Compiling Faust with Ondemand

Yann Orlarey

GRAME
CENTRE NATIONAL
DE CRÉATION
MUSICALE, LYON

June 2022

Identifiers and Marking

Fresh identifier representing memory locations

- $\mathsf{id}_\mathsf{s}[\![S.T]\!] = m$ unique scalar identifier for S in time context T ;
- $\mathsf{id}_\mathsf{v}[\![S.T]\!] = v$ unique vector identifier for S in time context T ;
- $\mathsf{id}_\mathsf{t}[\![T]\!] = t$ unique scalar identifier representing the current time in time context T.

Marking recursive definitions

- $\mathsf{mark}[\![Xi.T]\!] = \varnothing$: not yet marked ;
- $\mathsf{mark}[\![Xi.T]\!] \leftarrow v$: mark it with identifier v ;
- $\mathsf{mark}[\![Xi.T]\!] = v$: already marked with v.

Signal Compilation

Function $\mathsf{cs}[\![.]\!] : \mathbb{S} \times \mathbb{T} \rightarrow \mathbb{M} \times \mathcal{P}(\mathbb{I})$

Number

$$(\mathsf{num})\,\frac{}{\mathsf{cs}[\![k.T]\!] = k \times \varnothing}$$

User interface

$$(\mathsf{ctrl})\,\frac{}{\mathsf{cs}[\![u.T]\!] = u \times \varnothing}$$

Inputs

$$(\mathsf{input})\,\frac{}{\mathsf{cs}[\![\mathtt{I}_c.T]\!] = \mathtt{I}_c \times \varnothing}$$

Signal Compilation

Numerical Operation

$$\text{(nop)}\ \frac{\begin{array}{c}\mathsf{cs}[\![S_1.T]\!] = M_1 \times J_1 \\ \mathsf{cs}[\![S_2.T]\!] = M_1 \times J_2 \\ \vdots \\ \mathsf{id_v}[\![\star(M_1, M_2, ...)]\!] = m\end{array}}{\mathsf{cs}[\![\star(S_1, S_2, ...).T]\!] = m \times \{T \vdash m{:=} \star (M_1, M_2, ...)\} \bigcup_i J_i}$$

Signal Compilation

Downsampling

The downsampling $S_1 \downarrow S_2$ appears at the entrance of an ondemand. This means that compiling $S_1 \downarrow S_2$ into the $M_2.T$ time environment (where M_2 is the compiled version of S_2) is like compiling S_1 into the T time environment and using a variable to do the downsampling.

$$\text{(down)}\,\frac{\begin{array}{c}\mathsf{cs}[\![S_1.T]\!] = M_1 \times J_1 \\ \mathsf{cs}[\![S_2.T]\!] = M_2 \times J_2 \\ \mathsf{id_s}[\![M_1.T]\!] = m \\ J_3 = \{T \vdash m{:=}M_1\}\end{array}}{\mathsf{cs}[\![(S_1 \downarrow S_2).M_2.T]\!] = m \times J_3 \cup J_1 \cup J_2}$$

Signal Compilation

Upsampling

The $S_1 \uparrow S_2$ upsampling appears at the output of an ondemand. It is necessary to compile S_1 into the clock time reference S_2 (which is added to the current time reference). The signal S_1 must also be stored in a variable to do the upsampling.

$$\text{(up)}\,\frac{\begin{array}{c}\mathsf{cs}[\![S_2.T]\!] = M_2 \times J_2 \\ \mathsf{cs}[\![S_1.M_2.T]\!] = M_1 \times J_1 \\ \mathsf{id_s}[\![M_1.M_2.T]\!] = m \\ J_3 = \{M_2.T \vdash m{:=}M_1\}\end{array}}{\mathsf{cs}[\![(S_1 \uparrow S_2).T]\!] = m \times J_1 \cup J_1 \cup J_2}$$

Signal Compilation

Delay

$$\dfrac{\begin{array}{c}\mathsf{cs}[\![S_1.T]\!] = M_1 \times J_1 \\ \mathsf{cs}[\![S_2.T]\!] = M_2 \times J_2 \\ \mathsf{id_v}[\![M_1.T]\!] = v \\ \mathsf{id_t}[\![T]\!] = t \\ J_3 = \{T \vdash v[t,0]{:=}M_1\} \cup \{T \vdash t{:=}t+1\}\end{array}}{\mathsf{cs}[\![(S_1 @ S_2).T]\!] = v[t, M_2] \times J_3 \cup J_1 \cup J_2}\ \textsf{(up)}$$

Signal Compilation: recursion

First visit

If it is the first visit, we have $\mathsf{mark}[\![X_i.T]\!] = 0$:

$$(\mathsf{r1})\ \frac{\begin{array}{c}\mathsf{id_v}[\![X_i.T]\!] = v \\ \mathsf{mark}[\![X_i.T]\!] \leftarrow v \\ \mathsf{def}[\![X]\!] = (...,S_i,...) \\ \mathsf{cs}[\![S_i.T]\!] = M_i \times J_i \\ \mathsf{cs}[\![S_d.T]\!] = M_d \times J_d \\ \mathsf{id_t}[\![T]\!] = t \\ J_3 = \{T \vdash v[t,0]{:=}M_i\} \cup \{T \vdash t{:=}t+1\}\end{array}}{\mathsf{cs}[\![(X_i @ S_d).T]\!] = v[t, M_d] \times J_3 \cup J_i \cup J_d}$$

Signal Compilation: recursion

Next visits

If it is not the first visit, we have $\mathsf{mark}[\![X_i.T]\!] = v$:

$$\text{(r2)}\ \frac{\mathsf{cs}[\![S_d.T]\!] = M_d \times J_d \qquad \mathsf{id_t}[\![T]\!] = t}{\mathsf{cs}[\![(X_i@S_d).T]\!] = v[t, M_d] \times J_d}$$

Global compilation

$\mathsf{comp}[\![.]\!] : \mathbb{S}^n \to \mathcal{P}(\mathbb{I})$

Global compilation

$$\dfrac{\begin{array}{c}\vdots \\ \mathsf{cs}[\![S_i.1]\!] = M_i \times J_i \\ J'_i = \{1 \vdash \mathsf{o}_i{:=}M_i\} \cup J_i \\ \vdots\end{array}}{\mathsf{comp}[\![(\ldots, S_i, \ldots)]\!] = \ldots \cup J'_i \cup \ldots}\ (\mathsf{comp})$$