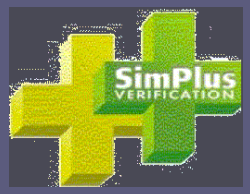
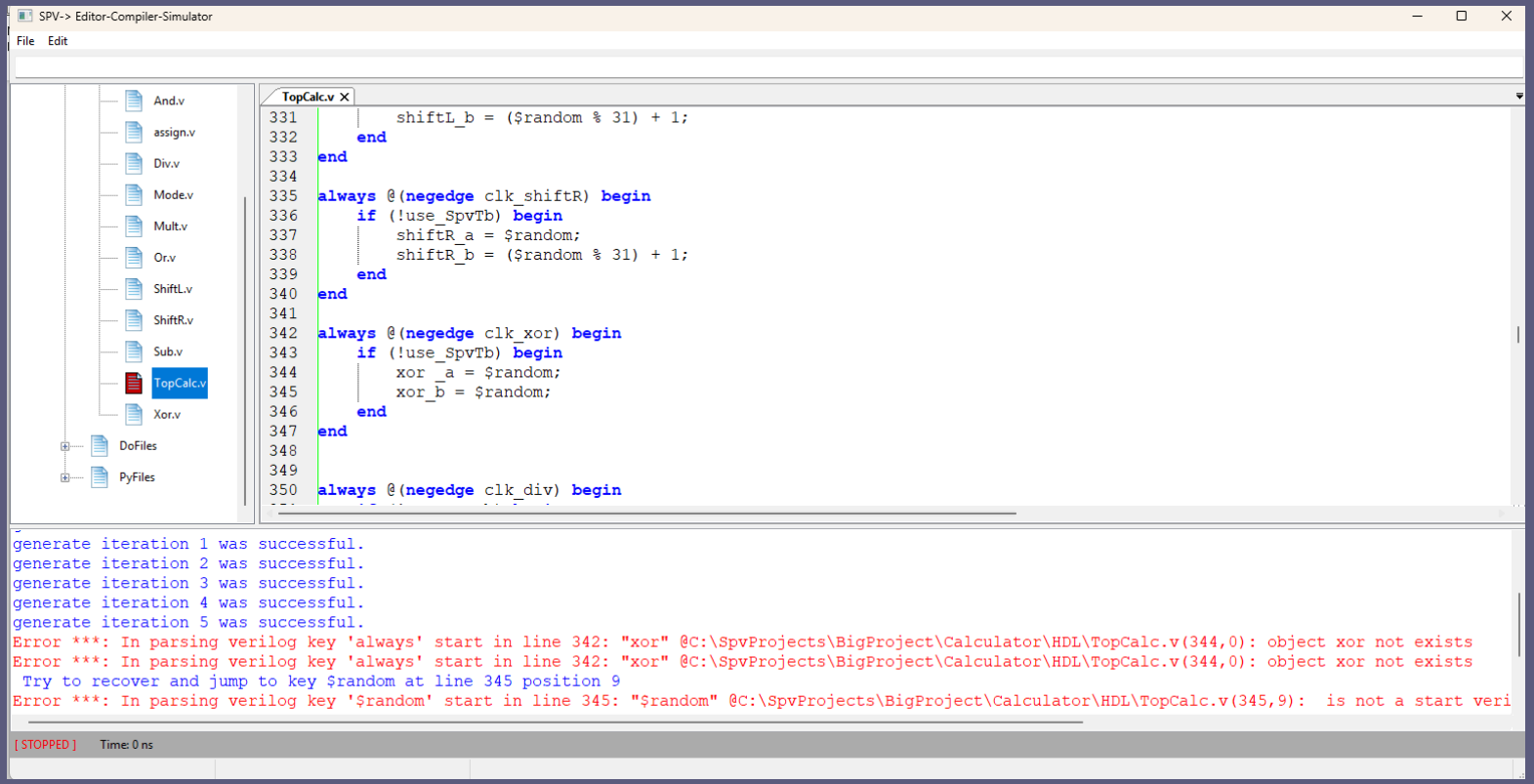
SPV

