HOMOMORPHIC INTEGER DIVISION FOR TFHE

Olivier Bernard Amit Deo Marc Joye

FHE.org 2024 · Toronto, March 24, 2024

MOTIVATION

- Division is one of the 4 basic arithmetic operations
- Arguably also the most difficult to implement

How can we implement it efficiently using TFHE?

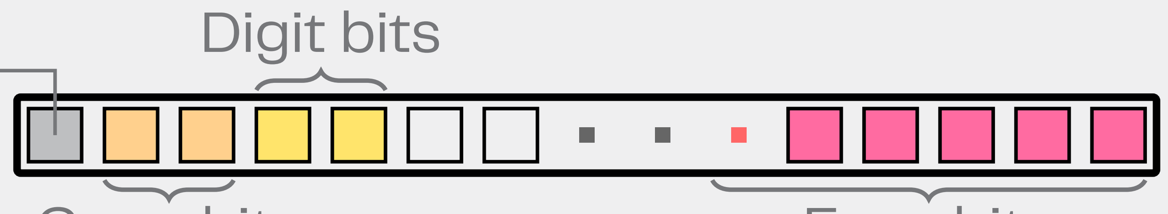
TFHE-rs Integer Format

- **Radix-4** decomposition

$$u = \sum_{i=0}^{m+n-1} u_i \cdot 4^i, \qquad u_i \in \{0, 1, 2, 3\}$$

- Each digit homomorphically encrypted in a 64 bits ciphertext:

$\mathbf{Enc}(u_i) =$



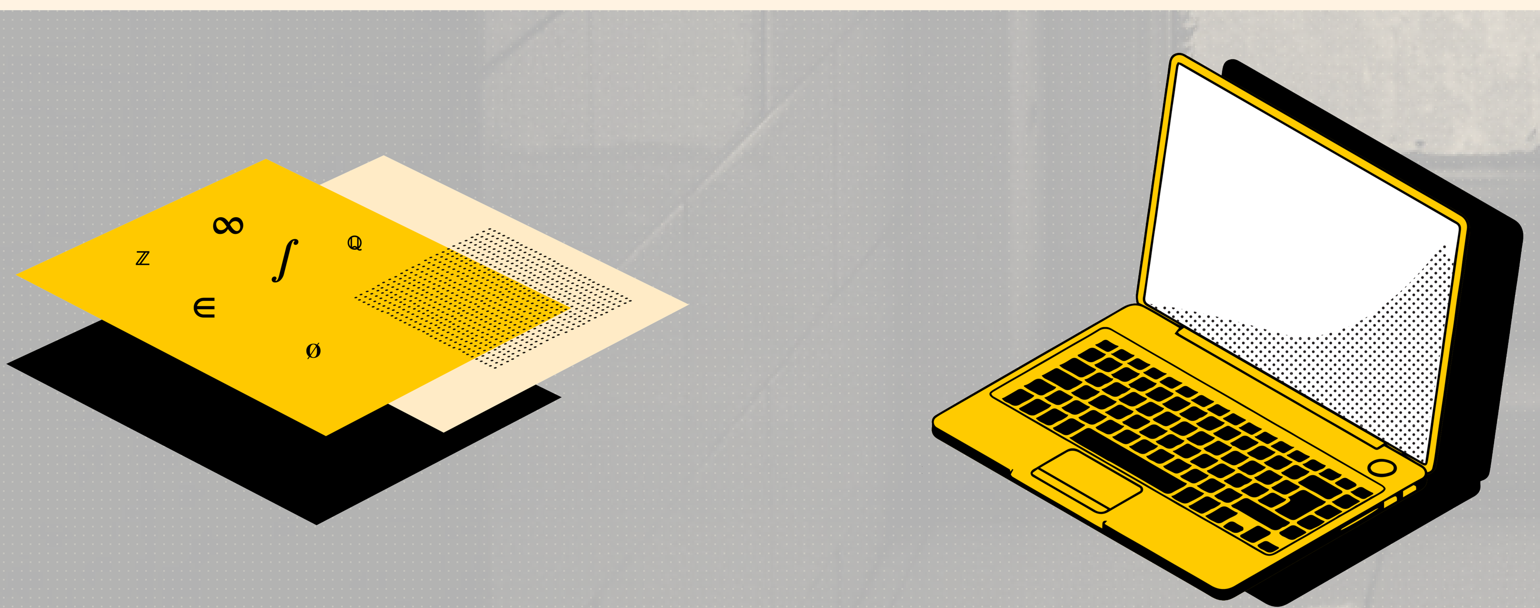
- Dividend u is a collection of $(m+n)$ ciphertexts
- Divisor v is a collection of n ciphertexts

