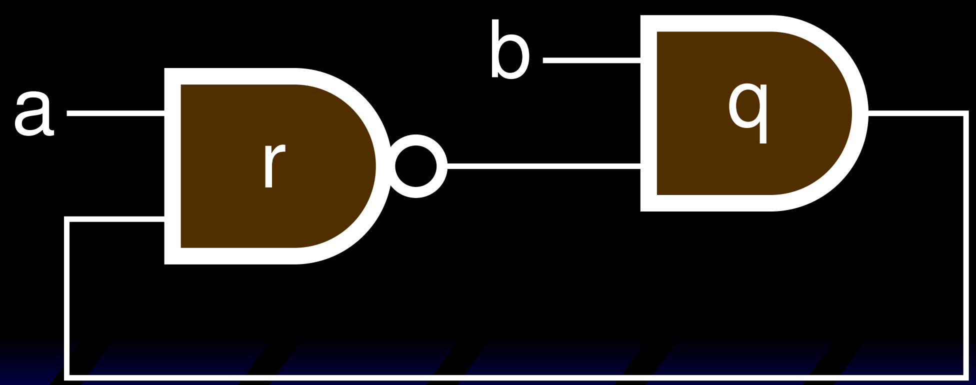
An Efficient Algorithm for the Analysis of Cyclic Circuits

Osama Neiroukh	Stephen A. Edwards	Xiaoyu Song
Intel	Columbia University	Portland State University

What is a Combinational Circuit?

Malik [ICCAD 93]: “A circuit is combinational *for an input pattern* if three-valued simulation starting from Xs converges to 0s and 1s.”

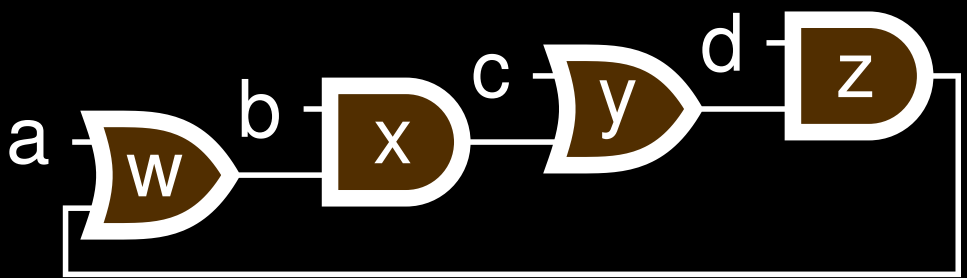
Shiple [96]: “Equivalent to stability in Brzozowski and Seger’s [95] model.”

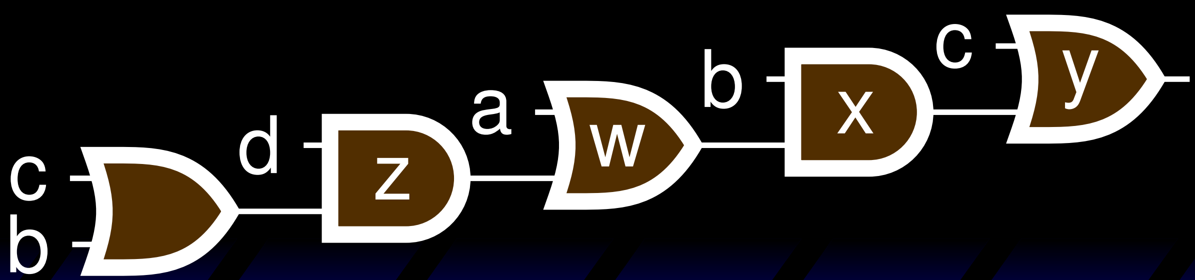


a	b	q	r
0	0	0	1
0	1	1	1
1	0	0	0
1	1	X	X

Goal

Given a cyclic circuit that is combinational for some inputs, create an acyclic circuit that computes the same combinational function.





this input not combinational

abcd	wxyz
0000	0000
0001	0000
0010	0010
0011	1011
0100	0000
0101	
0110	0010
0111	1111
1000	1000
1001	1000
1010	1010
1011	1011
1100	1110
1101	1111
1110	1110
1111	1111

Applications

Fixing cyclic circuits from high-level synthesis

Stok [ICCAD 92]: cycles from resource sharing

Berry [92]: cycles from Esterel programs

Acyclic circuits easier to simulate

Related Work

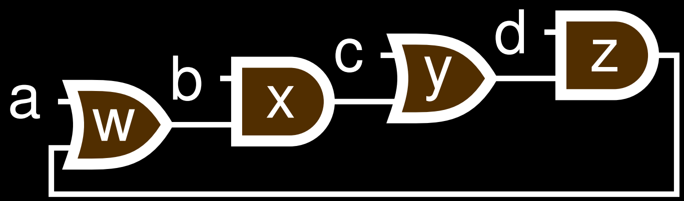
Malik [ICCAD 93]: basic definitions, unrolling

Edwards [DAC 03]: basis of our work

Gupta and Selvidge [ICCAD 05]: fix single loops

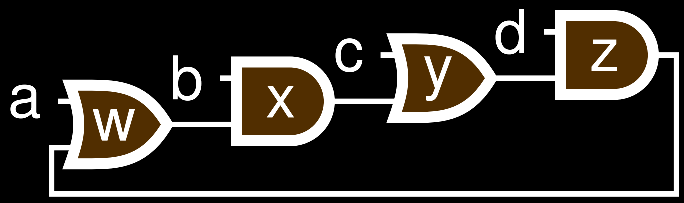
Riedel [DAC 03]: a technique for creating them

Approach [Edwards, DAC 03]

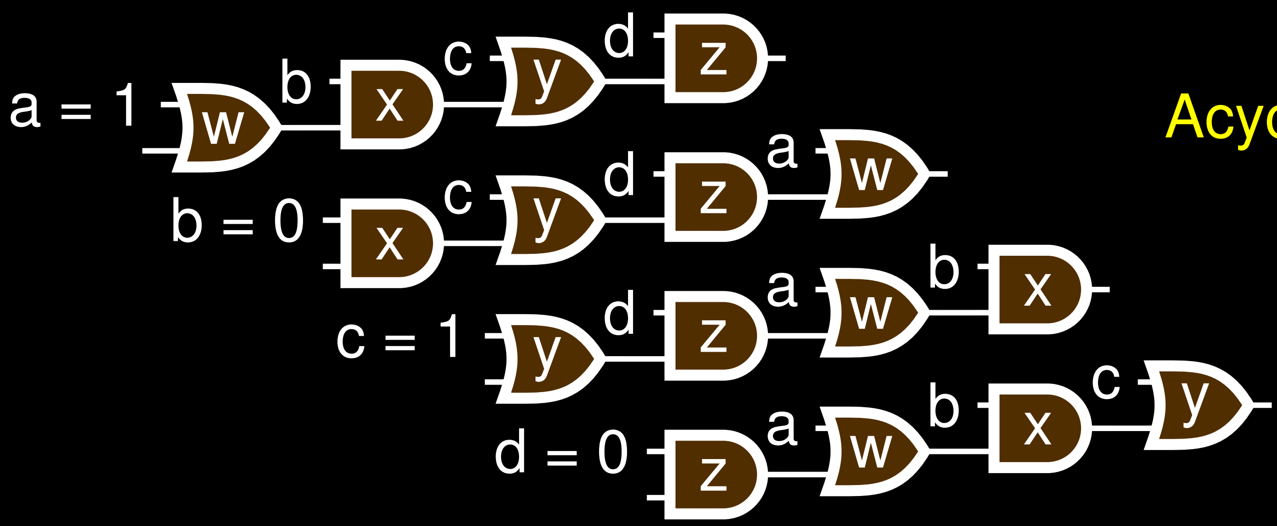


Original Circuit

Approach [Edwards, DAC 03]

