HARDWARE ESTIMATION

INSTRUCTORS
Prof. M.BALAKRISHANAN
Prof. ANSHUL KUMAR

SUBMITTED BY

RAMESH KUMAR(VLS01023)
ASHISH KUMAR SHARMA(VLS01009)

PART-1

INTRODUCTION:

Embedded system is an important field of in Computer Engineering.Embedded Systems can be found in a large variety of applications today like image processing,networking and wireless communication.They essentially coprise of aprocessor and some hardware built around it

MOTIVATION:

Embedded systems generally have a heterogeneous implementation.On one end performance and low power cosiderations encourage use of ASICS where as cost and flexibility requirements motivate the designer to use re-programmable components.

The purpose of Hardware Estimation is to estimate,without actually synthesizing the circuit,various parameters in circuit design like area,number of pins,executiontime,channel widths etc.Hardware Estimation is vital because of two reasons. First.it enables the designer to evaluate the cost and performance of hardware without actually doing the synthesis.Second,it enables the system designer to explore design alternatives by providing quick feedback for any design decision

HADWARE ESTIMATION MODEL:

Here a control unit/datapath model is for which various methods are used to compute the following quality metrics:

1.Clock Cycle estimation

Here two methods are used to caculate the clock cycle os the system.First one is Maximum-oprator-delay and second is Slack-minimization method. In slack-minimization we take appripiate clock cycle so that maximum utilization of the resourses.Various algorithm is known to minimize it.

2.Control step minimization

The number of control steps neede for a behivor’s execution can be estimated in several ways.Two algorithms are used here to calculate it.First one is Operator-use method and second one is Scheduling.First method partitions all statements ib the behavior into a set of nodes in such way thatr all statements in anodecould be executed concuurently.It can be extented to incorporate pipelined functional units and include memory accesses.

In second, each type of operation in the behavior ,the list scheduling algorithm maintains a priority list form which operations are assigned to cotrol steps.

3.Execution time estimation

If behavior is described by straight-line code,the star-to-finish execution time for the behavior depends on the number of control steps into which the behavior is scheduled.Probability-based flow analysis is used to calculate it.We can determine the execution frequencies of each of its nodes using flow analysis based on the branching probabilities.

4.Comunication rate estimation

Communoication channels may be either explicitly specified in the behavior description or created when avariable being accessed by aportion of yhr behavior is assigned to a different chip during system partitioning.

The total execution time for any behavior consists of two components:Compution time and Communication time.

5.Area estimation

The primary task in estimating the size of any design is determining the number and type of components that will empliment the given behavior.For an FPGA implementation,an estmate of the toatl number of CLBs(combinational logic blocks) used in the design can be obtained by summing the number of CLBs uesd for each component in the design.

The Datapath consists of three kinds of RT components: storage units,Functional units and interconnect units.Storage units can be caculated by either by **clique partitioning** or **left-edge algorithm**.Funtional units required to implement the behavior can be dtermined using the **forced-directed algorithm .**We can estimate the number of interconnect units,such as buse and multiplexers,which are required for interconnecting the storage and the functional units.

6.Datapath area estimation

The datapath is usually a bit-sliced stack in which the individual datapath components are placed in rows,one over the other.To determine the area of the datapath,we need to determine the total length Lbit and Hbit of a bit-slice.We first determine the number of transistors in each bit-slice of the datpath,tr(DP).

7.Control unit area estimation

The control unit consists of a state register,the control logic and the next-state logic.The stae register contains encoded states.The control logic defines the control signals for the datapath components as a funtion of current state. The datapath status lines communicate results of comparision operations in the datapath to the control logic.The contrl logic and next-state logic unit may be imlpementade as ROM or PLA. Total area of the control logic is then approximated as multimlication of transistor area coeficient and total number of transistor in control logic.

8.Pin estimation

The communication width or the number of wires at a behavior’s boundary required for accessing external data depends on how the communication is specified in the functional specification.Port declaration,communicatin,global datnad procedure calls.

BEHAVIORAL LEVEL POWER ESTIMATION AND EXPLORATION

High level synthesis has aroused considerable interest in the recent years. While a lot of effort has been put into synthesis for speed and area, power optimization has been explored only recently. Estimation of power con-sumption of a design is the first step towards integrating power minimization techniques into any synthesis system.