Modified radix-4 division

- Modified version of schoolbook division
- Partial remainders computed *via* full carry propagation

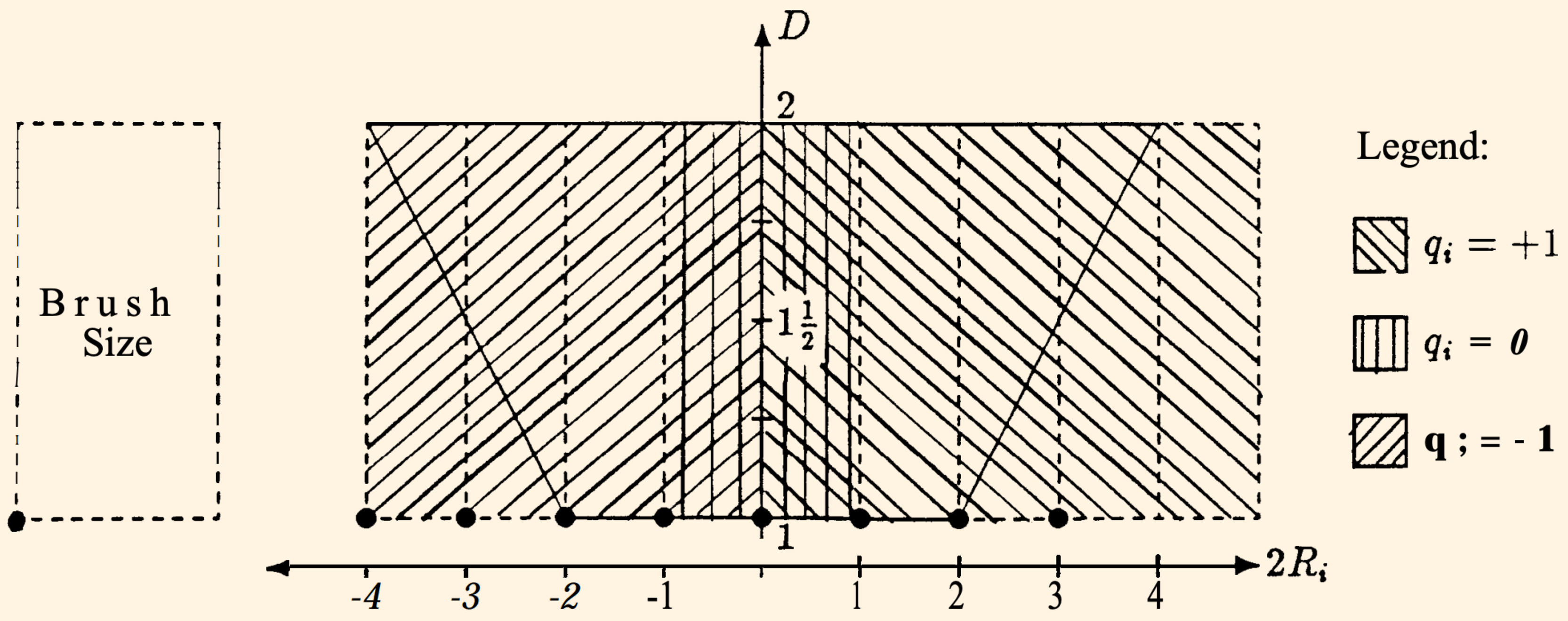
Input: Nonnegative integers $u = (u_{m+n-1}\, u_{m+n-2} \ldots u_0)_4$ and $v = (v_{n-1}\, v_{n-2} \ldots v_0)_4$ with $v_{n-1} \neq 0$
Output: $q = \lfloor u/v \rfloor$ and $r = u \bmod v$

