An assortment of my objections and comments and questions as I study calculus over at Khan Academy, plus a Reevaluation of the Input and Output of a Load-Bearing Konzen Simulation.

I'm skipping through a lot of Khan's videos and examples since they are so incredibly boring. My goal is to be able to understand the SDT and the DDT (IDT) functions which are available in Micro-Cap.

VINYASI
JUL 23, 2026

> I'm putting this article under my subheading of "Is Free Energy for Real?", rather than putting it under the other subheading I possess here at Substack, of "non-Euclidean Mathematics", because I'm apply my acquisition of calculus skill-building to the use which electronic simulators can put to their analysis toolkit. "non-Euclidean Mathematics" is a subheading devoted to my three-year study of the Infinite Range of Golden Ratios and its related topic of Silver Ratios conducted between 1994 and 1997.

All of these statements/complaints/etc, plus others from a previous date, can be found on my "Dashboard" page at Khan Academy.

> This is not a question. This is an objection.
> Why restrict ourselves with a straitjacket?
> Your stipulations require a rewrite.
> It is the magnitude which remains the same despite the discontinuities of polarity.
> Hence, we define the limit (in all of your examples) as being the absolute value of infinity.
> And then, we stipulate polarity according to context.
> It's that simple.

Introducing infinite limits (video) | Khan Academy

> Vertical asymptotes signify an absolute magnitude of infinity as its limit with an undefined polarity of sign. This constitutes my rewording of your statement that "Vertical asymptotes signify undefined limits, ...".
>
> It's not fair that we abuse infinity by ignoring it under certain inconvenient circumstances since it is unfair of us to use it for all of our inputs yet exclude it from some of those outputs.

Exploring infinite limits and asymptotes (video) | Khan Academy

---

Let's get on with it already ...

Integration tutorial using SPICE's SDT - Search

Gotcha — you want to **verify an** sdt() **integration result** by using ddt() (the derivative function) and some filtering to see if the math holds up in Micro-Cap SPICE.

Here's the idea:

- sdt() integrates a signal over time.
- ddt() differentiates a signal over time — in Micro-Cap (see last screenshot, below); idt() in other instances of SPICE electronic simulation.
- If you integrate a signal and then differentiate it, you should get back the original signal (within numerical limits).
- Filtering helps remove high-frequency noise or numerical jitter that can appear in derivatives.

🔹 Step-by-Step Verification Method

1. **Integrate the signal with** sdt()

Spice

1. Bint int 0 V = sdt(V(in))
2. **Differentiate the integrated signal with** ddt()

Spice

1. Bdiff diff 0 V = ddt(V(int))
   This should give you something very close to V(in).
2. **Compare original vs. recovered**
   - Plot V(in) and V(diff) in the waveform viewer.
   - Small differences are normal due to numerical precision.
3. **Apply filtering to smooth noise**

   Differentiation amplifies high-frequency noise, so you can pass the derivative through a low-pass filter:

Spice

1. Rf diff filt 1k
   Cf filt 0 10u

- This RC filter smooths the derivative output.
- Choose `Rf` and `Cf` so the cutoff frequency is well above your signal's main frequency but low enough to suppress noise:

$$f_c = \frac{1}{2\pi R_f C_f}$$

🔹 Example Circuit for Testing

Spice

```
V1 in 0 SIN(0 1 1k)
Bint int 0 V = sdt(V(in))
Bdiff diff 0 V = ddt(V(int))
Rf diff filt 1k
Cf filt 0 10u
.tran 0 5m
```

- `V(in)` → original signal
- `V(int)` → integrated signal
- `V(diff)` → derivative of integrated signal (raw)
- `V(filt)` → filtered derivative (cleaned up)