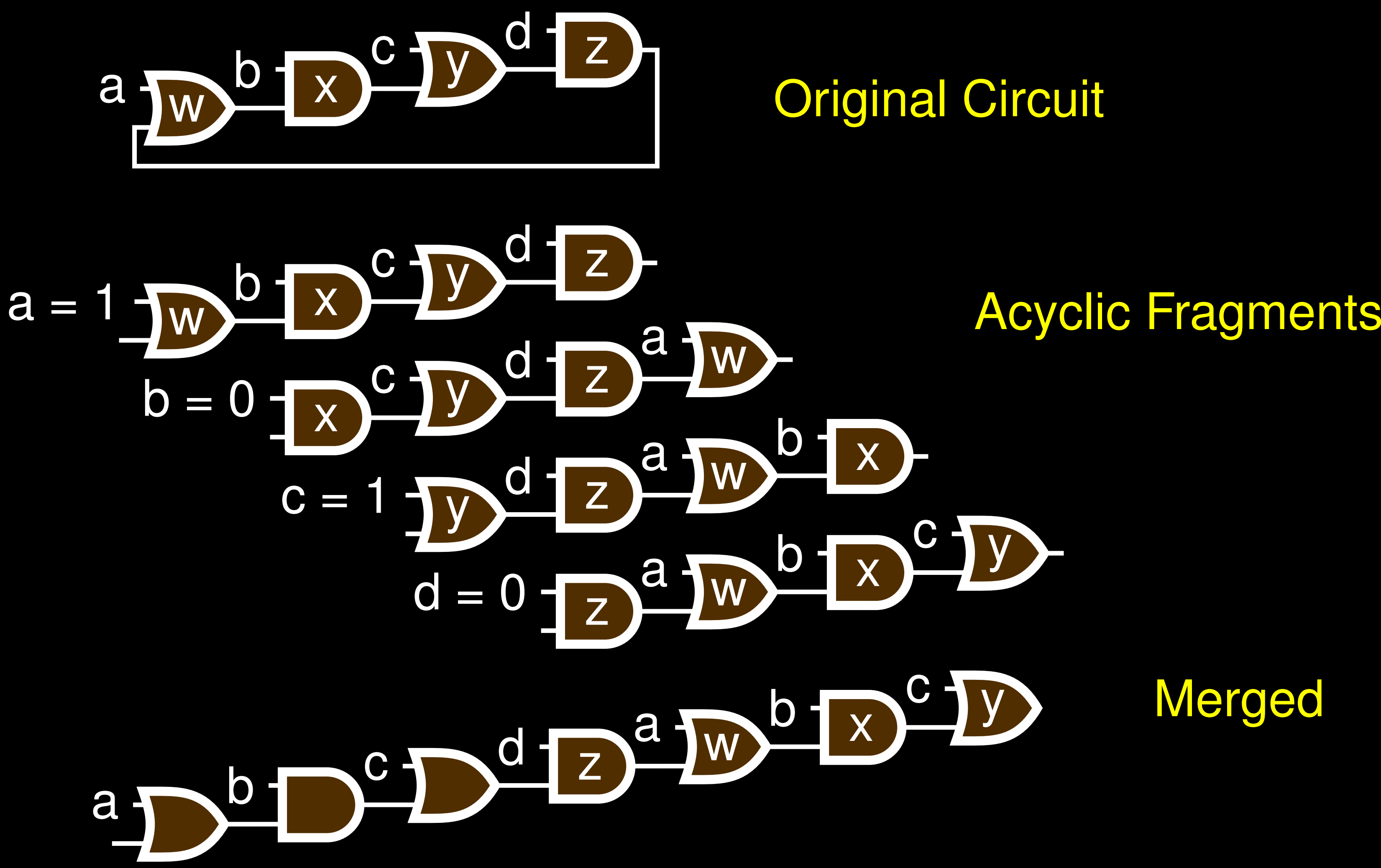


Original Circuit

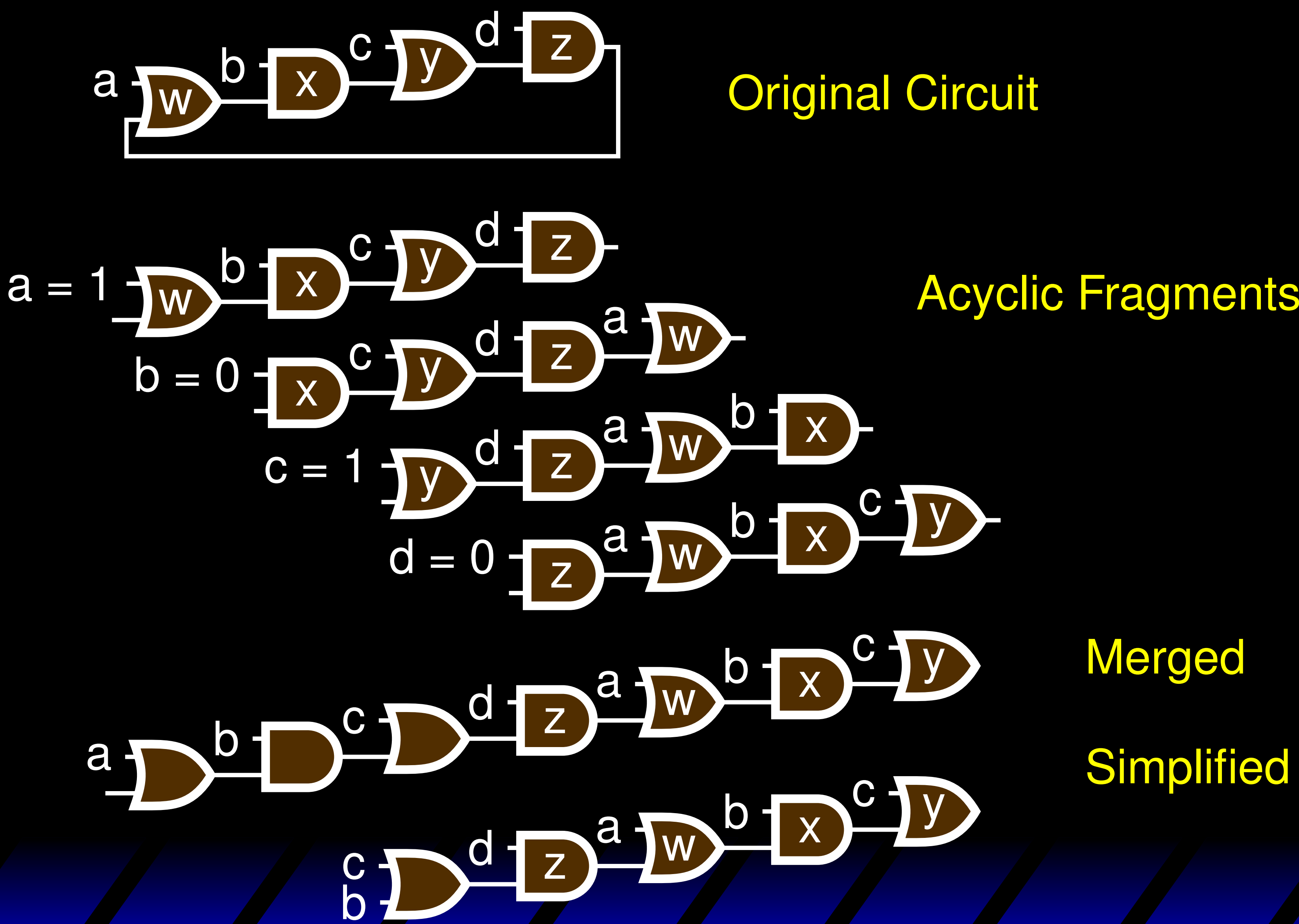


Acyclic Fragments

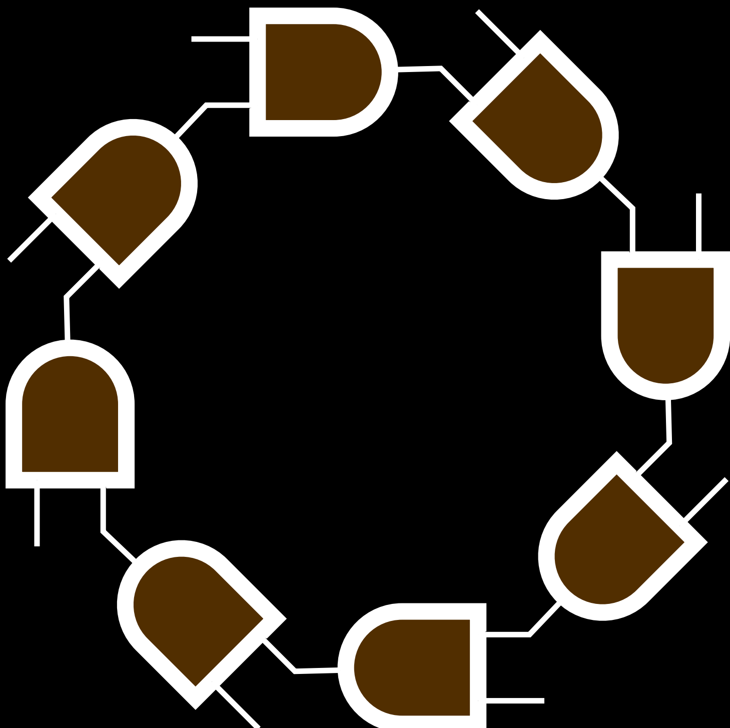
Approach [Edwards, DAC 03]



Approach [Edwards, DAC 03]



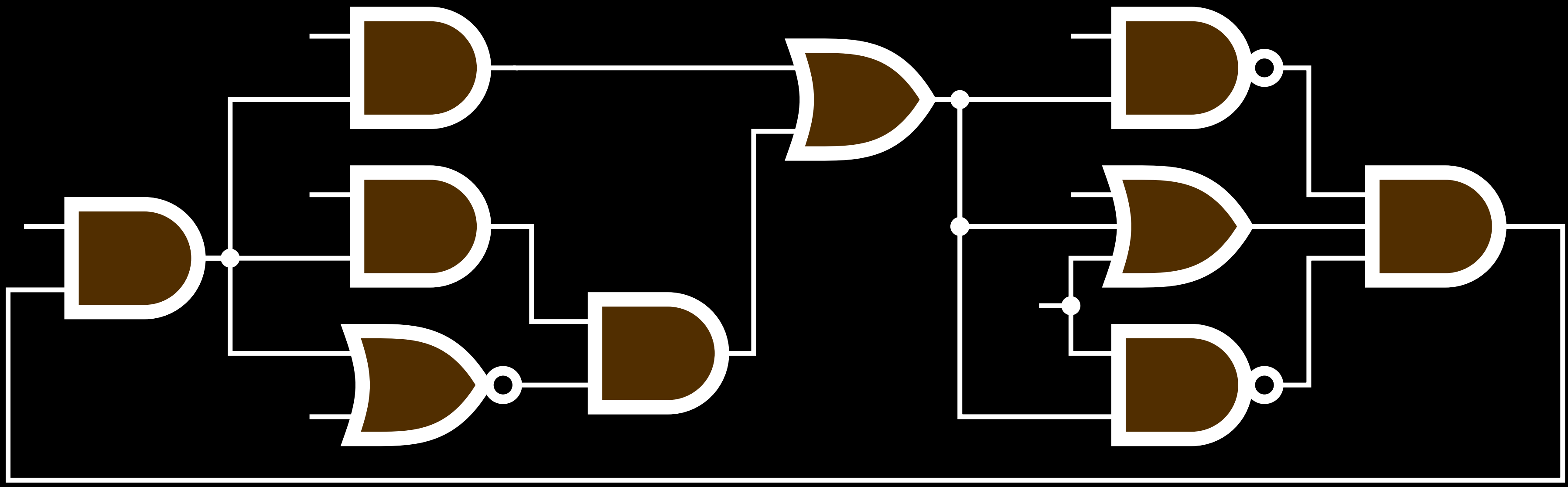
First Observation [Edwards 2003]



