Lecture 11
Examples of Z Specifications

David Garlan
Carnegie Mellon University

Models of Software Systems
15-671
Fall 1995

© Garlan, 1995

This Lecture

An Industrial Case Study

A Formal Specification of an Oscilloscope

Delisle and Garlan, *IEEE Software* (Sept 1990)

New concepts

- **Bridging the gap between theory and practice**
- **Axiomatic specifications**
- **Higher-order functional specifications**

But First ...
Two Useful Identities

1. $\forall$ **x: T | P(x) • Q(x)**

$\equiv$

$\forall$ **x: T • P(x)** $\Rightarrow$ **Q(x)**

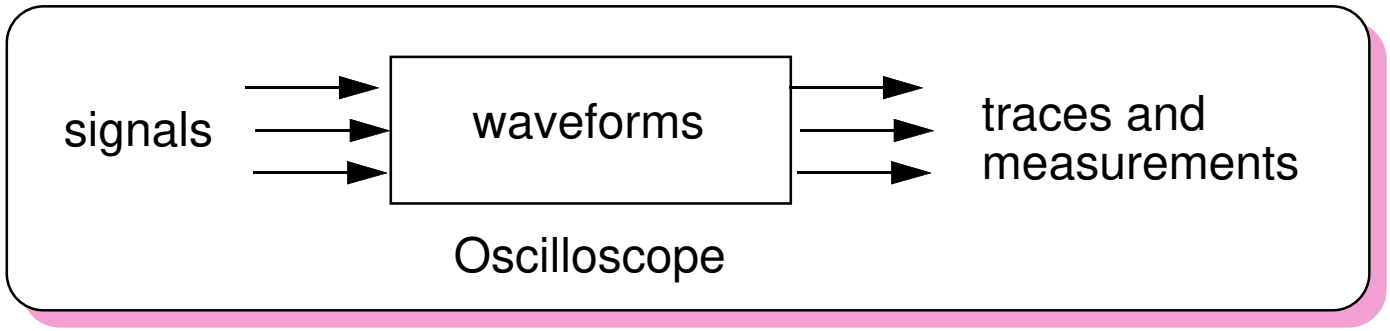
2. $\exists$ **x: T | P(x) • Q(x)**

$\equiv$

$\exists$ **x: T • P(x)** $\wedge$ **Q(x)**

Context

Oscilloscopes are simple in theory



But complex in practice

- **Complex software**
- **Multi-processing environment**
- **Special hardware**
- **Sophisticated user interface**
- **Many specialized modes**

The Problem

At Tektronix

- Many divisions building similar product
- No reuse among product efforts
- Long time-to-market (5 years)
- Each product typically built from scratch
- User interface increasingly important
- Software effort increasingly larger proportion of development costs

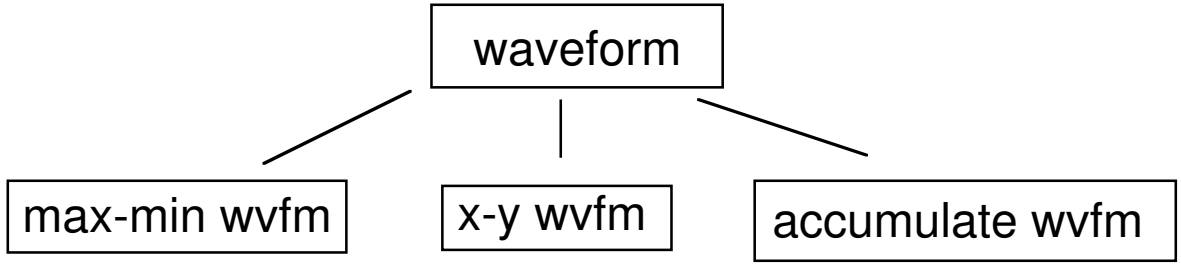
© Garlan, 1995

Goals

- Build next generation instrumentation systems
- Allow reuse between product divisions
- Support better interactive response to user changes
- Multiple hardware platforms for same user interface
- Multiple user interfaces for same platform (vertical markets)