💡 Practical Tip

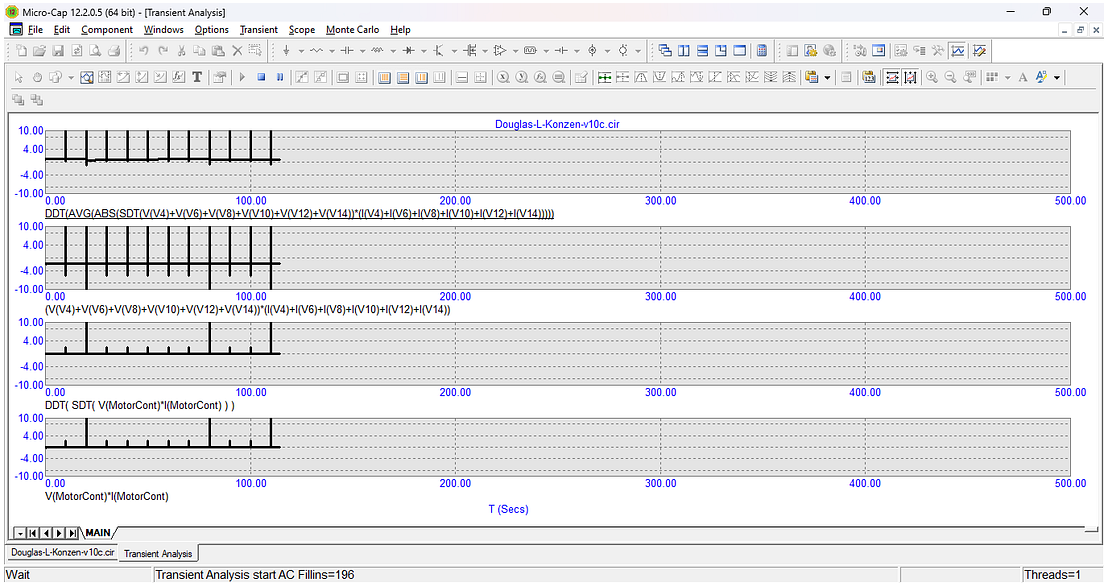
If `V(diff)` matches `V(in)` closely after filtering, your `sdt()` integration is behaving correctly. If there's a large mismatch, check:

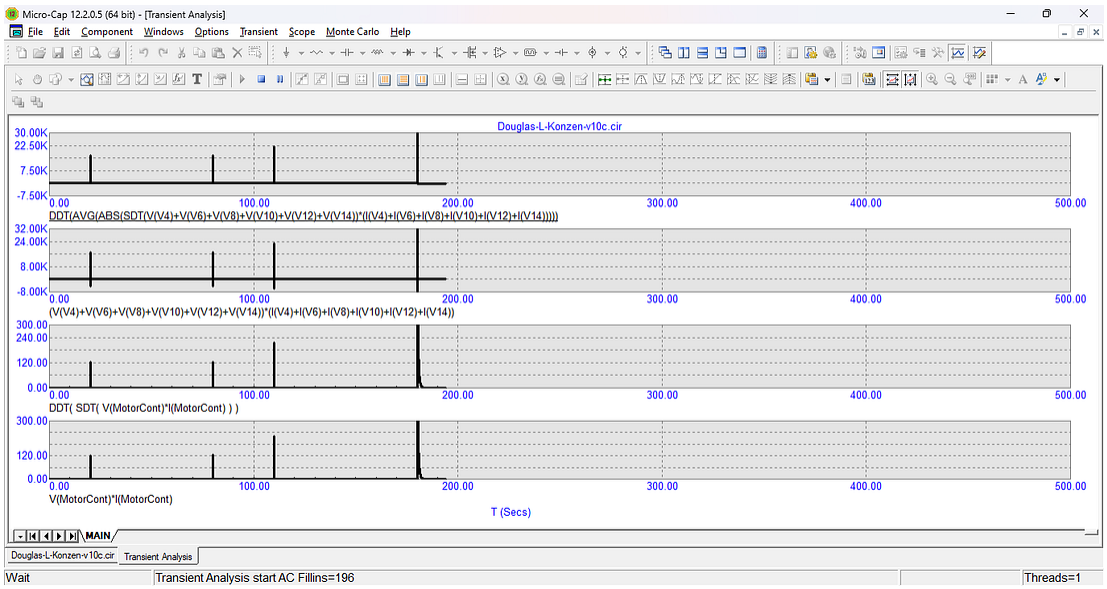
- Time step size (`.tran ... uic`) — smaller steps improve accuracy.
- Whether your signal has a DC offset (which can cause drift in integration).