SimPlus  Verification

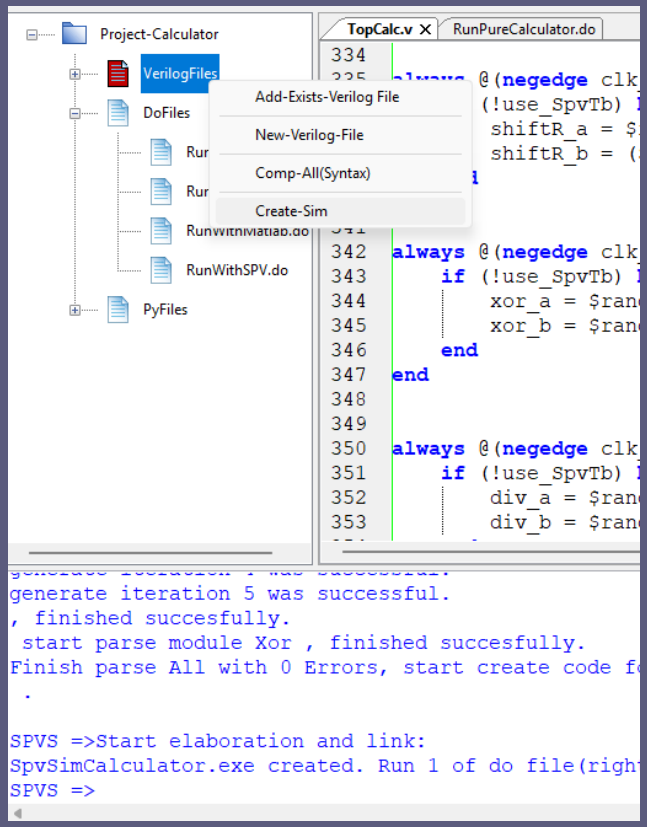
Integrated Environment for Digital Circuit Design and Verification