First Attempt

First attempt was an Object-Oriented Architecture

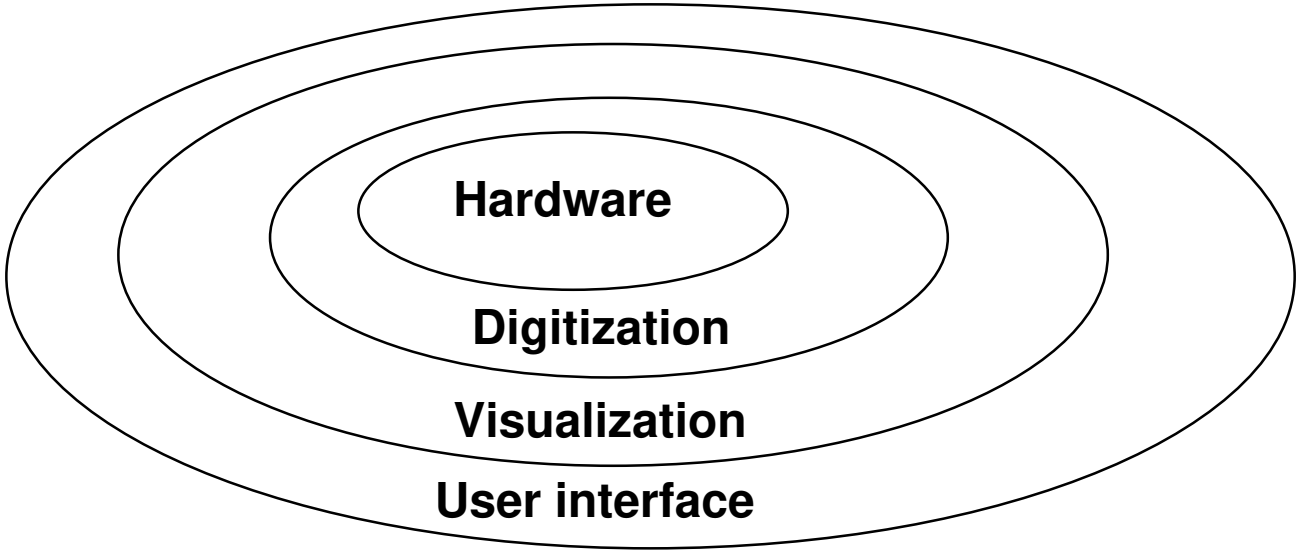


Result:

Hundreds of classes, little structure, no overall pattern

Layered View

Second attempt was Layered Architecture



Boundaries of abstraction not realistic

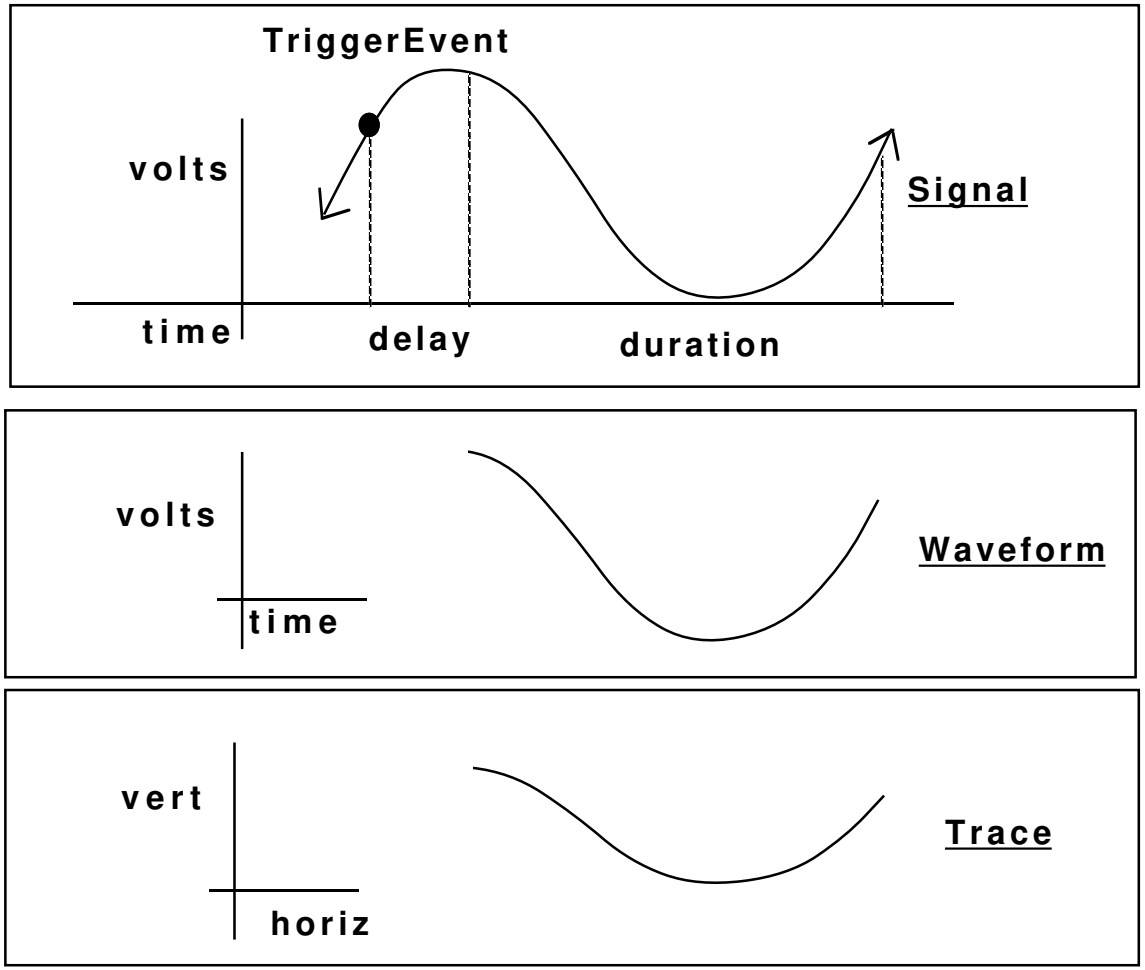
Functional View

Traces = Oscilloscope (Signals)

But:

1. **How to handle user input?**
2. **How can you decompose it into manageable pieces?**

