

# TIME:CODE

RUNNING NUMBERS GENERATOR ADD-ON FOR BLENDER



2023-08-14

ver: 1.1.2

by Saro Sahihi, SoundBits

**INTRODUCTION**..... 2  
 About this Project & me..... 2

**GENERAL**..... 3  
 Installation..... 3  
 Adaptivity..... 3  
 User Input..... 3  
 Custom Start- and Stop-Positions..... 4  
 Base-Clock Multiplier..... 4  
 Tooltips..... 4  
 Default- and Min-/Max- Values..... 4

**VISUAL STYLES**..... 5  
 Global Parameters [prefix: vis]..... 5  
 7 Material Slots..... 5  
 Just basic colors to maintain overview..... 5  
 Custom Object #1 to All..... 5  
 Transform..... 5  
 Select Visual Style..... 5  
 Don't Show Category Names..... 5  
 Hide the small introductory headlines that appear with each source..... 5  
 Style Specific Parameters (prefix: vis [#])..... 5  
**The 3 Visual Styles**..... 6  
 Cube Arrays [vis 1]..... 6  
 Led-Style Digits [vis 2]..... 6  
 Monotype Font [vis 3]..... 6

**SOURCE GENERATORS**..... 7  
 SMPTE Timecode [src 1]..... 7  
 Manual Timecode- / Clock-Mode..... 7  
 Adaptive Frame Counter [src 2]..... 8  
 Bars & Beats Counter [src 3]..... 8  
 EXTRA: Tempo-, Rhythm- and Timing Indicators..... 9  
 16-Step Sequencer..... 9  
 Musical Timeline Representation..... 9  
**Digital & Analog Chronometers [src 4]**..... 9  
 Manual Timecode/Clock Mode..... 10  
 EXTRA: Analog Clock Face..... 10  
 Countdown [src 5]..... 10  
 Extended Time/Timing Statistics..... 11

**PARAMETER OVERVIEW**..... 12

## INTRODUCTION

**About this Project & me**

Hi! I'm Saro Sahihi from Ludwigsburg, Germany. My field of work and passion, can best be described as everything multimedia.

My main focus is - or better was - on audio-related projects, like sound design and mixing for any kind of moving pictures, music production as well as (field-)recording, editing and mastering sound effect libraries, to distribute them through a well established network of independent platforms and partners. All this demands to cast one's net wide across many different fields.

This kept feeding the part of me that has always been keenly curious and intrigued by almost every aspect of 2D/3D design and animation.

About three years ago I decided to dive deeper into the waters of learning After Effects, Premiere and Blender as well as all the accompanying theoretical stuff, to one day reach the vibrant utopia of full pipeline media production.



The initial idea for this project came up, when I was working on the official advertorial video series “Nuendo | Exclusive Features for...” for Steinberg’s post-production DAW Nuendo.

For the intro animation, I had created a hyper realistic version of Nuendos transport panel, with the typical timecode and Bars & Beats displays. But instead of taking hi-res screen captures of the actual displays, to use them as textures, I had chosen to jump in at the deep end of Blenders Geometry Nodes.

And now, just a few months later, I'm happy and proud  
to present to you my first Blender Add-On:

**TI:ME:CO:DE**

**GENERAL**

Timecode is a running numbers generator, based on Blenders Geometry Nodes. It comes with five individual generators to either get an adaptive real-time numbers read-out of the current scene or to easily apply customizable running numbers wherever needed (e.g. radio alarm clocks, car stereo, sci-fi interfaces and -displays, time-bombs, timers, professional equipment, ....).

**Installation**

To work with Timecode in a new Blender project or to add it to the one that you're currently working on, simply link or append the collection or the object "TI:ME:CO:DE".

Click the corresponding object that has been added to the Outliner and navigate to the Modifiers tab. Here you'll find 63 carefully picked parameters to customize Timecode to your needs.

