. handling attention . When revising it, the old edition is abandoned.

Specifications

SC-HEAP E3
Python I/F Function specifications

Version 2.0-J

. Stencil paper inspection approval and end .

LLWEB-00105192		
MSS-SG-12-0062-02		
2012/9/5		
Software generalization part Soft tool		
Approval	Inspection	Making
Sato	---	Arai

Location of this book

This book **SC-HEAP_E3** [No] **Python I/F** The one that the specification was provided.

Refusal

The content of this book is subjected to variation later.
The patent search of the technology described in this book is not done.

INDEX

1. Outline

Book Python I/F To prospective ..peel.. , SC-HEAP_E3 Simulation mode where [ni] is installed Dynamic switching (High speed/Height Accuracy switch)A dynamic parameter setting is achieved. I/F ..going out.. .

1.1. Policy

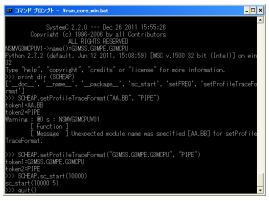
1 Python2.7 [Wo] is used.

2 Python [Ha] sc_main [Suru] of ..twining.. start.

```
sc_main{
    .
    mShPythonAPI.StartPy();
    . .
}
```

```
void ShPythonAPI::StartPy(){
    .
    if( gp_python_scr_file != stdin )
        PyRun_SimpleFile();
    else
        Py_Main( 1, argv );
    .
}
```

Script file input
or
Console start



3 Python The module name in which [ni] string is put up is "SC-HEAP"It makes it.

4 Python [Yo] comes to light twining. API It mounts by each module according to it is necessary to peel off.

1.2. Term explanation

Table1-1Term explanation

Words and phrases	Meaning
SC-HEAP_E3	RH850 [No]SystemC Simulator
Python	The NetherlandsGuido van Rossum Develop..open source. [E;kuto] aim script programming language.

2. Use class

2.1. ShPythonAPI

Explanation:

Python I/F Class.

Python [Wo]It controls. API [Wo] is offered.

Succession:

(It is not.)

ShPythonAPI

API	Outline
static void StartPy()	Python ..drinking.. start
static void DestructorPy()	Python ..drinking.. destructor
public Member variable	Outline
It is not.	

2.1.1. API

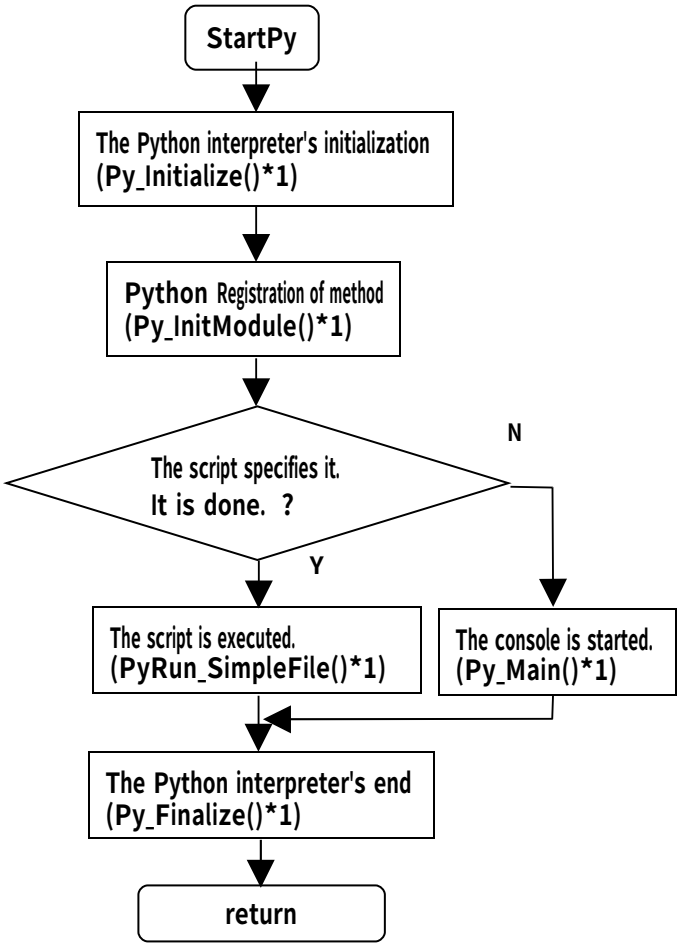
```
static void StartPy( int argc, char* argv[] )
```