Signals, Waveforms, Traces



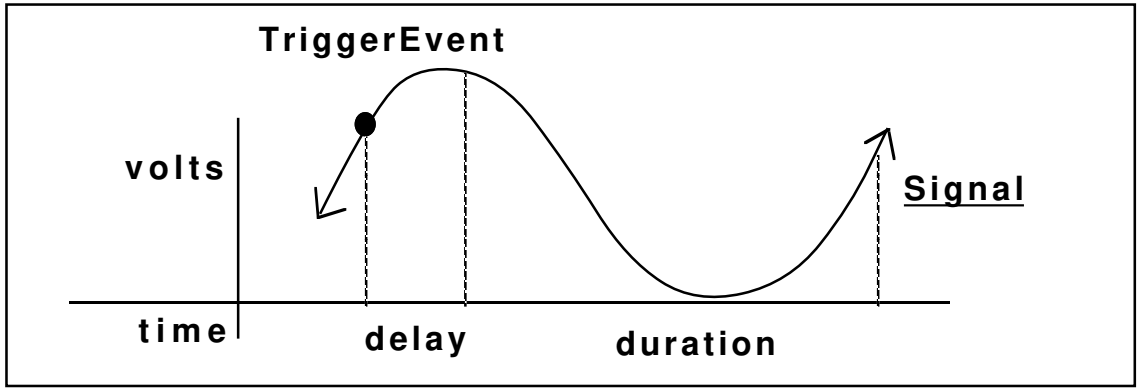
Basic Types

$$\begin{aligned} \textbf{AbsTime} &== \mathbb{N} \\ \textbf{RelTime} &== \mathbb{N} \\ \textbf{Volts} &== \mathbb{Z} \\ \textbf{Horiz} &== \mathbb{Z} \\ \textbf{Vert} &== \mathbb{Z} \end{aligned}$$