**Adaptivity**

The numbers for Timecode's dynamic and flexible time displays come from five different source generators that automatically adapt to many of Blender's playback preferences:

- The scene frame rate
- The first and last frame of the current scene as well as the resulting frame range
- The first and last frame, as well as the frame range and current state of Preview Range
- All of the above also work seamlessly in combination with various user inputs

**User Input**

Timecode gives you an effective set of customizable parameters at hand. The target of each parameter can be identified by the short labels in brackets:

---

<b>[src]</b>	Affects the behavior of most generators
<b>[src #]</b>	Only affects a specific or a selection of generator(s)
<b>[vis]</b>	Affects all three visual styles (e.g. materials)
<b>[vis #]</b>	Only affects one of the three visual styles
<b>[clk]</b>	Affects only the visual presentation of the the analog clock face
<b>[seq]</b>	Affects only the visual presentation of the the analog clock face
<b>[stat]</b>	This only affects the visual presentation of the extended statistics

---

## Custom Start- and Stop-Positions

You can determine specific frames where the timers should start counting from their relative 0-values and where they should stop running, while keeping their last value.

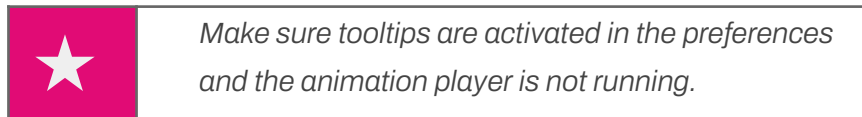
The **Custom Timer Start** frame number only has an effect, if it's greater than the first frame of the scene or preview range (if activated). Accordingly the **Custom Timer Stop** frame number only comes into play, when the custom value is less than the last frame of the scene or preview range (if activated).

## Base-Clock Multiplier

If the timers are running too slow for your needs, you can use the Base-Clock Multiplier to speed-up time up to factor 10,000 (!).

## Tooltips

While all of Timecode's parameter names are somewhat self-explanatory, each one of them also holds a tooltip with an informative short-description. Just hover above the input field.



## Default- and Min-/Max- Values

Every customizable parameter has a helpful default value to which you can reset it to, at any time. Additionally, the minimum and maximum values.

You can hot-switch between three different digit-looks that you can customize further to your needs.

**Global Parameters** [*prefix: vis*]

These parameters apply to all visual styles / sources:

**7 Material Slots**

Just basic colors to maintain overview



These materials are shared among all generators and styles. You can use these basic materials that are set as default for each slot or link your own materials. To find out which slots are used by a certain style, please check each section's specs table.

**Custom Object #1 to All**

If checked, the custom object linked in the “[vis 1] Custom Object” slot applies to all visual styles. This works, even if the other styles have their own custom object linked and activated, or not.

**Transform**

Position (x, y, z), Rotation (x, y, z), Scale (x, y, z)

**Select Visual Style**

Use numbers 1-3 to select one of three available visual styles.

**Don't Show Category Names**

Hide the small introductory headlines that appear with each source.

**Style Specific Parameters** (*prefix: vis [#]*)

Each style is created from an integrated specific base object but it can be replaced by a custom object from the outliner. Any adjustments to your custom object - including modifiers - apply to the currently selected digit display at runtime.

To link a **Custom Object** from your outliner, just use the eyedropper or drag & drop it onto the corresponding modifier input field and check the box below.

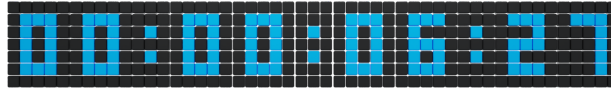
This replaces the standard objects ([vis 1], [vis 2]), or populates it as instances on the text-curve points ([vis 3]).

**The 3 Visual Styles**

**Cube Arrays [vis 1]**

Base-Object	Blender Standard Cube
Material Slot[s] #	1 / 2 / 3 / 4 / 5 / 6 / 7

Link Custom Object	Yes (Slot #1)
Instances Transformable	Yes (Scale x, y,z & Rotate x, y, z)



Numbers are being displayed by toggling materials from material slot #1 to material slot #3. Apply custom materials for both highlighted and non-highlighted elements.

