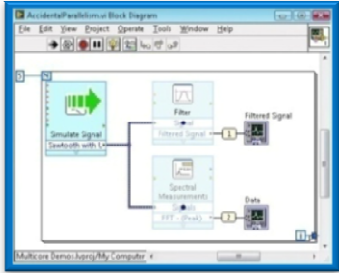


Desarrollando Aplicaciones con el LabVIEW Statechart Module

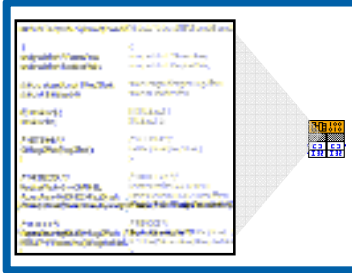


Herramientas de Desarrollo de Alto Nivel

Flujo de Datos



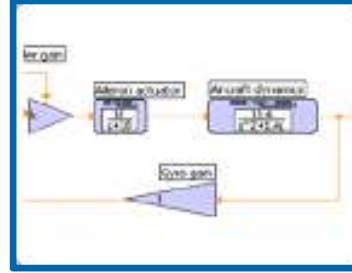
Código C



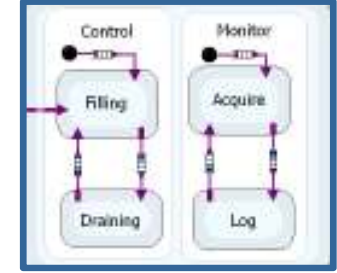
Matemáticas Textuales

```
1 c = 0.285 + 0.013i;  
2 [X Y] = meshgrid(x, y);  
3 z = X + i*Y;  
4 for k=1:30  
5     z = z.^2 + c;  
6 end
```

Modelado



Diagramas de Estado



NATIONAL INSTRUMENTS

LabVIEW™

Plataforma de Diseño Gráfico de Sistemas

Linux®



Macintosh



Windows

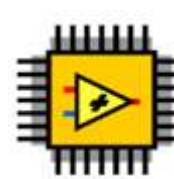


Plataforma de Escritorio

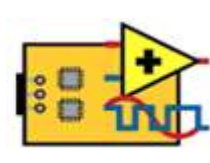
Tiempo Real



FPGA



Micro



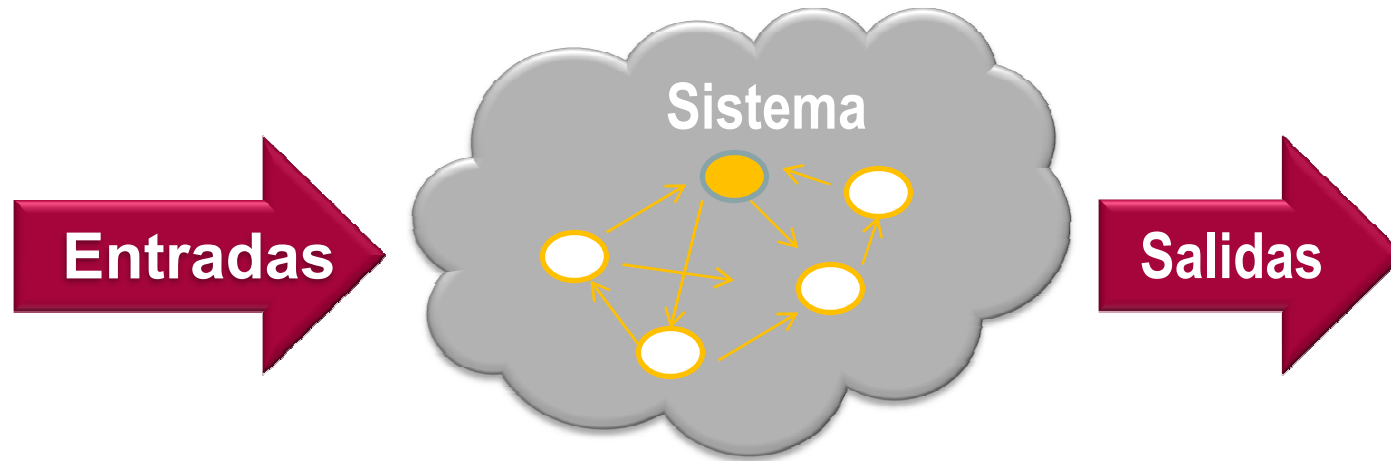
Plataforma Embebida

ni.com/statechart

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries. Tux penguin is courtesy of Larry Ewing.



¿Qué son los “Diagramas de Estado”?

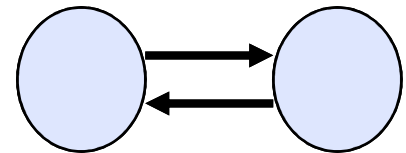


“Diagramas de Estado” es una representación visual de sistemas reactivos (basados en eventos).

Diferencias entre Diagramas de Estado y Máquinas de Estados Finitos

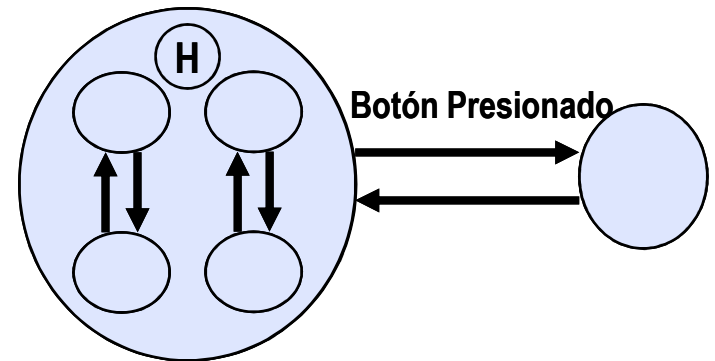
Ambos contienen los mismos conceptos básicos:

- Estados
- Transiciones



Diagramas agregan más conceptos:

- Jerarquía
- Concurrencia
- Paradigma basado en eventos
- Pseudoestados y conectores



Basado en la especificación UML del Diagrama de Estados

Sistemas Reactivos

- Sistemas de comunicaciones
- Protocolos digitales
- Aplicaciones de control
 - Lógica secuencial
 - Procesamiento por lote
 - Respuesta a eventos
 - Control no lineal
- Implementación de interfaz de usuario
- Modelación de sistemas para prototipos virtuales (simulación)