Signals, Waveforms, Traces

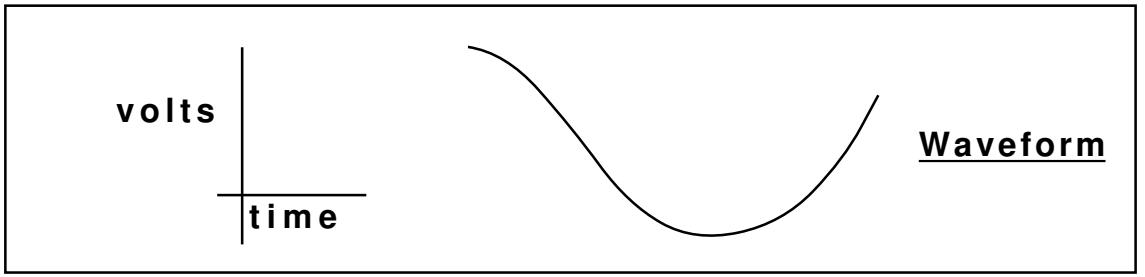
Signal ==

AbsTime → Volts



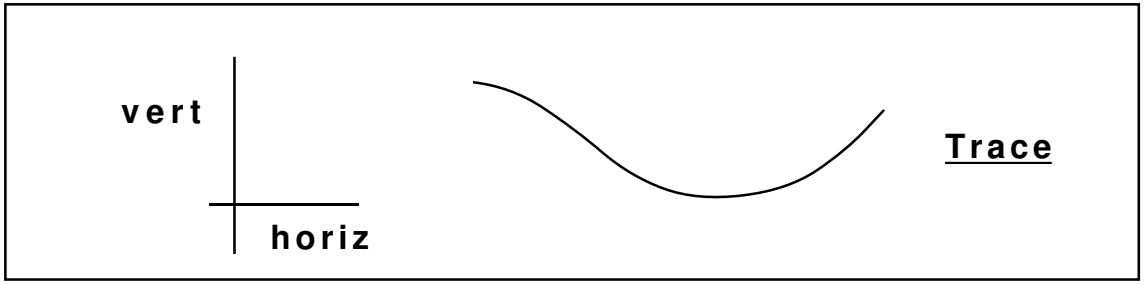
Waveform ==

AbsTime ⇸ Volts



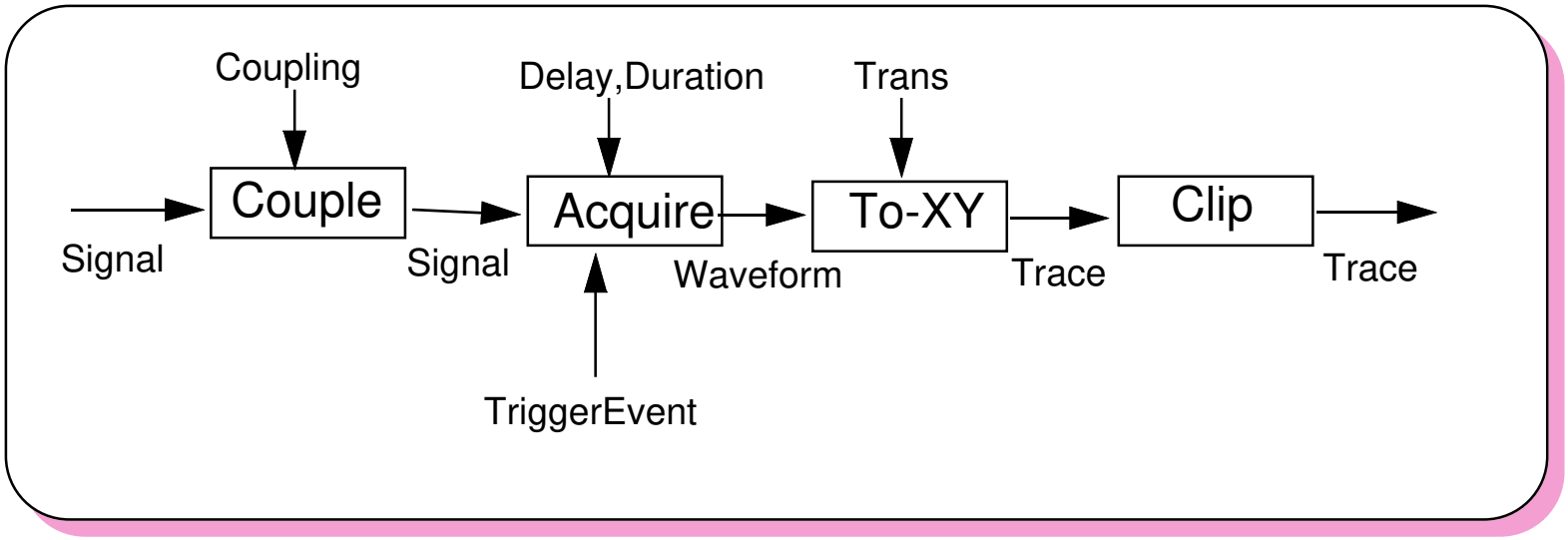
Trace ==

Horiz ⇸ Vert



© Garlan, 1995

Channel Subsystem



Coupling

Coupling ::= DC | AC | GND

Couple: Coupling → Signal → Signal

dc: Signal x AbsTime → Volts

Couple DC s = s

Couple AC s =

λ t: AbsTime • s(t) - dc (s, t)

Couple GND s =

λ t: AbsTime • 0

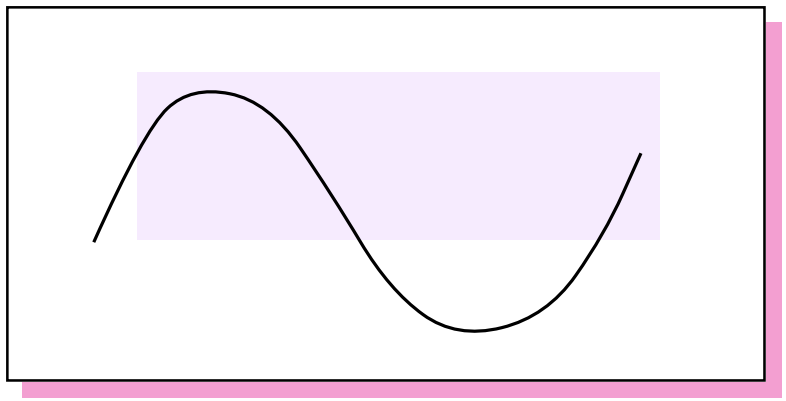
Acquisition

TriggerEvent == AbsTime

Acquire: RelTime x RelTime →
TriggerEvent → Signal → Waveform

Acquire (delay, dur) trig s =
{ t: AbsTime | trig + delay ≤ t ≤ trig + delay + dur } ◁ s

Clip



maxH: Horiz

maxV: Vert

Clip : Trace $\rightarrow$ Trace

Clip tr = (0 .. maxH) $_{\lhd}$ tr $_{\rhd}$ (0 .. maxV)

