Logic Design using EXORs

Gerhard Dueck

April 1, 2009

Outline

Expansion Theorem

Theorem

An arbitrary logic function $f(x_1, x_2, \ldots, x_n)$ can be expanded as

$$f(x_1, x_2, \ldots, x_n) = f_0 \oplus x_1 f_2 \quad (1)$$

$$f(x_1, x_2, \ldots, x_n) = \overline{x}_1 f_0 \oplus f_2 \quad (2)$$

$$f(x_1, x_2, \ldots, x_n) = \overline{x}_1 f_0 \oplus x_1 f_2 \quad (3)$$

where $f_0 = f(0, x_2, \ldots, x_n)$, $f_1 = f(1, x_2, \ldots, x_n)$, and $f_2 = f_0 \oplus f_1$.

These are **positive Davio**, **negative Davio** and **Shannon expansion.**

PPRM

Definition

By expanding f using 1 recursively, we have a logical expression with only un-complemented literals:

$$a_0 \oplus a_1 x_1 \oplus \cdots \oplus a_n x_n \oplus a_{12} x_1 x_2 \oplus a_{13} x_1 x_3 \oplus \cdots$$
$$\oplus a_{n-1,n} x_{n-1} x_n \oplus \cdots \oplus a_{12\cdots n} x_1 x_2 \cdots x_n$$

This is a **positive polarity Reed-Muller expression (PPRM).**

PPRM is a canonical expression, so no minimization problem exists.

Example

Represent $f = \overline{x}_1 \overline{x}_2 \overline{x}_3$ by a PPRM.

FPRM

Definition

By applying the positive Davio expansion or the negative Davio expansion to a given function *f*, we have a logical expression which has similar form as a PPRM. In this case, assume that we can use either un-complemented or complemented literals but not both for each variable. This is a **fixed polarity Reed-Muller expression (PPRM)**.

There are up to ?? different FPRMs for a given function.

Example

Represent $f = \overline{x}_1\overline{x}_2\overline{x}_3\overline{x}_3 \vee x_1x_2x_3x_4$ by a FPRM.

FPRM

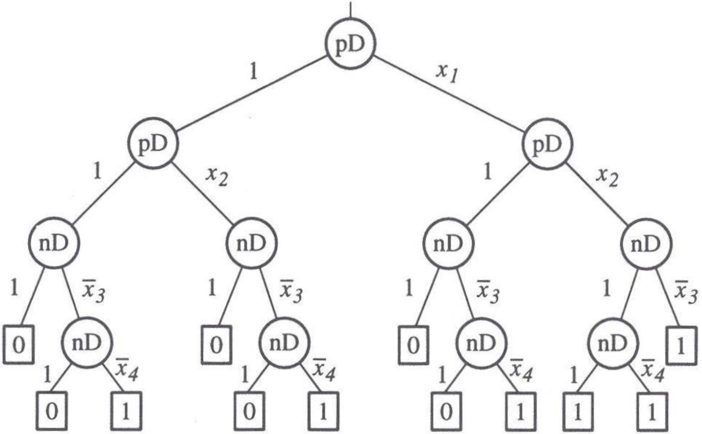


Figure 13.1 Expansion tree for FPRM of $x_1x_2x_3x_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4$.

Kronecker Expression (KRO)

Definition

When the given function f is expanded by either the positive Davio expansion, the negative Davio expansion, or the Shannon expansion, we have a logical expression that is a generalization of FPRMs. This is a **Kronecker Expression (KRO)**.

There are up to ?? different KROs for a given function.

Example

Represent $f = \overline{x}_1\overline{x}_2\overline{x}_3\overline{x}_3 \vee x_1x_2x_3x_4$ by a KRO.