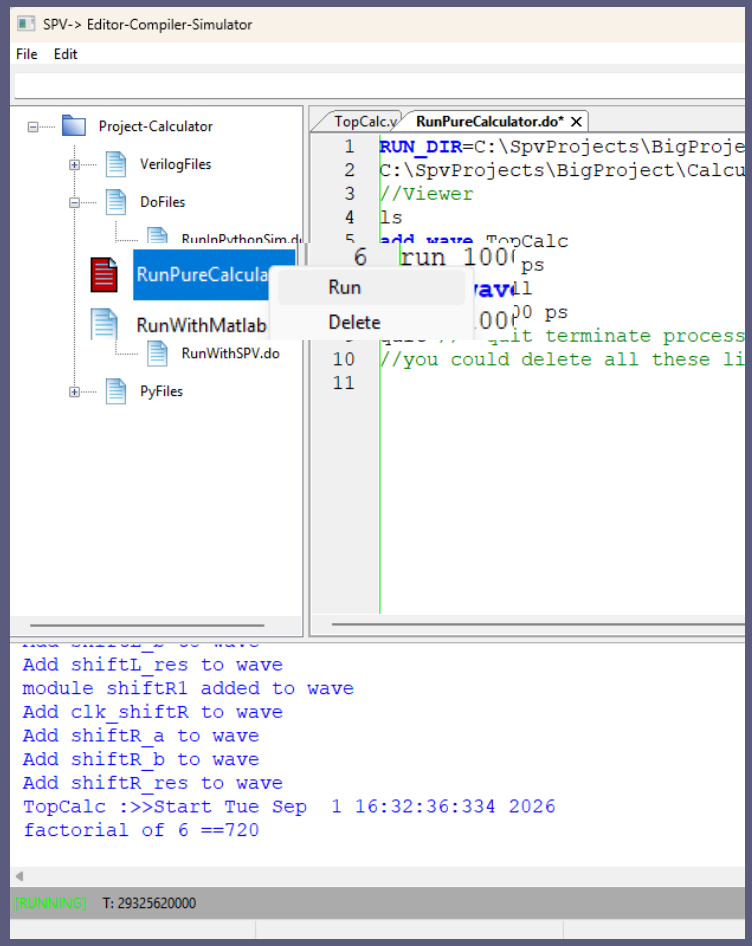
Shmuel Amrusi

Verilog Editor & Compilation Environment



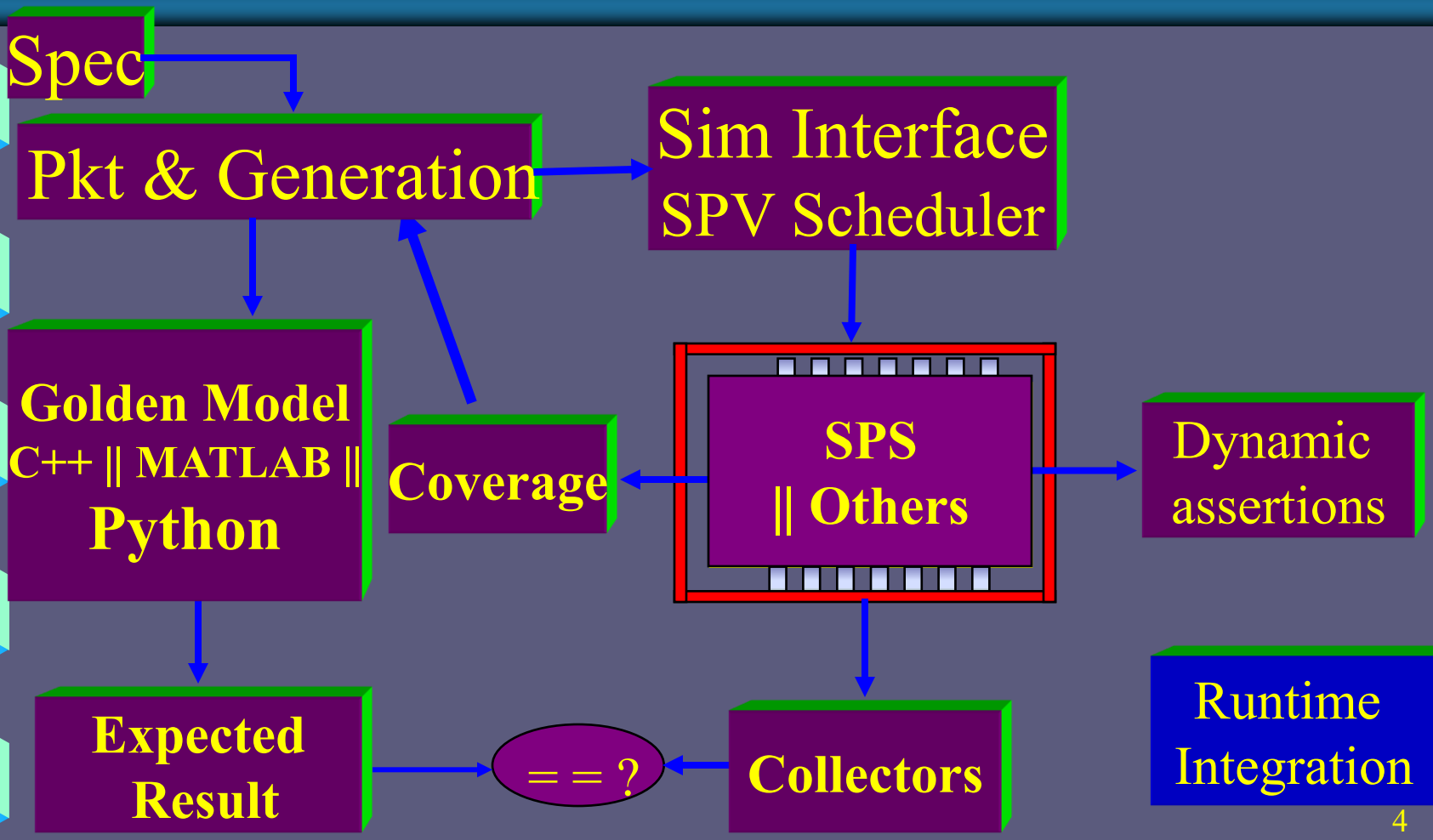
Unified Workspace for Code, Compilation, and Simulation