Channel Parameters

Channel Parameters

c: Coupling

delay, duration: RelTime

scaleH: RelTime

scaleV: Volts

posnV: Horiz

Channel Configuration

ChannelConfiguration: ChannelParameters →
TriggerEvent → Signal → Trace

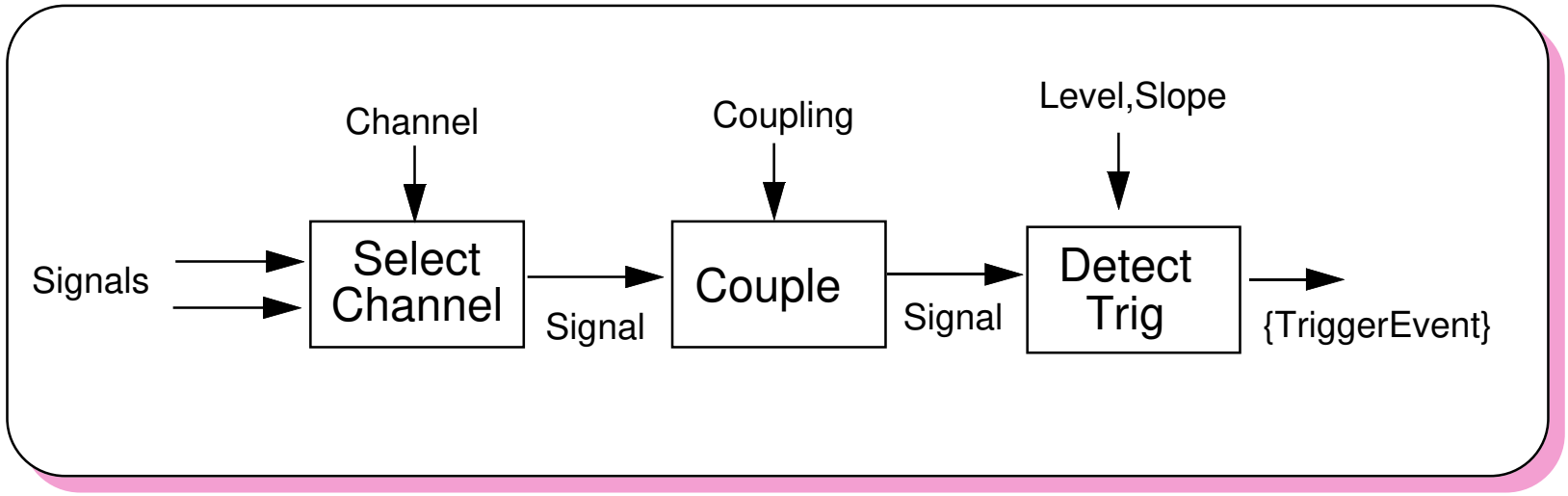
ChannelConfiguration p =

λ trig: TriggerEvent •

Clip ¤ WaveformToTrace (...) ¤

Acquire (p.delay, p.dur) trig ¤ Couple p.c

Trigger Subsystem



Channel Selection

Channel ::= CH1 | CH 2

SelectChannel: Channel →

Signal x Signal → Signal

SelectChannel Ch1 (s1, s2) = s1

SelectChannel Ch2 (s1, s2) = s2

Trigger Detection

Slope ::= POS | NEG

Level == Volts

DetectTrig: Level x Slope $\rightarrow$ Signal $\rightarrow$ $\mathbb{P}$ TriggerEvent

