TRUGEN – Truth Table Generator for 3 or 4 variables

Abstract

TRUGEN is a program written in 1980 for the SHARP PC-1210 / PC-1211 / PC-1212 pocket computers to help generate truth tables for logical expressions having 3 or 4 logical variables, recognizing tautologies and contradictions. Three worked examples included.

Keywords: *truth table, generator, logical expression, logical variable, SHARP pocket computers, PC-1210, PC-1211, PC-1212*

1. Introduction

TRUGEN is a very simple 8-line practice program I wrote in 1980 for the *SHARP PC-1210 / PC-1211 / PC-1212* pocket computers to help generate truth tables for logical expressions having 3 or 4 logical variables, recognizing tautologies and contradictions.

The logical expression can include all the usual logical operators such as *AND*, *OR*, *NOT*, *XOR*, *XNOR*, *NOR*, *NAND* and *Logical implication / Material conditional*, which the user must convert to a *BASIC*-language expression by simply replacing each *unary* and *binary* operator by the equivalent *BASIC* constructions featured in *Table 1: Operator Equivalences* just below, using parentheses if necessary to preserve precedence rules.

Table 1: Operator Equivalences

Logical operator	***PC-1211 BASIC equivalent***	
Logical conjunction (AND)	$p \wedge q$	*p * q*
	p & q	
	$p \cdot q$	
Logical disjunction (OR)	$p \vee q$	*p + q*
	$p \parallel q$	
	p + q	
Exclusive disjunction (XOR)	$p \oplus q$	*(SGN p <> SGN q)*
Logical negation (NOT)	$\neg p$	*(p = 0)*
	$\sim p$	
Logical implication (p implies q)	$p \Rightarrow q$	*(p = 0) + q*
Material conditional (if p then q)	$p \rightarrow q$	
Logical equality (XNOR)	$p \leftrightarrow q$	*(SGN p = SGN q)*
	p = q	
	$p \equiv q$	
Logical NAND	$p \uparrow q$	*(p = 0) + (q = 0)*
	p | q	
Logical NOR	$p \downarrow q$	*(p = 0) * (q = 0)*

Besides producing the truth table by rows, the program will also tally the number of *True* and *False* values, and additionally will declare it a *Tautology* (all values *True*) or a *Contradiction* (all *False*) when appropriate.

2. Program Listing

```
10: "A" A$(1)="F ",A$(2)="T ",N=0,K=8,K=16
20: FOR P=0 TO 1: FOR Q=0 TO 1: FOR R=0 TO 1: FOR S=0 TO 1
30: REM Z=F(P,Q,R,S)
40: Z=SGN Z,N=N+(Z<>0): PRINT A$(P+1);A$(Q+1);A$(R+1);A$(S+1);"  ";A$(Z+1)
50: NEXT S: NEXT R: NEXT Q: NEXT P
60: IF N=K THEN PRINT "ALL TRUE, TAUTOLOGY": END
70: IF N=0 THEN PRINT "ALL FALSE, CONTRADICTION": END
80: PRINT USING "##";N;" TRUE,";K-N;" FALSE"
```

Important note:

*- if the logical expression has **4** variables, (p, q, r, s) then **keep** the highlighted boxed statements in the listing.*

*- if the logical expression has **3** variables, (p, q, r) then **omit** the highlighted boxed statements from the listing.*

3. Usage Instructions

See the worked examples to understand how to use the program.

4. Examples

The following examples can be useful to check that the program is correctly entered and to understand its usage.

4.1 Example 1

Produce the truth table for the following logical expression: $[(p \wedge q) \Rightarrow r] \Rightarrow (p \vee r)$

*(this logical expression has **3** variables (p, q, r) so make sure to **omit** or delete the highlighted boxed statements from the listing)*

In **PRO** Mode, enter the following program line to define the logical expression above, which after substituting the corresponding equivalences from *Table 1* becomes:

```
30: Z=(((P*Q=0)+R)=0)+P+R
```

In **DEF** Mode, proceed as follows to produce the truth table (press **ENTER** after each row to continue):

SHFT **A**

p	*q*	*r*	***z***
F	F	F	**F**
F	F	T	**T**
F	T	F	**F**
F	T	T	**T**
T	F	F	**T**
T	F	T	**T**
T	T	F	**T**
T	T	T	**T**

6 TRUE, 2 FALSE

4.2 Example 2

Produce the truth table for the following logical expression: $[(p \wedge q) \Rightarrow p] \Rightarrow [(q \vee r) \wedge (\neg q \wedge \neg r)]$

*(this logical expression has **3** variables (p, q, r) so make sure to **omit** or delete the highlighted boxed statements from the listing)*

In **PRO** Mode, enter the following program line to define the logical expression above which, after substituting the corresponding equivalences from *Table 1* and removing unneeded parentheses, becomes:

```
30: Z=(((P*Q=0)+P)=0)+(Q+R)*(Q=0)*(R=0)
```

In **DEF** Mode, proceed as follows to produce the truth table (press **ENTER** after each row to continue):

SHFT **A**

p	*q*	*r*	***z***
F	F	F	**F**
F	F	T	**F**
F	T	F	**F**
F	T	T	**F**
T	F	F	**F**
T	F	T	**F**
T	T	F	**F**
T	T	T	**F**

ALL FALSE, CONTRADICTION

4.3 Example 3

Produce the truth table for the following logical expression: $(\sim p \vee q) \Rightarrow (\sim r \wedge s)$

*(this logical expression has **4** variables, (p, q, r, s) so make sure to **keep** or insert the highlighted boxed statements in the listing)*

In **PRO** Mode, enter the following program line to define the logical expression above, which after substituting the corresponding equivalences from *Table 1* becomes:

```
30: Z=((P=0)+Q=0)+(R=0)*S
```

In **DEF** Mode, proceed as follows to produce the truth table (press **ENTER** after each row to continue):

SHFT **A**

p	*q*	*r*	*s*	*z*	*p*	*q*	*r*	*s*	*z*
F	F	F	F	**F**	T	F	F	F	**T**
F	F	F	T	**T**	T	F	F	T	**T**
F	F	T	F	**F**	T	F	T	F	**T**
F	F	T	T	**F**	T	F	T	T	**T**
F	T	F	F	**F**	T	T	F	F	**F**
F	T	F	T	**T**	T	T	F	T	**T**
F	T	T	F	**F**	T	T	T	F	**F**
F	T	T	T	**F**	T	T	T	T	**F**

7 TRUE, 9 FALSE

Notes

1. The truth table has **8** (=2^3) rows for logical expressions having **3** variables and **16** (=2^4) rows for those having **4** variables.

2. The following rarely used logical operators aren't included in *Table 1*, namely: *Contradiction*, *Converse implication*, *Converse nonimplication*, *Material nonimplication*, *Projection functions (p, q)* and *Tautology.*

3. Logical implication (**p implies q**) is symbolized as **p ⇒ q** while the *material conditional* (**if p then q**) is symbolized as **p → q** , but actually they have the *same* truth table and thus the same equivalence in *Table 1*, namely: ***(p = 0) + q*** .

Copyrights