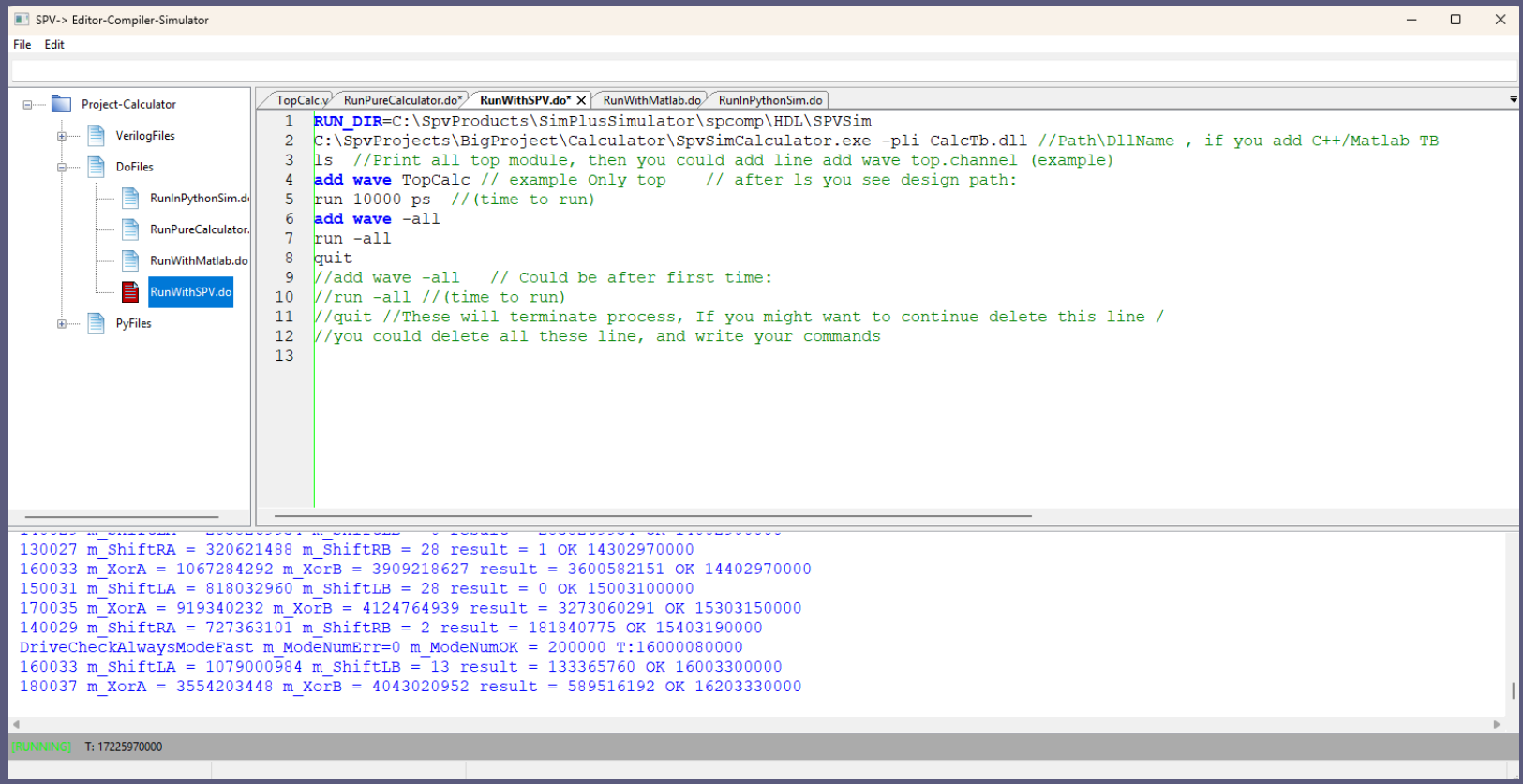


Testbench

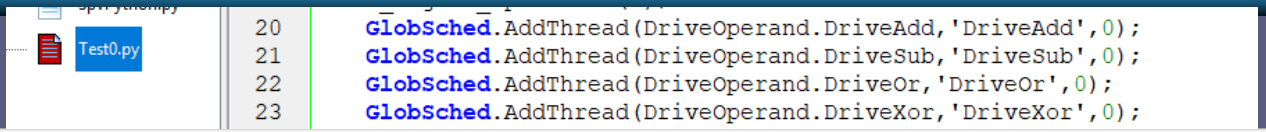
SPV Verification <=> Design



Unified Workspace for Code, Compilation Simulations, and Verification.



Scheduler and HDL interface Python



TopCalc.v | RunPureCalculator.do* | RunWithSPV.do* | RunWithMatlab.do | RunInPythonSim.do | DriveOperand.py*

```
#import SPSched
from SpvPython  import *
from AllSig     import *
import          random
from random import randrange #, uniform
from datetime import datetime
from SPV import *

LoopSize = 80001;
def DriveAdd(id: int):
    PR(threading.current_thread().name + " thread id = " + str(id) + " start at Time: " + str(SPV.SimTime()));
    m_AddClkNeg = SPSigEvent(m_Sigclk_add,AtNeg);
    GlobSched.WaitEv(m_AddClkNeg);
    numOk=0;numErr=0;
    for ActionInd in range (0,LoopSize):
        GlobSched.WaitEv(m_AddClkNeg);
        #GlobSched.WaitTime(40);
        if (ActionInd > 0) :
            resHdl  = (m_Sigadd_res.Val32())&0xFFFFFFFF;
            if (res != resHdl):
                PR("DriveAdd:   Error at ind " + str(ActionInd) + " adda = " + str(adda) + " addb=" + str(addb) + " res="
                numErr = numErr+1;
            else:
                numOk = numOk+1;
        adda = SPV.Gen()%0x6FFFFFFF;
        addb = SPV.Gen()%0x6FFFFFFF;
        res  = (adda + addb)&0xFFFFFFFF;#Expected
        m_Sigadd_a.Set(adda);
        m_Sigadd_b.Set(addb);
    PR("DriveAdd:   numErr=" + str(numErr) + " numOk=" + str(numOk));

def DriveSub(id: int):
    import pdb; pdb.set_trace();
```