Explanation:

Python Function for [wo] to start

Argument	Type	Explanation	Default value
1	int	Number of arguments	None
2	char*	Argument pointer array	None

Flow chart :



*1: Python [No] API

```
static void DestructorPy( void )
```

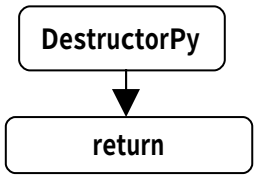
Explanation:

Python ..drinking.. destructor.

Especially, anything doesn't process the reality. The future **Python** When necessary processing is generated when ending, it describes it.

Argument	Type	Explanation	Default value
	void		

Flow chart :



3. It uses it. Python [De] is defined. API And, the type.

1. ShPythonAPI The classIt uses it. Python I/F When [wo] is constructed, it uses it. Python [De] is being offered. API Below [wo] It shows.

API	Explanation	Remarks
PyArg_ParseTuple (Pointer of argument, C++ ..drinking.. [kata], Variable poi that stores argument taken out [Nta],...)	Python ..drinking.. the argument of the method C++ It drinks and it takes it out with [kata].	It registers. Python In the method When the argument is analyzed, it uses it.
Py_BuildValue (Character string)	Python It drinks and it calls it from the method. It was done. C++ ..drinking.. function is [kae] as for the value. It uses it to do.	Book I/F [Nioiteha] and the argument are eternal. It uses it by [ni] ""

2. ShPythonAPI When the class is used, the user needs it. Python The type in which [de] is defined is shown below.

The following types Python.h It is possible to use it by doing [wo] include.

Type name	Explanation	Remarks
PyMethodDef	Python With ..drinking.. method C++ [No] API [Womusu] Structure that puts up [bi].	Python Insertion from console or script From the power [sareta] method C++ [No] API [Wo] call It uses it to do.
PyObject	Python It ..object.. receives it drinking. C++ ..drinking.. [kata].	Python と C++ Delivery of object drinking It uses it.

4. ShPythonAPI The class was used. Python I/F „drinking„ [jitsu] Present method

4.1. Preparation

4.1.1. [Kaku] IP [Deno]Preparation for method

1 The set function of the parameter is prepared.
2 The set function Python Trumpet of „twining„ [yoba] to compete API [Wo] is prepared.
3 The instance name and the instance pointer are named in the string.
4 Python With „drinking„ method API [Wo] string is put up.

1.1.1.1. BusSlave If Embodiment in class

STATIC The declaration part is described as follows.

BusSlave If.h

```
#include <Python.h>

template<unsigend int BUSWIDTH> class BusMSlaveIf {
.
static PyMethodDef mShApiDef[];
static std::map<std::string, BusSlaveIf<BUSWIDTH> *> mMapBusSlaveIf;
static PyObject* setReadLatencyPy( PyObject* self, PyObject* args );
void setReadLatency( sc_dt:uint64 latency );

     static void SetPyExtCmd( void );
.
};
```

Include

Python With „drinking„ method API Definition group ④ that relates [wo]

Name and poi of instance [Nta] is put up in the string. map③

Python [Yo] comes to light twining. Trumpet API②

The value is actually operated. Set function ①

mShApiDef [Wo] Python It registers as a command. Function ④

The declaration of substance is described as follows.