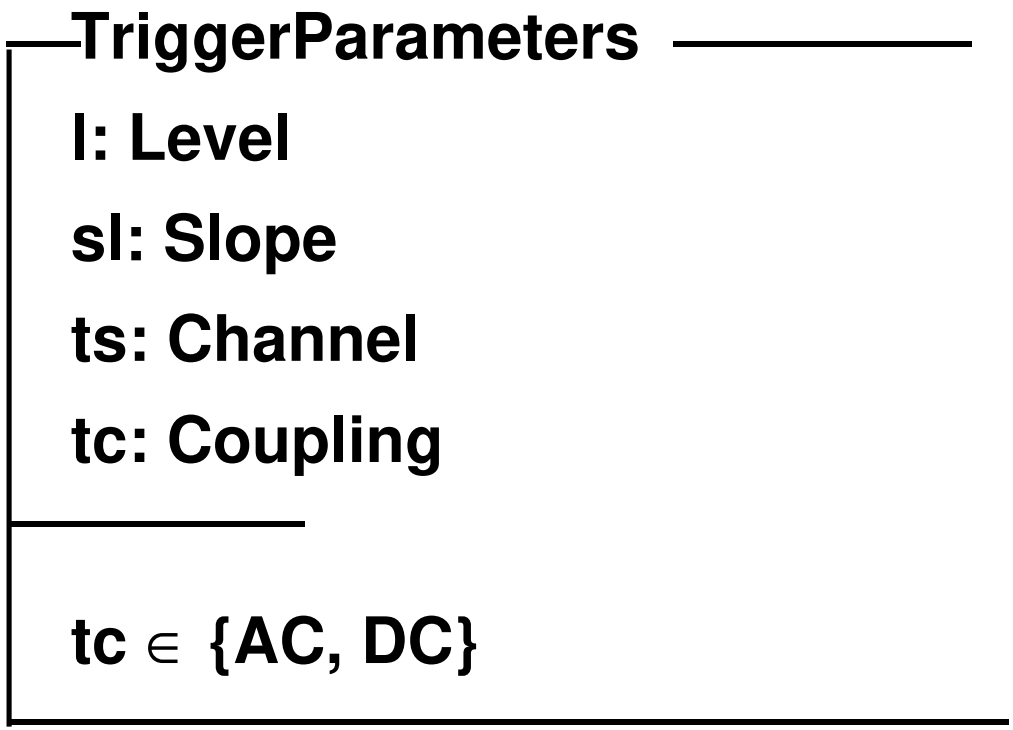
---

$\forall$ t: DetectTrig (l, sl) s •

(sl = POS) $\Rightarrow$ s (t - 1) $\leq$ l $\leq$ s (t)

(sl = NEG) $\Rightarrow$ s (t) $\leq$ l $\leq$ s (t - 1)

