6.12 UAL011 Read Device Variables (Command 9)

Checks operation of command 9. Issue command 0 to determine the number of variables supported by the device. Then issue command 9 for 0 to 8 device variables, using device variable code 0 and checking the error response for each.

For HART 7 and later devices, verify that the time stamp increments during the test.

Table 5 Universal Command 9 Response Data Bytes for Varying Device Variables

Number of Device Variables Requested	Byte Count (BC)	
	HART 6	HART 7
1	11	15
2	19	23
3	27	31
4	35	39
5	35	47
6	35	55
7	35	63
8	35	71

References:

Specification	Rev.	Sections
Universal Command Specification	6.0	6.10

Test Case A: Checking for Supported Device Variables

```
CALL IdentifyDevice
IF UNIV_REVISION < 6 THEN
   SEND Command 9 with no data bytes

   IF RESPONSE CODE != "Command Not Implemented"
      THEN Test result is FAIL                                    (3211)
   ELSE
      Abort (Test is not applicable to this device)               (5001)
   END IF
END IF
```

Read Command 3 to learn how many slots it supports and the unit codes of the Device Variables. While checking for dVar below we must find dVar using the unit codes returned in Command 3.

```
SET dynVarUnitsList = null
SET numDynVars = 0

SEND Command 3 to read DynVarUnits

SWITCH on (BYTE_COUNT)

CASE 11
   SET dynVarUnitsList = { pvUnits }
   SET numDynVars = 0
```

```
      CASE 16
         SET dynVarUnitsList = { pvUnits, svUnits }
         SET numDynVars = 1

      CASE 21
         SET dynVarUnitsList = { pvUnits, svUnits, tvUnits }
         SET numDynVars = 2

      CASE 26
         SET dynVarUnitsList = { pvUnits, svUnits, tvUnits, qvUnits }
         SET numDynVars = 3

      CASE DEFAULT
         Test result is FAIL                                          (3213)

      END SWITCH
```

If the DUT does not expose any Device Variables then it returns PV, SV, TV, and QV

```
      SEND Command 0 to read the = maxDeviceVars.
      IF (maxDeviceVars = 0)
         THEN SET maxDeviceVars = NumDynVars
      END IF
```

Find a supported Device Variable

```
      SET nVar = 0
      SET dVar = -1
      DO
         SET dVarFound = FALSE;
         DO
            INCREMENT dVar
            SEND Command 9 with one byte = dVar
            CALL TestValidFrame

            IF ( (RESPONSE_CODE == "Invalid Selection")
               THEN Test result is FAIL                               (3210)
            ELSE IF ( (RESPONSE_CODE != "Update Failure")
              AND (RESPONSE_CODE != "Success" )
              AND (RESPONSE_CODE != "Dynamic Variables Returned for
                  Device Variables" ))
               THEN Test result is FAIL                               (3220)
            END IF
            IF (((BYTE_COUNT != 11) AND (UNIV_REVISION == 6))
              OR ((BYTE_COUNT != 15) AND (UNIV_REVISION > 6)))
               THEN Test result is FAIL                               (3225)
            END IF
```

Make sure Command 9 returns the dVar we are looking for

```
            IF ("Slot 0: Device Variable Code" response != dVar)
               THEN Test result is FAIL                               (3223)
            END IF
```

If we get a NaN response make sure all the other fields are set correctly

```
            IF (dVar.Value == "7F A0 00 00"(NaN) THEN
               IF (dVar.Units != 250)
                  THEN Test result is FAIL                              (3226)
               END IF
               IF (dVar.Status != 0x30)
                  THEN Test result is FAIL                              (3227)
               END IF
               IF (dVar.Class != 0)
                  THEN Test result is FAIL                              (3228)
               END IF
```

