Vector Programming

Vipin Vasu

College Of Engineering,Trivandrum

Overview

Design Principles

- Vector Processors have vector registers.
- The length of the vector register is called **Vector Length**
- Consider addition of two arrays A(1:N) = B(1:N)+C(1:N).On a cache based microprocessor this would result in a loop over the elements of the arrays.
- For each index,two loads, one addition and one store operation would have to be executed.
- In a vector CPU a single instruction for the whole array can be used if the length of the array is less than the vector length of the processor.Refer fig 1.

Design Principles

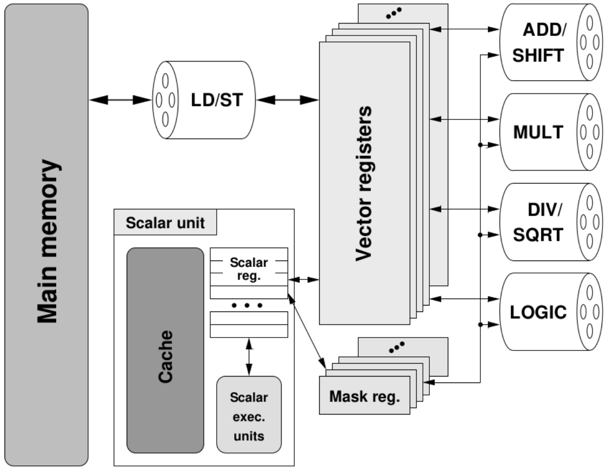
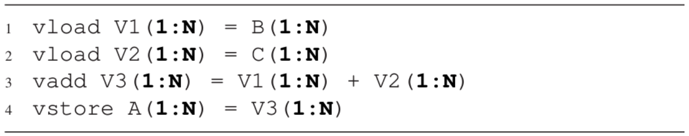


Figure: Vector Processor Block Diagram

Design Principles

- If the array length is larger than the vector length,the loop must be *stripmined*,i.e original array must be traversed as chunks of vector lenght.Refer fig 2.
- An operation like vector addition does not have to wait until its argument vector registers are completely filled but can commence after some initial arguments are available.This feature is called chaining and forms an essential requirement for different pipes (like MULT and ADD) to operate concurrently.
- More importantly, the speed of the load/store pipes is matched to the CPU's core frequency,so feeding the arithmetic pipes with data is much less of a problem.

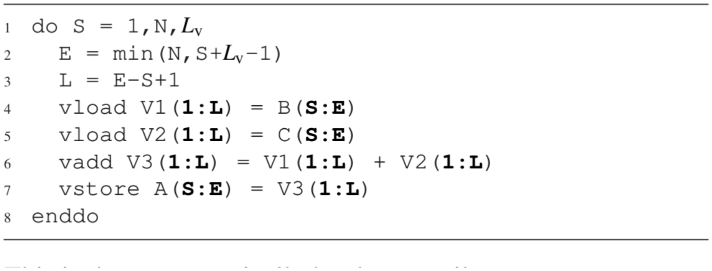
Design Principles



```
vload V1(1:N) = B(1:N)
vload V2(1:N) = C(1:N)
vadd V3(1:N) = V1(1:N) + V2(1:N)
vstore A(1:N) = V3(1:N)
```

Figure: Vector Programming

Design Principles



```
do S = 1,N,$L_v$
  E = min(N,S+$L_v$-1)
  L = E-S+1
  vload V1(1:L) = B(S:E)
  vload V2(1:L) = C(S:E)
  vadd V3(1:L) = V1(1:L) + V2(1:L)
  vstore A(S:E) = V3(1:L)
enddo
```

Figure: Strip Mining

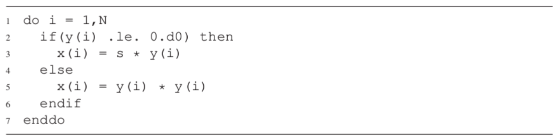
Maximum Performance Estimates

- The peak performance of a vector processor is given by the number of tracks for the ADD and MULT pipelines and its clock frequency. For example, a vector CPU running at 2 GHz and featuring four-track pipes has a peak performance of
  2 (ADD+MULT) x 4 (tracks) x 2 (GHz) = 16 GFlops/sec
- As for memory bandwidth, a single four-track LD/ST pipeline can deliver
  4 (tracks) 2 (GHz) 8 (bytes) = 64 GBytes/sec

Mask Registers

- *Mask Register*(essentially boolean registers with vector length) are provided that allow selective execution of loop iterations.
- A vector of boolean values is first generated from the branch conditional using the logic pipeline. This vector is then used to choose results from the if or else branch.
- Of course, both branches are executed for all loop indices which may be waste of resources if expensive operations are involved.However, the benefit of vectorization often outweighs this shortcoming.

Mask Registers



```
do i = 1,N
  if(y(i) .le. 0.d0) then
    x(i) = s * y(i)
  else
    x(i) = y(i) * y(i)
  endif
enddo
```

Figure: Need for mask register

Mask Registers

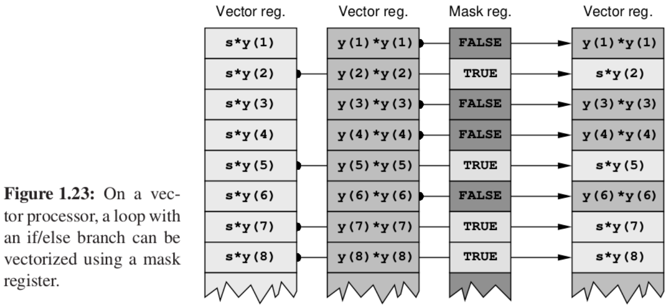


Figure 1.23: On a vector processor, a loop with an if/else branch can be vectorized using a mask register.

Figure: Mask Register

Gather & Scatter

- For a single branch (no else part), especially in cases where expensive operations like divide and square roots are involved, the gather/scatter method is an efficient way to vectorization.
- All needed elements of the required argument vectors are first collected into vector registers (gather), then the vector operation is executed on them and finally the results are stored back (scatter)

Gather & Scatter

```
do i = 1,N
  if(y(i) .ge. 0.d0) then
    x(i) = sqrt(y(i))
  endif
enddo
```

Figure: Program with if but no else

Gather & Scatter

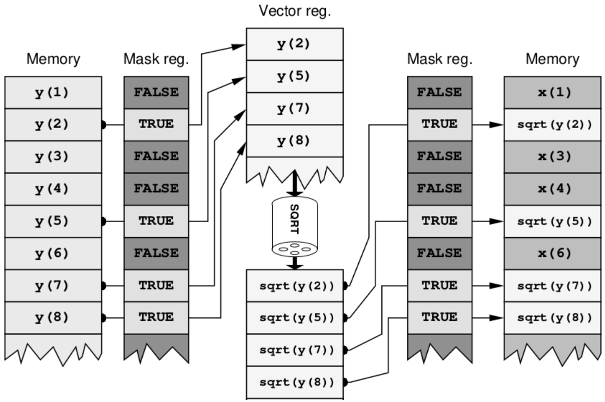


Figure: Gather Scatter

The End