Date: March 14, 1995  Bulletin No: 95001  Rev: 1.0a

Approvals: CSE  ENG  PPL

Products: MotionBASIC® MB2.1c through MB2.1i

Title: WAIT and WAIT UNTIL statements may not work as expected when used in EVENT subroutines.

Purpose:
To advise customers of the solution to a software problem that can cause WAIT and WAIT UNTIL statements used in an EVENT subroutine to finish earlier or later than expected.

Description:
When a WAIT [time] statement is executed, it launches a task which suspends program execution for the duration of the time delay. If an EVENT occurs during the delay and the program jumps to an EVENT subroutine, the WAIT [time] task keeps running. If the EVENT subroutine contains another WAIT [time] or WAIT UNTIL statement, this second WAIT statement will finish when the original time delay expires rather than when its own time delay or UNTIL condition is satisfied.

Part Numbers Affected:
All versions of MotionBASIC® from MB2.1c through MB2.1i (inclusive). Prior to MB2.1c, WAIT statements could not be interrupted by events and therefore could not experience this problem.

Work Around:
When a WAIT instruction is needed in an EVENT subroutine, substitute one of the following.

For a WAIT [time] statement, substitute:

\[
\text{DIO@}(X) = \text{desired delay} \\
\text{WHILE DIO@}(X) : \text{WEND}
\]

X can be any spare DIO point from 1 to 40. If you use a point between 17 and 40 your controller must have the /I option although you do not need to install an extended I/O rack unless you need it for other I/O points. In ether case, you do not have to install an OPTO22 module but you will need to set IO.MODE@(X)="O" to configure the point as an output. It
will also work with any PAMUX output however you will have to install a PAMUX rack and OPTO22 output module.

Or, if you do not have any spare DIO points to use, you can substitute the following which will work for time delays of up to 65635.

START.TIME&=TIMER AND &HFFFF&
ELAPSED&=0
WHILE ELAPSED<&desired delay
   IF (TIMER AND &HFFFF&)<START.TIME& THEN
      START.TIME&=START.TIME&-&HFFFF&
   ENDIF
   ELAPSED&=(TIMER AND &HFFFF&)-START.TIME&
WEND

If you are using a WAIT UNTIL [condition] statement in your EVENT subroutine, substitute:

   WHILE NOT condition :WEND

If you are using the WAIT [time] AFTER [condition] syntax in your EVENT subroutine, substitute:

   WHILE NOT condition :WEND
   DIO@(X)=desired delay
   WHILE DIO@(X) :WEND

Or:

   WHILE NOT condition :WEND
   START.TIME&=TIMER AND &HFFFF&
   ELAPSED&=0
   WHILE ELAPSED<&desired delay
      IF (TIMER AND &HFFFF&)<START.TIME& THEN
         START.TIME&=START.TIME&-&HFFFF&
      ENDIF
      ELAPSED&=(TIMER AND &HFFFF&)-START.TIME&
   WEND

**Warranty Considerations:**

This problem will be corrected in the next release of the MotionBASIC®. While not yet formally scheduled, release is expected in the fall of 1995. Since the work around does provide a reasonable way to solve this problem, it is not considered a warranty issue.