

---

Subject: Re: Assembly - Convert 16 bit integer to ascii help please  
Posted by [Anonymous](#) on Mon, 03 Jun 2013 09:50:44 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Originally posted by: Johann Klasek - swap last two domainparts

Groepaz <groepaz@gmx.net> wrote:

```
> rusure wrote:
>> On Sunday, January 13, 2013 12:12:30 AM UTC-7, Groepaz wrote:
>>> rusure wrote:
>>>> On Saturday, January 12, 2013 8:56:32 AM UTC-7, JB wrote:
>>>> > I am looking for a routine to convert a 16 bit integer into ascii
>>>> > (text)
>>>> >
>>>> >
>>>> > ex. convert $C000 (49152) to the characters "49152" stored in a
>>>> > memory address.
>>>> > Ideally I would like to have a routine that you can pass an option to
>>>> > either print leading 0's or strip them. But first things first I
>>>> > guess.
>>>> > :)
>>>>
>>>> A C64 systems routine at $bdcd prints the decimal equivalent of a two
>>>> byte
>>>> integer. Here is how to use that code:
>>>>
>>>> its not a very good idea to use direct calls into basic rom though =P
>>>>
>>>> Jim Butterfield agrees with you, but I can't figure out why.
>>>>
>>>> because you should generally see rom code as a "black box" and only ever use
>>>> the official entry points (the kernal jumtable). that way your program will
>>>> always work and doesnt need special case handling for when a modified basic
>>>> rom and/or kernal is present.
```

Yes, that's well known but portability has a complete other value in specific system environments like a C64. As you probably know the BASIC routines are not offered via a jump table or call interface, but it would be far to idealistic and wasteful to omit the valuable code in the ROM and reprogram or reload everything again and again just to hold the blackbox paradigm. Yes, you are right, but practical needs and common habit disobeyes blackboxing. In a C64 environment, even for several different hardware generations most - not to say all - well known and useful BASIC entries did not change through the time.

It's nice to mention the blackbox paradigm, but repeatedly replying on every post with a BASIC ROM routine reference becomes tiresome. It's not the todays programmer who needs this advice, but Microsoft and CBM had

the obligation to provide such a interface. Alas, the missed it and the time passes away ...

JeeK

---