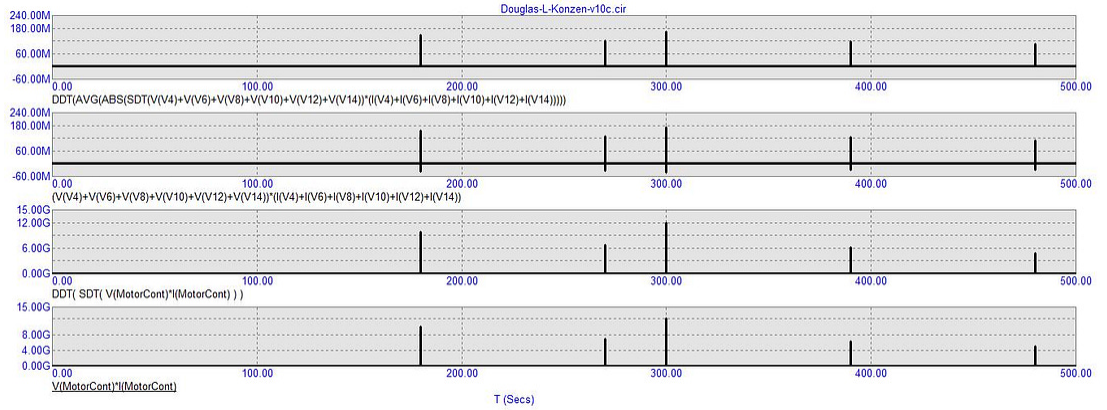
- If you need to reset integration periodically to avoid error buildup.

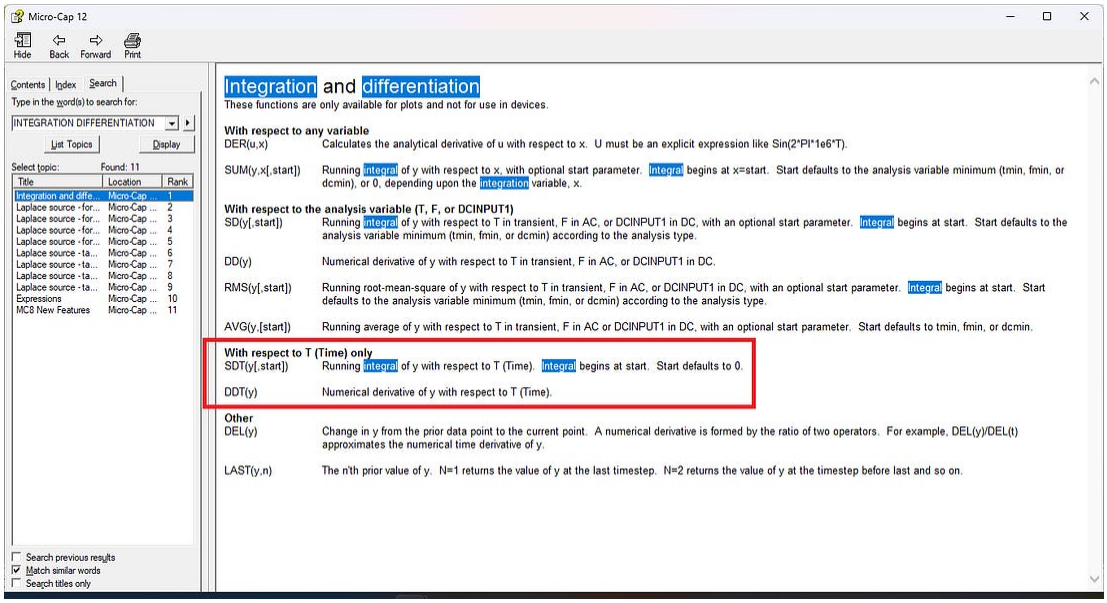
If you want, I can give you a **combined SPICE testbench** that integrates, differentiates, and overlays all signals for a visual accuracy check. That way you can instantly see if your integration is solid. Would you like me to prepare that?