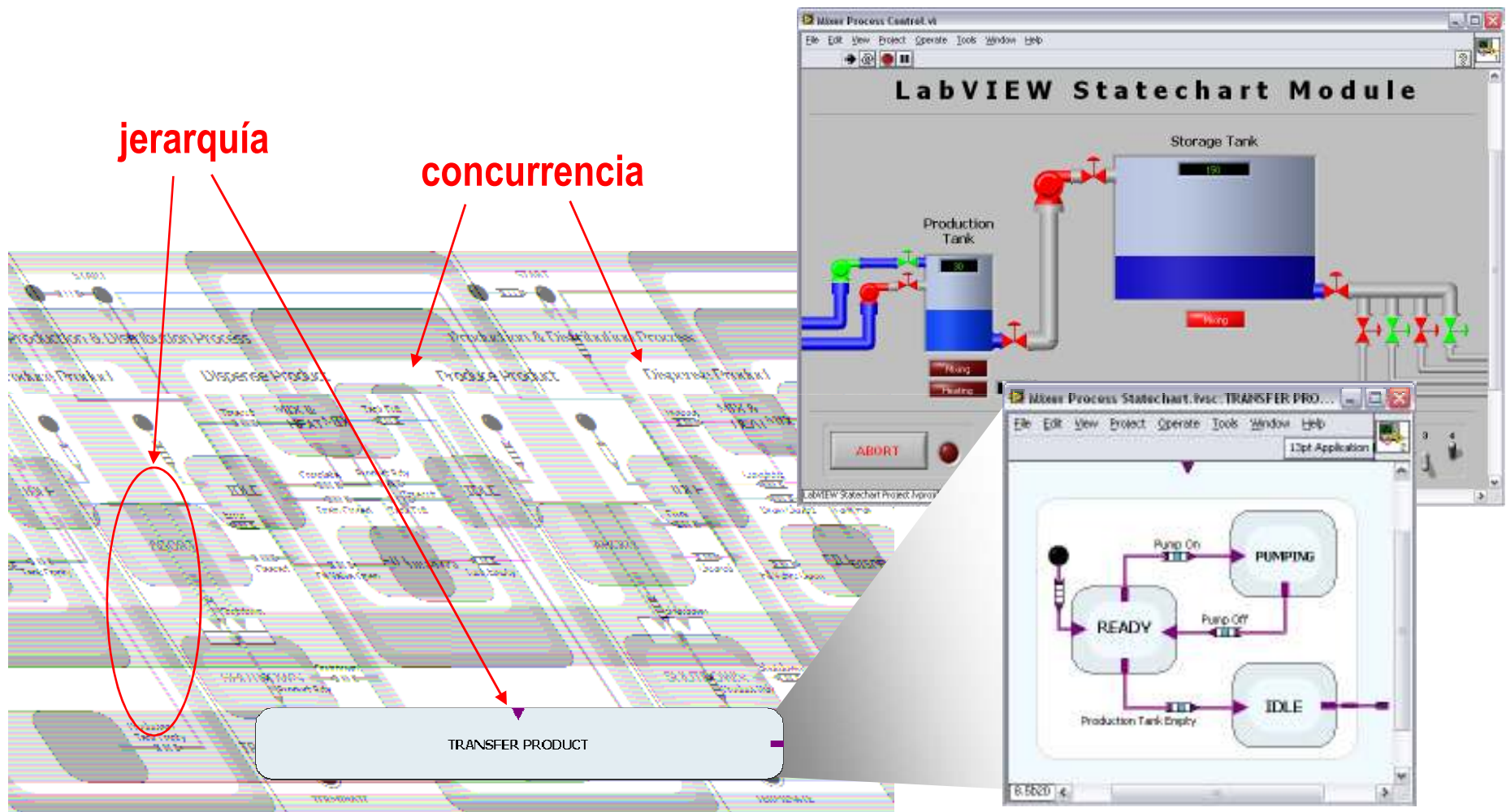


Máquinas de Estado

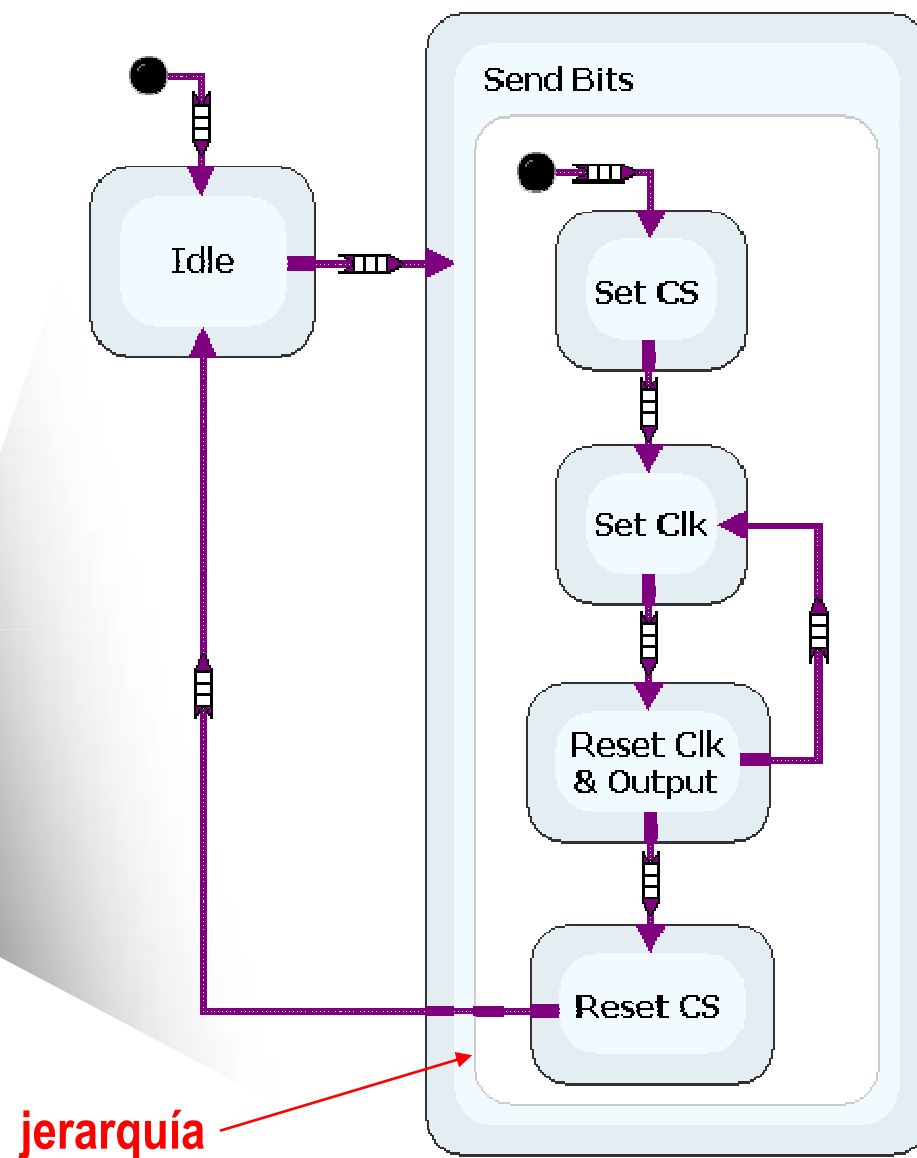
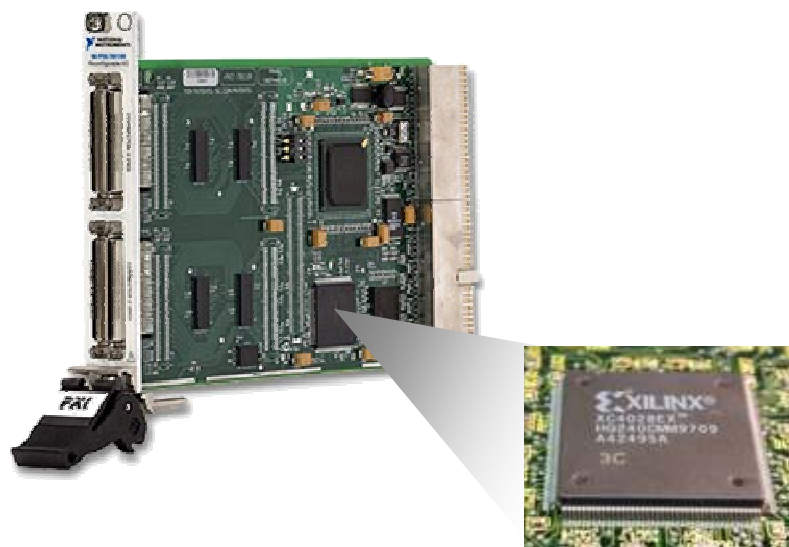
Beneficios de Diagramas de Estados