Scheduler and HDL interface C++

```
cout << " DO_FAST_PROC=FALSE Mode loopSize = " << loopSize << endl;
StartTProc(this, (SpvProcMethod)&Driver::DriveCheckAlwaysOrTP);
StartTProc(this, (SpvProcMethod)&Driver::DriveCheckAlwaysAddTP);
StartTProc(this, (SpvProcMethod)&Driver::DriveCheckAlwaysSubTP);
StartTProc(this, (SpvProcMethod)&Driver::DriveCheckAlwaysMultTP);
StartTProc(this, (SpvProcMethod)&Driver::DriveCheckAlwaysAndTP);
```

```
void        Driver::DriveCheckAlwaysAddTP()
{
    for (unsigned i = 0; i < loopSize; i++) {
        if (i) {
            Wait(m_ClkAddNeg);
            if (m_AddRes != SH::m_Sigadd_res) {
                cout << i << ": Error ***: m_AddA = 0x" << hex << m_AddA << " m_AddB = 0x" << m_AddB << " SPV result = 0x" << m_AddRes
                m_AddNumErr++;
            }
            else {
                m_AddNumOK++;
            }
        }
        m_AddA = GenUnsigned(); m_AddB = GenUnsigned(); m_AddRes = m_AddA + m_AddB;
        SH::m_Sigadd_a = m_AddA; SH::m_Sigadd_b = m_AddB;
    }
    cout << "DriveCheckAlwaysAddTP m_AddNumErr=" << m_AddNumErr << " m_AddNumOK = " << m_AddNumOK << endl;
}
```

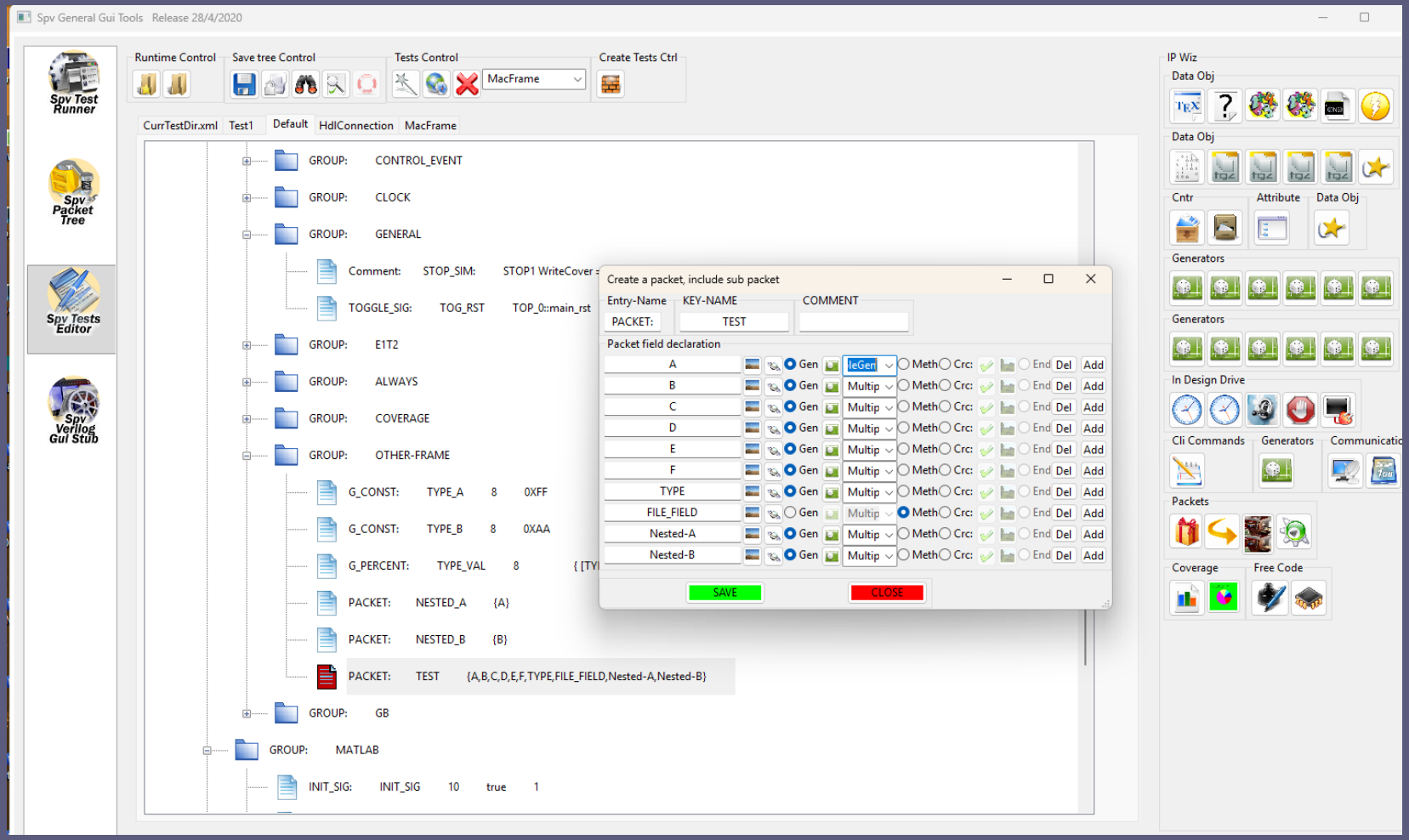
Scheduler and HDL interface MATLAB

```
bool MatCalcInf::PrepareAddVector(vector<unsigned>& a, vector<unsigned>& b, vector<unsigned>& results, unsigned numObj)
{
    m_MlNumObj[0] = numObj;
    m_MlResults.Destroy();
    bool isOk = mlfPrepareListAdd(1, m_MlResults, m_MlNumObj);
    if (!isOk)  return false;
    return PrepareData(a, b, results, numObj);
}
```