Behavioral level estimation is essential because lower level estimation tools are time consuming and their use precludes a complete exploration of the design space.

PREVIOUS WORK

Increased interest in low power designs has stimulated a lot of research activity in the area of power estimation.Traditionally, circuit level simulators like SPICE and IRSIM have been used for power estimation.

.

Power estimation

The power cosumed by general hardware resoures is given by

$$Power = \alpha C_1 V^2 f$$

where α is the activity factor, C_1 is the physical capac-itance of the component, V is the supply voltage and f is the clock frequency. In the behavioral context, this trans-lates to

Behavioral level estimation is essential because lower

$$Power = N_a C_2 V^2 f_s$$

where N_a is the number of accesses to the resource over the period of computation, C_2 is the average capaci-tance switched per access and f_s is the computation (or sampling) frequency.

The number of accesses to a hardware resource depends on the algorithm and the architecture. The esti-mation is targated for real time DSP application on ASIC architectures.

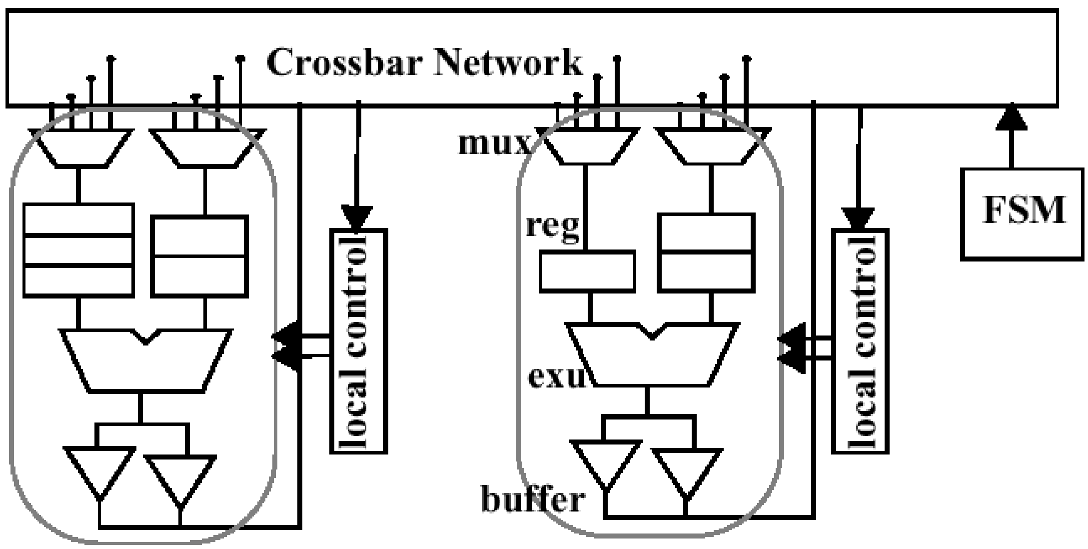


Fig. The targted architecture ASIC

The power consumed by an ASIC design comprises of contributions from the following hardware resources: datapath, interconnect, control and memory. The power consumed by some of these components can be esti-mated analytically from the control/data flow graph (CDFG) which is used to represent the design. Statistical models were developed to estimate those components for which adequate information is not available at this level. Power estimation of each of the components is dis-cussed in the following sections.

Datapath component

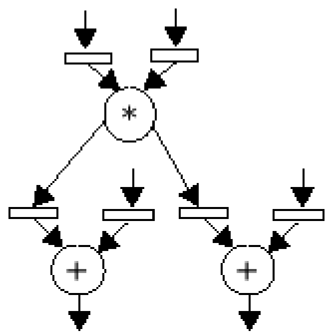
A general datapath consists of execution units, regis-ters, multiplexers and buffers (tri-stated and otherwise). Given a hardware library, the capacitance switched per access for datapath components is estimated as explained above. The number of accesses to each of the datapath components is determined as follows:

Execution units

Every node in the CDFG represents an operation to be performed on the corresponding hardware. The number of accesses to each type of hardware is exactly the num-ber of nodes of that type in the graph.