- Abstracción
 - Semántica simple para representar sistemas complejos
 - Vista a nivel sistema
 - Auto documentación

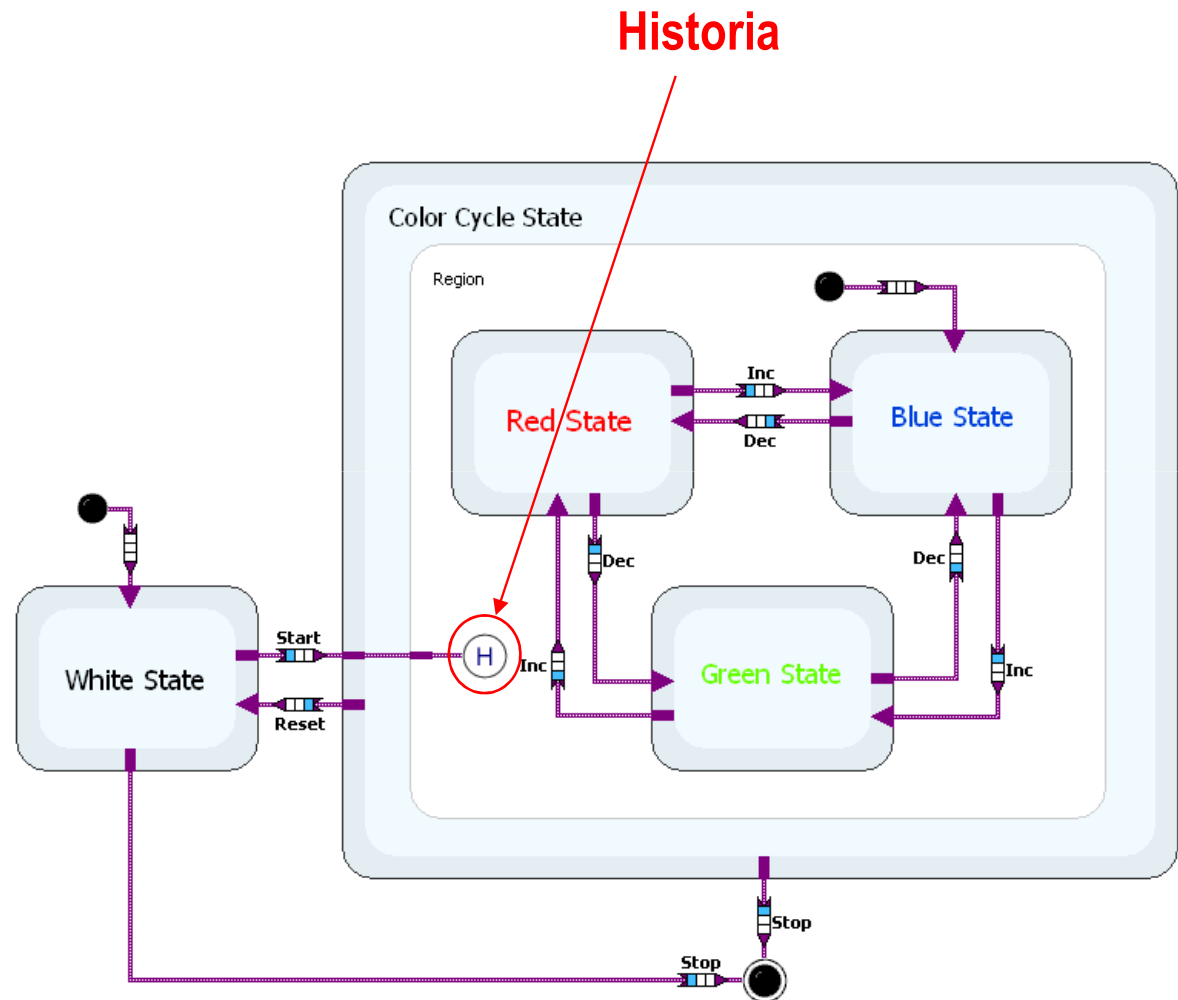
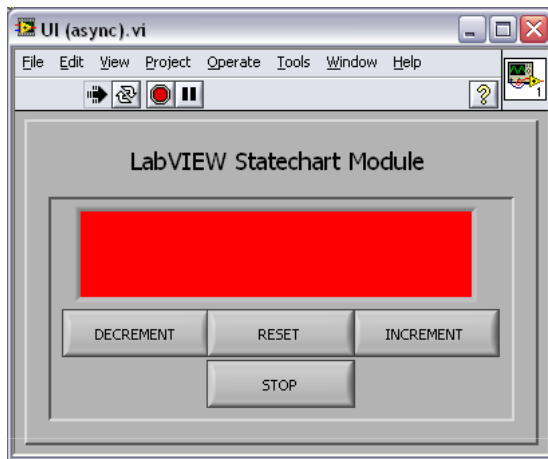
Control de Máquinas y Procesos



