

Chapter 55 - When BASIC Is Not Enough

BASIC is the right first language for Intuition Engine. It lets you see the bus, try registers, start sound, and build effects without a second tool. It is not always the right final language for every part of a demo.

The usual upgrade path is gradual. Keep the structure that works, then move the hot or awkward part.

55.1 Keep BASIC For Orchestration

BASIC is good at:

- Allocating buffers with MEMALLOC.
- Loading assets with BLOAD.
- Starting audio.
- Setting up VideoChip, copper, and blitter state.
- Describing the timeline.
- Calling into native parts when they exist.

Even a native-heavy demo can use BASIC as the front panel while the effect is being designed.

55.2 Use IE Mon For Small Patches

Use IE Mon when the change is small enough to enter and inspect:

```
(ie64)> A 1000
asm $000000000000001000> move.q r2,#$00100000
asm $000000000000001008> store.l r2,(r3)
asm $000000000000001010>
Exited IE64 assemble mode
(ie64)> d 1000 #2
```

IE Mon is best for short routines, register tests, and verifying an instruction sequence before it becomes part of a larger source file.

55.3 Use IE Script For Repetition

Use IE Script when the setup must be repeated exactly:

```
cpu.freeze()
mem.write_block(0x600000, sys.read_file("TEXTURE.RAW"))
cpu.resume()
sys.wait_frames(2)
sys.print(video.frame_hash())
```

That is not a replacement for the demo. It is how you stage, run, and measure the same experiment again.

55.4 Move Hot Loops To Native Code

Move a part out of BASIC when:

- It does heavy arithmetic every frame.
- It rebuilds large tables or textures.
- It needs exact instruction timing.
- It needs CPU-specific addressing or byte operations.
- It is stable enough that interactive editing is no longer useful.

The rotozoomer is the pattern. BASIC proves the effect. IE Script makes experiments repeatable. IE64 and IE32 make the core loop native. The other CPU versions show how the same VideoChip contract looks through different instruction sets.

55.5 Advanced Companion Variants

Some supplied rotozoomer variants run through operating-system-facing interfaces rather than the bare-machine path. Treat them as companion study material after the PRG path is understood.

| Variant | What to study |
|------------------------|--|
| GEM-style M68K version | How an OS-facing programme still reaches the same presentation idea. |
| API-style version | How a higher-level system call path wraps the same demo structure. |
| Hardware-style version | How direct hardware access compares with the API path. |

These variants are not the normal route through this guide. They belong after you understand the bare bus, VideoChip, file block, audio engines, IE Mon, and IE Script.

55.6 Decision Table

| Problem | First tool |
|----------------------------------|--|
| Learn a register | BASIC direct mode. |
| Try a short visual idea | BASIC programme. |
| Repeat a setup and capture proof | IE Script. |
| Patch a few IE64 instructions | IE Mon assemble mode. |
| Write a fast stable loop | Native CPU source assembled inside the machine where supported, or loaded as a prepared image. |
| Compare CPU styles | Read the six-CPU examples against the same MMIO contract. |

55.7 The Rule To Keep

Do not upgrade because a language seems more serious. Upgrade because a specific part has outgrown its current form.

The best IE demo code keeps the machine visible:

```
BASIC or script sets the stage.  
Native code does the hot work.  
VideoChip, audio, files, input, and coprocessors stay on the same bus.  
IE Mon proves what happened.
```

That is the end of the Part VII climb: from a moving block in BASIC to a complete intro structure and a clear path into native code.