈á yo⌡á arσ read· t∩ compilσ thσ res⌠ oµá thσ ì

☺ A>cc a99
☺ A>cc a99asmlnc
☺ A>cc a99evalc
☺ A>cc a99getc
☺ A>cc a99putc
☺ A>cc a99symbc

☺ Now you can link it all together.

☺ A>clink a99 -s

☺ *a99asmln
☺ *a99eval
☺ *a99get
☺ *a99put
☺ *a99symb
☺ *a99tbls

☺ And, as if by magic, you've got a99.com!

☺ Notσá thaƒá thσ linkagσ caε al∞ bσ donσ oεá onσ ì

è

☺ 1.9 Final Comments
☺ Happ·á assemblingíá Iµ yo⌡ havσ question≤ o≥ ì

è