Lógica FPGA



Interfaces de Usuario

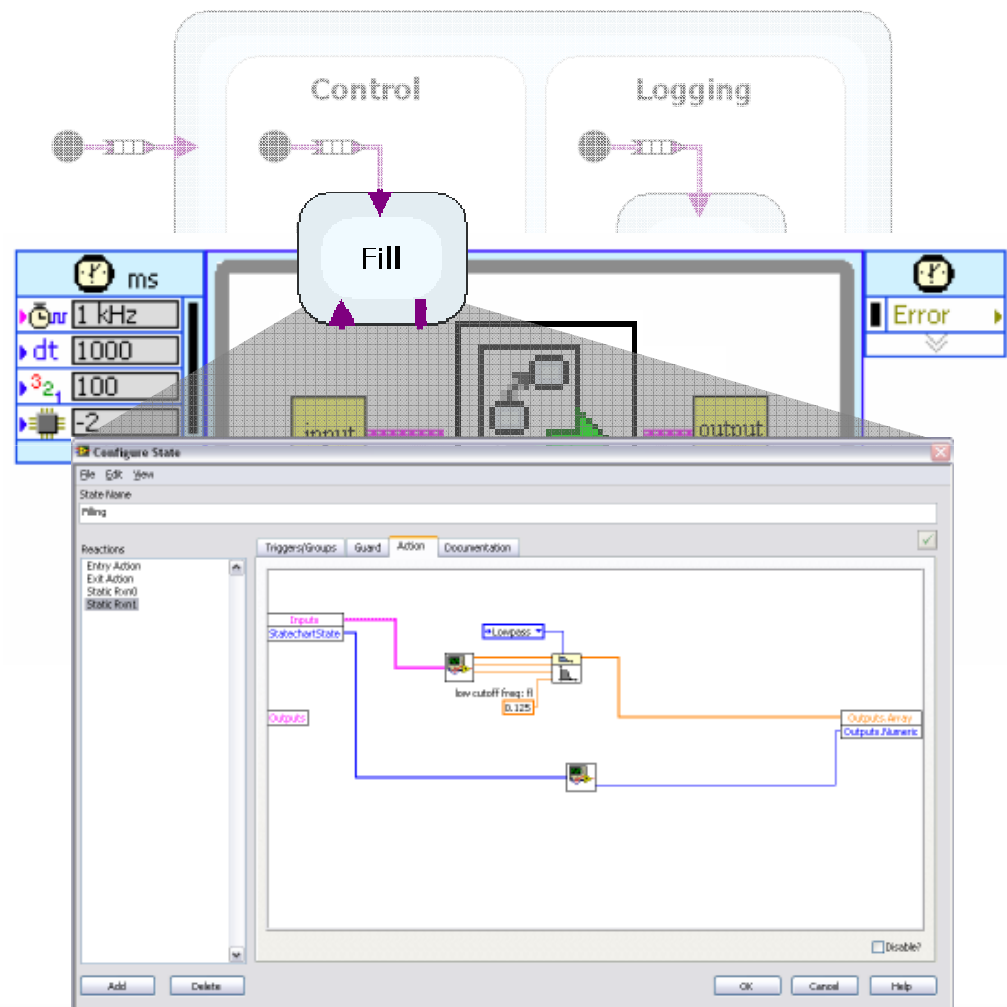


Beneficios de Diagrama de Estados

- Abstracción
 - Semántica simple para representar sistemas complejos
 - Vista a nivel sistema
 - Auto documentación
- Escalabilidad
 - Aplicaciones fácilmente extendibles
 - Plataforma de software abierta
- Generación Automática de Código
 - Tecnología embebida de LabVIEW

Desarrollo con el Módulo LabVIEW Statechart

1. Construir el diagrama de estados
2. Definir transiciones y estados
3. Generar subVI de diagrama de estados
4. Colocar en el diagrama de bloques de LabVIEW



Ejemplo – Ventilador

- Señales de disparo
 - Interruptor de alimentación
 - Interruptor de ventilador
 - Interruptor de luz
- Salidas
 - Luz
 - Velocidad de ventilador

Power		No Power	
Ventilador	Luz	Ventilador	Luz
Alto	Encendido	Apagado	
Medio			
Bajo	Apagado		
Apagado			



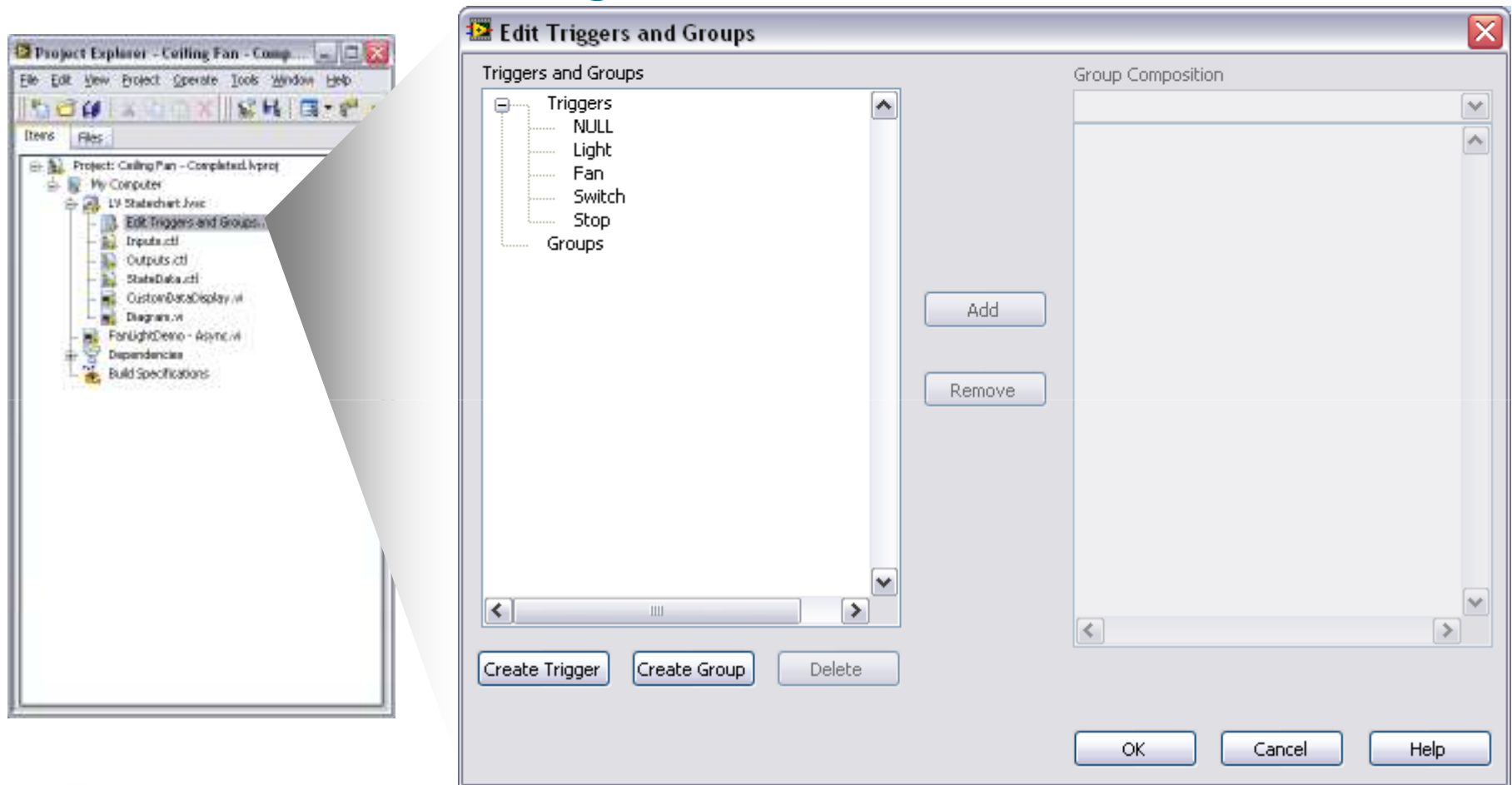
Ejemplo – Ventilador

- Señales de disparo
 - Interruptor de alimentación
 - Interruptor de ventilador
 - Interruptor de luz
- Salidas
 - Luz
 - Velocidad de ventilador
- Datos Internos
 - Velocidad de ventilador