Registers

The number of read accesses to registers is exactly equal to the total number of inputs for each node in the graph, under the assumption that no chaining of opera-tions is performed. The number of writes to registers, on the other hand, is dependent upon assignment . As shown in



Buffers

A buffer is utilized every time a bus is accessed. Unfortunately, the number of accesses to global buses is very difficult to estimate analytically. The number of bus accesses correlates lin-early to the total number of edges in the graph. This sta-tistical model was used to estimate the number of accesses to buffers.

INTERCONNECT

The power consumed by the interconnect network may be a significant percentage (10 - 30%) of the entire chip power. However, it is the most difficult component to estimate. The total interconnect capacitance switched is determined by the physical capacitance of the bus, times the number of accesses. It is only after a complete layout that the area of the chip and the average length of busses can be determined. Also, some busses are local to a data-path and do not contribute to interconnect capacitance.
As a result, both the physical capacitance of the global buses and the number of accesses to them is unknown. This warrants the need for statistical estimations.

$$Area = \alpha_1 + \alpha_2 A_{act} + \alpha_3 N_{bits} N_{bus} A_{act}$$

where *A act* is the sum of the areas of the execution units and registers, *N bits* is the average word length used and *N bus* is the number of buses.

CLOCK

The power consumed by the gates loading the clock lines is taken into account with the datapath power, since input capacitances are included in the module capaci-tances.

The only component of clock power that is not accounted for, is the interconnect wires which can be treated in the same way as any other bus, except that it is accessed every clock period.

Controller

The controller may consume 5-20% of the total power in a design It was found that the total capacitance switched for the finite state machine *(C FSM*) is related to the number of states per sample period *(N States*) as shown in

$$C_{FSM} = \alpha N_{States} + \beta$$

The sum of the capacitance switched in the local con-trollers found to approximate the following model.

$$C_{lc} = \beta_0 + \beta_1 N_{trans} + \beta_2 N_{States} + \beta_3 B_f$$

4. Design space exploration

High level decisions made early in the design cycle can greatly influence the quality of the final result. Since they limit the search space, bad decisions at this level may lead to non-optimal solutions. To provide the user with estimates of performance metrics over the entire design space, the estimation techniques developed in ***Explor*e.**

Given a CDFG description of an algorithm and a library of hardware modules, *Explore* selects the optimal hardware available for a range of clock periods and sup-ply voltages and estimates the corresponding power and area for a fixed throughput. The “optimal hardware” is chosen according to the following strategy: the hardware unit that can perform the operation in the lowest number of clock cycles is chosen. In case of conflicts, the one with smaller area is selected.

The effect of voltage reduction is to increase the delay of the unit, and hence the number of cycles required to perform a particular operation on that unit. As a result a different hardware unit may become optimal or the clock cycles allocated for the operation may increase.

Decreasing the clock period will increase the total number of clock cycles available for scheduling opera-tions (if the throughput is kept constant) and may reduce the number of resources required. On the other hand, it may increase the number of clock cycles that a unit needs to perform a particular operation resulting either in an increase in critical path (in terms of clock cycles) of the design or in the selection of different units.

It is clear that the choice of the supply voltage and the clock period affects the speed, area and power of the design in a complex way.

Transformation Selection

Transformations may effect the power consumed in several ways. Different transformations on a 14 th -order wavelet filter with a 20 ns clock period and 300 ns sample period. Substituting constant multiplications by shifts and adds results in a 50% decrease in power. The lowest possible supply voltage for both versions is 4.5V.

Pipelining the transformed version lowers the critical path of the design considerably, making it possible to lower voltages to 1.5 V.

