```
#!/usr/local/bin/perl5 -w

$input1 = 'chaintest.scan';
$output= "OUT";
$output2= "OUT.out";

open (INFILE,"$input1")||die "cannot open $input1";
open(OUTFILE,">$output")||die "canoot\n";
open(OUTFILE2,">$output2")||die "canoot\n";

#######################################################
        &preprocess;
open (INFILE2,"$output")||die "cannot open $input1";
        $/=";";
        $ct_scanout = 0;
        while (<INFILE2>){
        chomp;
        $ct_scanout = 1 if (/apply\s*\"grp[0-9]_unload\"/);
        $chain_test=1  if (/CHAIN_TEST/);

if (( /\t*chain\s+\"chain([0-9])\"/) && ($chain_test)){
                $chain_number = $1;
                &cleanup;
                        $chain_input = (split /=/,$_)[1];
                        $chain_input =~ tr/\"//d;
                SWITCH: {
                                $chain_number==1 && do {
                                @chain1 = split (//,$chain_input);
                                $scan_chain_length1 = @chain1; };
                                $chain_number==2 && do {
                                @chain2 = split (//,$chain_input);
                                $scan_chain_length2 = @chain2; };
                                $chain_number==3 && do {
                                @chain3 = split (//,$chain_input);
                                $scan_chain_length3 = @chain3; };
                                $chain_number==4 && do {
                                @chain4 = split (//,$chain_input);
                                $scan_chain_length4 = @chain4; };
                                $chain_number==5 && do {
                                @chain5 = split (//,$chain_input);
                                $scan_chain_length5 = @chain5; };
                                $chain_number==6 && do {
                                @chain6 = split (//,$chain_input);
                                $scan_chain_length6 = @chain6; };
                                $chain_number==7 && do {
                                @chain7 = split (//,$chain_input);
                                $scan_chain_length7 = @chain7;
                                &printout;
                                };
                }#END SWITCH
        }
} #end of While statement
sub printout {
        if ($ct_scanout ){
                        for ($i=0;$i<$scan_chain_length3;$i++){
               $chain1[$i] =~ s/0/L/g; $chain1[$i] =~ s/1/H/g;
               $chain2[$i] =~ s/0/L/g; $chain2[$i] =~ s/1/H/g;
               $chain3[$i] =~ s/0/L/g; $chain3[$i] =~ s/1/H/g;
               $chain4[$i] =~ s/0/L/g; $chain4[$i] =~ s/1/H/g;
               $chain5[$i] =~ s/0/L/g; $chain5[$i] =~ s/1/H/g;
               $chain6[$i] =~ s/0/L/g; $chain6[$i] =~ s/1/H/g;
               $chain7[$i] =~ s/0/L/g; $chain7[$i] =~ s/1/H/g;
                print OUTFILE2 "\n(ct_so
$chain1[$i]$chain2[$i]$chain3[$i]$chain4[$i]$chain5[$i]$chain6[$i]$chain7[$i]
 )";
```

```
                                                    }
                        $ct_scanout=0;
                }
                elsif ($ct_scanout==0){
                                for ($i=0;$i<$scan_chain_length3;$i++){
                        $chain1[$i] =~ s/X/0/g; $chain2[$i] =~ s/X/0/g;
                        $chain3[$i] =~ s/X/0/g; $chain4[$i] =~ s/X/0/g;
                        $chain5[$i] =~ s/X/0/g; $chain6[$i] =~ s/X/0/g;
                        $chain7[$i] =~ s/X/0/g;
                                print OUTFILE2 "\n(ct_si  $ct_si{tdi}
$chain1[$i]$chain2[$i]$chain3[$i]$chain4[$i]$chain5[$i]$chain6[$i]$chain7[$i]
 )";
                                                    }
                }
}
sub cleanup{
                         s/^\s+|\s*\n+$//g;
                         tr/\t  //d;
                         s/\n//g;
                         s/\\//g;
}
sub preprocess{
        while (<INFILE>){
                chomp;
        s/$/;/g if (/apply/);
        print OUTFILE ("$_\n");
        last if ( /^SCAN_CELLS/);
        }
        close OUTFILE;
}
```