**LEVEL 1:**

You should compare the test coverage and diagnosis resolution qualities of the 3 given TV sets (A-C, see files uploaded separately) for the bench circuit of C432.

**Part1: Quality Metrics**

Test Coverage:

Test Coverage was found by simulating the TV for the bench mark and analyzing the detected faults compared to total fault.

**Diagnosis Resolution:**

Diagnosis Resolution was found by reading the undetected faults list and entire fault list obtained from Each TV set. To understand true Resolution of the TVs. All three of them where compared using a Venn diagram.

Detailed resolution for each fault set was obtained.

**Part 2: Methodology of gathering data**

Test Coverage: Simulation was carried out on Atlanta-M on each TV set separately. Report File was analyzed for data interpretation.

Diagnosis Resolution: When simulation was carried out undetected fault list was obtained from .ud file. These .ud files where then compared with fault list for bench c432(.flt file) create various sets faults based on type of fault. This sorting was carried out in visual studio (c# compiler).The result was displayed in terms of Venn Diaagram.

**Experimental Setup:**

Windows OS, atalanta-M.exe, Visual Studio(C# compiler), cmd

**DATA GRAPH**

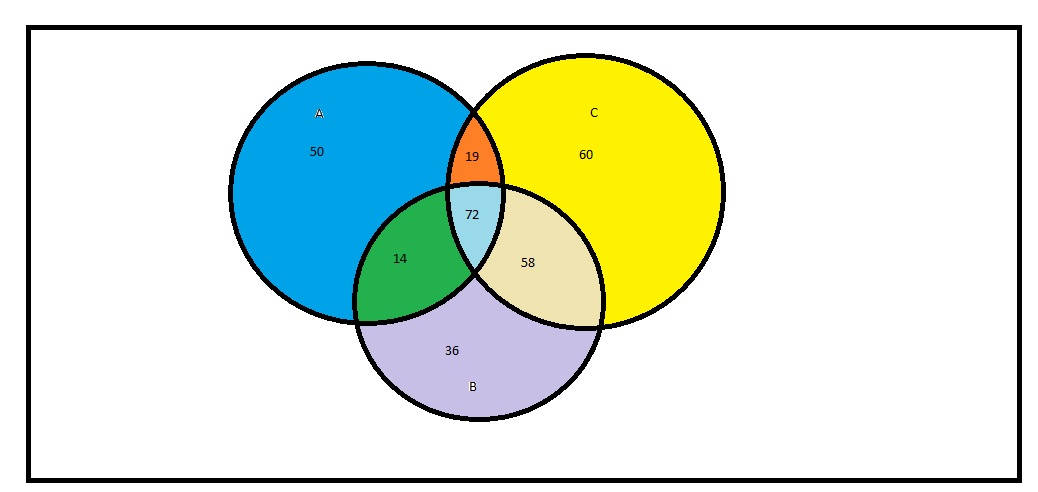
**TEST Coverage:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | TV Set A | TV Set B | TV Set C |
| Detected  Fault | 155 | 180 | 209 |
| Total Fault | 524 | 524 | 524 |
| Coverage % | 29.58 | 34.35 | 39.89 |

NOTE: One can see that Coverage of about 103 % is carried out by all TV Set combined. But it may be a wrong amount and we may still miss out some faults.

**Diagnostic Quality:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | SET A | SET B | SET C | Intersection(A,B) | intersection(B,C) | Intersection(A,C) | Intersection(A,B,C) |
| FAULT # | 155 | 180 | 209 | 86 | 130 | 91 | 154 |
| Coverage % | 29.58 | 34.35 | 39.89 | 16.4 | 24.8 | 7.36 | 29.33 |



**INFERENCE:**

TV SET C is the best and has highest Resolution.

Total detected fault by TVs= 251

Undetected Fault=3

**LEVEL 2:**

In addition to Level1, you should provide results for the test coverage and diagnosis resolution qualities of randomly generated test vectors for the 3 bench circuits of C880, C1355, and C7552. Particularly, you should examine whether (and how) enlarging the size of random TV set would (or would not) affect the qualities

**Part1: Quality Metrics**

**Test Coverage:**

Step1: Random Test Pattern generator was created by using VS C#. Three set of TV pattern was created each having 4, 16, 32 sets of TV.

Step2:Test Coverage was found by simulating the TV for the bench mark and analyzing the detected faults compared to total fault to each of three bench mark.

**Diagnosis Resolution:**

Diagnosis Resolution was found by reading the undetected faults list and entire fault list obtained from Each TV set for each Bench file. To understand true Resolution of the TVs. All three of them where compared using a Venn diagram. Three Venn diagrams were created.

Detailed resolution for each fault set was obtained for each Bench file.

