Máster Universitario en Ingeniería de Sistemas Electrónicos

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Sistemas Integrados Digitales

Tarea 3.1

Jose Luis, Rocabado Rocha

ÍNDICE

ÍNDICE 2
Introducción del ejercicio 4
Presentación de resultados 4
Método 0 4
Método 1 5
Método 2 6
Conclusión 7

ÍNDICE DE FIGURAS

Fig. 1. Frecuencia de operación máxima obtenida para el sistema base. 4
Fig. 2. Resultado del entrenamiento y del performace counter para el sistema base. 5
Fig. 3. Frecuencia de operación máxima obtenida para el sistema del método 1. 5
Fig. 4. Resultado del entrenamiento y del performace counter para el método 1. 6
Fig. 5. Frecuencia de operación máxima obtenida para el sistema del método 2. 6
Fig. 6. Resultado del entrenamiento para el método 2. 7

Introducción del ejercicio

Para la resolución de este ejercicio se pretende diseñar y verificar diferentes métodos de aceleración hardware aplicados, en particular, a una red neuronal. Estás aceleraciones se centrarán en la optimización de los cálculos realizados durante los procesos de *back y front propagation.*

Para ello utilizaremos la herramienta de Quartus, TimeQuest Timing Analyzer que nos proporcionará entre otras cosas, la frecuencia máxima a la que podría ejecutarse nuestro diseño teniendo en cuenta unas especificaciones indicadas a la herramienta.

Se debe remarcar que no siempre una frecuencia de reloj mayor conlleva una aceleración temporal. Además, es posible que, aunque el algoritmo se ha optimizado, nos introduzca un sesgo en el modelo de *Machine Learning* produciendo *overffiting.*

Presentación de resultados

A continuación, encontraremos los resultados de frecuencia máxima obtenidos con la herramienta TimeQuest para cada método realizado para la primera parte de la tarea.

Hay que tener en cuenta que la especificación de frecuencia de operación del reloj del sistema introducido en TimeQuest es de 150 MHz.

Método 0

Slow 1100mV 85C Model

	Fmax	Restricted Fmax	Clock Name	Note
1	113.25 MHz	113.25 MHz	altera_reserved_tck	
2	126.02 MHz	126.02 MHz	u0|pll|altera_pll_i|general[2].gpll~PLL_OUTPUT_COUNTER|divclk	
3	126.79 MHz	126.79 MHz	u0|pll|altera_pll_i|general[0].gpll~PLL_OUTPUT_COUNTER|divclk	

Fig. 1. Frecuencia de operación máxima obtenida para el sistema base.

Tras ajustar el emplazamiento y rutado del compilador de Quartus para favorecer un sistema con buena *performance* se comprueba que únicamente se puede alcanzar una frecuencia máxima de ~127MHz para el sistema base, sin ningún método de aceleración implementado. Será interesante también observar el resultado obtenido por el *"performance counter"* (fig 2) para posteriormente realizar la comparativa con los diferentes métodos de aceleración.

```
epoch = 200 RMS Error = 0.038226
pat = 1 actual = 0.950000 neural model = 0.927358
pat = 2 actual = 0.950000 neural model = 0.929626
pat = 3 actual = -0.950000 neural model = -0.893021
pat = 4 actual = -0.950000 neural model = -0.909125
--Performance Counter Report--
Total Time: 40.1414 seconds  (4014137741 clock-cycles)
+---------------+-----+-----------+---------------+-----------+
| Section       |  %  | Time (sec)|  Time (clocks)|Occurrences|
+---------------+-----+-----------+---------------+-----------+
|INICIAR        |0.121|    0.04849|        4849244|          1|
+---------------+-----+-----------+---------------+-----------+
|FASE FORWARD   | 39.1|   15.70559|     1570559030|        804|
+---------------+-----+-----------+---------------+-----------+
|FASE BACKWARD  | 7.15|    2.87210|      287209611|        804|
+---------------+-----+-----------+---------------+-----------+
|FASE UPDATE    |   14|    5.63417|      563417493|        804|
+---------------+-----+-----------+---------------+-----------+
|CALCULO ERROR  | 39.3|   15.79414|     1579413619|        201|
+---------------+-----+-----------+---------------+-----------+
|TEST FINAL     |0.201|    0.08058|        8058137|          1|
+---------------+-----+-----------+---------------+-----------+
```

Fig. 2. Resultado del entrenamiento y del performace counter para el sistema base.

Se observa que las predicciones del modelo no son muy alejadas del valor actual a predecir (se debería valorar el error máximo permitido) y se observa que se ha necesitado alrededor de 40 segundos para llevar a cabo el entrenamiento.

Método 1

Slow 1100mV 85C Model

	Fmax	Restricted Fmax	Clock Name	Note
1	97.06 MHz	97.06 MHz	u0|pll|altera_pll_i|general[2].gpll~PLL_OUTPUT_COUNTER|divclk	
2	112.51 MHz	112.51 MHz	altera_reserved_tck	
3	146.46 MHz	146.46 MHz	u0|pll|altera_pll_i|general[0].gpll~PLL_OUTPUT_COUNTER|divclk	

Fig. 3. Frecuencia de operación máxima obtenida para el sistema del método 1.

