

Calculo do deslocamento de corpos usando Quatérnions

Allan Ribeiro
2037807

Lucas Perin Silva Leyser
1860275

21 de Dezembro de 2022

1 Introdução

A intenção era fazer o uso de FPGA para cálculo do deslocamento de uma partícula no tempo fazendo uso da abstração de quaternions, que evita problemas algébricos, diferentemente da abordagem dos ângulos de Euler.

Fez-se o uso dos dados de giroscópio e acelerômetros de um smartphone, descritos em um csv, fazendo-se a amostragem em tempo da leitura da aceleração das variações, em eixo x, y e z. Os dados são então filtrados de string para float, processados por um socket, escrito em python, que por sua vez realizaria a conexão com a placa.

Na placa Cyclone II, foram escritos arquivos em VHDL que fazem a abstração do uso de registradores para guardar um quaternion, sendo descrito como o valor normalizado de dimensão única. Foram usados 2 registradores, um como buffer, e outro que acumula as operações subseqüentes, devolvendo o resultante da multiplicação de todos os quaternions recebidos.

Sem sucesso, o presente relatório dá visão para problemas que futuros desenvolvedores terão, se usada a mesma abordagem. Dentre os inúmeros, listam-se os empecilhos principais: estabelecimento de um socket de rede funcional e uma operação efetiva de multiplicação. Os procedimentos são detalhados abaixo.

Alguns pontos evoluídos no trabalho, importante que leitores saiam é que a estrutura para a multiplicação dos quaternions em si está pronta, sendo necessária a correção das operações. Um test-bench também foi elaborado para uso com os sources disponíveis, e está funcional. O parsing dos dados também foi realizado, e as respectivas funções estão prontas, além de um envio efetivo realizado.

2 Desenvolvimento do projeto

2.1 Conexão de rede

A conexão TCP/IP com a placa Altera DE2 Cyclone II foi projetada com base no projeto "Wave Gen" realizado pelo aluno Dennis Guilherme de Macedo Bragagnolo, no segundo semestre de 2020. A conexão não pôde ser estabelecida, pois os arquivos fonte disponíveis no projeto não eram os mesmo que geravam o código que roda na placa e efetivamente faz a conexão, impossibilitando a alteração do código para que o envio fosse do quaterno a ser calculado e, mesmo replicando o código utilizado em outro projeto, ainda não foi possível realizar a conexão com a placa para o envio dos pacotes com os quaternos. A origem do problema é incerta.

2.2 Multiplicação

Para o cálculo da multiplicação dos quaternos, foram utilizados sinais do tipo "sfixed" que utiliza um ponto fixo determinando qual a parte inteira e qual a parte decimal do valor recebido, que é então armazenado em um registrador de 32 bits. Isso resulta em uma precisão de 2^{-15} nas operações.

A operação, infelizmente, apresenta um problema: em teste de mesa, a tentativa de multiplicar 0.5 por 0.5 resulta no deslocamento errôneo dos bits.

```
0.5 * 0.5 = 0b0010 * 0b0010 = 0b0001 = 0.25 (caso ideal)
```

```
0.5 * 0.5 = 0b0010 * 0b0010 = 0b000001 = 0.0625 (caso real)
```

Na tentativa de resolver o problema, não foram feitos testes de outras operações, uma vez que os esforços passaram a se concentrar no envio dos dados à placa.

Devido à falta de suporte a esse tipo de sinal, foi preciso utilizar portas do tipo `std_logic_vector` para o recebimento dos valores e a própria FPGA faz a cópia desse valor para um sinal do tipo "sfixed". Partimos do pressuposto que, como os valores são sinalizados e entre 0 e 1, sempre teríamos dois bits iniciais, o primeiro para o sinal e o segundo para a parte inteira do valor, e o restante é a parte decimal do valor, ou seja, utilizando um sinal com o ponto fixo entre o segundo e terceiro MSB do `std_logic_vector` a cópia pode ser feita diretamente, bit a bit.

Foi-se feito a tentativa de abstrair outra forma de cálculo: receber os dados em binário, e fazer a operação bit a bit. Outro problema apareceu aqui através da Os principais problemas residem na abstração usada para fazer a multiplicação entre os pontos. No caso de um dos números serem negativos, a multiplicação em complemento de 2 não foi implementada com sucesso. Resolvido este problema, a multiplicação deve funcionar efetivamente.

Outra tentativa consistiu em trabalhar com inteiros, fazendo-se shift dos bits, multiplicando-os, e retornando à posição original. Ocorreram os mesmos problemas da abordagem por complemento de 2.

3 Conclusão e recomendações gerais

O projeto teve avanços, mas não está perto de estar completo. Novamente, a conexão de rede funciona, desde que o socket na placa esteja funcional. A multiplicação funciona, mas existem erros na operação.

Se possível, troque a versão da placa para evitar problemas de licença, problemas de compilação e geração de componentes/circuitos (Qsys).

A forma como a multiplicação foi feita é dependente da biblioteca `ieee_proposed`, representada pelos arquivos `fixed_float_types_c.vhdl`, `float_pkg_c.vhdl` e `fixed_pkg_c.vhdl`. Os três tem problemas com versões de `vhdl`, cujo dialeto deve ser 2008.

As bibliotecas deve ser inserida como arquivos do projeto no Quartus (caso simulação em RTL), no Qsys (para geração dos componentes), e compiladas internamente do ModelSim (para simulação em RTL).

A multiplicação em binário na FPGA é considerada, por muitas referências, como uma área obscura para se implementar. Sem sucesso, algumas recomendações são: implementar uma lógica de complemento de 2 funcional, uso de outro tipo preferível (real não é sintetizável).