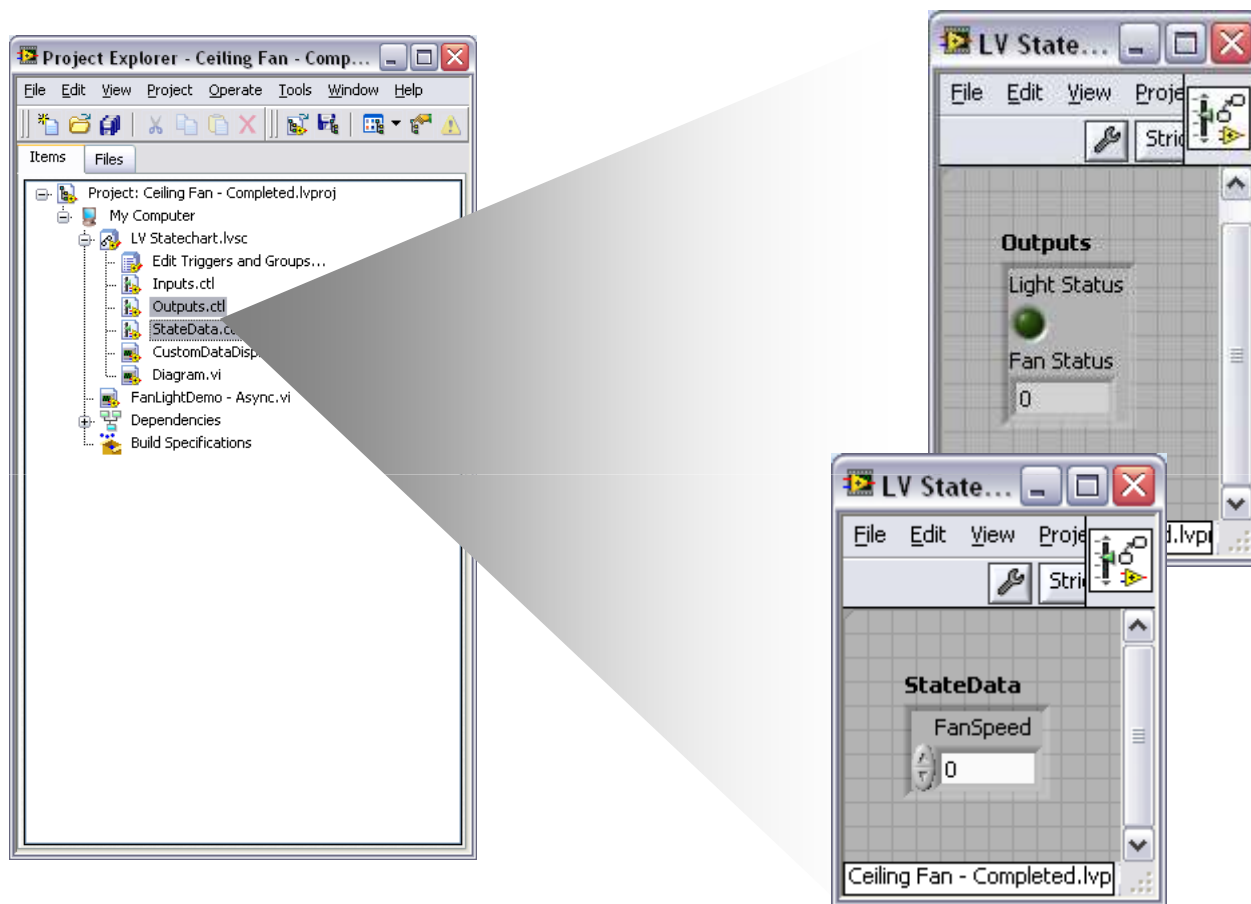
Power		No Power	
Ventilador	Luz	Ventilador	Luz
Encendido	Encendido	Apagado	
Apagado	Apagado		



