

Symbol table package. Saltzer.

~~TABLE~~ SEARCH @ Look up symbol.
 SYMDEF INSERT @ Insert new symbol.

SAVE @ Initialize table base.
 EAPbp @ Get symbol from caller.
 LDQ @ Create a random number.
 MPY @ Save number in case of failure.
 STQ @ Form probe index.
 MPY @ Make it even.
 ALS @ Clean up extra garbage.
 ANA @ Set table index pointer.
 FAX1 @ Get table entry.
 LDA bp!1,1 @ If zero, symbol is absent.
 TZE @ Something there, compare with symbol.
 CMPA @ Exit if successful.
 TZE @ Didn't work, reset generator.
 LDQ @ Try it again.
 TRA @

* FOUND: @ Symbol found, form value pointer.
 NOTIN: @ Give pointer to caller. even the symbol absent? In pointer pointing at value?
 * RHO: @ Constant which produces random numbers.
 TEMP @ Temporary storage for number generator.

* INSERT: @ Initialize table base.
 SAVE @ Set up table pointer for subroutine.
 EAPbp @ Get symbol to be inserted.
 STPbp @ Save for subroutine call.
 LDA @ LOC @ Get location to insert symbol.
 STA SYMBOL @ Non-zero lg means symbol found.
 CALL SEARCH(TABLE, SYMBOL, FLAG, LOC) @ Skip if found.
 SNZ @ Check table count to make sure that
 TNZ @ there is room for this symbol.
 LDA bp!0 @ Move count to limit position.
 ADD =2 @
 ALS 18 @

UNDEFINED ?
 TPL @ Exit if table full. [Leave one hole.] Why?
 AOS @ There is room, increment table size.
 LDX1 @ SEARCH returns location to store symbol.
 LDA @ Get symbol.
 STA bp!1,1 @ Store it.
 LDA ap!4,* @ Get value of symbol.
 STA bp!2,1 @ Store value.
 RETURN @ Ignore multiply defined symbols.

* MULTI: @ Storage for subroutine call.
 TEMP @
 TEMPD @

*don't you have to move some symbols
 toward end of tbl? or you'll be
 replacing rather than inserting. (?)*