Trigger Parameters



Trigger Configuration

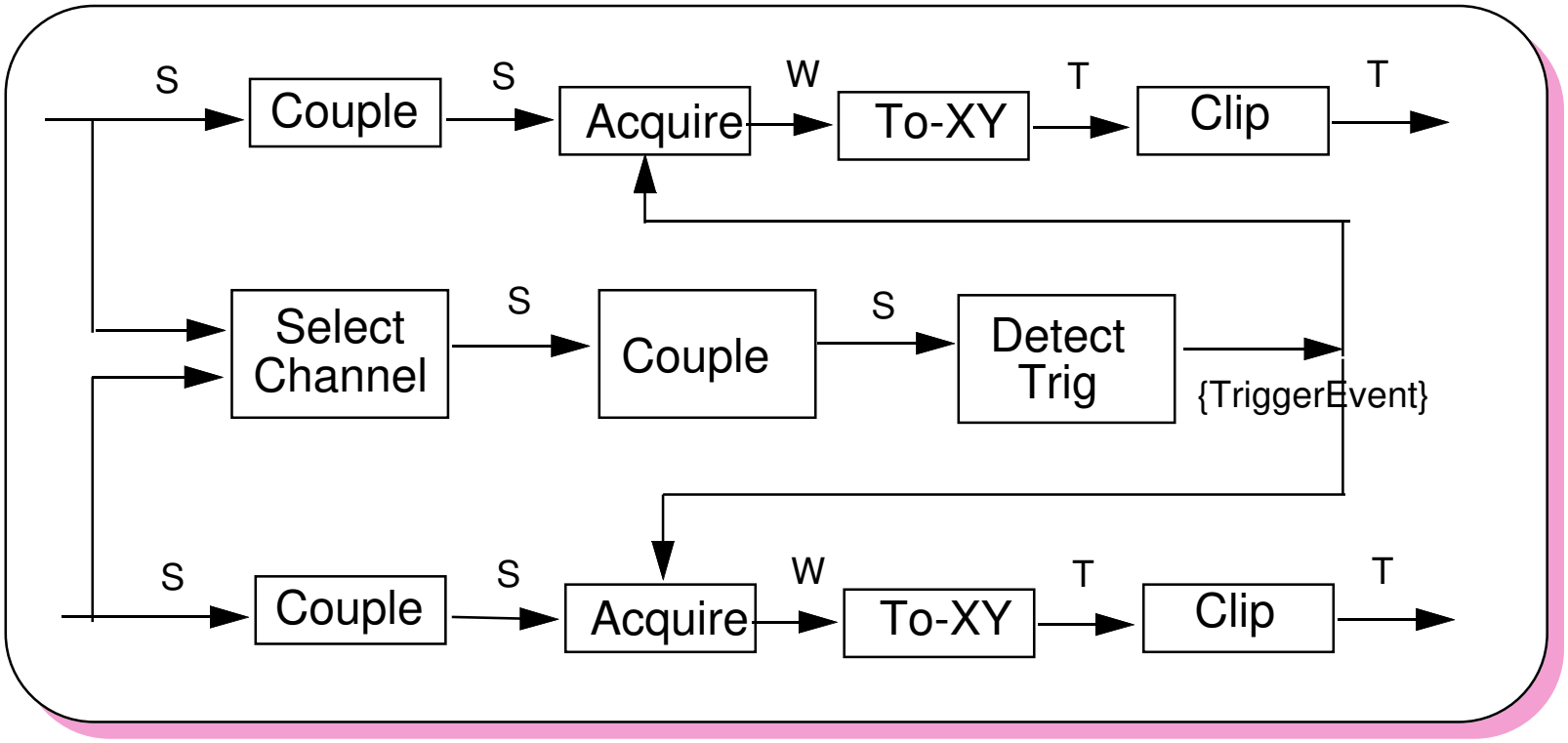
TriggerConfiguration: TriggerParameters $\rightarrow$ Signal
x Signal$\rightarrow$ P TriggerEvent

TriggerConfiguration p =

DetectTrig (p.l, p.sl) ¤ Couple p.tc ¤

SelectChannel p.ts

The Whole System



Oscilloscope

Oscilloscope

$s1, s2 : Signal$

$cp1, cp2 : ChannelParameters$

$tp : TriggerParameters$

$ts1, ts2 : \mathrm{seq}\ Trace$

$\forall\ t : \mathrm{ran}\ ts1 \bullet$

$\quad \exists\ trig : TriggerConfiguration\ tp\ (s1, s2) \bullet$

$\quad\quad t = ChannelConfiguration\ cp1\ trig\ s1$

$\forall\ t : \mathrm{ran}\ ts2 \bullet \ldots$

Role of Formalism

- What was formalized
  - User level model (higher-order P/F)
  - Connection framework (colored P/F)
  - User interface architecture
- Benefits of formalism
  - Motivated and constrained architectures
  - Conceptual prototyping
  - Communication medium
- Non-benefits
  - Correctness
  - Completeness
  - Adoption of formal methods within Tektronix

Some Morals

- Success requires
  - Domain experts and system builders
  - Expertise in abstraction
  - Skill in manipulating formal models
  - Willingness to abandon old implementation patterns
  - Willingness to reject inappropriate models
- Tools
  - Not needed during conceptualization
  - Essential during development
- Management
  - Must keep hands off the process initially
  - Must help inforce standards later