Here are the screenshots confirming the accuracy of my integral calculations:

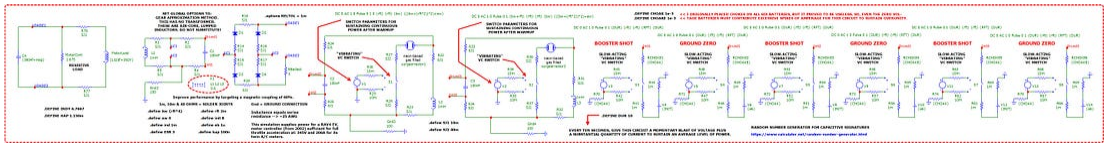



Micro-Cap 12.2.0.5 (64 bit) - [Transient Analysis]
File Edit Component Windows Options Transient Scope Monte Carlo Help
Douglas-L-Konzen-v10c.cir
30.00K
22.50K
7.50K
-7.50K
0.00
100.00
200.00
300.00
400.00
500.00
DDT(AVG(ABS(SDT(V(V4)+V(V6)+V(V8)+V(V10)+V(V12)+V(V14))*(I(V4)+I(V6)+I(V8)+I(V10)+I(V12)+I(V14)))))
32.00K
24.00K
8.00K
-8.00K
(V(V4)+V(V6)+V(V8)+V(V10)+V(V12)+V(V14))*(I(V4)+I(V6)+I(V8)+I(V10)+I(V12)+I(V14))
300.00
240.00
120.00
0.00
DDT(SDT(V(MotorCont)*I(MotorCont)))
300.00
120.00
0.00
V(MotorCont)*I(MotorCont)
T (Secs)
MAIN
Douglas-L-Konzen-v10c.cir
Transient Analysis
Wait
Transient Analysis start AC Fillins=196
Threads=1


Douglas-L-Konzen-v10c.cir
240.00M
180.00M
60.00M
-60.00M
0.00
100.00
200.00
300.00
400.00
500.00
DDT(AVG(ABS(SDT(V(V4)+V(V6)+V(V8)+V(V10)+V(V12)+V(V14))*(I(V4)+I(V6)+I(V8)+I(V10)+I(V12)+I(V14)))))
240.00M
180.00M
60.00M
-60.00M
(V(V4)+V(V6)+V(V8)+V(V10)+V(V12)+V(V14))*(I(V4)+I(V6)+I(V8)+I(V10)+I(V12)+I(V14))
15.00G
12.00G
6.00G
0.00G
DDT(SDT(V(MotorCont)*I(MotorCont)))
15.00G
8.00G
4.00G
0.00G
V(MotorCont)*I(MotorCont)
T (Secs)

