An Improved Micro-Architecture for Function Approximation Using Piecewise Quadratic Interpolation

Shai Erez & Guy Even

School of EE

Tel-Aviv Univ.

ICCD-08

Problem Description

Design a HW circuit that evaluates arithmetic functions.

- Wide variety of functions

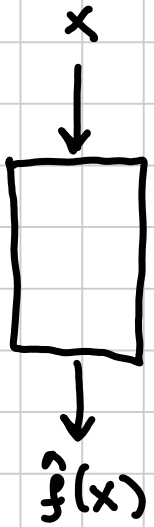
$f(x) \in \{ 1/x, \sqrt{x}, 1/\sqrt{x}, e^x, \log x, \sin x \ldots \}$

- many applications require (only) small precision

Single precision floating point : 23 bits for fraction

- Single cycle computation

function approximation



Specification (min-max norm):

$$\forall x: \quad |f(x) - \hat{f}(x)| < \text{Unit last position } (i.e.\ 2^{-24})$$

Good News: In single precision, we can validate a design by exhaustive testing (i.e. simulate all the possible inputs).

Focus: we discuss $f(x) = \frac{1}{x}$ for $x \in [1,2)$

$|x| = 24$ bits.

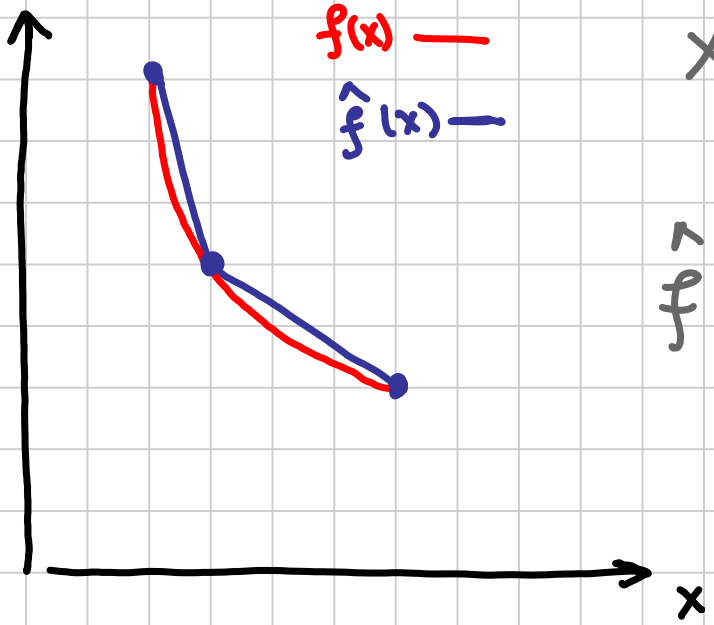
Overview: Methods For Computing 1/x

* "long division", SRT, division by recurrence.
  cheap but too slow

* multiplicative methods: Newton iterations, etc.
  fast but costly

* table based methods: linear approximation + modifications
  & quadratic approximations (polynomial approx.)
  good for small precision.

linear piecewise approximation



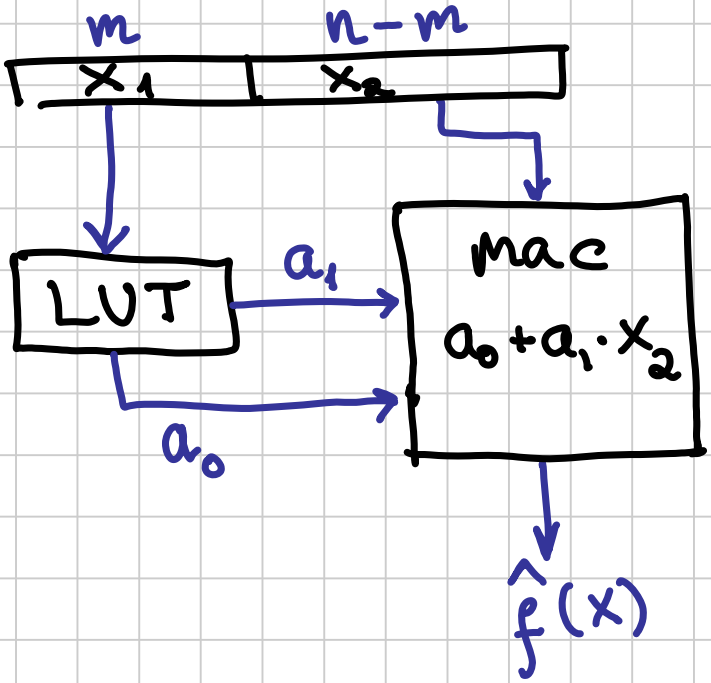
$x =$ 

x_1 = determines segment

x_2 = determines offset in segment

$$\hat{f}(x) = a_0 + a_1 \cdot x_2$$

micro architecture :