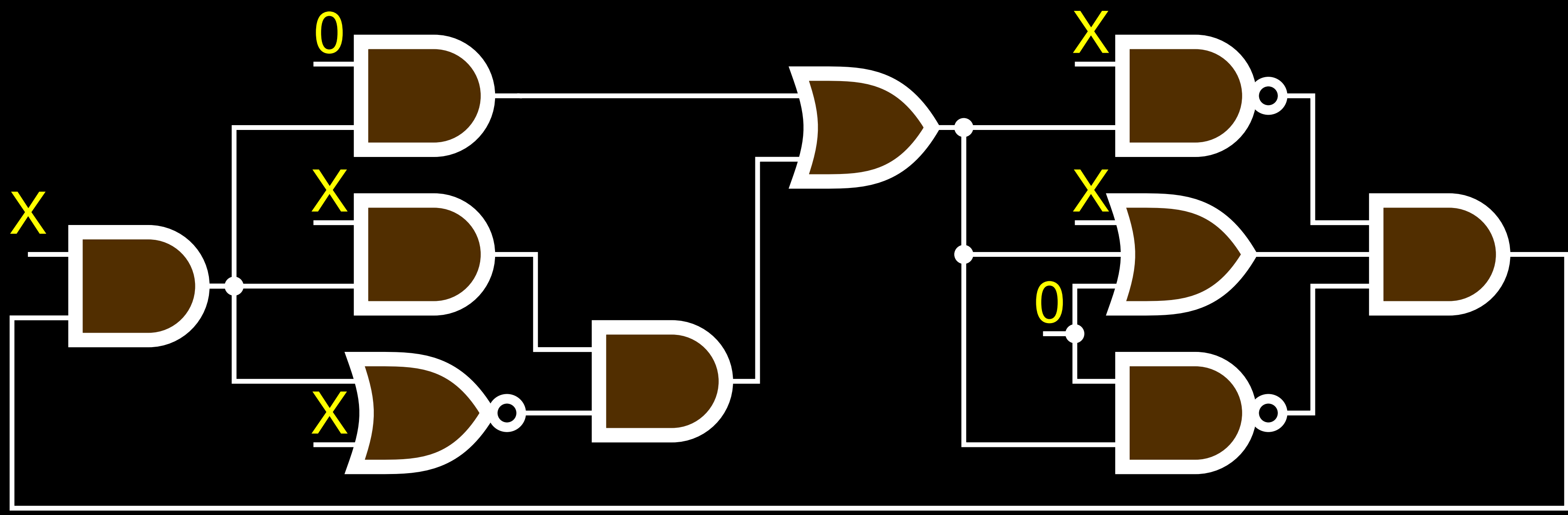
For an input pattern to be combinational, at least one input coming from outside each strongly-connected component must have a controlling value.

If all external inputs were non-controlling, the gates in the SCC would stay at X.

