ECE 695R: System-on-Chip Design

Module 2: HW/SW Partitioning
Lecture C: Automatic Custom Instruction Generation: Re-use

Anand Raghunathan
raghunathan@purdue.edu

Fall 2014, ME 1052, T Th 12:00PM-1:15PM
Automatic Custom Instruction Generation: Phased Approach

- Challenge: HUGE design space!
- Phased approach
  - Local generation: Focus on small kernels ("hot-spots") in the code and generate custom instructions
  - Re-use: Identify opportunities for re-use in other parts of the program
  - Global selection: Select best combination of custom instructions for the entire program
Re-use of Custom Instructions

OK, we can automatically generate Instruction-Set Extensions

Can we re-use them elsewhere in the program?
Instruction Re-use

• Given a code segment:

\[ \begin{align*}
Y_1 &= a \cdot R_1 + b \cdot G_1 + c \cdot B_1 + d; \\
Y_2 &= a \cdot R_2 + b \cdot G_2 + c \cdot B_2 + d; \\
Y &= Y_2 + q \cdot (Y_1 - Y_2);
\end{align*} \]

• And having identified special instructions:

\[ \begin{align*}
\times \\
\times \\
+ \\
s \\
t \\
w
\end{align*} \]

\[ \begin{align*}
\times \\
foo \\
w
\end{align*} \]

• Find out all places where the instruction can be used in the code:
  - **Maximise** usage—and hence **advantage**
  - Increase **robustness to changes** in the application
Instruction Re-use

RISC + "foo":
15 operations =
11 instructions =
~11 cycles

Best?!
Using Simple Algebraic Properties

- Identity elements
  
  \[ c = c \times 1 \]

- Associativity, distributivity, etc.
  
  \[ (a + b) c = a c + b c \]
Using Simple Algebraic Properties

19 operations
9 instructions
~9 cycles

Best?!
More Use of Algebra

• Given a code segment:

\[
\begin{align*}
Y_1 &= a \cdot R_1 + b \cdot G_1 + c \cdot B_1 + d; \\
Y_2 &= a \cdot R_2 + b \cdot G_2 + c \cdot B_2 + d; \\
Y &= Y_2 + q \cdot (Y_1 - Y_2);
\end{align*}
\]

Formulate one polynomial for the basic block

\[
Y = a \cdot R_2 + b \cdot G_2 + c \cdot B_2 + d + q \cdot a \cdot R_1 + q \cdot b \cdot G_1 + q \cdot c \cdot B_1 - q \cdot a \cdot R_2 - q \cdot b \cdot G_2 - q \cdot c \cdot B_2;
\]

Perform various transformations; factor \( p = q - 1 \)

\[
Y = (p \cdot R_2 + q \cdot R_1) \cdot a + (p \cdot G_2 + q \cdot G_1) \cdot b + (p \cdot B_2 + q \cdot B_1) \cdot c + d;
\]
Rewriting Polynomials to Expose Re-Use

- 17 operations
- 7 instructions
- ~7 cycles

Peymandoust, Pozzi, Ienne, De Micheli – ASAP 2003

• Can be automated using Gröbner bases
• Available in Maple (simplify), Mathematica (AlgebraicRules),…