PART –2

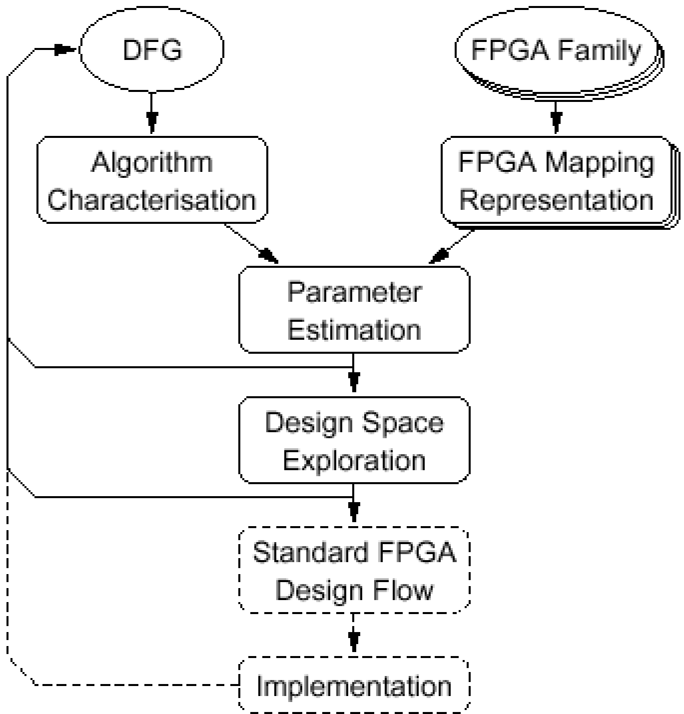
High-Level Area and Performance Estimation of Hardware Building Blocks on FPGAs

The technology development together with the growing acceptance of FPGAs makes the implementation of more complex designs both possible and interesting. Especially for applications that require extensive data processing.The benefit of such an aid is twofold.On one hand,it allows to roughly quantify an FPGA design and therefore enables early comparisons to traditional approaches.On the other hand,it supports especially less experienced designers and thus,can help to make FPGAs more popular and regarded.

Starting with a register transfer level (RTL)description,the area is estimated by predicting look-up table (LUT) mapping,con .gurable logic block (CLB)construction,and placement.The sub- sequent timing estimation is based on predicting CLB delay,wiring delay,and input-to-output delay.

Estimation Methodology

As shown in below figure outlines,the proposed estimation methodology consists of four major steps:algorithm characterisation,FPGA mapping representation,parameter es timation,and design space exploration.The basic idea behind the concept is to separate the algorithm description from the in .uence of the FPGA architecture.



Algorithm Characterisation

This step characterises the algorithm on a high level,independent from any implementation issue.For that purpose,a DFG,a block diagram,or any comparable description is suitable.A task is characterised by its inputs and outputs, its operations,its extra registers (i.e.not pipeline registers),its inherent structure,and the number of iterations it is run through.The Characterisation Vector shown in Table 1 represents the characteristics of an algorithm,whereby several simplifications are used:

–Not each element s considered separately.Elements of the same types are combined and the average word length over all the instances is used.

–The precise structure of the task is neglected.Instead,the degree of parallelism is defined,which provides a coarse idea of the task 's composition.We propose the heuristic

Degree of parallelism =Total number of operations/No.of operations in the longest data path .

FPGA MAPPING REPRESENTATION

This step considers the in .uence of the FPGA architecture,especially on the mapping of the operations.Therefore,a mapping model is speci .ed for each type of operation,from which the area and timing parameters are derived.

Description		Element
Data inputs	Number	C_1
	Average word length	C_2
Data outputs	Number	C_3
	Average word length	C_4
Control I/Os	Number	C_5
	Average word length	C_6
Adders	Number	C_7
	Average word length	C_8
Array multipliers	Number	C_9
	Average input word length	C_{10}
Multiplexers	Number	C_{11}
	Average fan-in	C_{12}
	Average word length	C_{13}
Bit-wise logic operations	Number	C_{14}
	Average fan-in	C_{15}
	Average word length	C_{16}
LUTs	Number	C_{17}
	Average fan-in	C_{18}
	Average word length	C_{19}
Extra registers	Number	C_{20}
	Average register length	C_{21}
Inherent degree of parallelism		C_{22}
Number of iterations		C_{23}

Characterisation Vector Table

Parameter estimation

Based on the antecedent steps of algorithm characterisation and FPGA mapping representation,the parameters area,timing (propagation delay),number of iterations,and number of I/O pins are estimated for the overall task.The following equations show the calculation rules.