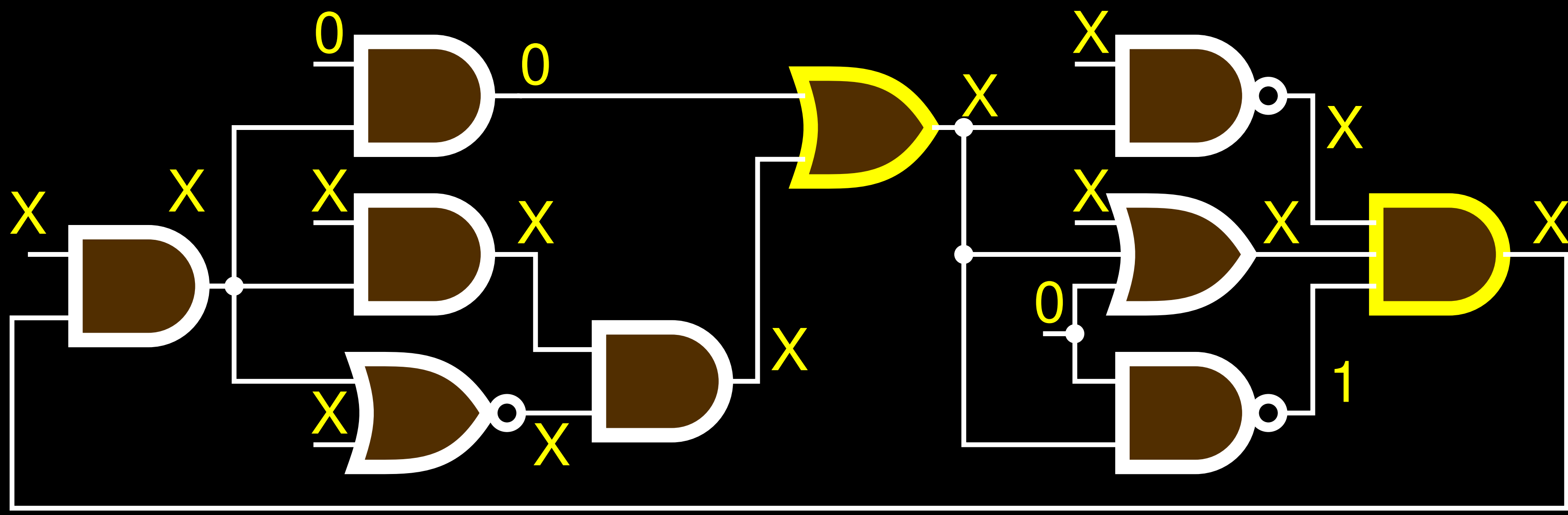
Second Observation



Second Observation

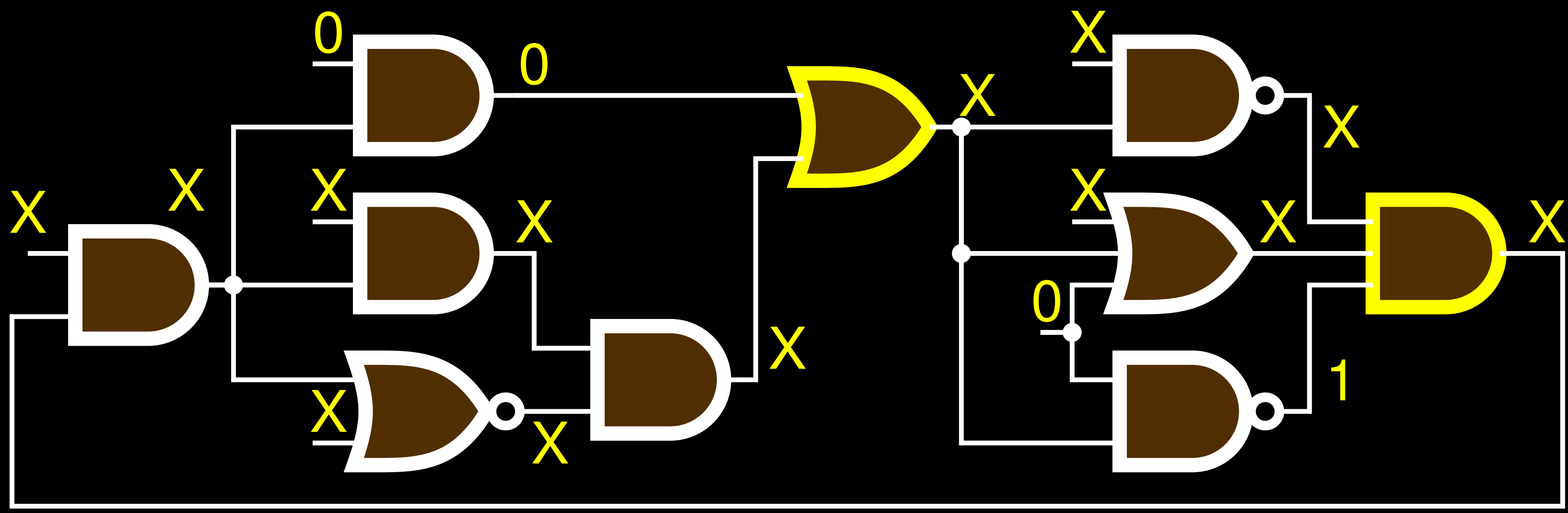


Second Observation



Frontier gate: some inputs defined, output remains X

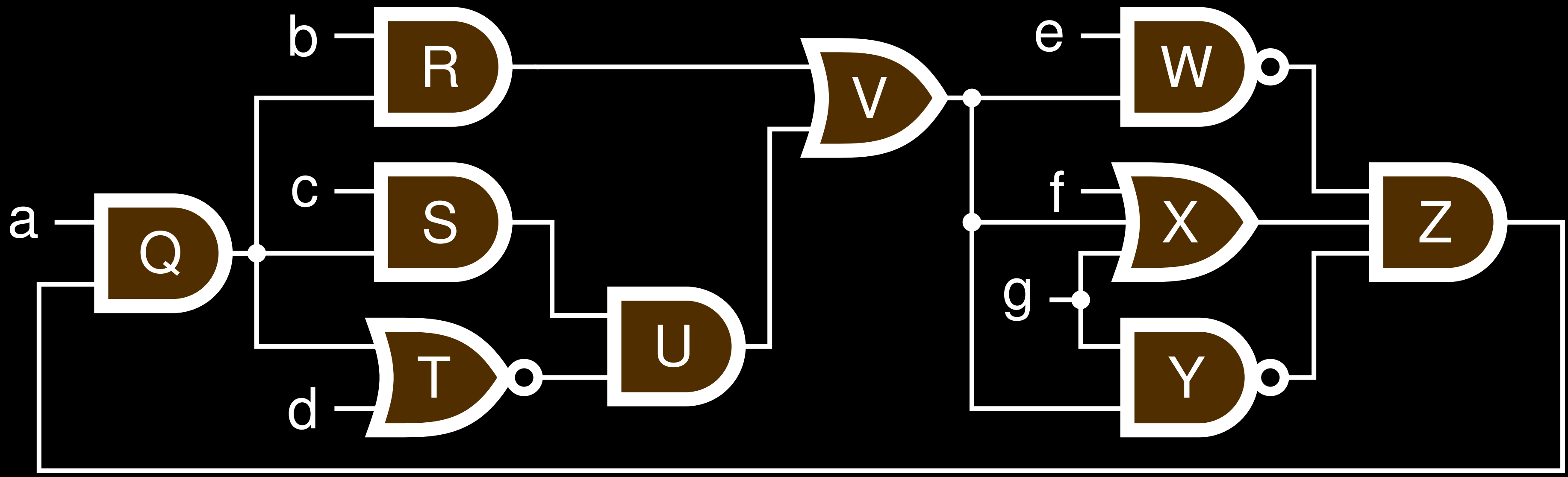
Second Observation



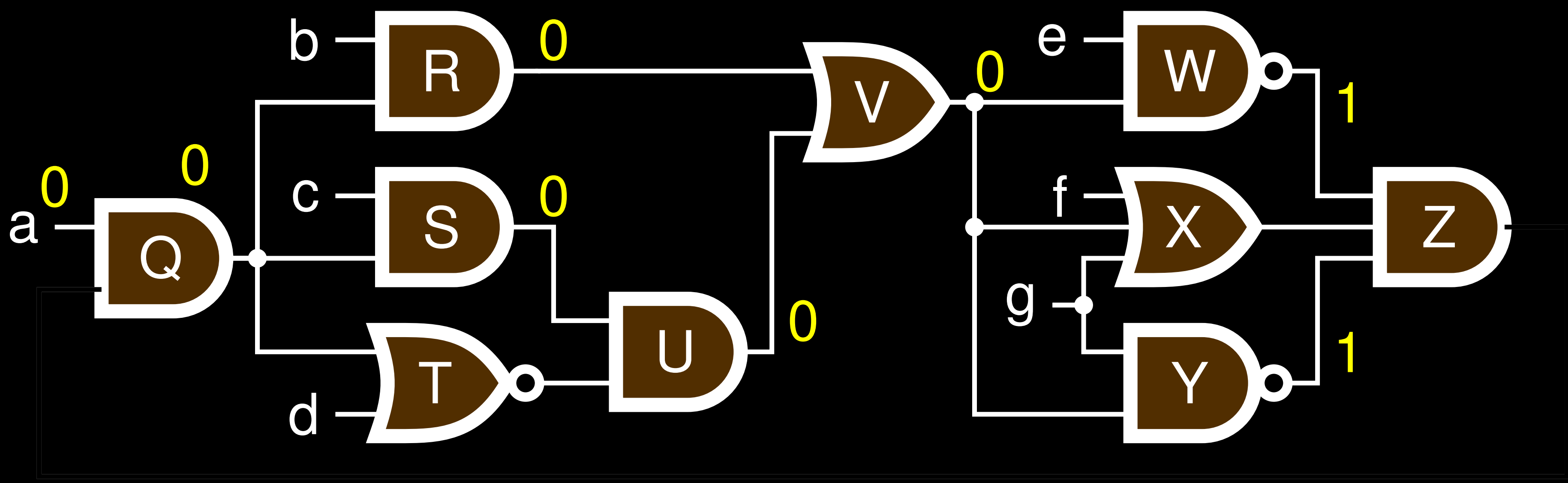
Frontier gate: some inputs defined, output remains X