**LED-Style Digits [vis 2]**

Base-Object	Mirrored Trapezoids
Material Slot #	1 / 2 / 3 / 4 / 5 / 6 / 7
Link Custom Object	Yes (Slot #2)
Instances Transformable	Yes (Scale x, y,z & Rotate x, y, z)



The displayed numbers are generated at runtime, by toggling the visibility of seven, pre-laid-out mirrored trapezoid curve-objects, according to the input values.

**Monotype Font [vis 3]**

Base-Object	(Meshed) Monotype Strings
Material Slot(s) #	1 / 2 / 3 / 4 / 5 / 6 / 7
Link Custom Object	Yes (Slot #3)
Instances Transformable	Yes (Scale x, y, z & Rotate x, y, z), Instances Amount & -Alignment



Compared to the other visual styles, this one seems to be a bit simple. It's basically just "values to strings to curves" but when populating a Custom Object as instances on its curve points, you can get quite abstract and playful results. If you are familiar with Geometry Nodes, it's easy to replace the font for curve generation with any other font you have installed.

**SOURCE GENERATORS**

In the first input field you can easily toggle between all five Source Generators by numeric input (1-5). This works independently from the currently active visual style.

**SMPTE Timecode [src 1]**

Format	hh:mm:ss:ff
Adapts to Base Clock Multiplier	Yes
Adapts to Custom Frame Offset	Yes
Adapts to Custom Start / Stop	Yes
Source Generator #	1



This shows a typical 4-segment SMPTE-Timecode that runs in sync to the scene timeline. It automatically adapts to Scene Frame Rate, Scene Start Frame, Preview Start, -End and -Range, Custom Timer Start- and End-Frame and Base-Clock Multiplier.

When the timecode reaches 23:59:59:29 (at 30 fps), the next frame will reset it to 00:00:00:00.

**Manual Timecode- / Clock-Mode**

If activated, the timecode won't run by itself but you can manually input any number for hours, minutes, seconds and frames, in the corresponding fields below, to display the exact timecode you want/need. Of course, you can keyframe each of the three parameters - or just keyframe the seconds, minutes and hours will follow, when it's time.

This way, you can easily maintain continuity - even with complex time leaps.

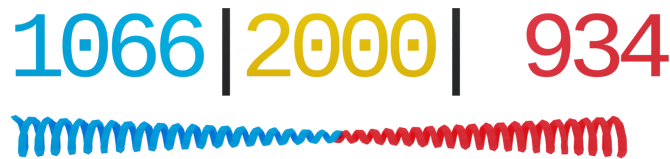
All numbers sum up correctly to the next higher unit (e.g. an input of 1900 frames @30fps is displayed as 00:01:03:10). Additional inputs for seconds and/or minutes add up correctly to the calculated numbers.

**Adaptive Frame Counter [src 2]**

Format	Current Frame / Total Frames / Frames left
Adapts to Base-Clock Multiplier	No
Adapts to Custom Frame Offset	Yes
Adapts to Custom Start / Stop	Yes




Source Generator #	2
--------------------	---



The Frame Counter shows the current frame, the total frame range and the frames to go till the end. The numbers adapt automatically to the state of the **Preview Range** and **Frame Range**, the **Frame Rate**, **Custom Start- and Stop Frame** and **Custom Frame Offset**.






### Bars & Beats Counter [src 3]

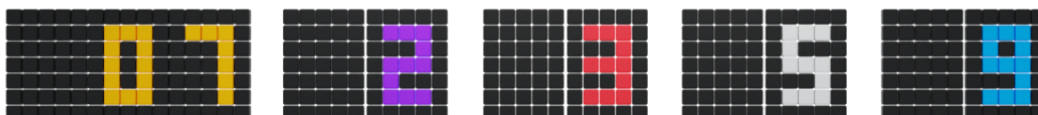
Format		<i>Full.Half.4th.8th.16th Notes</i>
Adapts to Base Clock Multiplier	No	
Adapts to Custom Frame Offset	No	
Adapts to Custom Start / Stop	Yes	
Source Generator #	3	

If you want to work beat-sync to music, playing in Blender’s Video Sequencer, you can fill in its song tempo (in BPM, Beats per Minute) and time signature, to get several little helpers for timing, rhythm and music related timeline position.

