Introduction to DSP

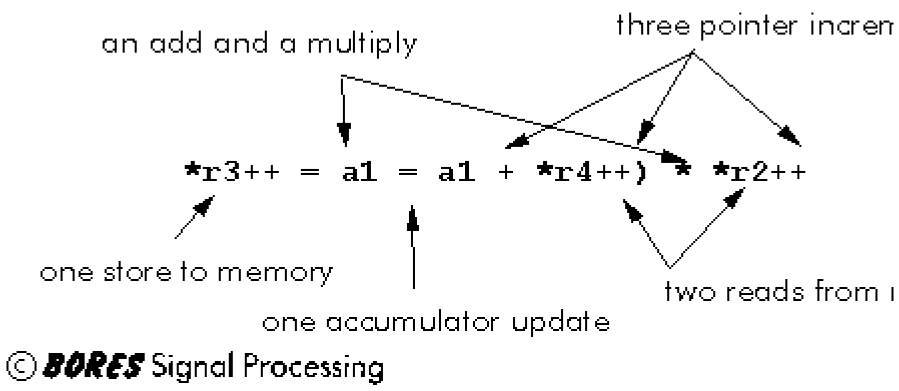
Programming a DSP processor: MIPS, MOPS an

The development of efficient assembly language code shows how e processor can be: each assembler instruction is performing several But it also shows how difficult it can be to program such a specialis efficiently.

```
temp = *c_ptr++) * *x_ptr--);
a1  =  *r3++  *  *r4--
for (k = 1; k < N-1; k++)
do 0,r1
temp  = temp +    *c_ptr++ * *x_ptr--)
a1  = a1 +   *r3++ *  *r4--
*y_ptr++ = temp
 *r2++  = a1
```

Bear in mind that we use DSP processors to do specialised jobs fas then it may be permissible to throw away processor power by ineff that case we would perhaps be better advised to choose an easier in the first place. A sensible reason to use a DSP processor is to pe lowest cost, or at highest speed. In either case, wasting processor for more hardware which makes a more expensive system which le expensive final product which, in a sane world, would lead to loss c competitive product that was better designed.

One example shows how essential it is to make sure a DSP process efficiently:



- two arithmetic operations (an add and a multiply)
- three memory accesses (two reads and a write)
- one floating point register update

- three address pointer increments

All of these operations can be done in one instruction. This is how 1
made fast. But if we don't use any of these operations, we are thrc
potential of the processor and may be slowing it down drastically. (
instruction can be translated into MIPS or Mflops.

The processor runs with an 80 MHz clock. But, to achieve four mem
instruction it uses a modified von Neuman memory architecture wh
divide the system clock by four, resulting in an instruction rate of 2
manic marketing mode, we can have fun working out ever higher M
as follows:

80 MHz clock

20 MIPS = 20 MOPS

but 2 floating point operators per cycle = 40 MOPS

and four memory accesses per instruction = 80 MOPS

plus three pointer increments per instruction = 60 MOPS

plus one floating point register update = 20 MOPS

making a grand total MOPS rating of 200 MOPS

Which exercise serves to illustrate three things:

- MIPS, MOPS and Mflops are misleading measures of DSP pow
- marketing men can squeeze astonishing figures out of nothir

Of course, we omitted to include in the MOPS rating (as some man
possibility of DMA on serial port and parallel port, and all those ass
DMA address pointers, and if we had multiple comm ports, each wi
really wild...

Apart from a cheap laugh at the expense of marketing, there is a v
be drawn from this exercise. Suppose we only did adds with this pı
Mflops rating falls from a respectable 40 Mflops to a pitiful 20 Mflop
the memory accesses, or the pointer increments, then we can cut 1
200 MOPS to 20 MOPS.

It is very easy indeed to write very inefficient DSP code. Luckily it i
with a little care, to write very efficient DSP code.

BORES Signal processing

BORES Signal Processing: Fordwater Pond Road Woking, Surrey GU22 0JZ United Kingdom
telephone:01483 740138 **fax:** 01483 740136 **email:** bores@bores.com