Input is combinational $\Leftrightarrow$ frontier is empty

Our Algorithm Step 1: Apply a Controlling Value to Each Input

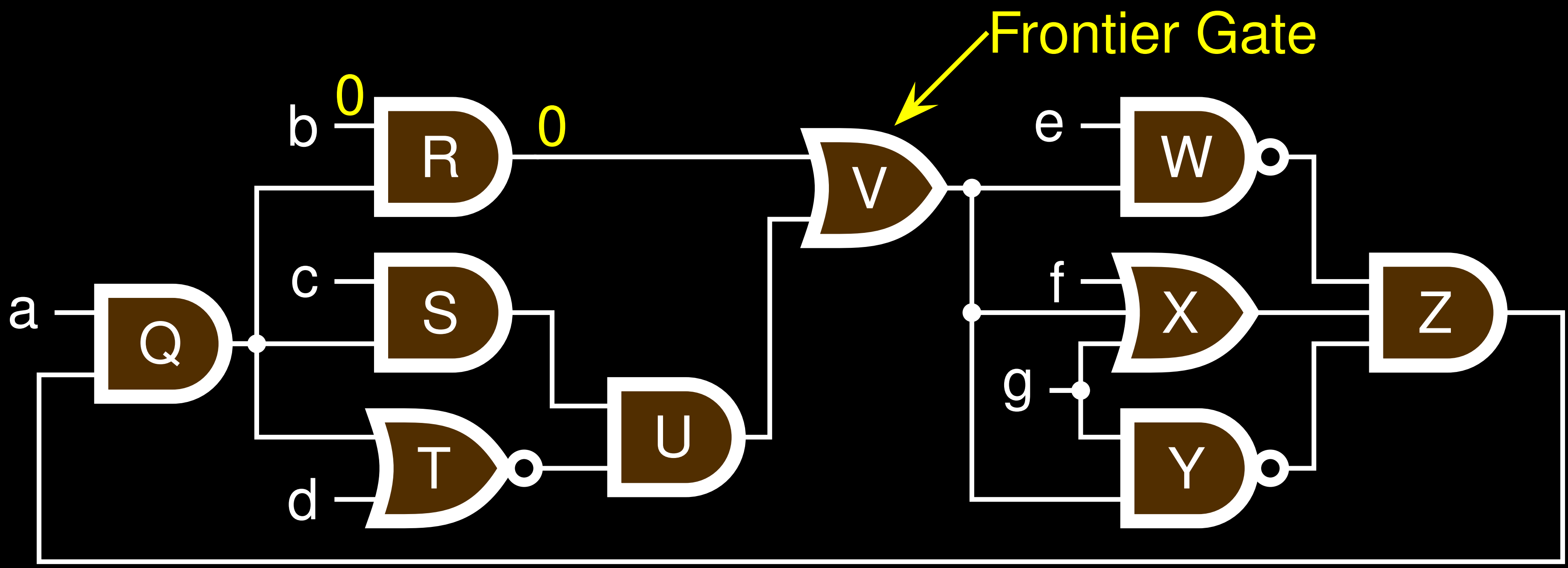


Our Algorithm Step 1: Apply a Controlling Value to Each Input



a = 0 acyclic

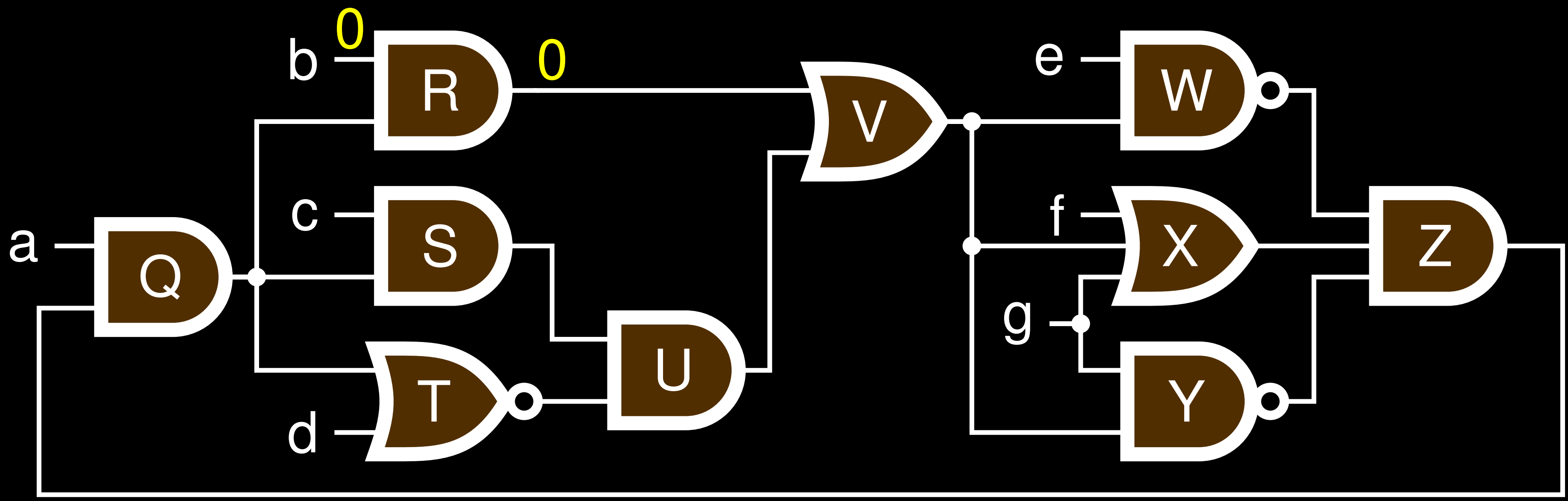
Our Algorithm Step 1: Apply a Controlling Value to Each Input