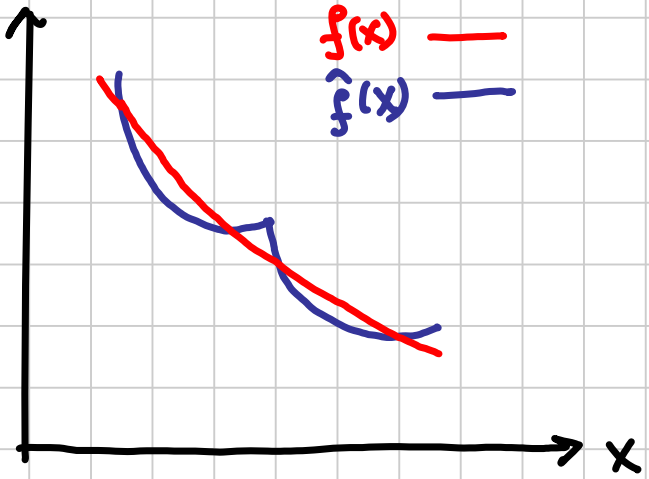
To achieve 2^{-24} prec:

⇒ big table (> 50 Kb)

because $m \geq 11$

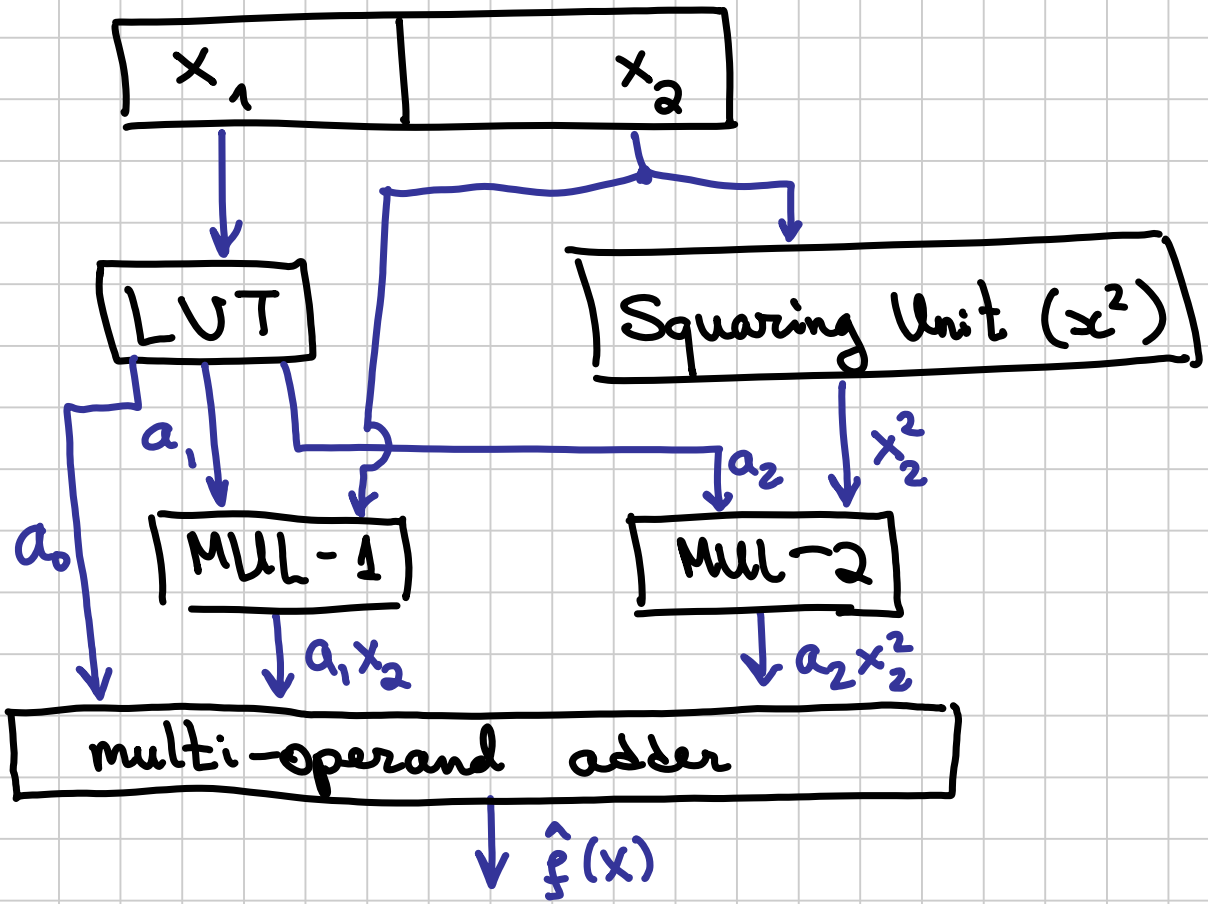
and $|a_1| + |a_0| \geq 36$

Quadratic piecewise approximation



$$x = x_1 + x_2$$

$$\hat{f}(x) = a_0 + a_1 x_2 + a_2 x_2^2$$