En cuanto al método 1, se pretende introducir una instrucción adicional que permitirá a la ALU trabajar con números de coma flotante acelerando los cálculos pues no es necesario realizar conversiones de tipo para realizar las operaciones.

Se observa satisfactoriamente en la figura 3 que se ha conseguido aumentar en 20MHz la frecuencia máxima de operación del nuevo sistema.

Por otro lado, conviene comprobar los resultados obtenidos por el *performance counter* para comprobar si realmente hemos conseguido acelerar el sistema base.

```
epoch = 200 RMS Error = 0.038232
pat = 1 actual = 0.950000 neural model = 0.927353
pat = 2 actual = 0.950000 neural model = 0.929621
pat = 3 actual = -0.950000 neural model = -0.893015
pat = 4 actual = -0.950000 neural model = -0.909120
--Performance Counter Report--
Total Time: 26.8684 seconds  (2686840466 clock-cycles)
+---------------+-----+-----------+---------------+-----------+
| Section       |  %  | Time (sec)|  Time (clocks)|Occurrences|
+---------------+-----+-----------+---------------+-----------+
|INICIAR        |0.138|    0.03703|        3702874|          1|
+---------------+-----+-----------+---------------+-----------+
|FASE FORWARD   | 45.4|   12.18783|     1218782869|        804|
+---------------+-----+-----------+---------------+-----------+
|FASE BACKWARD  | 2.12|    0.57055|       57055394|        804|
+---------------+-----+-----------+---------------+-----------+
|FASE UPDATE    |  6.8|    1.82618|      182618173|        804|
+---------------+-----+-----------+---------------+-----------+
|CALCULO ERROR  | 45.3|   12.17745|     1217745355|        201|
+---------------+-----+-----------+---------------+-----------+
|TEST FINAL     |0.234|    0.06296|        6295593|          1|
+---------------+-----+-----------+---------------+-----------+
```

Fig. 4. Resultado del entrenamiento y del performace counter para el método 1.

Si comparamos con el modelo base se puede comprobar que el error RMS al final del entrenamiento (*epoch* 200) se encuentra en el mismo orden al obtenido en el sistema inicial. Sin embargo, es realmente notable que el tiempo necesario para el entrenamiento se ha reducido a ~27 segundos.

Método 2

Slow 1100mV 85C Model

	Fmax	Restricted Fmax	Clock Name	Note
1	123.61 MHz	123.61 MHz	u0|pll|altera_pll_i|general[0].gpll~PLL_OUTPUT_COUNTER|divclk	
2	123.82 MHz	123.82 MHz	altera_reserved_tck	
3	131.03 MHz	131.03 MHz	u0|pll|altera_pll_i|general[2].gpll~PLL_OUTPUT_COUNTER|divclk	

Fig. 5. Frecuencia de operación máxima obtenida para el sistema del método 2.

En el tercer método se quiere realizar la implementación de un periférico con comportamiento Avalon MM Slave que realizará el cálculo de la tangente hiperbólica de los datos almacenados en el registro 3 y devolverá el resultado en el registro 0.

Esta IP se añadirá al proyecto de QSys que contiene la instrucción de operaciones en coma flotante del método 1 para comprobar si se consigue acelerar aún más el sistema base.

A primera vista, se observa que la frecuencia máxima de reloj del sistema es aproximadamente 3MHz inferior al sistema de referencia. Esto se podría a llegar a explicar al hardware adicional utilizado en el conjunto de los dos métodos.

Por otro lado, si revisamos el resultado del *performance counter* podemos observar como a pesar de haber obtenido una frecuencia de operación máxima inferior a la del punto de partida, el entrenamiento de la red neuronal ha sido completada en alrededor de 4 segundos.

```
epoch = 200 RMS Error = 1.374296
pat = 1 actual = 0.950000 neural model = 0.993068
pat = 2 actual = 0.950000 neural model = 0.993068
pat = 3 actual = -0.950000 neural model = -1.000000
pat = 4 actual = -0.950000 neural model = -1.000000
```

Fig. 6. Resultado del entrenamiento para el método 2.

Sin embargo, se comprueba que el error obtenido (fig. 6) es realmente incoherente con lo esperado. Se puede apreciar una convergencia o sesgo del error lo que nos permite deducir que la red neuronal no está "aprendiendo".

Se decide continuar con el desarrollo de la práctica, pero sería conveniente revisar nuevamente la creación de la IP y poder encontrar a que se debe el fallo dado que este mismo comportamiento, se produce sin introducir el método 1 el sistema de QSys.

Conclusión

Tras realizar la primera parte de la Tarea, se ha podido comprobar el desarrollo, implementación y funcionalidad de los diferentes métodos de aceleración hardware propuestos por el docente en el guion de prácticas.

Durante el desarrollo del ejercicio, se encuentra un problema de convergencia durante el entrenamiento del modelo predictivo del método 2 que podría deberse a algún problema con el IP generado, aunque no se ha realizado el *debug* apropiado.

Para finalizar, se observa como realmente se obtiene una reducción en el tiempo de ejecución del entrenamiento de la red neuronal para ambos métodos y se comprueba que un aumento en la frecuencia máxima de operación del sistema no conlleva necesariamente una mejora en el rendimiento del sistema.