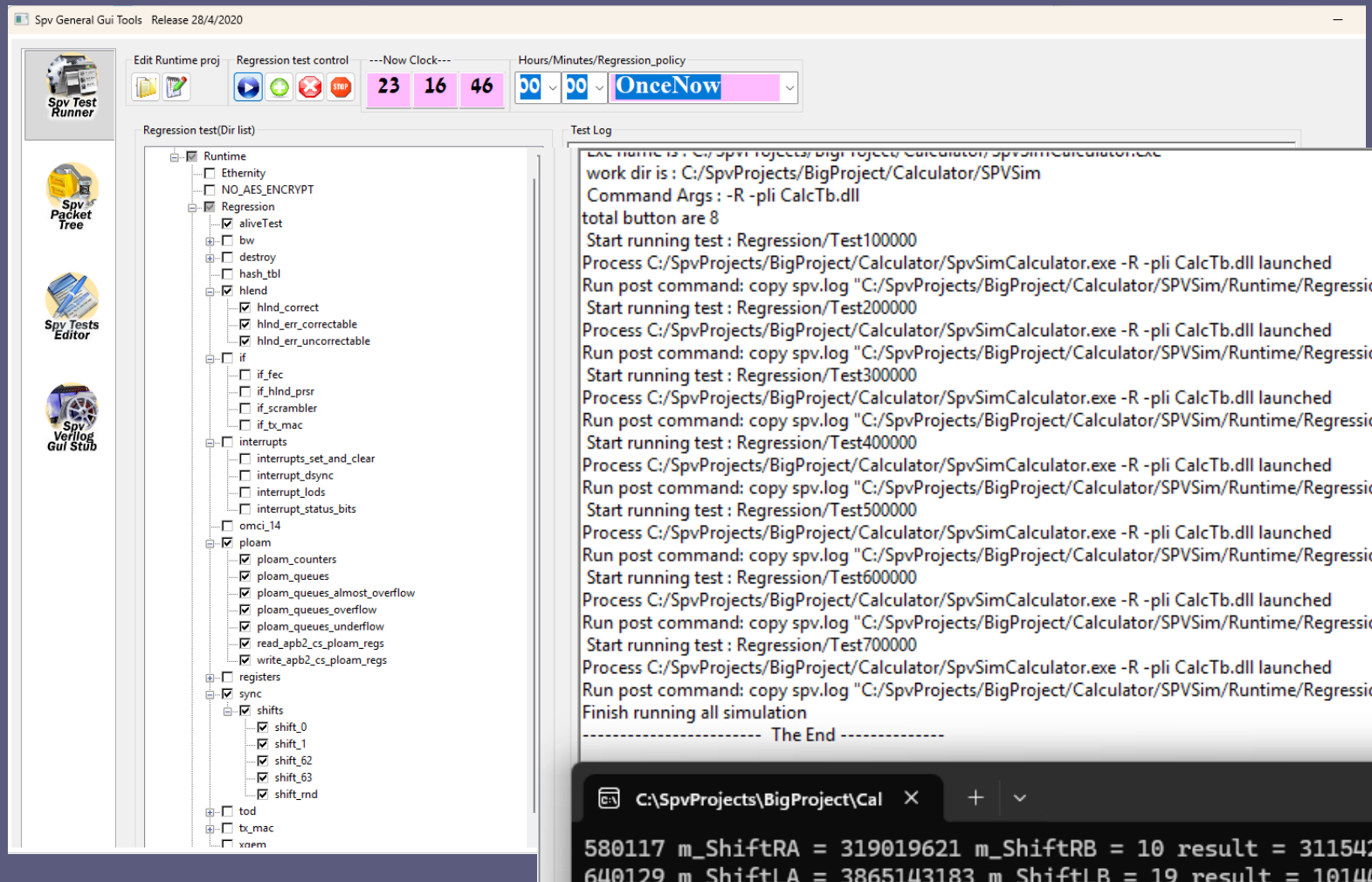
```
void     Driver::DriveCheckAlwaysOrFast()
{
    static unsigned cnt = 0, ind = 0;
    if (cnt == 0) m_MatCalcInf->PrepareOrVector(m_OrAVec, m_OrBVec, m_OrResVec, m_VecSize);

    if (cnt++) {
        if (m_OrResVec[ind] != SH::m_Sigor_res) {
            cout << cnt << ": Error ***: m_OrA = 0x" << hex << m_OrAVec[ind] << " m_OrB = 0x" << m_OrBVec[ind] << " SPV result = 0x"
            m_OrNumErr++;
        }   else {
            m_OrNumOK++;
        }
        if (++ind == m_VecSize) {
            m_MatCalcInf->PrepareOrVector(m_OrAVec, m_OrBVec, m_OrResVec, m_VecSize);
            ind = 0;
        }
    }
    //Set data and calculate expected for next negedge.
    SH::m_Sigor_a = m_OrAVec[ind]; SH::m_Sigor_b = m_OrBVec[ind];
    if (cnt > loopSize) {
        CurrAProc()->Kill();
        cout << "DriveCheckAlwaysOr m_OrNumErr=" << m_OrNumErr << " m_OrNumOK = " << m_OrNumOK << endl;
    }
}
```