Micro-Arch. (POMB-05
WS-05)

* Squaring unit optimized

* Small LUT (<15Kb)

proposed: micro-architecture

Horner's Method :

$$\hat{f}(x) = a_0 + a_1 x_2 + a_2 x_2^2$$

$$= a_0 + x_2 (a_1 + a_2 x)$$

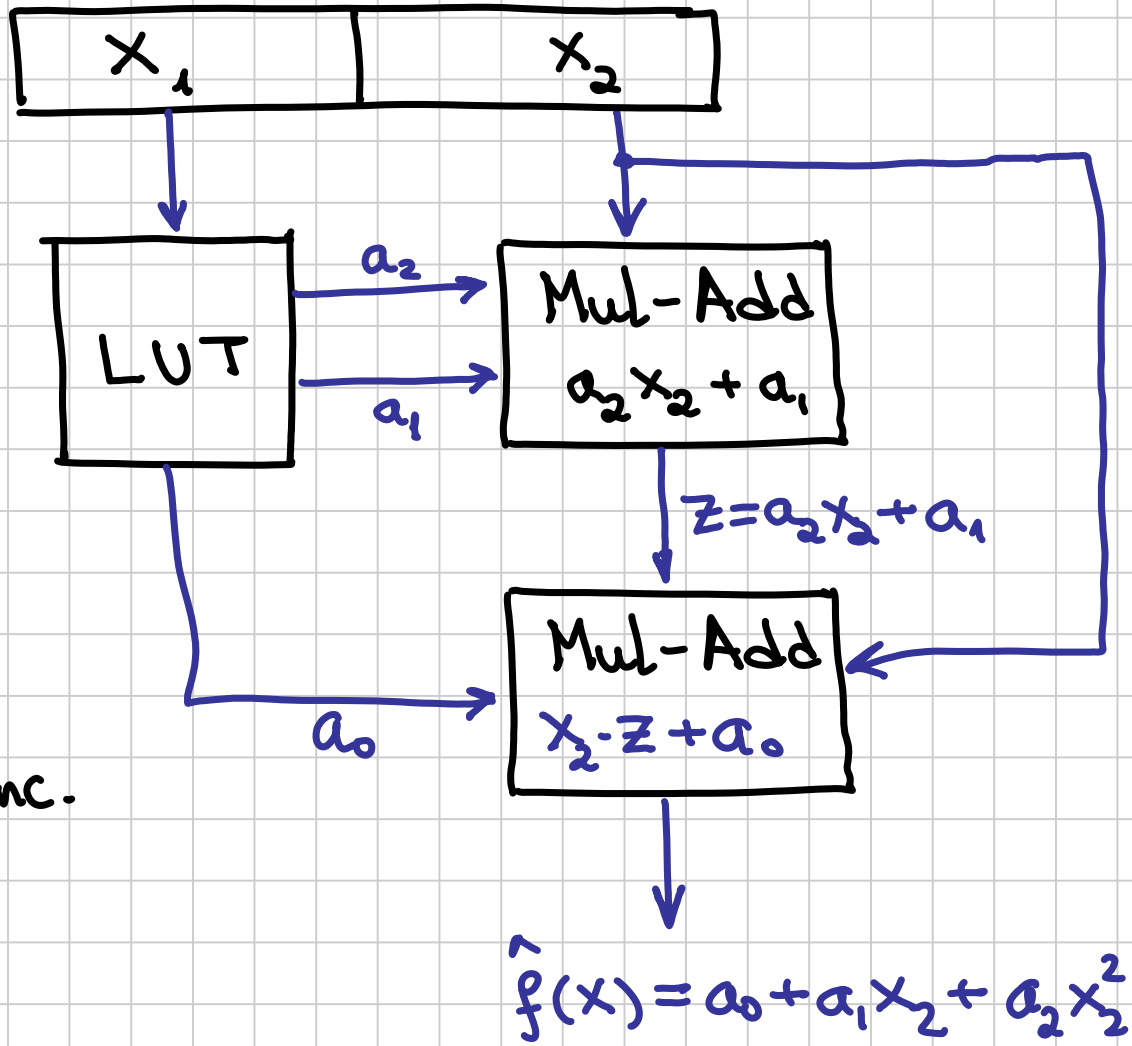
Advantages:

* 2 mul-add vs. 3 mul.
* fewer partial products (324 vs. > 400)
* suitable for pipelining with very short clk periods.

proposal : micro-architecture (cont.)

* Optimize Mul-Add:
  * Booth radix 4
  * truncated multipliers
  * represent Z in redundant rep. (CS).

direct CS → Booth-4 enc.



Systematic Design Procedure

* determine $m = |X_1|$ (2^m = no. LUT entries)

  we chose $m = 7$.

* compute coefficients a_0, a_1, a_2 for every X_1.
  - min length of coefficients.
  - effect { cost (LUT size, # partial products)
    delay (length $Z = a_2 X_2 + X_1$)
  - round coefficients and meet prec. specs.

* truncate multipliers : reduce cost

  (usually : truncated multiplies XOR Booth-4 because of error analysis - we do both.)

Comparison

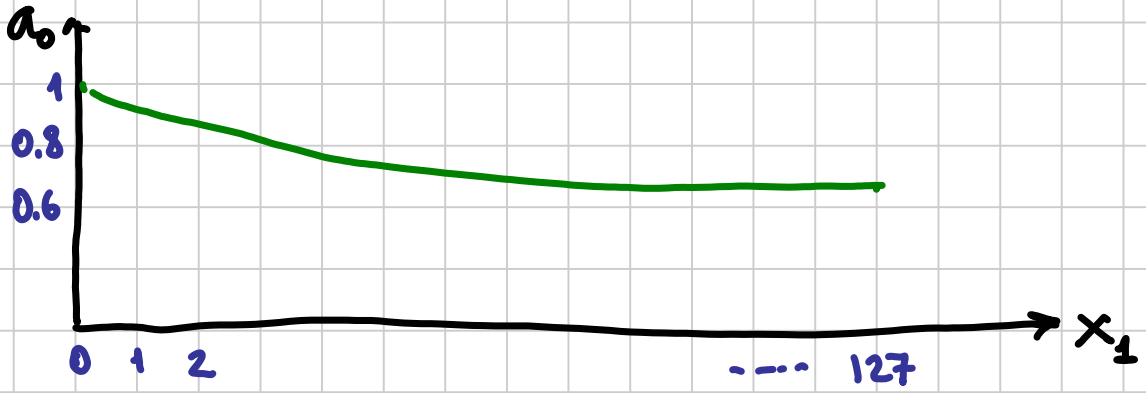
	m	$\lvert a_0 \rvert$	$\lvert a_1 \rvert$	$\lvert a_2 \rvert$	delay FA	#partial products
POMB-05	7	26	16	10	14.5	445
OUR	7	26	18	10	19	324

- cheaper
- suitable for pipelining $2 \times 10_{FA}$.

Further Work (compress tables)

$2^7 = 128$ table entries for a_0, a_1, a_2.

a_0 is a function of x_1 where : $|x_1| = 7$ bits, $|a_0| = 26$ bits



IDEA:

Compute a_0 using linear approx.

- smaller table
- reuse Mul-add units.

Summary

1) Micro-architecture for small precision function evaluation.

2) Systematic method for computing table entries, multiplier sizes, multiplier truncation.

3) Design : cheaper, slower : $\Phi = 19$ or 10 FAs.

4) Exhibit tradeoffs between delay & cost.

Thanks!