a = 0 acyclic

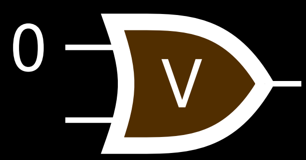
b = 0

Our Algorithm Step 1: Apply a Controlling Value to Each Input

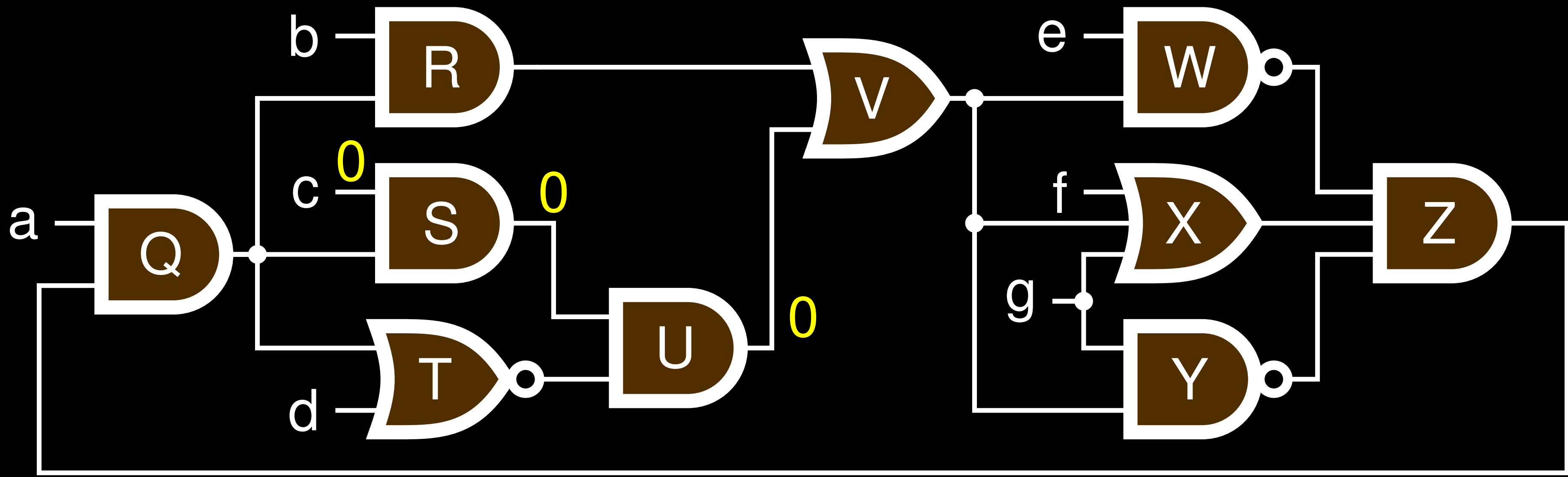


a = 0 acyclic

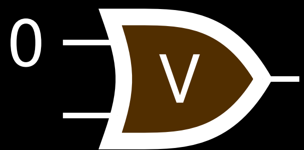
b = 0



Our Algorithm Step 1: Apply a Controlling Value to Each Input

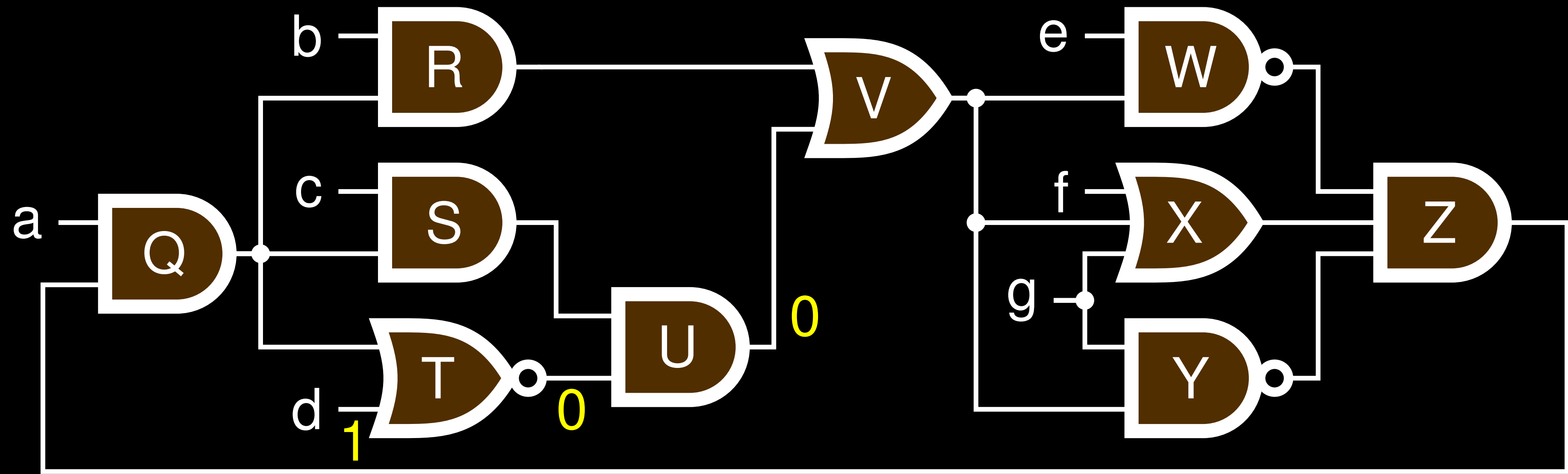


a = 0 acyclic

b = 0 

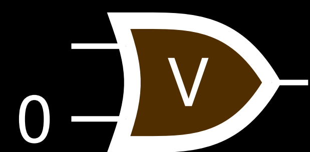