Visual Interface: Test Editor and Regression Manager



Visual Interface: Test Editor and Regression Manager



Debug C++
Debug Python

```
129  void     Driver::DriveCheckAlwaysOrFast()
130  {
131      static unsigned cnt = 0;
132      if (cnt++) {
133          if (m_OrRes != SH::m_Sigor_res) {
134              cout << cnt << ": Error ***: m_OrA = 0x" << hex << m_OrA << " m_OrB = 0x" << m_OrB << " SPV result = 0x" << m_OrRes << " HDL m_Sigor_res
135              m_OrNumErr++;
136          }
137          else {
138              m_OrNumOK++;
139          }
140      }
141      //Set data and calculate expected for next negedge.
142      m_OrA = GenUnsigned(); m_OrB = GenUnsigned(); m_OrRes = m_OrA | m_OrB;
143      SH::m_Sigor_a = m_OrA; SH::m_Sigor_b = m_OrB;
144      if (cnt > loopSize) {
145          CurrAProc()->Kill();
146          cout << "DriveCheckAlwaysOr m_OrNumErr=" << m_OrNumErr << " m_OrNumOK = " << m_OrNumOK << " T:" << SimTime() << endl;
147      }
148  }
```

119 % | No issues found | Ln: 143 Ch: 15 Col: 18

Watch 1

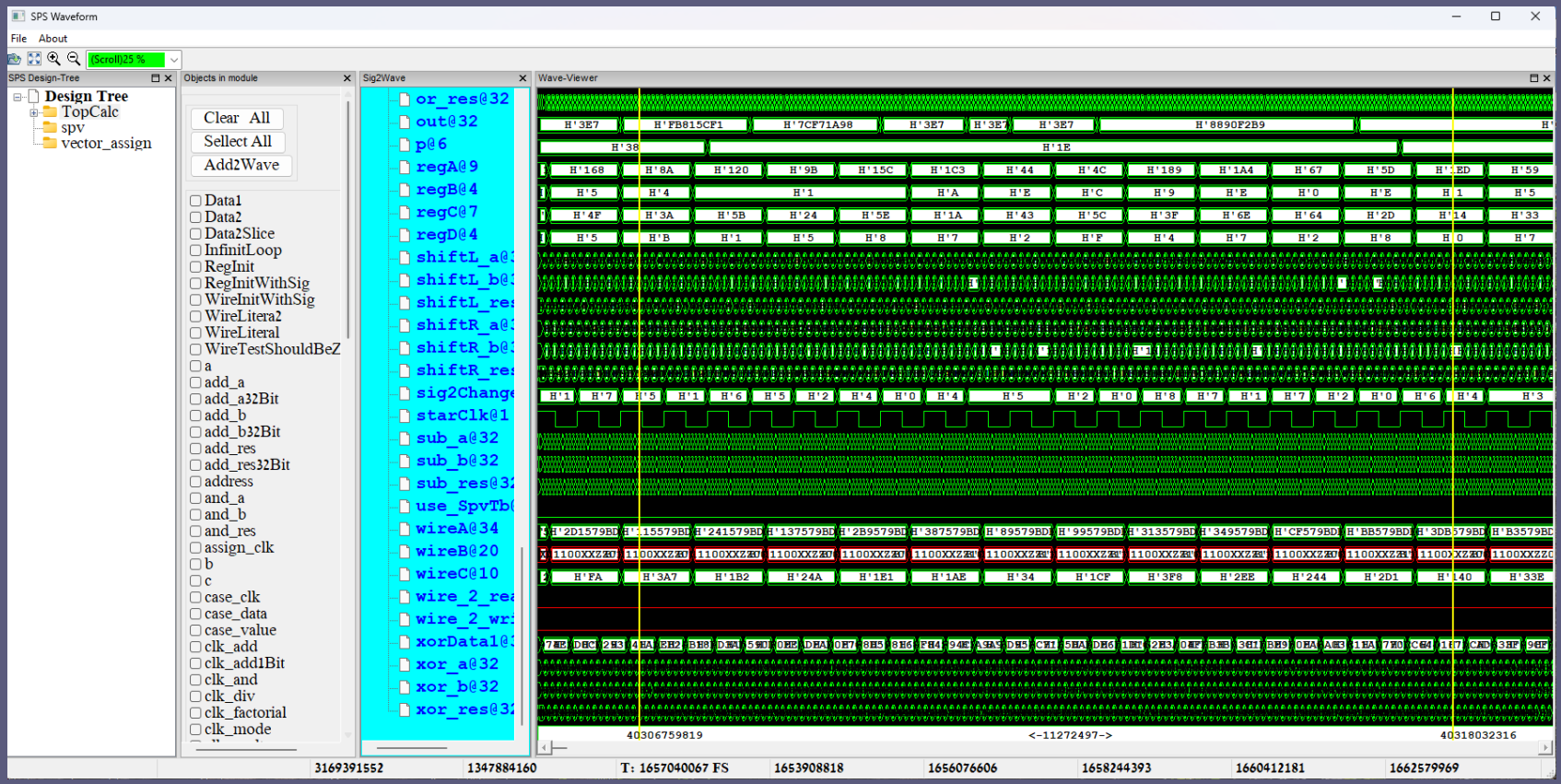
Search (Ctrl+E) | Search Depth: 3

Name	Value	Type
SimTime()	30000	const unsigned __int64 &
SH::m_Sigor_a	{my_Name="TopCalc.or_a" pSim_Type=0x0000023533cde470 {mIsSigExist=true my_Father=0x00007fffc1dc6f30 {CalcTbDB.dll!SpvSig SH::m_Sigor_a} {...} ...} ...}	SpvSig
SpvMonitor	{m_IsSig=true }	SpvMonitor