To adjust the Bars & Beats display to your needs, you can apply an offset in bars / full note intervals.

### Customizable Parameters

-  ● Song Tempo in BPM
-  ● Bar / Full-Note Offset
-  ● Time Signature: Counter & Denominator (*only works for 4/4 and 3/4 at the moment*)
-  -- Show Music Timeline
-  ● Show 16-Step Sequencer



### EXTRA: Tempo-, Rhythm- and Timing Indicators

#### 16-Step Sequencer

The 16 steps change their material to material slot #3 synchronously to the 16th notes of the numeric counter above, showing the exact 16th note position within the current bar. The

accented steps 1, 5, 9 and 13, which represent the quarter notes of each bar, are clearly emphasized.



**Musical Timeline Representation**

A thin line represents the range of the scene timeline, divided in full-note lengths, showing the total amount of measures/bars, sub-divided in quarter note lengths, based on the determined BPM. Three position-indicators represent bars, quarter notes and 16th notes.



**Digital & Analog Chronometers [src 4]**

Format	hh:mm:ss / clock face
Adapts to Base-Clock Multiplier	Yes
Adapts to Custom Frame Offset	Yes
Adapts to Custom Start / Stop	Yes
Source Generator #	4

The Digital Clock comes in the classic numeric format, showing hours, minutes and seconds separated by colons, flashing once per second. If time is running too slow for you, just turn-up the Base-Clock Multiplier to see hours passing like seconds.

But what happens once the 12th hour is through? Well, it depends!

You can easily toggle between the **12h- and 24h system**. For the 12h-System, an additional **AM/PM** indicator shows up, to prevent confusion.



**Customizable Parameters**

- Toggle 12h- and 24h time system
- Toggle standard- and manual mode
- In Manual Mode: Hours, Minutes, Seconds  
 Frames → only applies to the Timecode Generator

**Manual Timecode/Clock Mode**

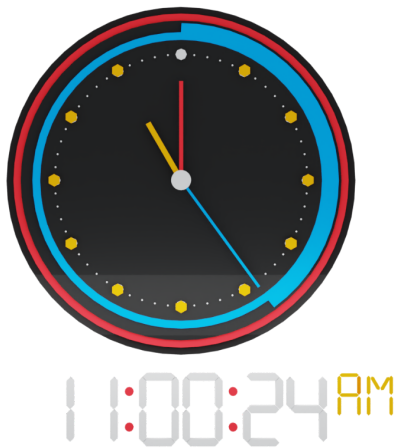
In this mode, the clock stands still, even if the animation player is running - only the colon-delimiters keep flashing at one second intervals but it allows you to manually input static values for hours, minutes and seconds, to display the exact time you want/need.

If you still want the clock to run, you can keyframe each of the three parameters - or just keyframe the seconds, minutes and hours will follow, when it's time. This way, you can easily maintain continuity - even with complex time leaps or go backwards in time.

All values sum up correctly to the next higher unit (e.g. an input of 125 seconds is displayed as 00:02:05).

**EXTRA: Analog Clock Face**

Typical clock-face with hands for hours, minutes and seconds as well as an additional time indicator in circles around the clock's rim. The analog clock also follows the Base-Clock Multiplier and runs in sync with the Digital Clock. Obviously, switching from 12h- to 24h-System has no effect on the analog clock.