c = 0 

Our Algorithm Step 1: Apply a Controlling Value to Each Input

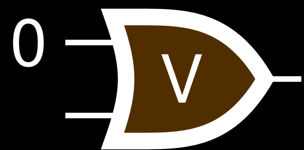


a = 0 acyclic

d = 1

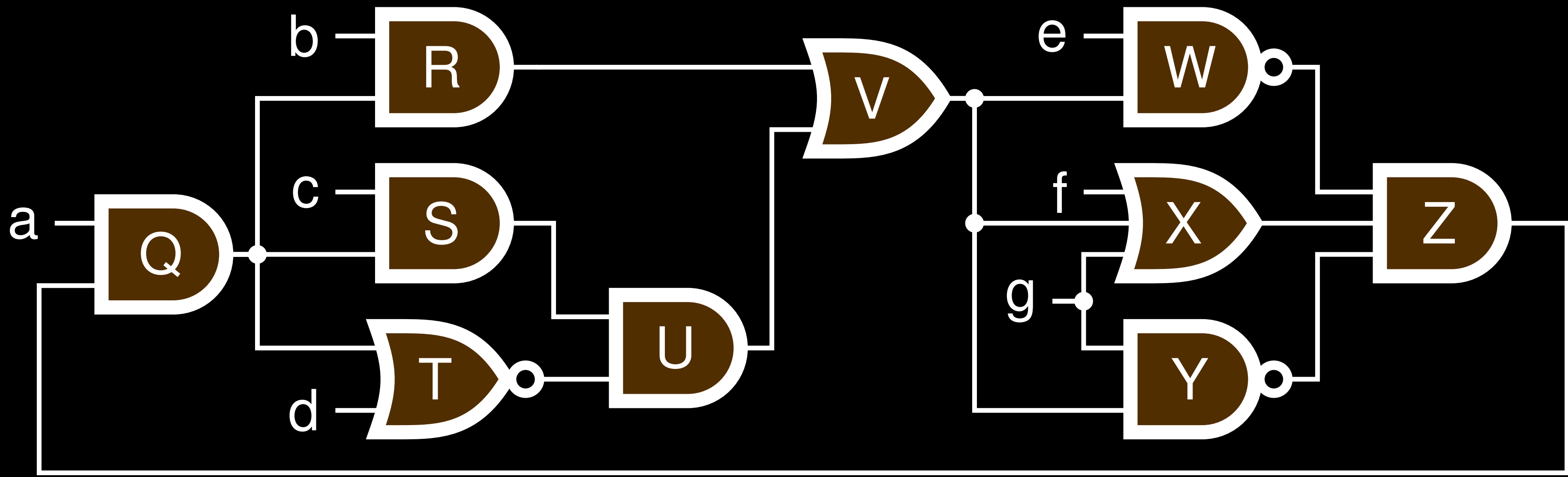


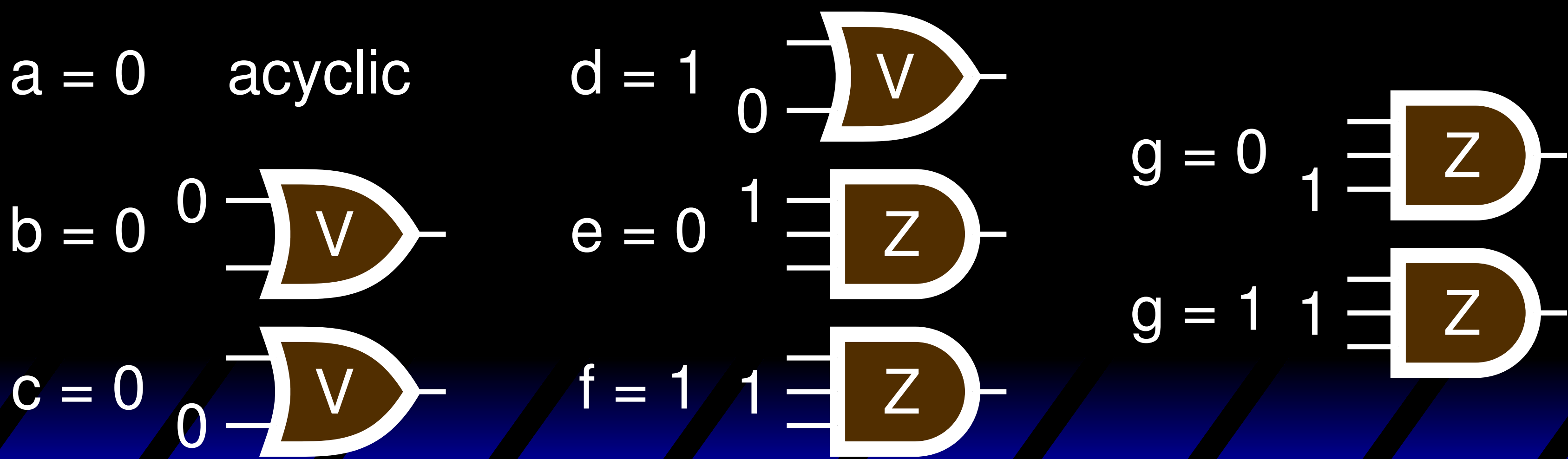
b = 0



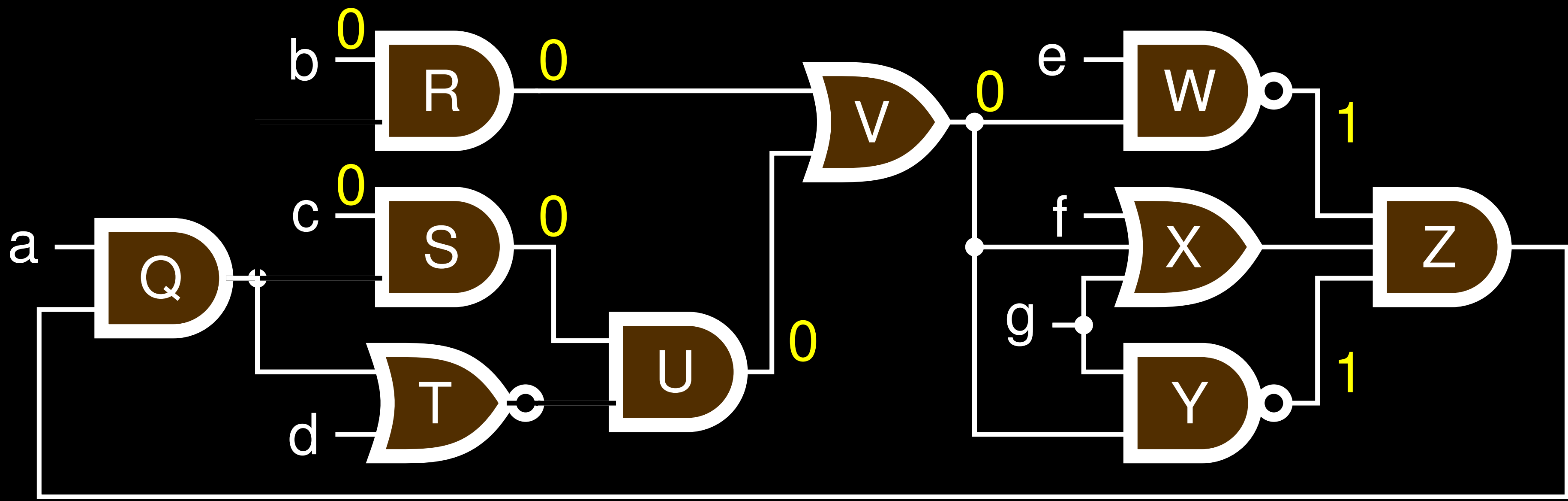
c = 0

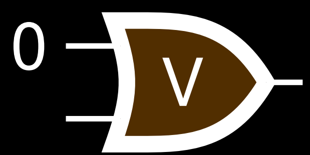
Our Algorithm Step 1: Apply a Controlling Value to Each Input

