A DFA for binary representation of integers divisible by 3

Lemma 1: The binary representation of every natural number *n* divisible by 3 has the following property:

$$(N_e - N_u) \bmod 3 = 0$$

where N_e is the number of 1:s in even positions (the rightmost least significant bit is nr 0) and N_u is the number of 1:s in odd positions. The reverse is also true.

Proof:

From its binary representation we can write *n* as

$$n = \sum_{i=u_1,\ldots,u_{N_u}} 2^i + \sum_{i=e_1,\ldots,e_{N_e}} 2^i$$

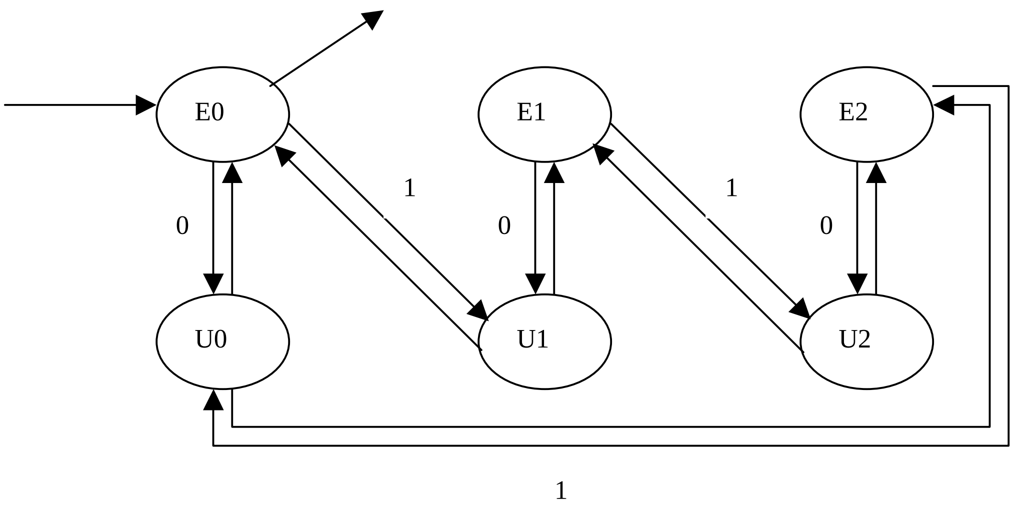
where $u_1,\ldots,u_{N_u}$ are the odd positions with 1:s and $e_1,\ldots,e_{N_e}$ are the even positions with 1:s

Now, since $2^0 = 1$ it follows that all *even* powers of 2 = 1 *mod* 3. Why? Because if you have a number $3m + 1$ and multiply by $2^2 = 4$ you get $12m + 4 = 3(4m + 1) + 1$. By the same reasoning all odd powers of 2 = 2 *mod* 3. Thus the expression for *n* can be written

$$n = \sum_{i=u_1,\ldots,u_{N_u}} (3m_i + 2) + \sum_{i=e_1\ldots e_{N_e}} (3m_i + 1)$$
$$= 3M + 2N_u + N_e$$

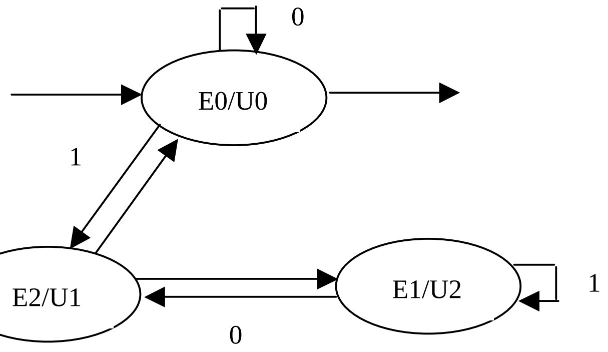
for some integers m_i and M . If *n* is divisible by 3, so is this expression, and thereby also $2N_u + N_e$. Subtract by $3N_u$ and the result follows. It is also clear that if *n* not is divisible by 3, then $(N_u - N_e) \bmod 3 \neq 0$, which proves the reverse.

Now we use this result to construct a DFA which accepts all strings with this property (and no other strings). We index the states with E/U depending on whether the *next* character is to be placed at an even or odd position respectively. They are also indexed by $(N_e - N_u)$ mod 3. i.e. the surplus of even numbered 1's *mod* 3.



Note: this DFA builds strings with the least significant bit first.

A minimization gives the following equivalent DFA:



The language of the DFA can distinguish the strings *0, 1* and *10.*

0 and *1*	by	*0*
0 and *10*	by	*0*
1 and *10*	by	*1*

So the language of this DFA distinguishes three strings and is therefore minimal.

The languge the DFA defines is described by the regular expression

$$L = (0 \cup (01^{*}0)^{*})^{*}$$

Question: Can we find a DFA which builds strings with this property but with the most significant bit first?

Answer: Indeed. Every string which has the property from Lemma 1 also has it if you exchange most/least significant order. If the number of bits are odd, N_e and N_u are unchanged. If the number if bits are even N_e and N_u exchange values, but the crucial property still holds true. Thus the DFA above can be used this way too.