BusSlaveIf.h [1]

```
// Constructor
template<unsigned int BUSWIDTH> BusSlaveIf()
{
    .
mMapBusSlaveIf[this->name()] = this;
.
}

// Python  With „drinking„ method API Definition group storage array to which [wo] is related 4.1.1
PyMethodDef BusSlaveIf<32>::mShApiDef[] = {
{. BusSlaveIf32_setReadLatency",setReadLatencyPy,METH_VARARGS," ~ "},
{NULL,NULL,0,NULL}
};

//  With the instance name this  The pointer is put up in the string. map4.1.1
template<unsigned                                    int                                    BUSWIDTH>
std::map<std::string,BusSlaveIf<BUSWIDTH>*>
BusSlaveIf<BUSWIDTH>::mMapBusSlaveIf;

// Python  Trumpet of „twining„ [yo] that comes to light API 4.1.1
PyObject* BusSlaveIf<32>::setReadLatencyPy(PyObject* self, PyObject*
args)
{
char *token;
sc_dt:uint64 *latency;

if( PyArg_ParseTuple( args, "si", &token, &latency ) ){
if( mMapBusSlaveIf.count( token ) ){
(mMapBusSlaveIf[token1])->setReadLatency( latency );
    }
}
else{
//parse error
}

return Py_BuildValue("");
};

//  Set function that actually operates value 4.1.1
void BusSlaveIf<32>::setReadLatency( sc_dt:uint64 latency )
{
    ~ Omission~
};

// mShApiDef  [Wo] Python  Function ④ registered as command
void BusSlaveIf<32>::SetPyExtCmd( void )
{
Py_InitModule( . SCHEAP", mShApiDef );
}
```

With the instance name **this** String putting up ③ of the pointer.

Method name (Class name _API It is assumed the name. [Ru].)

API

help Character string for which [de] is used (It reflects in the example for the convenience of space. Abbreviation)

Instance name (第 1 Argument) Inn of „twining„ [po] [Ta] is calculated, and the set function is called.

The parameter setting processing is described.

Book I/F Then, the module name :. **SCHEAP** It makes it.

[1] BusSlaveIf Because „peel„ [tenpure-tokurasu] header [Ni] has been described, and substance : for a usual class. .cpp [Ni] is described.

1.1.1.2. main Embodiment when [ni] method is registered

extern The declaration part is described as follows.

main.h

```
extern void Se    tPyExtCmd( void );
```

mShApiDef [Wo] Python
It registers as a command.
External reference declaration ④ of function

The declaration of substance is described as follows.

main.cpp

```
#include <Python.h>

// Python   With „drinking„ method API Definition group storage array to which [wo] is related 4.1.1
static PyMethodDef mShApiDef[] = {
{. sc_start",sc_startPy,METH_VARARGS,"sc_start"},
{NULL,NULL,0,NULL}
};

// Python   Trumpet of „twining„ [yo] that comes to light API 4.1.1
static PyObject* sc_startPy(PyObject* self, PyObject* args)
{
int cycle_i;

PyArg_ParseTuple (args, "i", &cycle_i);

sc_start( cycle_i*glb_cleck_period, glb_cleck_time_unit );
     return Py_BuildValue("");
} ;

// mShApiDef   [Wo] Python  Function ④ registered as command
void SetPyExtCmd( void )
{
Py_InitModule( . SCHEAP", mShApiDef );
}
```

Include

Method name
(Class name _API It is assumed the name.
[Ru].)

API

help Character string for which [de] is used

The argument C++ To „drinking„ [kata]
It converts and it takes it out.

sc_start() [Wo] call

Book I/F Then, the module name :. SCHEAP It makes it.

4.1.2. ShPythonAPI The method is registered to the class.

1 The method is registered.

1.1.1.3. Embodiment

STATIC The declaration part is described as follows.

ShPythonAPI. cpp

```
void ShPythonAPI::LoadShCommandDefPy(void)
{
     SetPyExtCmd( void );
BusSlaveIf<32>::SetPyExtCmd( void );
.
};
```

The method is registered ①.

4.1.3. main [Niokeru] Python Preparation for start part

1 Python I/F The substance of the class is made. 。
2 Python The interpreter is started. 。
3 Python After the interpreter is ended, the destructor is called.

