What is lut9.txt about?

I have uploaded the contents of a lookup table which may help you with Programs 2 and 3. For each possible (nonzero) LFSR starting state, it will give you the nine possible 10th LFSR states. You can use this information to figure out which LFSR starting state I used in the testbench.

Addressing it:

Contents of address [(P-1)\*9 + T] will give you, for starting pattern P and tap pattern index T, the corresponding 10th LFSR state.

Possible use: implement as a LUT:

data\_out = core[address\_in];

// I have given you the numbers to load into the core on a $readmemb command

To use this lookup table:

1) extract and store the (starting) LFSR state, “P,” that lies in data\_mem[64];

you will have to invert bit[5] by XORing this value with 0x20 (ASCII space – why?)

2) extract and store the 10th LFSR state that lies in data\_mem[73];

you will have to invert bit[5] again

(You could also redo my table to avoid these two flip steps. (How?))

3) Now read values located at [(P-1)\*9] through [(P-1)\*9+8] and compare each in succession to the content of data\_mem[73] (again with bit [5] flipped)

4) As soon as you find a match, the T portion of that address pointer will be the index number of the LFSR tap pattern I used in creating the encrypted message