KRO

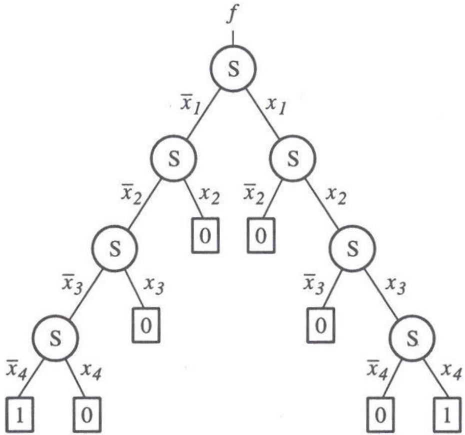


Figure 13.2 Expansion tree of KRO for $x_1x_2x_3x_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4$.

Pseudo Reed Muller Expression (PSDRM)

Definition

For a given function f, if f is expanded by the positive or negative Davio expansion, then we have two sub-functions. Each of these sub-functions, may be expanded by positive or negative Davio (not necessarily the same.) We have a logical expression that is a generalization of FPRMs. This is a **Pseudo Reed Muller Expression (PSDRM)**.

There are up to ?? different PSDRMs for a given function. (Assuming the order of the variable expansion is fixed.)

Example

Represent $f = \overline{x}_1\overline{x}_2\overline{x}_3\overline{x}_3 \vee x_1x_2x_3x_4$ by a PSDRM.

PSDRM

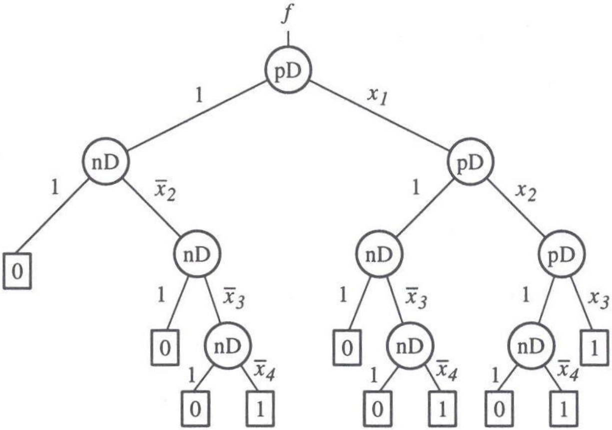


Figure 13.3 Expansion tree of PSDKRO for $x_1x_2x_3x_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4$.

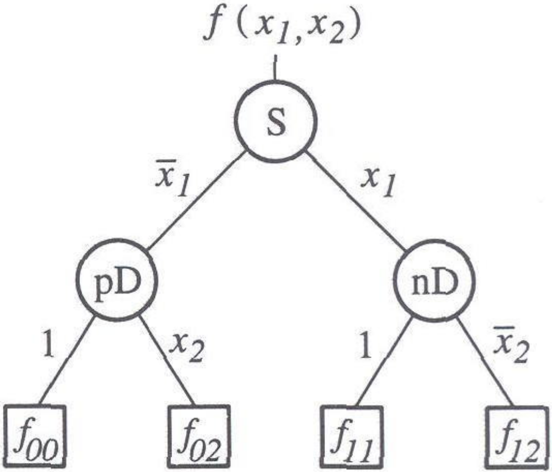
Pseudo Kronecker Expression (PSDKRO)

Definition

For a given function f, if f is expanded by the positive Davio expansion, negative Davio expansion, or the Shannon expansion, we have two sub-functions. Each of these sub-functions, may be expanded by the positive Davio expansion, negative Davio expansion, or the Shannon expansion (not necessarily the same.) We have a logical expression that is a generalization of KRO. This is a **Pseudo Kronecker Expression (PSDKRO)**.

There are up to ?? different PSDKRO for a given function. (Assuming the order of the variable expansion is fixed.)

PSDKRO



Generalized Reed Muller Expression (GRM)

Definition

Given the PPRM expansion, suppose that the polarity of each literal may be chosen arbitrarily. This is a **Generalized Reed Muller Expression (GRM**.

There are up to ?? different GRM for a given function.
(Assuming the order of the variable expansion is fixed.)

Cohn's Conjecture

Conjecture (Cohn)

Given a function f, there exists a GRM expansion with no more than

$$\binom{n}{\lfloor n/2 \rfloor}$$

terms.