Area [CLBs]:

$$A = \frac{C_7 C_8}{2} + 2C_9 {C_{10}}^2 + \frac{C_{11}C_{12}C_{13}}{4} + \frac{C_{14}C_{15}C_{16}}{8} + \frac{C_{17}C_{18}C_{19}}{8} + \frac{C_{20}C_{21}}{2}$$

Timing (propagation delay) [s]:

$$T = \frac{1.5}{C_{22}} \left(C_7(d_r C_8 + d_c) + 2d_c C_9 C_{10} + d_c C_{11} \log_2(C_{12}) + d_c \frac{C_{14}C_{15}}{4} + d_c C_{17} \right)$$

Number of iterations:

$$I = C_{23}$$

Number of I/O pins:

$$P = P_{\mathrm{Din}} + P_{\mathrm{Dout}} + P_{\mathrm{Ctrl}} = C_1 C_2 + C_3 C_4 + C_5 C_6$$

Operation	Substitution
Subtracter	Adder
Comparator	Adder
Shifter Barrel-shifter	Multiplexer neglected (since hard-wired)
Multiplier	▷ Array multiplier ▷ KCM[a]: 1 LUT (fan-in ρ), $2^{log_2(\rho)-2} - 1$ adders (word length 2ρ) ▷ Serial: 1 adder, 1 MUX (fan-in 4), 1 logic operation (fan-in 2)

[a] ρ: word length of the original multiplier

For area and timing the approach s basically the same:the contributions of each operation type are summed up and for the latter divided by the degree of parallelism.

Design Space Exploration

The presented estimation methodology allows to investigate area vs.performance trade-offs on a high level.The standard methods for that purpose are pipelining, replication and decomposition.Pipelining inserts S registers into the data path in order to shorten the critical path length.Replication intends to copy the building block R times.Decomposition breaks up the building block into a sequence of D identical subtasks,which are then iteratively reused D times.

Table shows the final estimation metrics for a building block as well as the extension for design space exploration.Due to the FPGA 's CLB structure, pipelining can be taken into account rather accurately.Similar to area and timing,the number of potential pipeline stages can be predicted to

		Estimation	Design Space Exploration		
		Building Block	Pipelining	Replication	Decomposition
Area	[CLBs]	A	A	$R \cdot A$	$\frac{1}{D} \cdot A$
Frequency	[Hz]	$\frac{1}{T}$	$\frac{S}{1.3} \cdot \frac{1}{T}$	$\frac{1}{T}$	$D \cdot \frac{1}{T}$
Throughput	[bit/s]	$\frac{P_{\mathrm{Din}}}{T \cdot I}$	$\frac{S}{1.3} \cdot \frac{P_{\mathrm{Din}}}{T \cdot I}$	$R \cdot \frac{P_{\mathrm{Din}}}{T \cdot I}$	$\frac{P_{\mathrm{Din}}}{T \cdot I}$
Latency	[cycles]	I	$S \cdot I$	I	$D \cdot I$
I/O	[pins]	P	P	$R \cdot P$	P

COCLUSIONS:

Hardware estimation is important in designing of Embedded System.In above we discussed to calculate thedifferent quality metrics of the Hardware .Also i discussed for ASICs and FPGAs.I presented Power estimation in ASICs.The methodology allows early and fast parameter estimation based on a high-level description of the algorithm.A combination of analytical and statistical technique for behavioral level power estimation have been proposed.Thus,early design space exploration is facilated and basic trade-offs can be investigated on a high level.Beneficially,the runs throuh the standard FPGA design flow are minimsed.

REFERENCES

1.Specification and Design of Embedded Systems by
-Daniel D.Gasjki,Frank Vahid,Sanjiv Narayan,Jie Gong

2.Proc.First International Workshop on Low Power Design,Napa Valley,CA,pp.197-202 April 1994.”Behavioral Level Power Estimation and Exploration”.By
-Renu Mehra and Jan Rabaey

3.”High-Level Area and Performance Estimation Of Hardware Building Blocks On FPGAs”
-Rolf Enzler, Tobias Jeger, Didier Cottet, and Gerhard Tr¨oster
Swiss Federal Institute of Technology (ETH), Electronics Laboratory
CH-8092 Zurich, Switzerland
PPT file:
R.W. Hartenstein and H. Gr¨unbacher (Eds.) FPL 2000, LNCS 1896, pp. 525–534, 2000. c_Springer-Verlag Berlin Heidelberg 2000

4.www.google.com

5.www.altavista.com