

Re: Faster way to write in a file

Source: http://coding.derkeiler.com/Archive/C_CPP/comp.lang.c/2008-02/msg01427.html

- *From:* Ben Bacarisse <ben.usenet@xxxxxxxxx>
 - *Date:* Wed, 13 Feb 2008 22:08:22 +0000
-

LilacSkin <lpaulo07@xxxxxxx> writes:

```
#include <stdio.h>

void convert(long long v,char *s,int sz,int dp)
{ char *p = s + sz;
  long long x = 1LL;
  int sign = v < 0.0;

  if (sign) v = -v;
  while (dp-->0) x *= 10;
  x = x * v;

  *p-- = '\0';
  do
  { *p-- = '0' + (x % 10);
    } while ((p >= s) && (x /= 10));

  while (p >= s) *p-- = '0';

  if (sign) *s = '-';

  }

int main(void)
{ long long test = -123456789012345;
  char buffer[64];

  convert(test,buffer,25,5);
  printf("%s\n",buffer);
  // then fwrite(buffer...)

  return 0;

  }
```

You may want to re-consider. Numbers are usually converted to a

Re: Faster way to write in a file

decimal string representation so the we (humans) can read them. If your program is producing such a volume of data that it is hard to get converted output fast enough, will the resulting data ever be read by a person? I doubt it. You'd need a program to scan just one second's worth.

It might pay to store the data as native binary numbers. That is probably what was being suggested when someone said use "fwrite".

Of course, if the output *has* to be processed by an existing program that needs decimal input, then the conversion has to happen some time, but it could be done later, at leisure, so to speak. It might even be possible to process into decimal only the portion you are interested in. Maybe the binary data could be indexed for rapid access to specific parts. Without knowing the ultimate fate of this data, it is hard to be more specific, but I suggest you look beyond your current headache of not being able to use `fprintf` because it seems too slow.

--

Ben.

.