**Part 2: Methodology of gathering data**

Test Coverage: Using Random Pattern generator program based in VS C#, 3 pattern files were obtained. These patterns were Simulated on each Bench file and Report was generated. Report File was analyzed for data interpretation.

**Diagnosis Resolution** (this Step was carried out three Times for each Bench File): When simulation was carried out, undetected fault list was obtained from .ud file. These .ud files were then compared with fault list for bench fault file (.flt file) to create various sets faults based on type of faults. The sorting was carried out in visual studio (c# compiler)program. The result was analyzed and displayed in terms of Venn Diagram.

**Experimental Setup:**

Windows OS, atalanta-M.exe, Visual Studio(C# compiler), cmd

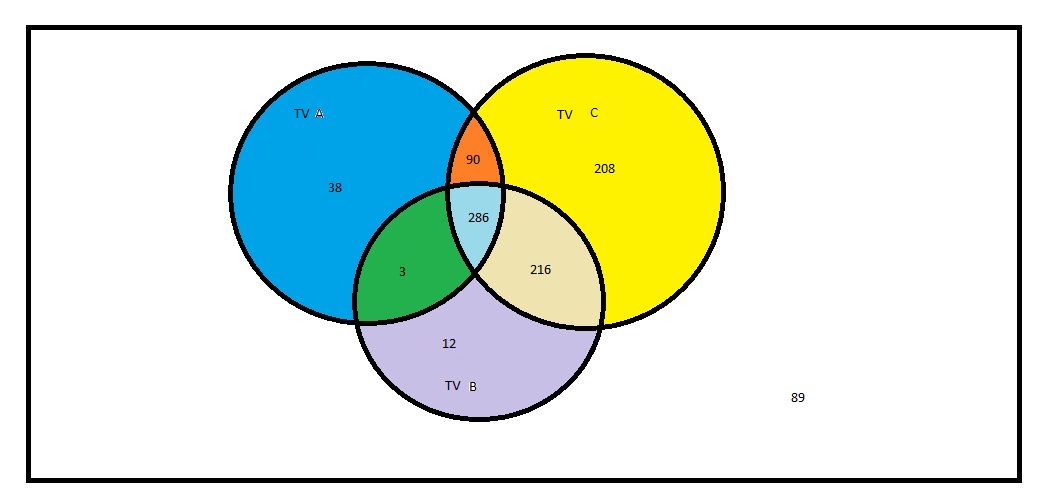
**DATA GRAPH**

**TEST Bench c880**

**TEST Coverage:**

NOTE: One can see that Coverage of about 184 % is carried out by all TV Set combined. But it may be a wrong amount and we may still miss out some faults.

**Diagnostic Quality:**

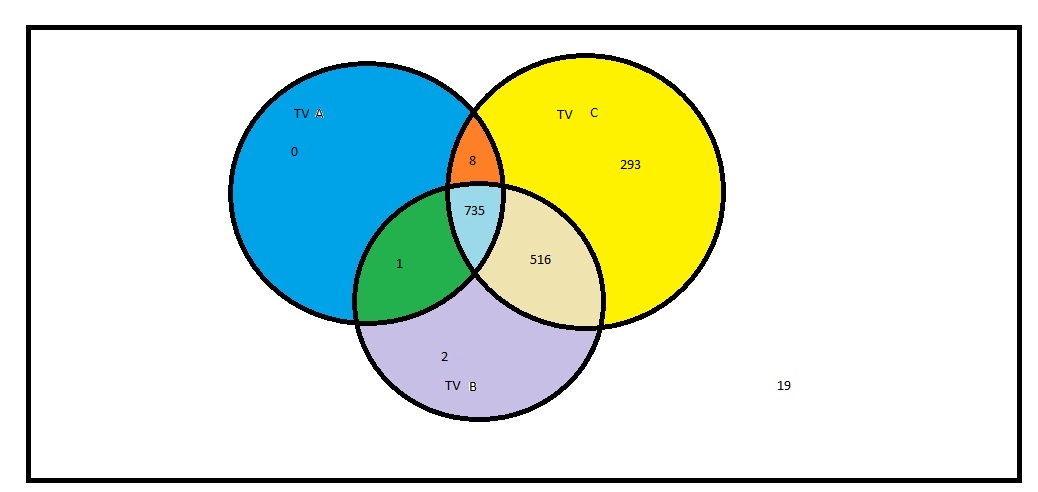


Undetected faults: 89

**TEST Bench c1355**

**TEST Coverage:**

Diagnostic Quality:



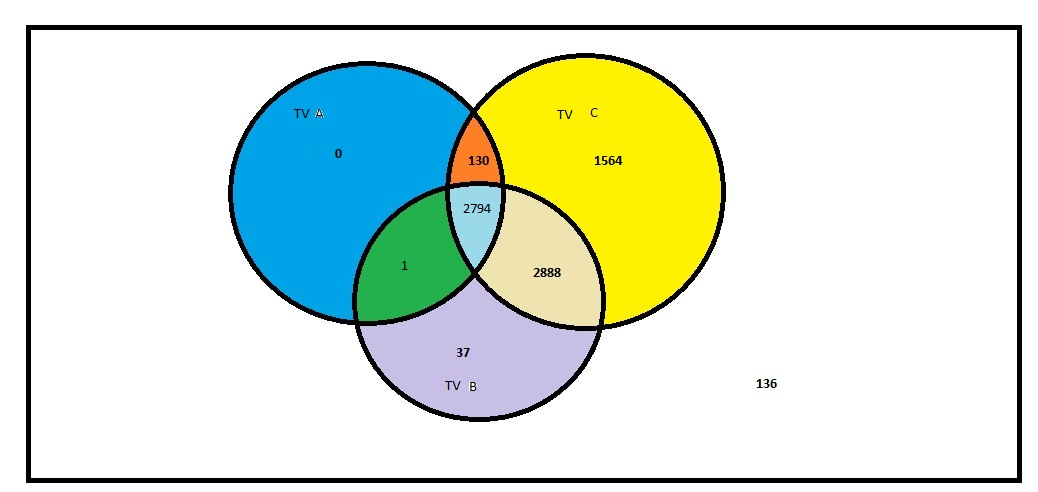
Undetected faults: 19

**TEST Bench c7552**

**TEST Coverage:**

NOTE: One can see that Coverage of about 184 % is carried out by all TV Set combined. But it may be a wrong amount and we may still miss out some faults.

Diagnostic Quality:



Undetected faults: 136

INFERENCE:

NOTE:

TV sets applied were in increasing order to check

TV SET C is the best and has highest Resolution.

Total detected fault by TVs= 251

Undetected Fault=3

LEVEL 3:

Come up with your own algorithm(s) of generating various sized TV sets of high quality in test coverage and diagnosis resolution, respectively. You should show results on the same 3 bench files of Level 2, and make meaningful comparisons to the data of random TV sets obtained in Level 2.

ALGORITHM:

IDEA:

TV Set can be of the size 2n where n is number of inputs. So we need to find a smart way to eliminate TV set from above pool. I suggest that a relationship table should be created between each i/p and o/p based on which i/p are most likely to affect a particular o/p. This is another way of seeing observability in the entire circuit. There are various observation that one can make after seeing the relationship within i/p and o/p. Test vector pattern can be then decided (or at least reduced largely).

Method:

This algorithm is divided into 2 steps:

Step 1: Generating a relationship table

First I read the bench file and generate Input, output and gate files which has the information about the respective inputs output and gate relation in terms of gate inputs and outputs. Then we create input to output relation as to how which input actually affects the output gate (it may be possible that few inputs may be unrelated for the output). The result we get has relation of output to input as well as how many times the input may affect the output. A simple way to imagine this part is by imagining a fan-out condition.

We also find for each output, lines that are responsible for alternating the output in some way. Again this relation may have multiple instances of the wire due to fan out. Having multiple instances of wire in some way gives how much number of times that line will be used to propagate the fault to that output.

Next from above to relation we calculate, final relation, wherein we find relation between each line and outputs viz. what are different ways to propagate faults into various outputs.

Now I group these faults based on the same pattern. Underlying principle being, faults having same pattern (most probably) have same TV sets. Another assumption that one can also make is that similar looking patterns(with difference of one-twos numbers) also most probably have same sets of test vectors (But we don’t exploit this factor).

Group is created based on the same patterns. Now we decide to make decision regarding the pattern.

Step 2: deciding TV based on the groups:

Groups that are generated have the lines which have same pattern. The highest number of value within that pattern is picked and chosen to be output where this faults will be sent in from.

Now input relation to that output is known. Each number corresponds to the input variability. We see the more numbers in the those patterns. Here highest number will have high variability while low numbers will have less variability for that faults.

There are some inferences that we can make, like if the number is 0 then obtaining fault at that particular output is independent of this input. While lower number represents that value of the input will not probably change a lot for faults coming out of that output.

So we pick higher values and vary them and create a subset of values possible for them. So for eg. if we select 4 inputs having higher readings then basically TV pool for those group will drop from 2n to 24 combination since all the other inputs are randomly selected to be constant(either 1 or 0)

For our case I only consider 1 TV for each group where in either less variable input are all 1 to simplify things for me. High variable inputs are randomly created (so that I randomly select from the subpool).