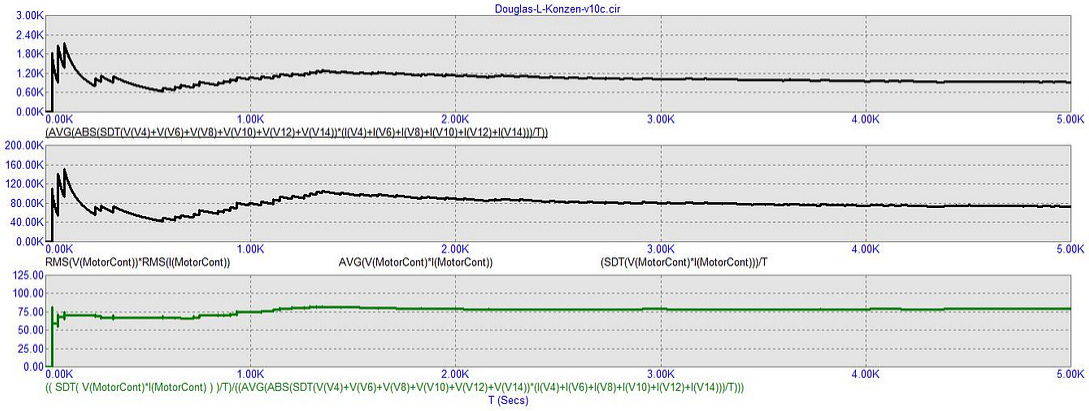

Micro-Cap 12
Hide
Back
Forward
Print
Contents
Index
Search
Type in the word(s) to search for:
INTEGRATION DIFFERENTIATION
List Topics
Display
Select topic:
Found: 11
Title
Location
Rank
Integration and diffe... Micro-Cap ... 1
Laplace source - for... Micro-Cap ... 2
Laplace source - for... Micro-Cap ... 3
Laplace source - for... Micro-Cap ... 4
Laplace source - for... Micro-Cap ... 5
Laplace source - ta... Micro-Cap ... 6
Laplace source - ta... Micro-Cap ... 7
Laplace source - ta... Micro-Cap ... 8
Laplace source - ta... Micro-Cap ... 9
Expressions Micro-Cap ... 10
MC8 New Features Micro-Cap ... 11
Search previous results
Match similar words
Search titles only
Integration and differentiation
These functions are only available for plots and not for use in devices.
With respect to any variable
DER(u,x) Calculates the analytical derivative of u with respect to x. U must be an explicit expression like Sin(2*PI*1e6*T).
SUM(y,x[,start]) Running Integral of y with respect to x, with optional start parameter. Integral begins at x=start. Start defaults to the analysis variable minimum (tmin, fmin, or dcmin), or 0, depending upon the integration variable, x.
With respect to the analysis variable (T, F, or DCINPUT1)
SD(y[,start]) Running Integral of y with respect to T in transient, F in AC, or DCINPUT1 in DC, with an optional start parameter. Integral begins at start. Start defaults to the analysis variable minimum (tmin, fmin, or dcmin) according to the analysis type.
DD(y) Numerical derivative of y with respect to T in transient, F in AC, or DCINPUT1 in DC.
RMS(y[,start]) Running root-mean-square of y with respect to T in transient, F in AC, or DCINPUT1 in DC, with an optional start parameter. Integral begins at start. Start defaults to the analysis variable minimum (tmin, fmin, or dcmin) according to the analysis type.
AVG(y,[start]) Running average of y with respect to T in transient, F in AC or DCINPUT1 in DC, with an optional start parameter. Start defaults to tmin, fmin, or dcmin.
With respect to T (Time) only
SDT(y[,start]) Running Integral of y with respect to T (Time). Integral begins at start. Start defaults to 0.
DDT(y) Numerical derivative of y with respect to T (Time).
Other
DEL(y) Change in y from the prior data point to the current point. A numerical derivative is formed by the ratio of two operators. For example, DEL(y)/DEL(t) approximates the numerical time derivative of y.
LAST(y,n) The n'th prior value of y. N=1 returns the value of y at the last timestep. N=2 returns the value of y at the timestep before last and so on.

