



// SECTION 01 • SETUP GUIDE

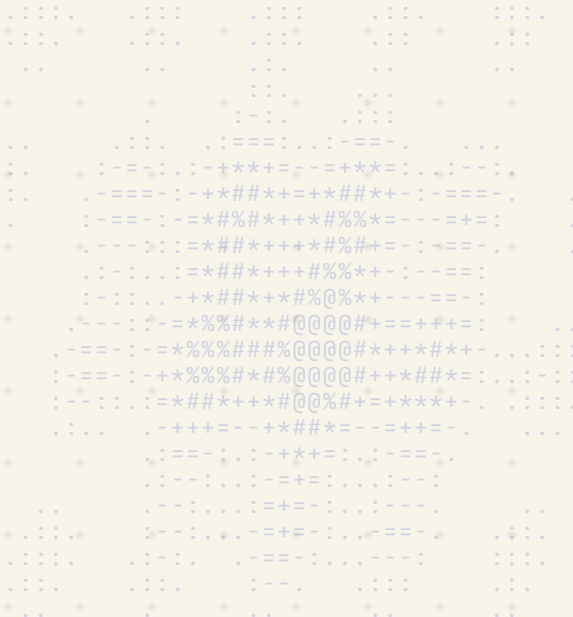
REMOTION IN CLAUDE



How to wire up Remotion, the React video framework, so Claude Code writes and edits your videos while Remotion Studio previews every change live. From an empty folder to a rendered MP4.

```
> boot opusjake_os
> resource: remotion-in-clause
> version: v1.0 • 2026
> status: ready_
```

Remotion turns video into React code, which makes it the rare creative tool Claude Code can drive directly. This is the setup I run, from an empty folder to a rendered MP4.



SCAFFOLD IT

You need Node 18 or newer and Claude Code installed. Check the version, then let Remotion build the project for you.

```
node --version  
npx create-video@latest
```

Pick the Hello World template when it asks. Remotion scaffolds a working project with one example composition and installs the dependencies for you. Then move into the folder it created.

```
cd your-project-name
```

That folder is now a normal React codebase. Everything from here is editing files Claude already knows how to read.

TWO WINDOWS

The whole workflow is two windows side by side: Remotion Studio showing the video, Claude Code changing the code.

Start the Studio from the project folder. It is the live preview.

```
npx remotion studio
```

It opens at localhost:3000 with the canvas, a frame-by-frame timeline you can scrub, and a props panel on the right. Leave it running.

Now open a second terminal in the same folder and start Claude Code.

```
cd your-project-name  
claude
```

The loop runs itself from here. You describe a change in plain English, Claude edits the files in src/, and the Studio reloads the instant the file saves. You watch the video change while you type.

THE MENTAL MODEL

Five ideas make your prompts land. Learn these and Claude handles the syntax.

- **COMPOSITION.** Every video is registered in `src/Root.tsx` with an `id`, a `size`, an `fps`, and a `length`. This is the file Claude edits to add or resize a video.
- **FRAMES, NOT SECONDS.** Time is counted in frames. At 30 fps, frame 90 is the three-second mark. Length is set as `durationInFrames`.
- **`useCurrentFrame()`.** The one hook everything animates from. It returns the frame being drawn right now.
- **`interpolate()` and `spring()`.** Map the current frame to any value: opacity, position, scale. `spring()` gives natural, physical motion.
- **Sequence.** Wrap an element to control when it starts and how long it stays on screen.

You do not have to write any of it. You just have to ask for it in these terms.

YOUR FIRST EDIT

Keep the Studio open beside you and paste this into Claude.

```
In the main composition, show a centered title
that reads OPUSJAKE on a #F7F4EA background in
#12121C. Fade it in over the first 15 frames and
slide it up 20px with spring(). Use AbsoluteFill,
useCurrentFrame, interpolate, and spring.
```

Watch the Studio. The title appears, fades in, and settles. Now iterate in plain English: "make it two seconds longer", "add a subtitle under it", "soften the easing". Each prompt is one diff you can read and undo.

RENDER IT

When it looks right, render to a file. Remotion runs the React in headless Chrome, captures every frame, and stitches them together with ffmpeg.

```
npx remotion render
```

It asks which composition to use, then writes an MP4 into out/. To target one by id and name the file yourself:

```
npx remotion render <CompositionId> out/video.mp4
```

For a single frame instead of a clip, swap in npx remotion still. That is the whole pipeline: scaffold, preview, prompt, render.

TALK IN OUTCOMES

The setup rewards one habit. Keep the Studio open and tell Claude what you want, not how to code it.

YOU WANT	TELL CLAUDE
A new video	"Add a composition, 1920x1080, 30fps, 5 seconds"
Motion	"Animate the logo in with a spring over 20 frames"
Timing	"Start the subtitle at the one-second mark"
Sound	"Play audio.mp3 from public/ under the clip"
The file	"Render it to out/ as an MP4"

One honest note: Remotion is free for individuals and teams of up to three. Past that, companies need a paid license at [remotion.pro](#), so check it before you build a business on it. For more build-with-AI playbooks, follow [opusjake.ai](#).