

Outline

- * BDD Apply: Circuit Representation
- * A Demo-BDD-WEB Package: TUDD
- A BDD Package: CUDD
 Simple Usage
- * A BDD Package: CUDD
 - ♦ Hints for Advanced Usage: How to use CUDD
 ♦ Problem Solving
 - ♦ BDD calculator and equivalence verifier ♦ N-queen Problem (select one)

BDD - Apply: Circuit Representation: A Demo-BDD-WEB Package: TUDD A BDD Package: CUDD Simple Usage A BDD Package: CUDD Hints for Advanced Usage: How to use CUDD Problem Solving BDD calculator and equivalence verifier N-queen Problem gelect one) Increasing difficulty (solutions available on the teacher WEB page) (sop when desired ... have fun!!)

BDD – Apply: Circuit Representation

TUDD

* Main features

- Stephan Horeth University TU Darmstadt
- http://marple.rs.e-technik.tu-darmastadt.de/~sth/demo.html
- Integrates different decomposition type
- Demo-WEB page with good graphical interface
- Package on request
- * Laboratory duty
 - "Play" with TUDD, i.e., select
 - ♦ Function
 ♦ Variable Order
 - Variable Order
 Decomposition Type
 - ... toy-tool ... have fun ...

CUDD

Main features

- Fabio Somenzi Boulder/Colorado WEB page
- <u>http://vlsi.colorado.edu/~fabio</u>
 (copy in <u>http://www.polito.it/~guer/teaching/phd/fv/laib</u>)
- Most widly used BDD package
- Integrates BDD, ADD (Algebraic Decision Diagram), ZDD (Zero-Suppresed Decision Diagrams)
- Very Efficient
- Many Realeases over the years ... now version -2.3.1
- (Includes the dddmp package from G. Cabodi and S. Quer)

Simple Usage

- * Laboratory Duty
 - Grab and uncompress it
 - Compile it

 - See Makefile in the root directory Small modification IFF necessary (architecture parameters, directory positions, etc.) "Run" ∻
 - ∻
 - Make
 - Check main features out
 - See documentation
 - Directory cudd/doc File cudd.doc (text file)

- Build BDD for standard ISCAS benchmarks (with c17.blif, c..., etc. Sequential benchmarks s713.blif #PI=35, #PO=23, #FF=19, #Gate=393 s1512.blif #PI=29 , #PO=21, #FF=57 , #Gate=780 s1423.blif #PI=17, #PO=5, #FF=74, #Gate=657 (from now on <c>)
 - s298, s1196, s1238, , s1488, s1494 save output results ...

Advanced Usage

- ✤ Hints to use CUDD:
 - DD Manager
 - Type
 - DdManager * ♦ Functions
 - Cudd_Init
 - Elementary BDD Variables Type
 - DdNode * ♦ Functions (somehow similar)
 - Cudd_bddIthVar
 - Cudd_bddNewVar

Build BDDs

- Start from constant one (get zero from "not" (one)) ♦ Start from constant one (yet 2
 ♦ Functions (somehow similar)

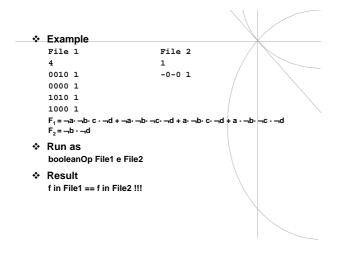
 - Cudd_ReadOne
 - DD_ONE
- Proceed through the "circuit/function"
 - Cudd_Not
 - Cudd_bddAnd _
 - Cudd bddOr etc
- Each new BDD has to be referenced Cudd Ref
- ∻ Useless node must be dereferenced
- Cudd RecursiveDeref

- Check Results and Statistics Functions
 - Cudd CountMinterm
 - Cudd PrintMinterm _
 - Cudd_DagSize
- Quit the manager
 - ♦ Function
 - Cudd_quit

Problem solving 1 Boolean Function Manipulation and (combinational) Equivalence Checker

- * Write Program
 - booleanOp
- Run it as
- booleanOp <fileName1> <op> <fileName2> where
- ٠

 - cop> is the operation a stands for and o stands for ard x stands for xor e stands for xor
- Result
 - report statistics on resulting function (e.g., print out BDD minterms, PLA format)





The N-quenn problem (chess puzzle)

- Write Program
- 8aueen
- Run it as ♦ 8quenn <N>
- where
 - <N> specifies the board size (N x N)
- Result
 - Report number of solutions
 - (Eventually) the solution themselves (somehow coded) ٠

Coding the Problem

- * Chess Board NxN
- ✤ For each position in the variable create variable x_{i,j} (i row index, j column index, from 1 to N)/
- Relations (constraints no queen in conflict)
 - Iff there is a queen in i, j no other queen in row i ٠ $X_{i,j} \Rightarrow \prod_k \neg X_{i,k}$, with $k = [1, N], k \neq j$
 - Iff there is a queen in i, j no other queen in column j
 - $X_{i,j} \Rightarrow \prod_{k} X_{k,j}$, with $k = [1, N], k \neq i$ Iff there is a queen in i, j no other queen in same diagonal ٠
 - $\mathbf{X}_{i,j} \Rightarrow \prod_k \neg \mathbf{X}_{k,j+k-i} \quad , \quad \text{with } \mathbf{k} = [1, \mathbf{N}], \, j+k-i = [1, \mathbf{N}], \, \mathbf{k} \neq i$ Iff there is a queen in i, j no other quenn in same inverse ٠ diagonal
 - $X_{i,j} \Rightarrow \prod_k \neg X_{k,j+i\cdot k}$, with $k = [1, N], j+i-k = [1, N], k \neq i$

Relations (constraints – enough queens)

- There must be a queen for each row $X_{i,1} \lor X_{i,2} \lor X_{i,3} \lor \ldots \lor X_{i,N} \quad , \quad \text{for all row } i = [1,N]$
- Final Relation
 - Taking the conjunction of all the previous ones ٠ When true there is a solution ٠

N.B. АВ а⇒в

- 0 0 1 1
- 0 1
- 1 0 0
- 1 1 1 $A \Rightarrow B = \neg A \lor B$