Please take note of the following problem in using the TM, an integrated development environment:

- On using the inspector

1. **Products and Versions Concerned**
   
   TM V.3.00 through 3.20 used in any of the following C compiler packages:
   
   - M3T-NC308WA V.3.00 Release1 and later
     (for the M32C/90, M32C/80 and M16C/80 series of MCUs)
   - M3T-NC30WA V.4.00 Release1 and later
     (for the M16C/60, M16C/30, M16C/Tiny, M16C/20 M16C/10 and R8C/Tiny series of MCUs)

2. **Description**
   
   If a source file in a project contains an enum-type variable declared using typedef, the inspector is abnormally shut down as soon as it is invoked.

3. **Conditions**
   
   This problem occurs if the following conditions are all satisfied:
   
   (1) This problem occurs if the following conditions are all satisfied:
   
   (2) The type name defined in (1) is re-defined as another using typedef.
   
   (3) Another variable is defined using the type name re-defined in (2).

   Example:
   
   ```
   typedef enum _NUMBER{NUM_0, NUM_1, NUM_2} NUMBER; /* Condition (1) */
   ```
typedef NUMBER NUM; /* Condition (2) */
NUM number; /* Condition (3) */

4. Workaround
This problem can be circumvented either of the following ways:
(1) Don't use the type name of an enumeration type defined using typedef.

Example:

typedef enum _NUMBER{ NUM_0,NUM_1,NUM_2} NUMBER;
typedef enum _NUMBER NUM; /* Type name of enum type defined using typedef */
NUM number; /* NUMBER not used */

(2) Don't re-define the type name of an enum-type variable as another using typedef.

Example:

typedef enum _NUMBER{ NUM_0,NUM_1,NUM_2} NUMBER;
/* typedef NUMBER NUM; */
/* NUM number; */
NUMBER number;

5. Schedule of Fixing the Problem
We plan to fix this problem in our next release of the product.