ESOP

Definition

A logical expression that combines arbitrary product terms by EXORs is an **exclusive-OR sum-of-products expression (ESOP)**.

Theorem

An arbitrary n-variable symmetric function $(n = 2r)$ *is represented by an ESOP with at most* $2 \cdot 3^{r-1}$ *products.*

ESOPs for symmetric functions

by induction.

When $r = 1$, it holds.
When $r \geq 2$ we can write

$$f(x_1, x_2, \ldots, x_n) = \overline{x}_1\overline{x}_2 f_{00} \oplus \overline{x}_1 x_2 f_{01} \oplus x_1\overline{x}_2 f_{10} \oplus x_1 x_2 f_{11}$$

where

$$\begin{aligned} f_{00} &= f(0, 0, x_3, \ldots, x_n), \\ f_{01} &= f(0, 1, x_3, \ldots, x_n), \\ f_{10} &= f(1, 0, x_3, \ldots, x_n), \\ f_{11} &= f(1, 1, x_3, \ldots, x_n) \end{aligned}$$

□

ESOPs for symmetric functions

continued.

Due to symmetry, $f_{01} = f_{10}$. Thus we have

$$f(x_1, x_2, \ldots, x_n) = \overline{x}_1\overline{x}_2 f_{00} \oplus (x_1 \oplus x_2) f_{01} \oplus x_1 x_2 f_{11}$$

Givin $x_1 \oplus x_2 = 1 \oplus \overline{x}_1\overline{x}_2 \oplus x_1 x_2$, we have

$$f(x_1, x_2, \ldots, x_n) = \overline{x}_1\overline{x}_2(f_{00} \oplus f_{01}) \oplus f_{01} \oplus x_1 x_2 (f_{01} \oplus f_{11})$$

f_{00}, f_{01}, f_{11} are $(n-2)$-variable symmetric functions. What about $f_{00} \oplus f_{01}$?
By hypothesis each subfunction can be represented with at most $2 \cdot 3^{r-2}$ products. Thus

$$3 \cdot (2 \cdot 3^{r-2}) = 2 \cdot 3^{r-1}$$

ESOP Example

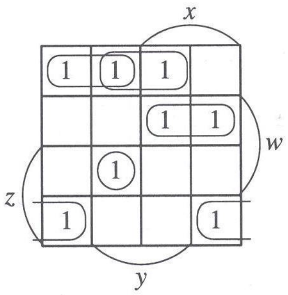


Figure 13.6 Representation using SOP.

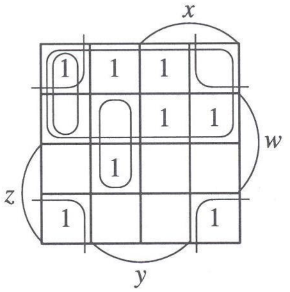


Figure 13.7 Representation using ESOP.

EXMIN2

- EXMIN2 is a heuristic minimization algorithm
- The following operation are used iteratively

$$
\begin{aligned}
\text{X\_MERGE} &: x \oplus x \rightarrow 0, x \oplus \overline{x} \rightarrow 1, \\
&\quad x \oplus 1 \rightarrow \overline{x}, 1 \oplus \overline{x} \rightarrow x \\
\text{RESHAPE} &: xy \oplus \overline{y} \rightarrow \overline{yx} \oplus x \\
\text{DUAL\_COMPLEMENT} &: x \oplus y \rightarrow \overline{y} \oplus \overline{x} \\
\text{X\_EXPAND\_1} &: x\overline{y} \oplus \overline{x}y \rightarrow x \oplus y \\
\text{X\_EXPAND\_2} &: xy \oplus \overline{y} \rightarrow 1 \oplus \overline{x}y \\
\text{X\_REDUCE\_1} &: x \oplus y \rightarrow x\overline{y} \oplus \overline{x}y \\
\text{X\_REDUCE\_2} &: 1 \oplus \overline{x}y \rightarrow xy \oplus \overline{y}
\end{aligned}
$$