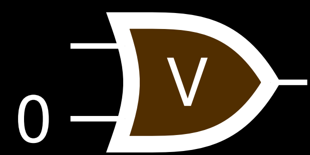




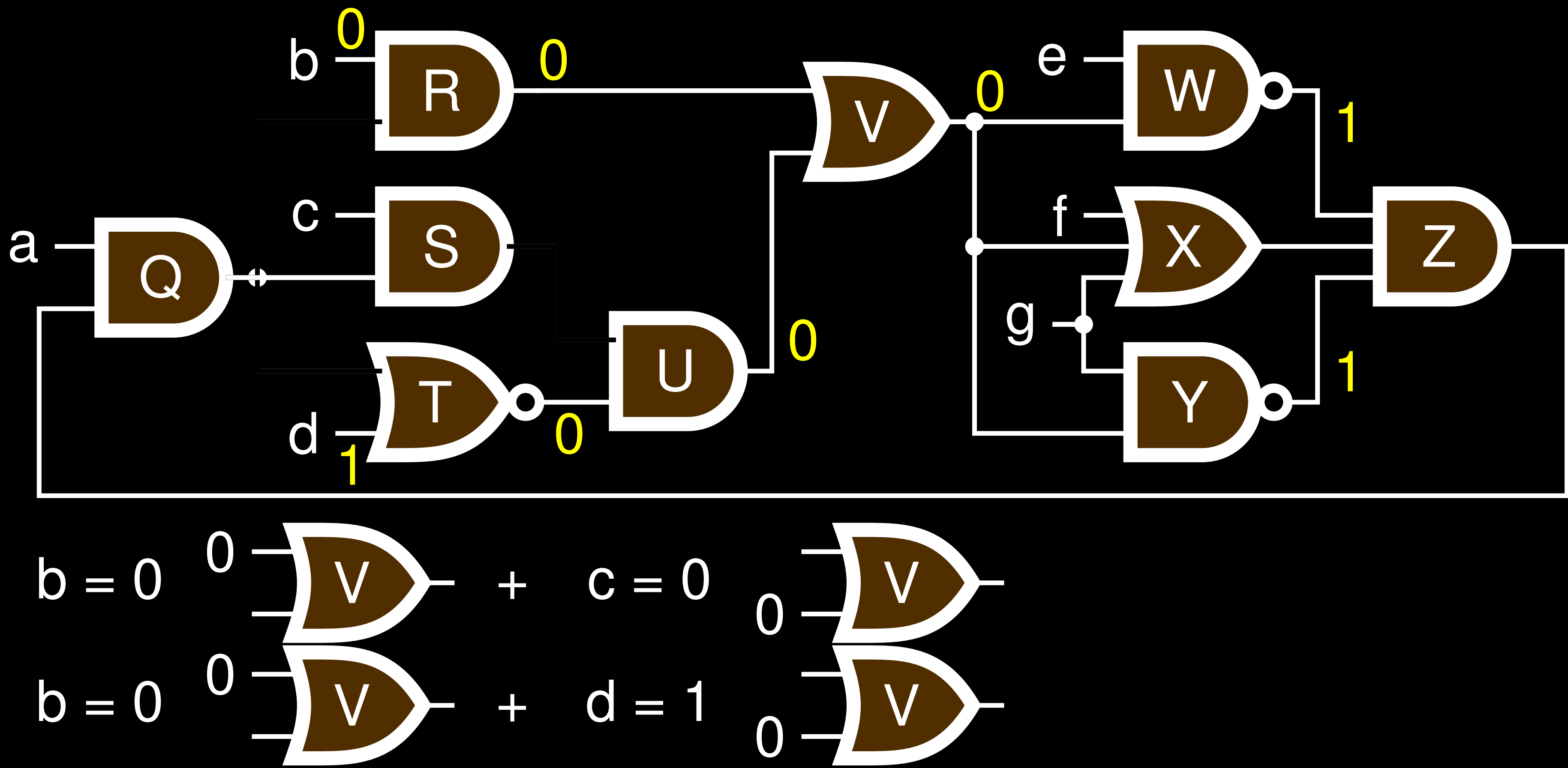
Our Algorithm Step 2: Attack Frontier Gates with Combinations



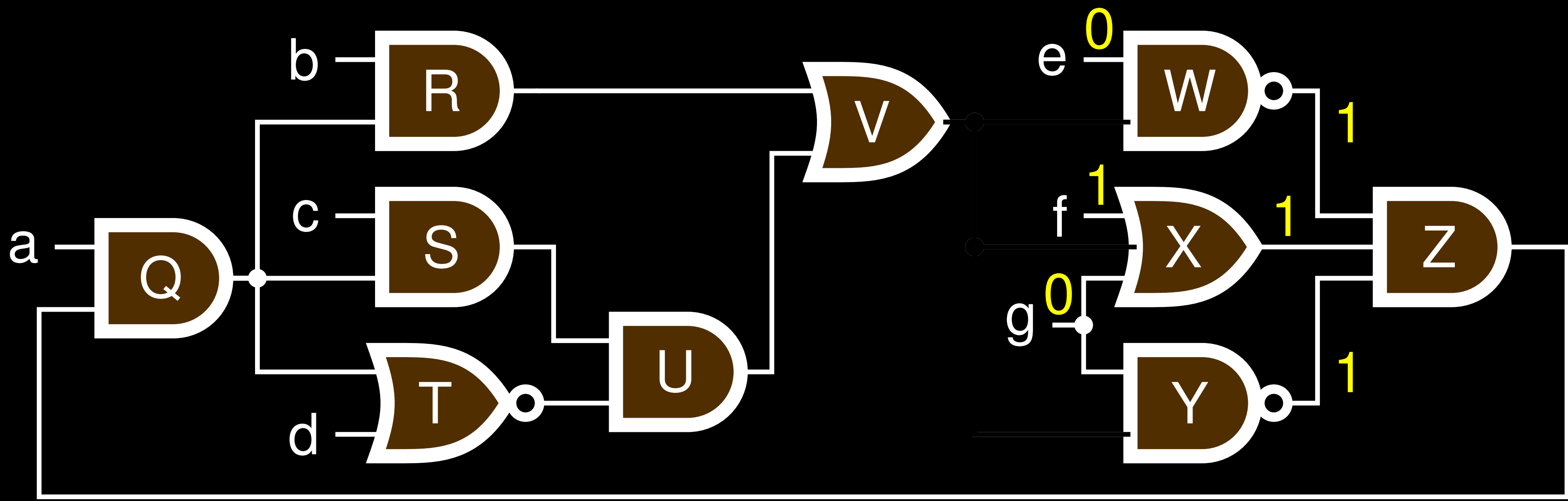
b = 0 

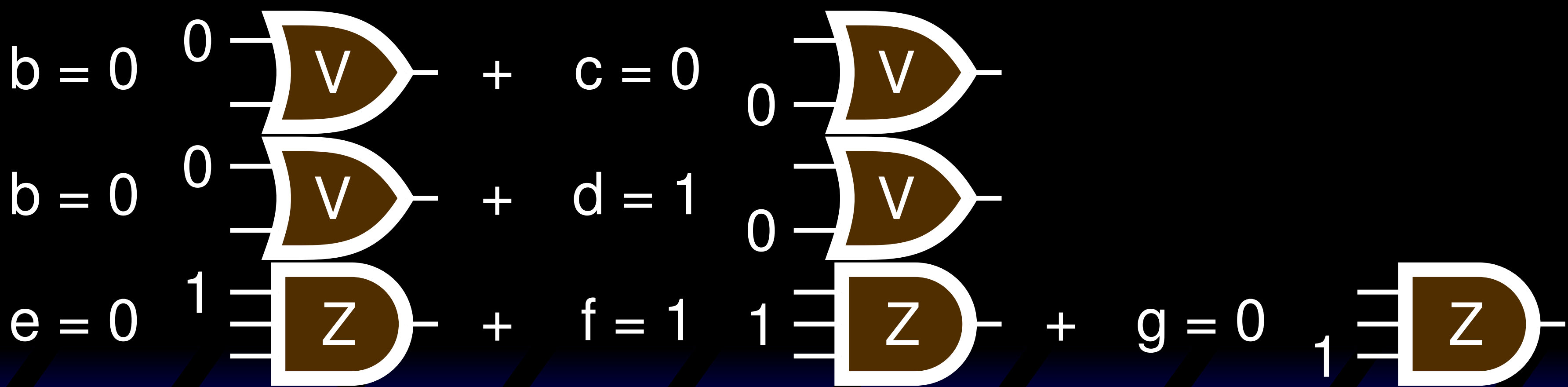
+ c = 0 

Our Algorithm Step 2: Attack Frontier Gates with Combinations



Our Algorithm Step 2: Attack Frontier Gates with Combinations





Experimental Results

Circuit	Netlist Gates	SCC Gates	DAC 03 PAs	DAC 03 time	Ours PAs	Ours time	Acyclic PAs
arbiter5	213	25	257	1.3	25	0.1	14
arbiter6	248	30	745	8	29	0.1	16
arbiter7	283	35	2205	69	33	0.2	18
arbiter8	318	40	6581	656	37	0.3	20
exp	124	69	54517	2868	23260	2.0	338
ex1	150	47	43777	2341	232	1.0	10
gary	177	32		∞	290	0.6	11
planet	253	51		∞	1489	0.3	22
s1488	272	61		∞	588	0.2	89
table3	311	49		∞	3604	1.0	38

Conclusions

- More focused exploration of search space
- Idea: combine partial assignments to attack frontier gates
- Exponential improvement compared to Edwards [DAC 03]
- Future work
  - Even better pruning
  - Symbolic approach?