CS223 : Assignment 7, Date of Demonstration 20th April 2018, Lab Timing (2PM-5PM), 12% weight

(Deadline for submission of assignment by sending zip file (of HDL codes, UCF/contrints file, C files, test bench file) to aryabartta<@>gmail.com is 18th April 2018 (Strictly upto 19th Morning 9.00AM IST).

Please ensure that you do not submit plagiarized code from your peer group or indirectly from any other website. If two group copies code from same external website and submit then both groups will get F grade. Result of plagiarism test will be informed after the demonstration.

Design and implement High-speed Intrusion Detector (ID system). Intrusion detector takes a stream of online words/characters and count the number of times the sensitive words occurs in the stream.

List of sensitive/restricted words can be “Bomb”, “bomb”, “Terror”, “terror”, “Kill”, “kill”, “gun”, “Gun”, “attack”, “Attack”, “blow”, “Blow”, “explode”, “Explode”, “violence”, “hijeck”, “Hijeck”, “Bomb”, “Bomb”, “shot”, “Shot”, …...

Assume you have M (say upto 256 or even higher) sensitive words. You send a stream of character from PC to FPGA and FPGA counts the number of each sensitive words. Your ID system on FPGA should be able to clear the word count, display the sensitive word count (in decimal format on 7 Seg LEDs of FPGA) by selecting the sensitive word using switches of FPGA.

From PC, you should be able to configure the sensitive words and put into FPGA before sending the online stream of char to FPGA.

You should take benefits of parallelism to speed up the matching and use up to M parallel matchers/word comparator. You can think of the stream of online characters get replicated and feed to all the M parallel word comparators. It is mandatory to instantiate many word comparators in this assignment.

C type peudocode is given bellow:

```
===========================================================================
char Text[M][16]; //Assume all Restricted Word are of size 16 characters
Initialize(RestrictedWords);//By sending Data (M*16 bytes) from PC to FPGA
for(i=0;i<INIFINITY;i++){ // Text is T[0 to INIFINITY]
      y=T[i]; //or y=getchar(fp) and send through USB to FPGA;
      for(j=0;j<M;j++){ // Inner loop can be done in parallel
            Copy(Text[j],T[i-16],16);//Copy T[i-16:i] to Text[j]
                                   // Same can be done with sifting also
            if(Compare(RestrictedWords[j], Text[j])==TRUE) Ctr[j]++;
      }
      Display(Ctr[InputSwitchValue]);
      if (CLR) for(j=0;j<M;j++) Ctr=0; //Done in Parallel
}
============================================================================
```

As you need to implement this in hardware, you need to think efficient way to implement

- Copy(Text[j],T[i-16],16),
- Compare(Restricted[j], Text[j])
- Efficiently implement M increment operation Ctr[j]++ using k (<<M) adder as your FPGA may not have M inbuilt adder unit (DSP48).

You may safely assume at a time (for given value of *i*) number of true outcome of Compare(Restricted[j], Text[j])==TRUE is less than a small number k (say 10) as the number of restricted words in the document will be very small but important. Probability of Ctr[j]++ operation is small, so try to share a ADDER operator many Ctr[j]++ and use at max k ADDERs.

```
===============================================================
BIT[256]={0}//Indicate the outcome of comparison
for(j=0;j<M;j++){ // Inner loop can be done in parallel
            Copy(Text[j],T[i-16],16);//Copy T[i-16:i] to Text[j]
                                   // Same can be done with sifting also
            if(Compare(RestrictedWords[j], Text[j])==TRUE) BIT[j]=1; else
BIT[j]=0;
}
for(j=0;j<M;j++) {
//Doing all INCREMENT operation serially. Require only one ADDER, may be too
slow
      if (BIT[j]==1) Ctr[j]++;
}
==============================================================================
```

Efficiently implement Compare(RestrictedWords[j], Text[j]) can be done in full parallel or partially parallel or serially.

Size of the restricted word (RW) are upto 16 character, suppose restricted word have 6 characters (example “terror”), rest can be padded with “Dont care”. So the word terror can be represented as “terror☒☒☒☒☒☒☒☒☒☒”. In binary it can be represented as 16 characters along with 16 bit data (BIT DC[16]) indicated that some are don’t care character. Comparison of the restricted word RW with text word S can be in parallel or in serial manner

```
bit MI=1;//match Indicator
for(i=0;i<15;i++){   //Do in parallel, and can be designed a digital circuit
for the same
      MI= MI AND ( (RW[i] == S[i]) || DC[i]);      //XNOR operation and OR
operation
}
==============================================================================
```

A sample input file is given

http://jatinga.iitg.ernet.in/~asahu/cs223/ABC.txt

Read character by character from this file and send to FPGA.

==

Evaluation Procedure

All the member of the group need to be present at the time of Demonstration of the assignment. All the absent members will be awarded 0 marks for the assignment. Please **show your ID card** at the time of demonstration (as it is difficult to remember faces of all the 90 students of your batch).