Extension of VHDL to support multiple-byte characters

Kiyoshi Makino
Seiko Instruments Inc.

VHDL Project Group
EDA Technical Committiee
EIAJ

VHDL International Users Forum 1998 Fall

Agenda

- Requirement for multiple-byte character
- Character set standards
- Encoding methods
- IEEE 1076-1993 character set definition
- Predefined type -- *character*
- Predefined type -- *string*
- Required change of IEEE 1076 Std.

Requirement for multiple-byte character

- a) VHDL Identifier

```
signal 同期信号 : std_logic;
```

- b) CHARACTER/STRING literal

```
assert false report "信号処理を間違えています";
```

- c) As comment text

```
Y <= not A;  -- 日本語のコメント
```


Character set standards

- Japanese Character Set Standards
  - JIS X 0201-1976 63 characters
  - JIS C 6226-1978 2965 + 3384 + 453 characters
  - JIS X 0208-1983 2965 + 3388 + 524 characters
  - JIS X 0208-1990 2965 + 3390 + 524 characters
  - JIS X 0212-1990 5801 + 266 characters
- Asian Character Set Standards
  - KS C 5601-1992 2350 + 4888 + 986 characters Korean
  - GB 2312-80 3755 + 3008 + 683 characters Mainland China
  - Big Five 5401 + 7652 + 470 characters Taiwan
- Unicode/ISO-10646
  - Unicode v2.0 - 1993 ... 20902 characters
  - Unicode v2.1 - 1998? Bug fix, Euro sign,etc.

Encoding methods

a) JIS (7bit, Modal)

a	b	c	<ESC>	$	B	日		本		語		<ESC>	(	J	a	b	c
61	62	63	1B		24		42	46-7C		4B-5C		38-6C	1B		28		4A	61	62	63

b) Shift-JIS (SJIS) (8bit, variable width, Nonmodal)

a	b	c	日		本		語		a	b	c
61	62	63	93-FA		96-7B		8C-EA		61	62	63

c) EUC (Extended Unix code) (8bit, variable width, Nonmodal)

a	b	c	日		本		語		a	b	c
61	62	63	C6-FC		CB-DC		B8-EC		61	62	63

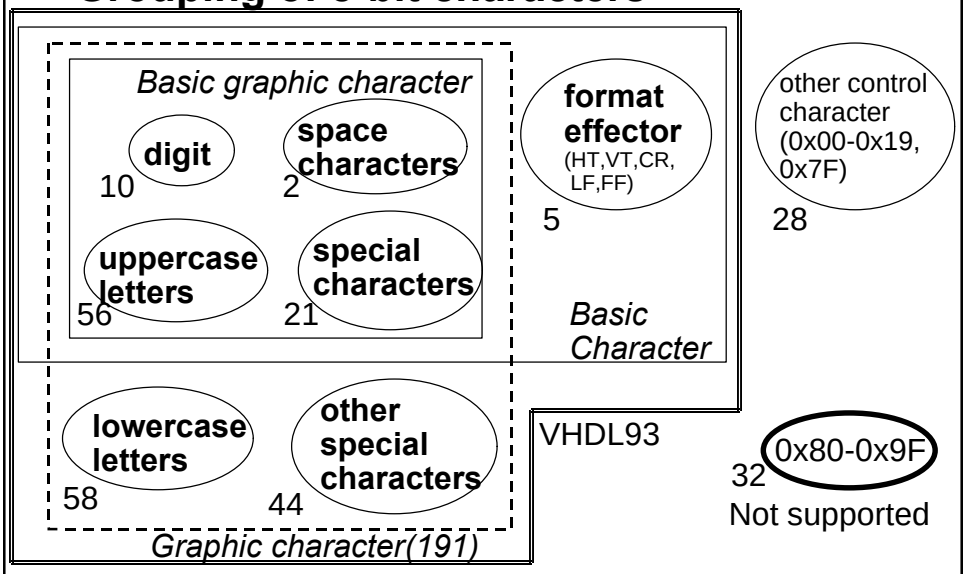
d) Unicode (8bit, fixed width, Nonmodal)

a		b		c		日		本		語		a		b		c	
00-61		00-62		00-63		65-E5		67-2C		8A-9E		00-61		00-62		00-63	