$u_{m+n} \leftarrow 0$; $\overline{v} \leftarrow \mathrm{CPL}^{(4)}_n(v)$; $\overline{V} \leftarrow 2\overline{v}$
for $j = m$ **downto** 0 **do**
  $(w_n\, w_{n-1} \ldots w_0)_4 \leftarrow (u_{j+n}\, u_{j+n-1} \ldots u_j)_4 + \overline{V}$
  $\beta \leftarrow w_n \bmod 2$; $\rho \leftarrow w_n - \beta$
  if $(\rho = 2)$ **then** $(u_{j+n}\, u_{j+n-1} \ldots u_j)_4 \leftarrow (\beta\, w_{n-1} \ldots w_0)_4$
  $(w_n\, w_{n-1} \ldots w_0)_4 \leftarrow (u_{j+n}\, u_{j+n-1} \ldots u_j)_4 + \overline{v}$
  $\beta \leftarrow w_n$
  if $(\beta = 1)$ **then** $(u_{j+n}\, u_{j+n-1} \ldots u_j)_4 \leftarrow (0\, w_{n-1} \ldots w_0)_4$
  $q_j \leftarrow \rho + \beta$
end
$q \leftarrow (q_m\, q_{m-1} \ldots q_0)_4$; $r \leftarrow (u_{n-1}\, u_{n-2} \ldots u_0)_4$
return q, r



SRT division with radix-4 inputs

- Quotient base $\beta = 2$ with **redundant** digit-set $\{0, \pm 1\}$
- Redundancy ⇝ use **estimates** of current remainder/divisor
  ⇝ **small lookup tables** for quotient digit selection
- Lookup tables from SRT diagrams [WH86]:



Input: Nonnegative integers $u = (u_{m+n-1}\, u_{m+n-2} \ldots u_0)_4$ and $v = (v_{n-1}\, v_{n-2} \ldots v_0)_4$ with $v_{n-1} = 1$
Output: $q = \lfloor u/v \rfloor$ and $r = u \bmod v$

Params: SRT radix $\beta = 2$, look-up table $\mathrm{LUT}'\colon \{0, 1, 2, 3\} \rightarrow \{-1, 0, 1\}$

$u_{m+n} \leftarrow 0$; $q_0 \leftarrow 1$; $\bar{v} \leftarrow \mathrm{CPL}^{(4)}_{m+n}(v \cdot 4^m)$
/* sum and carry digits $R_1 = u - q_0 v$ */
$(s^1_{m+n}, \ldots, s^1_0)_4 \leftarrow (u_{m+n}, \ldots, u_0)_4 \oplus_4 \bar{v}$
$(c^1_{m+n}, \ldots, c^1_0)_4 \leftarrow (\lfloor (u_j + \bar{v}_j)/4 \rfloor)_{j=m+n \text{ to } 0}$
for $i = 1$ ***to*** $2m$ **do**
  /* sum, carry, partial carry prop. of βR_i */
  $\bar{s} \leftarrow s^i \oplus_4 s^i \oplus_4 \mathrm{shift}(c^i) \oplus_4 \mathrm{shift}(c^i)$
  $\bar{c} \leftarrow (\lfloor (s^i_j + s^i_j + \mathrm{shift}(c^i)_j + \mathrm{shift}(c^i)_j)/4 \rfloor)_{j=m+n \text{ to } 0}$
  $(r_1, r_0) \leftarrow (\bar{s}_{m+n}, \bar{s}_{m+n-1})_4 + (-\bar{c}_{m+n-1}, \bar{c}_{m+n-2})_4$
  /* quotient digit selection */
  if $(r_1 = 0)$ **then** $q_i \leftarrow 1$ **else** $q_i \leftarrow \mathrm{LUT}'(r_0)$
  if $(q_i = 1)$ **then** $\bar{v} \leftarrow \mathrm{CPL}^{(4)}_{m+n}(v \cdot 4^m)$
  else if $(q_i = -1)$ **then** $\bar{v} \leftarrow (v_{n-1}, \ldots, v_0, 0^m)_4$
  /* sum and carry digits of $R_i = \beta R_{i-1} - q_i v$ */
  $s^{i+1} = \bar{s} \oplus_4 \bar{c} \oplus_4 \bar{v}$
  $c^{i+1} = (\lfloor (\bar{s}_j + \bar{c}_j + \bar{v}_j)/4 \rfloor)_{j=m+n \text{ to } 0}$
end
/* … now make sure remainder is ≥ 0 and */
/* calculate $q = 4^m \cdot \sum_{i=0}^{2m} q_i \cdot \beta^{-i}$ */

Cost

$(m+1) \cdot (9 + 2\log(n))$ sequential PBS with 2-bit LUT

Cost

$\approx 16m$ sequential PBS with 2-bit LUT

Summary

- SRT method outperforms basic radix-based division
- Number of sequential bootstraps independent of the number of digits
- Based on small look-up tables ⇝ works nicely with TFHE (and its PBS)

[WH86] Ted E. Williams and Mark Horowitz. SRT division diagrams and their usage in designing custom integrated circuits for division. Technical Report CSL-TR-87-326, Stanford University, Computer Systems Laboratory, 1986