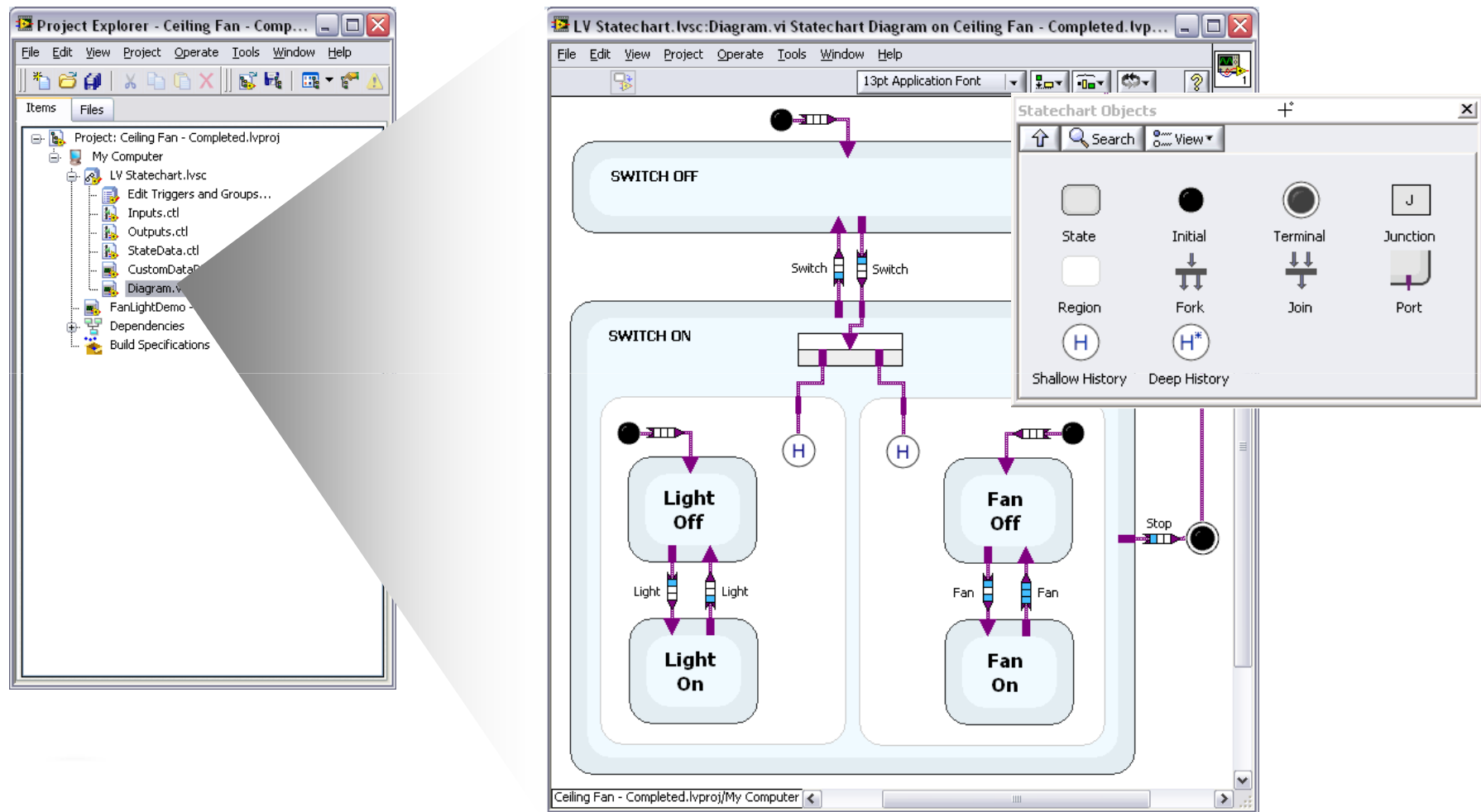
1. Construir Diagrama de Estados



1. Construir Diagrama de Estados



1. Construir Diagrama de Estados



2. Definir Transiciones y Estados

- Cada transición contiene tres componentes
 - **Disparo** – eventos que causan una transición
 - **Guardia** – lógica que puede prevenir una transición
 - **Acción** – que ocurre cuando se hace transición



***Si el timbre suena y
un adulto esta en
casa, abrir la puerta.***

Estado Actual- puerta cerrada
Disparo - timbre de la puerta
Guardia - ¿adulto en casa?
Acción - abrir puerta
Estado Nuevo - puerta abierta

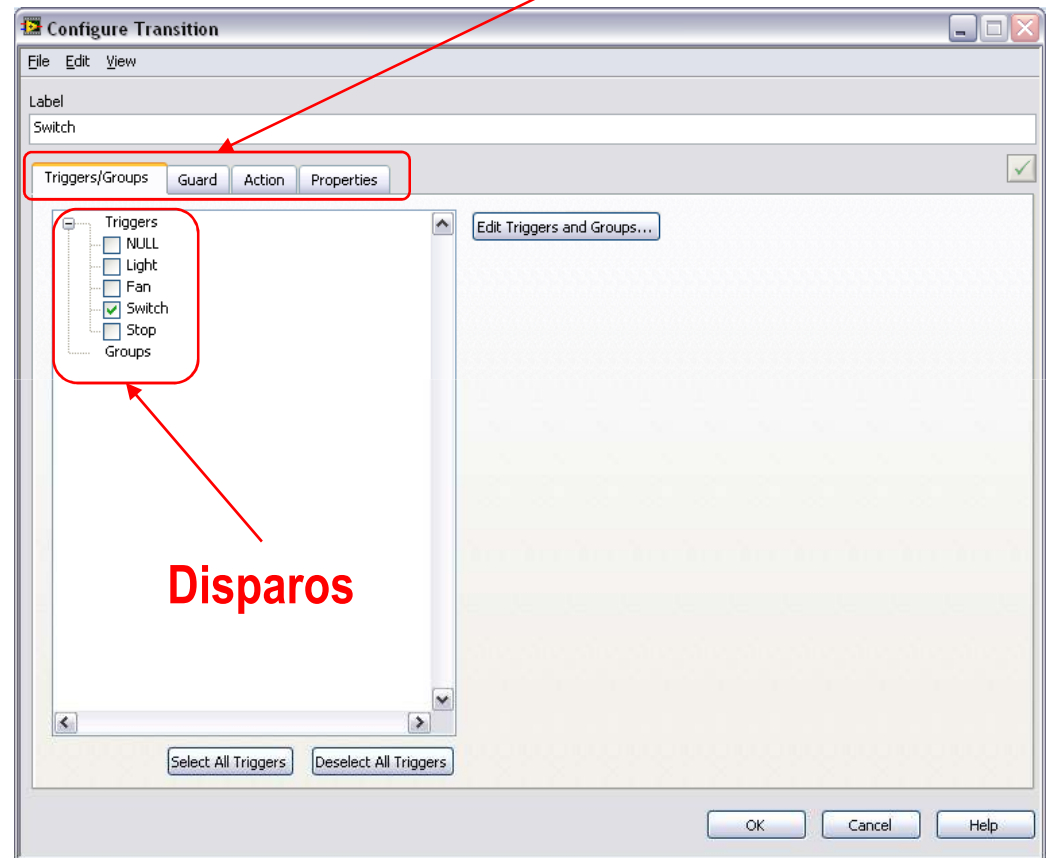
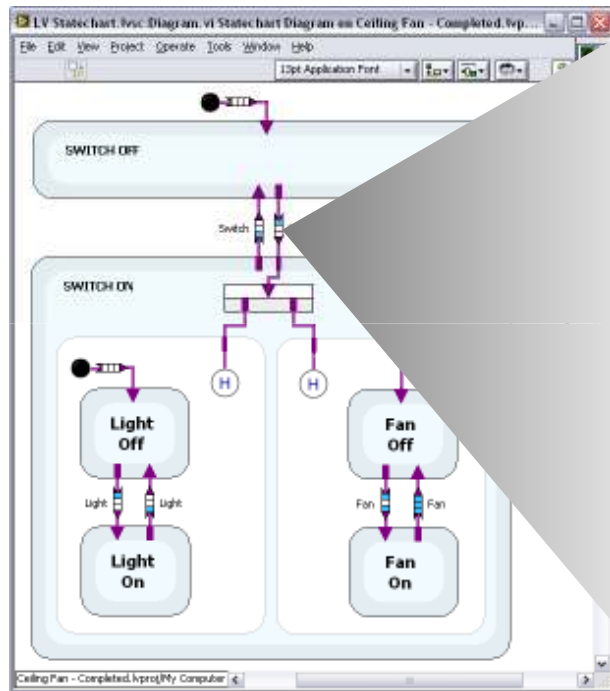
2. Definir Transiciones y Estados

- Cada transición contiene tres componentes
 - **Disparo** – eventos que causan una transición
 - **Guardia** – lógica que puede prevenir una transición
 - **Acción** – que ocurre cuando se hace transición
- Cada estado contiene tres tipos de acciones
 - **Entrada** – lo que sucede cuando llega
 - **Salida** – lo que sucede cuando sale
 - **Estática** – lo que sucede mientras está en él