**Customizable Parameters [clk]:**

- Show Analog Clock
- Position (x, y, z)
- Rotation (x, y, z)
- Scale (x, y, z)
- Show Hands
- Show Rings
- Show Dots
- Showq Base

**Countdown [src 5]**

Display Format	ssss
Adapts to Base-Clock Multiplier	Yes
Adapts to Custom Frame Offset	No
Adapts to Custom Start / Stop	Yes

Source Generator #	5
--------------------	---



This is a classic numeric Countdown at one second intervals from any integer(max. 9,999) and a calculated decimal.

It also “listens” to the Base-Clock Multiplier, thus you can break down the countdown intervals down to tiny fractions of a second, if you like.

**Customizable Parameters**

- Countdown start-value in seconds (max. 9999)

**Extended Time/Timing Statistics**

Display Format	Various
Adapts to Base-Clock Multiplier	Yes
Adapts to Custom Frame Offset	Yes
Adapts to Custom Start / Stop	Yes
Customizable	Yes, Line-Height, Character Spacing, Position, Rotation, Scale

This is a comprehensive statistics read-out with additional calculations of many important time- and timing-relevant parameters. You can show/hide the time related stats with just one click and add the musical info with another.

**Time Related Info**

- First/last frame and Frame Range
- Custom Timer Start- and Stop-Frame
- Timer Range / Duration
- First/Last Frame, Range and State of Preview Range
- Total duration of the Scene in **mm:ss.s**
- Custom Frame Offset / resulting Time Offset
- Frame Rate of the Scene
- Frame Skip on/off | Frame Step Rate

**Music Related Info**

- Applied Song Tempo
- Applied Time Signature
- Scene Length in Bars
- Frames per Beat / Bar
- Milliseconds per Beat / Bar
- Applied Bar Offset

PARAMETER OVERVIEW					
1	[src] Select Source (1-5)	26	[clk] Show Analog Clock	51	[stat] Toggle Time / Music
2	[src] Start Timer at	27	(clk) Position	52	[stat] Position
3	[src] Stop Timer at	28	[clk] Rotation	53	[stat] Rotation
4	[src 1,4,5] Base Clock Multiplier	29	[clk] Scale	54	[stat] Scale
5	[src 1, 2] Frame Offset	30	[clk] Show Hands	55	[stat] Line Height:
6	[src 3] Song Tempo (BPM)	31	[clk] Show Rings	56	[stat] Character Spacing
7	[src 3] Time Sig. (Counter)	32	[clk] Show Dots	57	[vis] Material 1
8	[src 3] Time Sig.(Denominator)	33	[clk] Show Base	58	[vis] Material 2
9	[sre 3] Bar Offset	34	[vis 2] Custom Obj.	59	[vis] Material 3
10	[src 4] Toggle 12h224h	35	[vis 2] Use Custom Obj.	60	[vis] Material 4
11	[src 1, 4] Use Manual Clock	36	[vis 2] Scale Instances	61	[vis] Material 5
12	[sre 1, 4] Manual Clock hh	37	vis 2] Rotate Instances	62	[vis] Material 6
13	[sre 1, 4] Manual Clock mm	38	[seq] Show Sequence A	63	[vis] Material 7
14	[sre 1, 4] Manual Clock ss	39	[seq] Show Sequence b		
15	[sre 1] Manual Clock ff	40	[seq] Position		
16	[sre 5] Countdown from	41	[seq] Rotation		
17	[vis] Select Digit Style (1-3)	42	[seq] Scale		
18	[vis] Don't Show Category Names	43	[vis 3] Custom Obj.		
19	[vis] Position	44	[vis 3] Use Custom Obj		
20	[vis] Rotation	45	[vis 3] Instance Amount		
21	[vis] Scale	46	[vis 3] Instance Alignment		
22	[vis 1] Custom Obj.	47	[vis 3] Scale Instances		
23	[vis 1] Use Custom Obj.	48	[vis 3] Rotate Instances		
24	[vis 1] Scale Instances	49	[vis] Custom Obj. 1 to All		
25	[vis 1] Rotate Instances	50	[stat] Hide / Show		
Parameter Types					
Integer	Vector	Value			
Boolean	Object	Material			

THANK YOU FOR BUYING!