

1. INTRODUCTION

1.1 Matrix multiplication

Before explaining why we chose a matrix multiplication, we will show how a matrix A and B are multiplied in equation 1.1

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} * \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 1 * 5 + 2 * 7 & 1 * 6 + 2 * 8 \\ 13 * 5 + 4 * 7 & 3 * 6 + 4 * 8 \end{bmatrix} = \begin{bmatrix} 19 & 22 \\ 93 & 50 \end{bmatrix} \quad (1.1)$$

When a problem is hard, people commonly divide the hard problem in multiple smaller/easier problems. Using matrices is one of those easier ways to deal with data, that's why the most common tools in electrical engineering and computer science are rectangular grids of numbers known as matrices. The numbers in a matrix can represent data, and they can also represent mathematical equations. In many time-sensitive engineering applications, multiplying matrices can give quick but good approximations of much more complicated calculations [?]. So the first reason why a matrix multiplication was chosen is the importance of calculations with matrices.

The second reason consist of the increasing computations when a matrix grows. The relation between the size of the matrix and the calculation is the following:

$$amountOfCalculations = (2 * size - 1) * size^2 \quad (1.2)$$

$$M = \begin{bmatrix} A & B \\ .. & .. \end{bmatrix} \quad (1.3)$$

$$N = \begin{bmatrix} F & .. \\ G & .. \end{bmatrix} \quad (1.4)$$

$$R1 = \begin{bmatrix} A * F + B * G & .. \\ .. & .. \end{bmatrix} \quad (1.5)$$

Deriving equation 1.2. First we count the amount of calculations to calculate the first element of the result matrix R1, when processing the matrix multiplication M * N. The amount of calculations is 3, 1 addition of 2 multiplications. When processing

the same calculation for a matrix of a bigger size, for example size 5, result matrix 1.6 will be calculated. In result matrix R2, 4 additions and 5 multiplications are calculated. Resulting in 4 * 5 calculations in the first element of result matrix R2 with size 5. The number of elements in the matrix is equal to size * size. When you put this all in a formula, we get equation 1.2.

$$R2 = \begin{bmatrix} A * F + B * G + C * H + D * I + E * J & .. & .. & .. & .. \\ & .. & & & \\ & & .. & & \\ & & & .. & \\ & & & & .. \end{bmatrix} \quad (1.6)$$

1.2 Parallel computing

1.3 SoC

1.3.1 FPGA

1.3.2 HPS

1.3.3 Bridges

lwHPS2FPGA

HPS2FPGA

FPGA2HPS

1.3.4 Quartus

Linux RHEL

Windows 10

1.3.5 Bluetooth

Master

slave

Bit error detection

Bit error correction

1.3.6 Web socket

2. IMPLEMENTATION

2.1 SoC

2.1.1 coding

C++ host

OpenCL kernel

C++ to bridge registers

Character error detection

2.1.2 Bluetooth connection

Verilog Bluetooth module

Websocket server