Re: Boolean algebra "2 nots" problem

Source: http://sci.tech−archive.net/Archive/sci.math/2004−06/5730.html

---

From: Keith A. Lewis (*lewis_at_PROBE.mitre.org*)
Date: 06/25/04

Date: Fri, 25 Jun 2004 01:59:17 +0000 (UTC)

Don Reble <djr@nk.ca> writes in article <40DB690D.E5370346@nk.ca> dated Thu, 24 Jun 2004 17:51:41 −0600:
>> Given three inputs A, B, C, generate three outputs A', B', C' using
>> any number of AND and OR operations, but a maximum of 2 NOTs.
>
>Ok.
>
>There are eight possible input states:
> ABC, ABC', AB'C, AB'C', A'BC, A'BC', A'B'C, A'B'C'.
>There are 256 possible functions; a function maps each of the eight
>possible input states to either of two output states. I name each
>function with an eight bit string; the function produces the n'th bit of
>the string, in response to the n'th input state.
>
>The given functions are A=11110000, B=11001100, and C=10101010.
>Do these combinations:

I'll see if I can translate into boolean algebra.

X = NOT((A|(B&C))&(B|C))
Y = NOT((A|(B&X)|(C&X))&((A&B&C)|X))

A' = ((B&C)|((Y|(B&X)|(C&X))&X))&(Y|(B&X)|(C&X))
B' = ((A&C)|((Y|((A|(C&X))&X))&X))&(Y|((A|(C&X))&X))
C' = ((A&B)|((Y|((A|(B&X))&X))&X))&(Y|((A|(B&X))&X))

>As you see, there are two NOT's, and the last group has A', B', and C'.

I hope I did that right.

−−Keith Lewis klewis {at} mitre.org
The above may not (yet) represent the opinions of my employer.