1.1.1.4. Embodiment

Python Interpreter's start-end part It describes it below [wo].

main.cpp

```
#include . ShPythonAPI.h"

int sc_main( int argc, char **argv ){
      .
ShPythonAPI mShPythonAPI;
.
      mShPythonAPI.StartPy( int argc, char *argv[] );
      mShPythonAPI.DestructorPy();
.
};
```

Python I/F Substance ① of class

Python Interpreter start ②

Python Interpreter Destructor ③

4.2. Execution Call at time Relation

1 From the console or the script file Python The method is input.
2 mShApiDef Trumpet to which [de] is related API However, the call is done.
3 Trumpet API The function of the more set is called.

4.2.1. main Example of method of mounting [de]

console

```
>>> SCHEAP.sc_start(10000)
```

①

main

```
PyMethodDef mShApiDef[] = {
{      . sc_start",sc_startPy,METH_VARARGS,"sc_start"},
{NULL,NULL,0,NULL}
};
```

main

```
static PyObject* sc_startPy      ( PyObject* self, PyObject* args )
{
    .
sc_start( cycle_i*glb_cleck_period, glb_cleck_time_unit );
.
}
```

②

SystemC [No]
sc_start() [Wo]
Call ③

4.2.2. BusSlave If Example of mount in class method

console

```
>>> SCHEAP. BusSlaveIf32_setReadLatency("RH850.GAPB_SLAVE_SG0.ts_if",10)
```

①

BusSlaveIf

```
PyMethodDef mShApiDef[] = {
{      . BusSlaveIf32_setReadLatency",setReadLatencyPy,METH_VARARGS,"~"},
{NULL,NULL,0,NULL}
};
```

Instance name

BusSlaveIf

```
static PyObject*     setReadLatencyPy( PyObject* self, PyObject* args )
{
    .
mMapBusSlaveIf[token1]->setReadLatency( latency );
.
}
```

②

From the instance name
The pointer is calculated.
The set function is called.

BusSlaveIf

```
void setReadLatency    ( sc_dt:uint64 latency )
{ ... }
```

③

5. Notes

1 Python [Nioite] PyMethodDef The method of registering [ni]class. It will specify it for a method name. 、 object. It is not possible to specify it, that is, the method name. When paraphrasing it PyMethodDef [Ha] static Should the array. 、 object [Ni] method name cannot be registered by arranging it. The object name like the example of the following therefore 1 In the argument The method for passing it is adopted.

. . . . . SCHEAP.setTlmTransType(. G3MSS.G3MPE.G3MCPU.CAISS.PE1_isl_if","SIM_MODE_CA")

2 Python With ..drinking.. method API [Wo] string is put up. PyMethodDef ..drinking.. variable the name mShApiDef[] It is assumed [de] fixation.

3 Python [Yo] comes to light twining. API It is classing ..principle ".. name as ..peel.. become unique ,. _API D o .

6. Error message

It is not.

7. Limitations

(1) MULTI When connecting it, only the command input from the script is supported. Moreover, this command input is a simulator. It starts. 1 It is assumed only times.

(2) MULTI The script file name of the connection cannot be specified. [2]
The script file name scheap.py It becomes fixation.

[2] MULTI When connecting it, the simulator. (sim.x/.exe The user must not start directly but ..peelr.teserv2 The [sarerutame] of ..twining.. start start option The purpose is not to be able to give it.)

8. Reference literature

It is not.

Number of versions	Date	Revision reason	Approval	Making
V1.0	2012.7.31	1st issue	Sato	Arai
V2.0	2012.9.5	Python When you register the enhancing command Py_InitModule() [Wo] ShPythonAPI From the method called directly in the class, Python In the enhancing command use class Py_InitModule() [Wo] piece It [ru]s it. set The definition perilla of the function. set The function ShPythonAPI To call it in the class, it changes.	Sato	Arai

Revision history

VSB NPB The remainder one cycle 12