# Direct Programming Interface for VHDL 201x Proposal Update:

Purpose of update is to make DPI universal connection interface in mixed designes. It makes VHDL201x more effective

Mapping types:

| VHDL 201x | C99 | SystemVerilog | SystemC |
|---|---|---|---|
| BOOLEAN | unsigned char | bit | sc_signal<bool> |
| BIT | unsigned char | bit | sc_signal<sc_bit> |
| CHARACTER | char | byte | sc_signal<char> |
| INTEGER | long int | int | sc_signal<int> |
| REAL | double | real | sc_signal<double> |
| TIME | long long int | longint | sc_signal<uint64_t > |
| STRING | const char * | string | C99 type used ; sc_string is deprected |
| STD_LOGIC | vhLogicVal | logic/reg | sc_signal<sc_logic> |
| STD_ULOGIC | vhLogicVal | logic/reg | sc_signal<sc_logic> |
| BIT_VECTOR | vhBitVecVal | bit vector (i.e bit [x:0] b) | sc_signal<sc_bv< > > |
| STD_LOGIC_VECTOR | vhLogicVecVal | logic/reg vector (i.e reg [x:0] b) | sc_signal<sc_lv< > > |
| STD_ULOGIC_VECTOR | vhLogicVecVal | logic/reg vector (i.e reg [x:0] b) | sc_signal<sc_lv< > > |
| INTEGER_VECTOR | vhOpenArray | Fixed size array (numer of elements must match) or container (number of dimensions must match, excluding associative arrays). | C99 type used |
| REAL_VECTOR | vhOpenArray | Fixed size array (numer of elements must match) or container (number of dimensions must match, excluding associative arrays). | C99 type used |
| TIME_VECTOR | vhOpenArray | Fixed size array (numer | C99 type used |

| | | | |
|---|---|---|---|
| | | of elements must match) or container (number of dimensions must match, excluding associative arrays). | |
| BOOLEAN_VECTOR | vhOpenArray | Fixed size array (numer of elements must match) or container (number of dimensions must match, excluding associative arrays). | C99 type used |
| UNSIGNED (NUMERIC_BIT) | vhUBitVecVal | bit vector (i.e bit [x:0] b) | sc_signal<sc_bv< > > |
| SIGNED (NUMERIC_BIT) | vhBitVecVal | signed bit vector (i.e bit signed [x:0] b) | sc_signal<sc_bv< > > |
| UNSIGNED (NUMERIC_STD) | vhULogicVecVal | logic/reg vector (i.e reg [x:0] b) | sc_signal<sc_lv< > > |
| SIGNED (NUMERIC_STD) | vhLogicVecVal | signed logic/reg vector (i.e reg signed [x:0] b) | sc_signal<sc_lv< > > |
| UNRESOLVED_UNSIGNED (NUMERIC_STD) | vhULogicVecVal | logic/reg vector (i.e reg [x:0] b) | sc_signal<sc_lv< > > |
| UNRESOLVED_SIGNED (NUMERIC_STD) | vhLogicVecVal | signed logic/reg vector (i.e reg signed [x:0] b) | sc_signal<sc_lv< > > |
| record | struct | unpacked struct | C99 type used |
| array | vhOpenArray | Fixed size array (numer of elements must match) or container (number of dimensions must match, excluding associative arrays). | C99 type used |

```
typedef int32_t vhBitVecVal
typedef uint32_t vhUBitVecVal
typedef int32_t vhLogicVal
typedef uint32_t vhULogicVal

typedef struct{
        vhLogicVal * vec;
        uint32_t size;
}vhLogicVecVal;

typedef struct{
        vhULogicVal * vec;
        uint32_t size;
}vhULogicVecVal;
```

where vhLogicVal is one of (vhpiU,vhpiX,vhpi0,vhpi1,vhpiZ,vhpiW,vhpiL,vhpiH,vhpiDontCare) . It may be typedef to vh type

i.e vhpiU <=> vhU

```
typedef void* vhOpenArray;


int vhLeft(const vhOpenArray h, int d);
int vhRight(const vhOpenArray h, int d);
int vhLow(const vhOpenArray h, int d);
int vhHigh(const vhOpenArray h, int d);
int vhIncrement(const vhOpenArray h, int d); /*TO or DOWNTO, return 0 for
                                                 REAL/TIME/INTEGER/BOOLEAN vectors */
int vhSize(const vhOpenArray h, int d);
int vhDimensions(const vhOpenArray h);
```

where d is number of argument (d=0 refers to first dimension i.e for array of STD_LOGIC_VECTOR it will be STD_LOGIC_VECTOR , for REAL_VECTOR → REAL)

Additional functions:

```
void *vhGetArrayPtr(const vhOpenArray); /* returns pointer to first array first element for
                                          read or write values */
int vhSizeOfArray(const vhOpenArray); // total size in bytes
```

In case of arrays and records number of elements and type must match. In Case of SystemC C99 types may be used.