my_Name	"TopCalc.or_a"	std::string
pSim_Type	0x0000023533cde470 {mIsSigExist=true my_Father=0x00007fffc1dc6f30 {CalcTbDB.dll!SpvSig SH::m_Sigor_a} {...} ...}	spv_Sim_Type *
m_Uval	0x0000023533cde498 {0}	unsigned int *
m_64Val	0x0000023533cde498 {0}	unsigned __int64 *
m_Size	32	unsigned int

Output | Watch 1

```
> c:\spvprojects\demoldir\pysim\driver.py(25)StartDrive()
-> for ActionInd in range (0,LoopSize):
(Pdb) n
> c:\spvprojects\demoldir\pysim\driver.py(26)StartDrive()
-> genVal = SPV.Gen();
(Pdb) n
> c:\spvprojects\demoldir\pysim\driver.py(27)StartDrive()
-> m_Sigbusa.Set(genVal);
(Pdb) p genVal
1003066593
(Pdb) p LoopSize
10000
(Pdb)
```

Wave viewer


SPS Waveform
File
About
SPS Design-Tree
Design Tree
TopCalc
spv
vector_assign
Objects in module
Clear All
Sellect All
Add2Wave
Data1
Data2
Data2Slice
InfinitLoop
RegInit
RegInitWithSig
WireInitWithSig
WireLitera2
WireLiteral
WireTestShouldBeZ
a
add_a
add_a32Bit
add_b
add_b32Bit
add_res
add_res32Bit
address
and_a
and_b
and_res
assign_clk
b
c
case_clk
case_data
case_value
clk_add
clk_add1Bit
clk_and
clk_div
clk_factorial
clk_mode
Sig2Wave
or_res@32
out@32
p@6
regA@9
regB@4
regC@7
regD@4
starClk@1
sub_a@32
sub_b@32
wireA@34
wireB@20
wireC@10
xorData1@1
xor_a@32
xor_b@32
Wave-Viewer
40306759819
<-11272497->
40318032316
3169391552
1347884160
T: 1657040067 FS
1653908818
1656076606
1658244393
1660412181
1662579969

The End
Thanks