**Steps for Generating the CUDA version of the Circuit Satisfiability Problem Using IPT**

c557-003$ IPT circuit.c

NOTE: We currently support only C and C++ programs.

Please select a parallel programming model from the following available options:

1. MPI

2. OpenMP

3. CUDA

**3**

Please enter the function in which you wish to insert the kernel call(or parallelize the for-loop).

Please choose the function that you want to parallelize from the list below

1 : main

2 : circuit\_value

3 : i4\_to\_bvec

4 : timestamp

**1**

Would you like

1. For-loop

2. TBD

**1**

for (i = 1; i <= n; i++) {

ihi = (ihi \* 2);

}

Is this the for loop you are looking for?(y/n)

**n**

OK - will find the next loop if available.

for (i = 1; i <= (n / 2); i++) {

printf("\ntest\n");

}

Is this the for loop you are looking for?(y/n)

**n**

OK - will find the next loop if available.

Note: With your response, you will be selecting or declining the parallelization of the outermost for-loop in the code region shown below. If instead of the outermost for-loop, there are any inner for-loops in this code region that you are interested in parallelizing, then, you will be able to select those at a later stage.

for (i = 0; i < ihi; i++) {

i4\_to\_bvec(i,n,bvec);

value = circuit\_value(n,bvec);

if (value == 1) {

solution\_num = (solution\_num + 1);

printf(" %2d %10d: ",solution\_num,i);

for (j = 0; j < n; j++) {

printf(" %d",bvec[j]);

}

printf("\n");

}

}

Is this the for loop you are looking for?(y/n)

**y**

Do you want to perform reduction on any variables ? (Y/N)

**y**

Please enter the variables to reduce ([ format: 1,2,3 etc. ] with 1 is for the first variable, 2 is for the second variable and so on).

Possible variables to reduce are:

1. n type is int

2. value type is int

3. solution\_num type is int

4. print\_statement\_deleted\_here type is int

**3**

Please enter the reduction operation for variable [solution\_num]. Possible reduction operations are:

1. Sum

2. Product

**1**

Is the following array [ bvec ]

1. Input , 2. Output 3. Input/Output 4. Neither Input nor Output

**4**

Running Consistency Tests