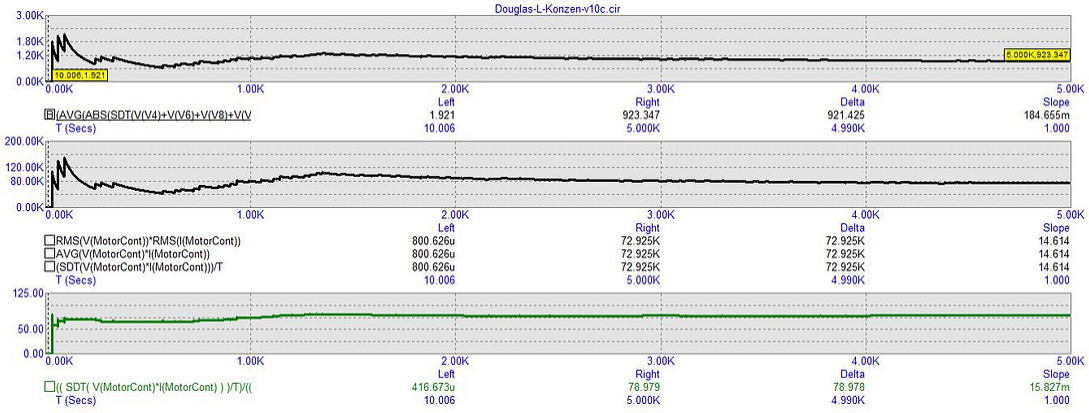
Schematic for this test:



Download these files.

I had to re-evaluate the input and output tracings since I made a mistake in where I placed the parameter of time. This had the consequence of increasing the integration of the input power over time which had the additional consequence of reducing the COP to **79 to 1** for a 5ks runtime.



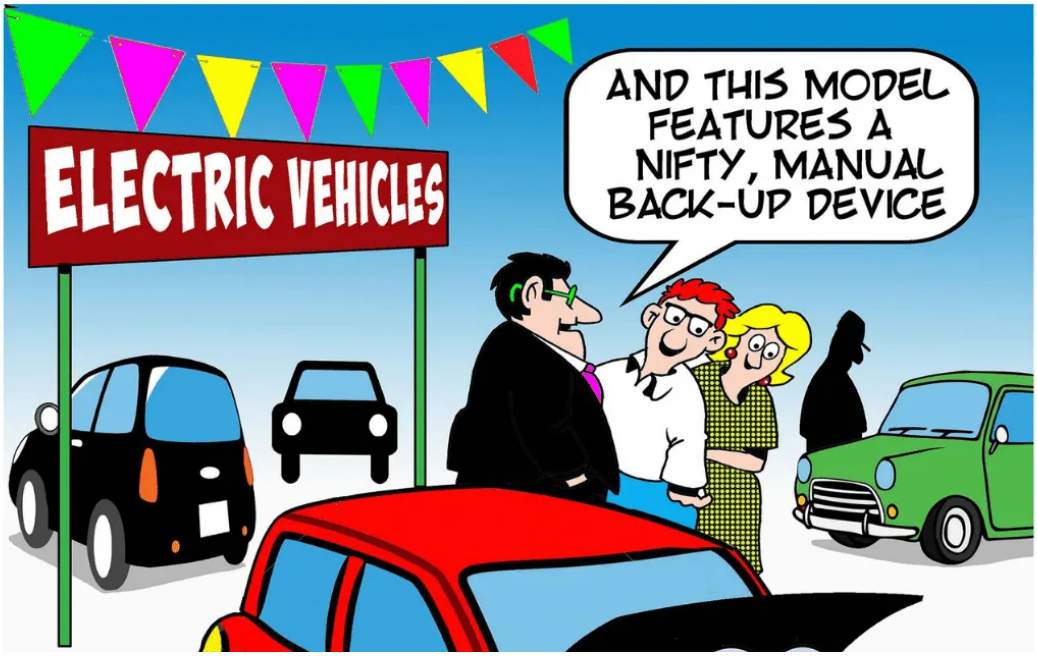


Download these files.

This post is an extension of this post:

Load-Bearing Konzen

VINYASI • JUL 21



The reason why my overunity simulations (over the past several days) of Douglas L. Konzen's Hunt for Free Energy have been so successful is due to a lack of any substantial load bearing down upon the...

Read full story