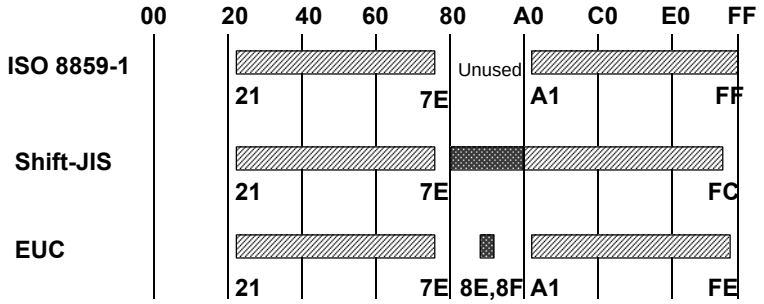
IEEE 1076-1993 definition of character set

- ***The only characters allowed in the text of a VHDL description are the graphic characters and format effectors. Each graphical character corresponds to a unique code of the ISO eight-bit coded character set[ISO 8859-1:1987(E)], and is represented (visually) by a graphical symbol. (by VHDL93 LRM)***
- **Japanese character use character which exceed ISO 8859-1 specification (0x80-0x9F)**

Grouping of 8-bit characters

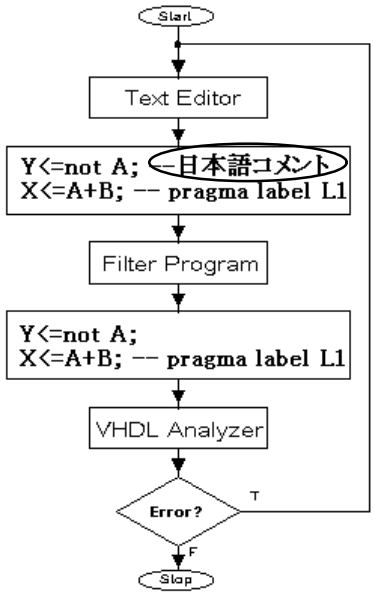


Encoding range



Current workaround

Delete Japanese comment before VHDL Analyzer



Not compliant

Compliant

Predefined type -- *character*

- **VHDL93 type *character* allows all 256 characters**
  - **Character'pos(character'high) becomes 256**
  - **191 graphic characters (ISO 8859-1)**
    - » **0x20 -- 0x7E (ASCII)**
    - » **0xA0 -- 0xFF (Extended ASCII)**
    - » **Character literal can be used. Ex. c1 := 'A';**
  - **65 non-graphic characters**
    - » **0x00 -- 0x19 (Control characters. Defined as NUL, SOH, STX...)**
    - » **0x7F -- DEL**
    - » **0x80 -- 0x9F (not used. Defined as C128, C129, C130...,C159)**
    - » **Character literal can not be used. Instead of it, enumeration type. Ex. c1 := NUL; c2 := C128;**

Predefined type -- *string*

- ***A string literal is formed by a sequence of graphical characters*** **(by VHDL93 LRM)**
  - **STR2 := "日本語" ... EUC OK (Except 0x8E,0x8F)**
  - **STR2 := "日本語" ... Shift-JIS ERROR**
    - » **If Japanese string use 65 non-graphical character, then it is an error like follows**

      ERROR: String Literal has a character not in the enumerated type
- **Possible workaround**

```
STR2 :=   character'VAL(16#93#)
        & character'VAL(16#FA#)
        & character'VAL(16#96#)
        & character'VAL(16#7B#)
        & character'VAL(16#8C#)
        & character'VAL(16#EA#);
```

Note:
Type *String* can hold Shift-JIS characters.
String Literal does not support Shift-JIS.

How to solve the character/string literal?

- **Expand *character* type to enumeration type of each Japanese/etc.. character**
  - **Huge standard package.... not acceptable.**
- **Introduce new character type and refer to the existing character set standard**
  - **Ex. type multiple-character is ISO-something**
  - **Difficult to make consensus to select one character set.**
- **Change string literal definition**
  - **Affect the VHDL analyzer too much(?)**
- **We can not find the way to solve this issue now.**

Required change of IEEE Std 1076

- **Section 13.1 Character set**
  - Allow character between 0x80-0x9F as VHDL description
- **Section 13.8 Comment**
  - Allow character between 0x80-0x9F as comment text explicitly
- **Section 13.5 String literals (If possible)**
  - Allow character between 0x80-0x9F in string literals