Response is "Success" or "Update Failure", not a NaN, and the right Byte Count. I think we have it!

```
            ELSE
               IF (dVar.Units == [250, 252, 255] )
                  THEN Test result is FAIL                              (3222)
               ELSE IF (dVar.Units == [251, 253] )
                  THEN PRINT "INSPECT: Possible illegal use of Unit Code
                     %uCode in Device Varialble %dVar"
                  END IF
               END IF
```

Prune the dynVarUnitsList removing the unit codes as we find them in the dVar list. If we find all the units then dynVarUnitList will become empty

```
               SET dynVarUnitIndex = length of dynVarUnitsList
               IF (dynVarUnitIndex > 0)
                  SET elementRemoved = FALSE
                  DO
                     DECREMENT dynVarUnitIndex
                     IF ( dynVarUnitList[dynVarUnitIndex] == dVar.Units)
                        REMOVE list element dynVarUnitList[dynVarUnitIndex]
                        SET elementRemoved = TRUE
                     END IF
                  WHILE (dynVarUnitIndex >= 0) AND (!elementRemoved) )
               END IF
```

Signal we found a dVar

```
               dVarFound == TRUE
            END IF
         WHILE ( (dVar < 239) AND (!dVarFound) )
```

We are out of the Do-Loop (either exhausted Device Variables or we found one) If we found one, check the classification for validity

```
         IF (dVarFound) THEN
            SET nVar = nVar + 1
            IF (slot 0 variable classification == [1-63 or 240-255] )
               THEN Test result is FAIL                              (3230)
            END IF
```

Is this a legal Device Variable ?

```
            IF (dVar > maxDeviceVars)
               THEN Test result is FAIL                              (3235)
            END IF
```

Send Command 9 with varying number of data bytes. Check command response length

```
            IF UNIV_REVISION > 6 THEN
               maxCnt = 7
               length[] = {23, 31, 39, 47, 55, 63, 71, 71}
               nBytes[] = { 2,  3,  4,  5,  6,  7,  8,  9}
            ELSE
               maxCnt = 3
               length[] = {19, 27, 35, 35}
               nBytes[] = { 2,  3,  4,  5}
            END IF

            FOR ( iter = [0 to maxCnt] )
               SEND Command 9 with nBytes[iter] of dVar
               CALL TestValidFrame

               SWITCH on RESPONSE_CODE

                  CASE RESPONSE_CODE == "Success"
                     IF (BYTE_COUNT != length[iter])
                        THEN Test result is FAIL                    (3251)
                     END IF

                  CASE RESPONSE_CODE == "Update Failure"
                     IF (BYTE_COUNT != length[iter])
                        THEN Test result is FAIL                    (3252)
                     END IF

                  CASE RESPONSE_CODE == "Command Response Truncated"
                     IF (nBytes < 5) OR (UNIV_REVISION < 7)
                        THEN Test result is FAIL                    (3253)
                     END IF

                     IF (BYTE_COUNT != {39, 47, 55, or 63} )
                        THEN Test result is FAIL                    (3254)
                     END IF

                     IF ( nBytes[iter] < (maxDeviceVars + 1) )
                        THEN Test result is FAIL                    (3274)
                     END IF

                  CASE RESPONSE_CODE == "Dynamic Variables Returned
                       for Device Variables"
                     IF (BYTE_COUNT != length[iter])
                        THEN Test result is FAIL                    (3250)
                     END IF

                  CASE DEFAULT
                     Test result is FAIL                            (3256)
               END SWITCH
            END FOR

         END IF
      WHILE (dVar < 239)
```

Must be at least the Dynamic Variables

```
IF (nVar = 0)
   THEN Test result is FAIL                                    (3212)
END IF

IF (length of dynVarUnitsList != 0)
   THEN Test result is FAIL                                    (3214)
END IF
```

Check "Invalid Selection" Response Code

```
SEND Command 9 with 4 data bytes of 0xFF
CALL VerifyResponseAndByteCount("Invalid Selection", 2)
CALL TestValidFrame
END TEST CASE
```