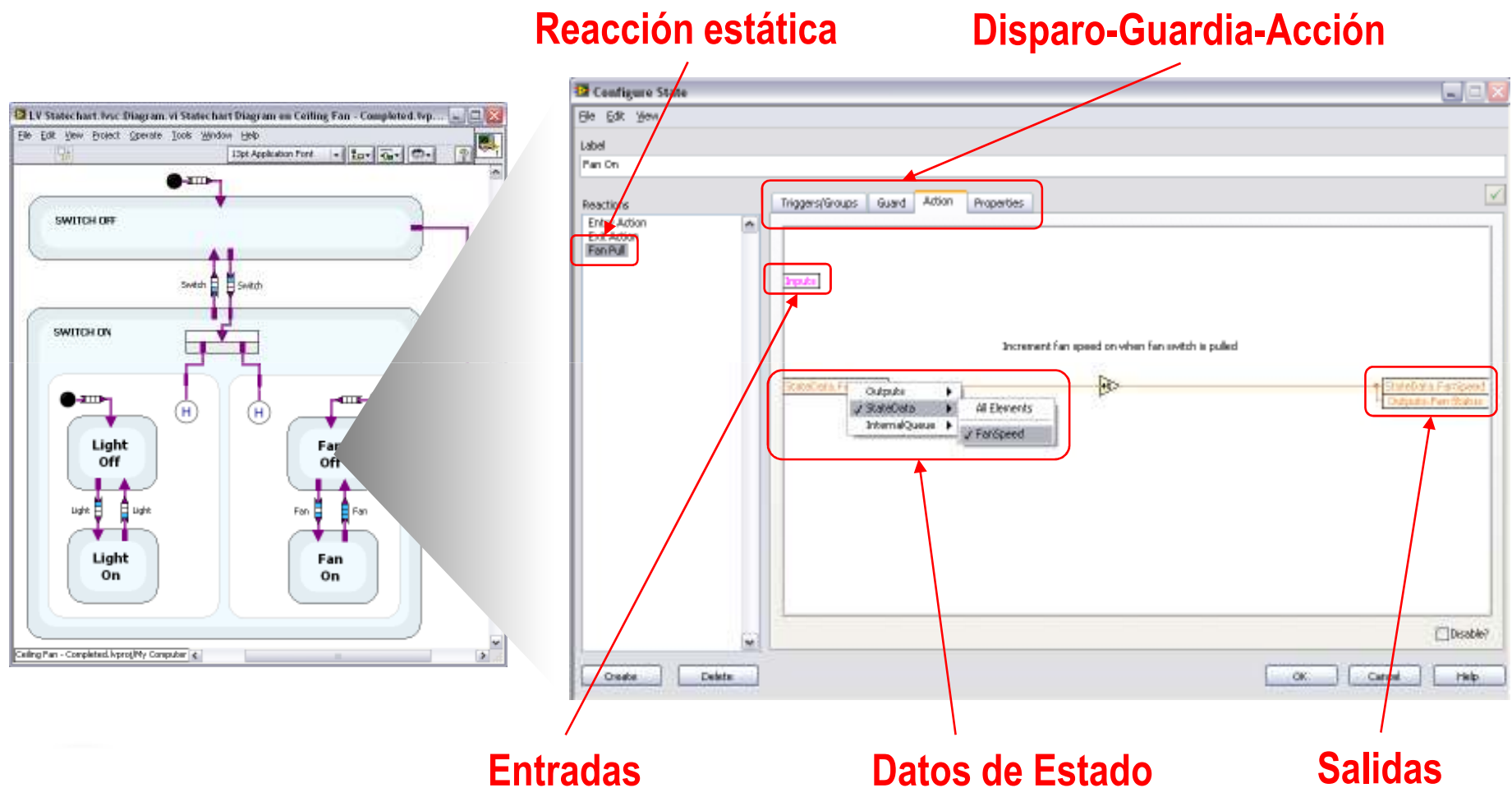
2. Definir Transiciones y Estados

Disparo-Guardia-Acción

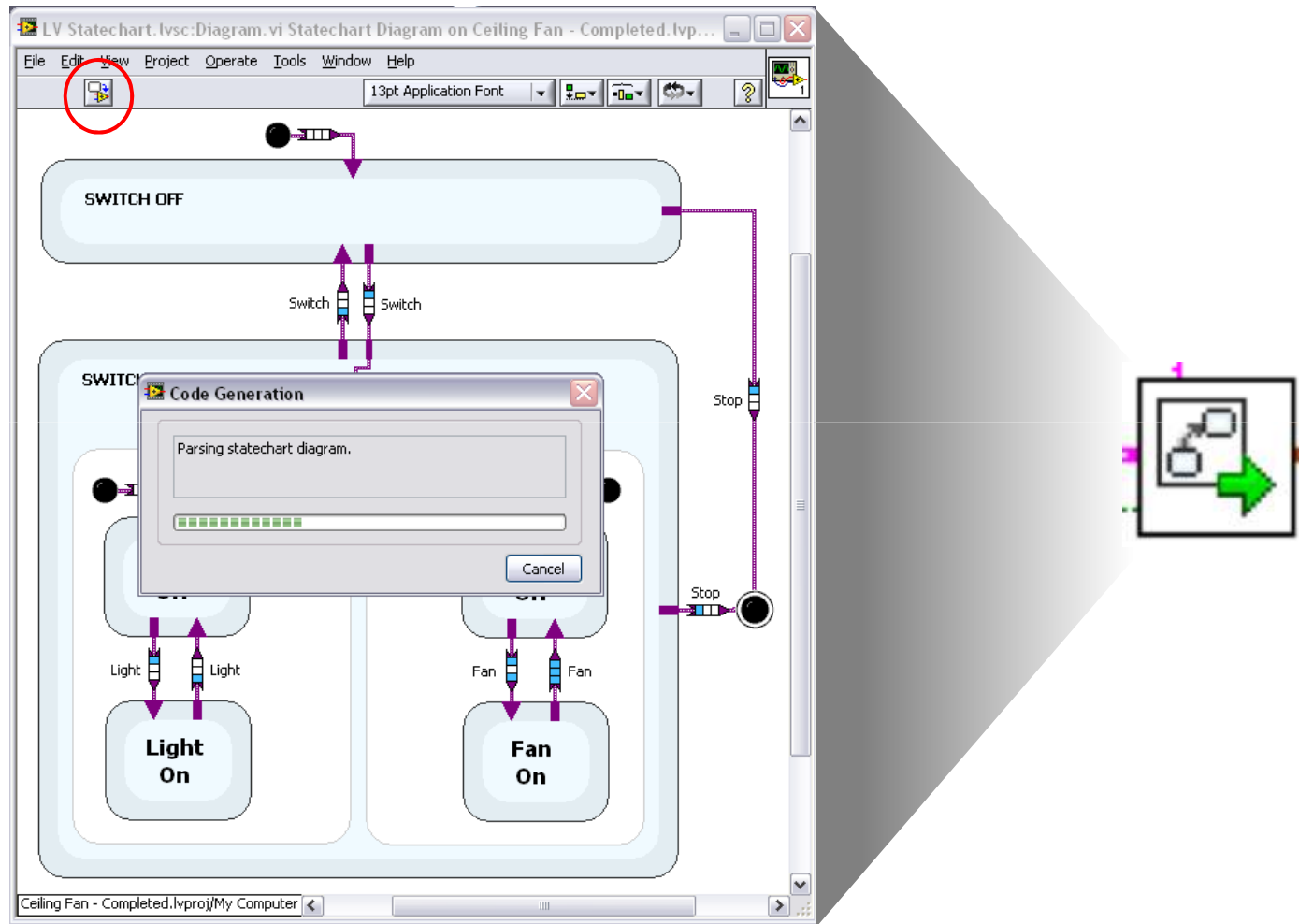


Disparos

2. Definir Transiciones y Estados

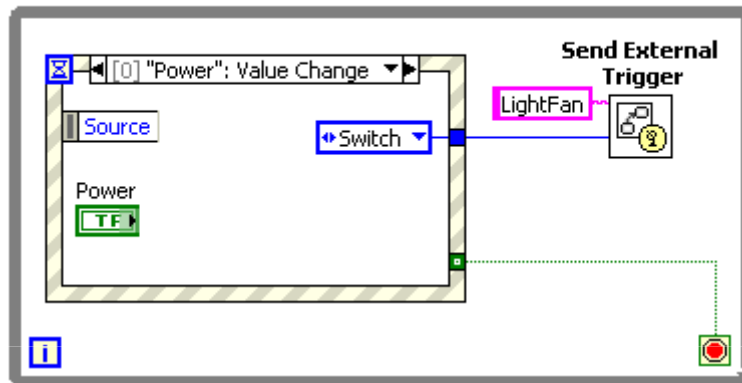


3. Construir SubVI de Diagrama de Estados

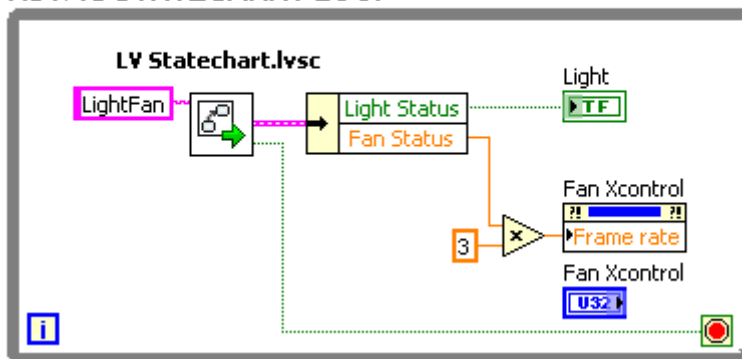


4. Colocar en el Diagrama de Bloques de LabVIEW

EVENT LOOP



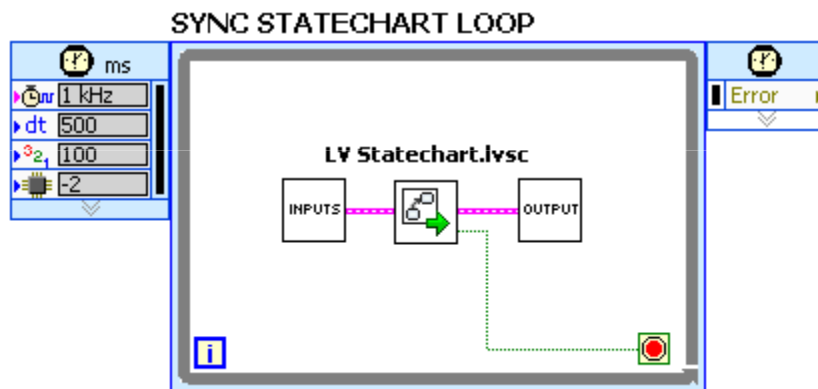
ASync STATECHART LOOP



Uso asíncrono

- Interfaz de Usuario
- Manejo de interrupciones
- Modelado de sistemas basados en eventos

4. Colocar en el Diagrama de Bloques de LabVIEW

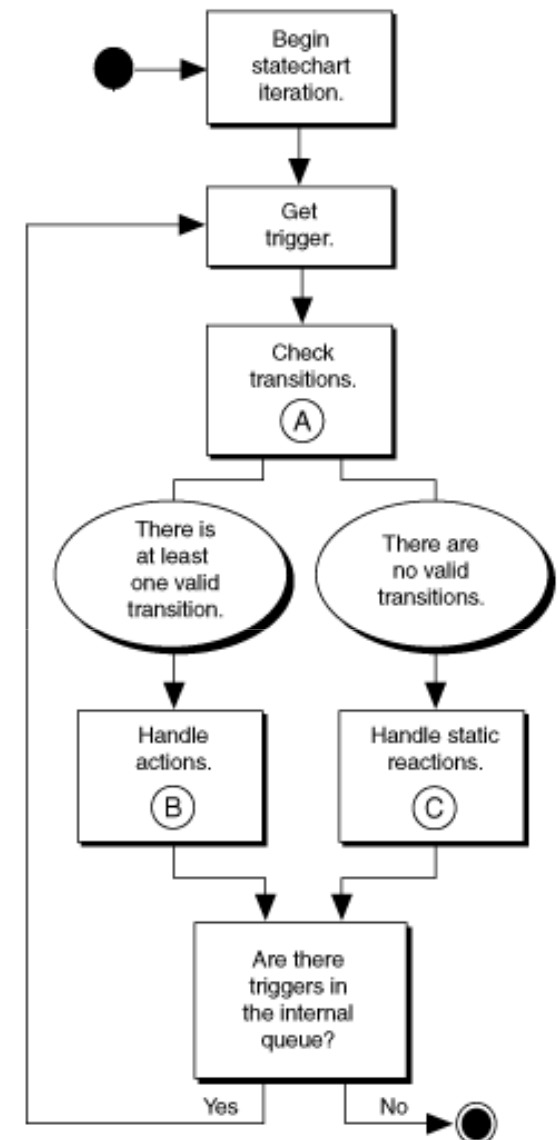


Uso síncrono

- Aplicaciones embebidas
- Protocolos de comunicación
- Implementaciones de control

Ejecución del Diagrama de Estado

- Evaluar la lógica de disparo/guardia para las transiciones que salen del estado actual
- En la primer transición válida:
 - Ejecutar la(s) acción(es) de salida para estado(s) actual(es)
 - Ejecutar la acción de transición
 - Ejecutar las acciones de entrada para todos los estados a los cuales se están cambiando
- Si no hay transiciones válidas:
 - Evaluar la lógica disparo/guardia para todas las reacciones estáticas configuradas para el estado actual
 - Ejecutar el código acción para todas las reacciones válidas



Siguiente paso?

- Visite **ni.com/statechart**
 - Videos con demostraciones
 - Documentos prácticos de diagramas de estado
 - Notas técnicas de diagramas de estado con LabVIEW FPGA
 - Pruebe el módulo de LabVIEW Statechart en línea
- Demostración de un Ingeniero de Campo local