

Sincronización y monitoreo de procesos secuenciales

Juan Pablo Aguilera Alvarez

Instituto Tecnológico de Celaya

fuanitoaguileral@gmail.com

Horacio Orozco Mendoza

Instituto Tecnológico de Celaya

horacio.orozco@itcelaya.edu.mx

Resumen

En la actualidad existen un sin número de diferentes soluciones para la programación de procesos secuenciales y automatización en la industria sin importar el giro de esta, además la sincronización adecuada para cada una de las plataformas de desarrollo depende de un método diferente asociado a cada una de dichas plataformas. En este artículo se tratará de introducir al lector en el método de programación denominado máquinas de estados, y una estructura de sincronización de múltiples sistemas que obedecen a un proceso maestro encargado de dicha sincronización. Se empleará como ejemplo por su simplicidad un proceso conocido por la mayoría de las personas, los semáforos, utilizando el software de desarrollo para instrumentos virtuales LabVIEW. A su vez el lector notará la versatilidad y simplicidad que las herramientas del software nos proporcionan. Además este artículo guiará a quienes están en busca de las herramientas que proporciona LabVIEW para redes industriales utilizando el protocolo de comunicación TCP/IP para monitoreo de los procesos creando HMIs (Interfaces Hombre Maquina) simples.

Palabra(s) Clave(s): HMI, LabVIEW, monitoreo, sincronización.

1. Introducción

En todo proceso de fabricación ya sea en la industria o con fines de investigación el monitoreo de variables de control es importante para la confiabilidad, repetitividad y reproductividad de dicho proceso, es por eso que un sistema de monitoreo confiable y en tiempo real es necesario para la elaboración de productos de buena calidad.

La sincronización correcta de los subprocesos que en conjunto colaboran para la fabricación del producto ayuda a optimizar los tiempos y movimientos del proceso en general.

Un sistema o proceso es secuencial cuando su funcionamiento es la evolución a través de una serie de estados conocidos. Cada estado debe ser único y representar un comportamiento estable y observable del sistema durante un instante de tiempo. La transición de un estado a otro es regida por el cambio en las variables internas o externas al sistema. Estos cambios deben considerarse discretos, es decir del tipo “todo o nada”, “abierto o cerrado”, “0 ó 1”. Un sistema con las características mencionadas puede tratarse como una máquina de estado. [1]

LabVIEW es una herramienta de programación gráfica que originalmente estaba orientada para aplicaciones de control de equipos electrónicos usados en el desarrollo de sistemas de instrumentación, lo que se conoce como instrumentación virtual. Por este motivo los programas creados en LabVIEW se denominan instrumentos virtuales o VI's por sus siglas en ingles. De forma similar a un instrumento real, un programa o instrumento virtual tiene dos componentes, un panel frontal donde estarán sus botones, pantallas, etc. y una circuitería interna. En LabVIEW reciben el nombre de panel frontal y diagrama de bloques respectivamente. [2]

Las variables compartidas (Shared Variables) introducidas en la versión 8 de LabVIEW son una forma más genérica aún de compartir información. Incluyendo la comunicación entre programas que pueden estar en ordenadores distintos. También se suelen utilizar para comunicar dispositivos como controladores (compactRIO o FieldPoint) con una computadora. [2]

Con las variables compartidas el traspaso de información entre los distintos programas es transparente para el programador, que no tiene que preocuparse por el protocolo de transmisión y otros detalles. [2]

Adicionalmente, para equipos móviles se dispone de la herramienta Data Dashboard que permite crear vistas personalizadas y portátiles de las aplicaciones del software LabVIEW de National Instruments. Con esta aplicación se pueden crear paneles de control de datos para mostrar los valores de las variables compartidas publicados en la red y desplegar los servicios de Web de LabVIEW sobre indicadores, tales como gráficos, cuadros de texto, LEDs, etc. [3]

Empleando estas herramientas, se simulará el control de un sistema de tres semáforos posicionados en una avenida de forma consecutiva, los cuales estarán sincronizados de tal manera que si hubiera tráfico de automóviles ocurriría con un flujo constante. Con fines de mostrar la metodología, se implementará una opción para deshabilitar el semáforo intermedio, pudiéndolo activar en el momento que se desee, provocando así una desincronización en los tres semáforos. Un botón extra ayudara a sincronizarlos nuevamente. Este sistema de tres semáforos es suficiente para mostrar los puntos importantes de la metodología y la estructura de programación propuesta.

La metodología de programación propuesta ayuda a simplificar los problemas de sincronización, sin tener que modificar los procesos individuales, facilitando además, la monitorización de las variables y los elementos de control.

2. Estructura del programa

Para mostrar el funcionamiento de las herramientas de LabVIEW, el método de programación por máquinas de estados y la sincronización del sistema, se decidió implementar el control de un proceso conocido, por lo que con fines prácticos se seleccionó la programación para un sistema de semáforos.

Lo primero que se definió es el diagrama de transición de estados, el cual es una herramienta gráfica que consiste en representar a cada estado mediante una burbuja. El cambio o transición entre estados se manifiesta por medio de una flecha unidireccional que conecta a los estados consecutivos. [1]

En la Fig. 1, se muestra el diagrama de transición de estados utilizado para cada uno de los semáforos, tres en total.

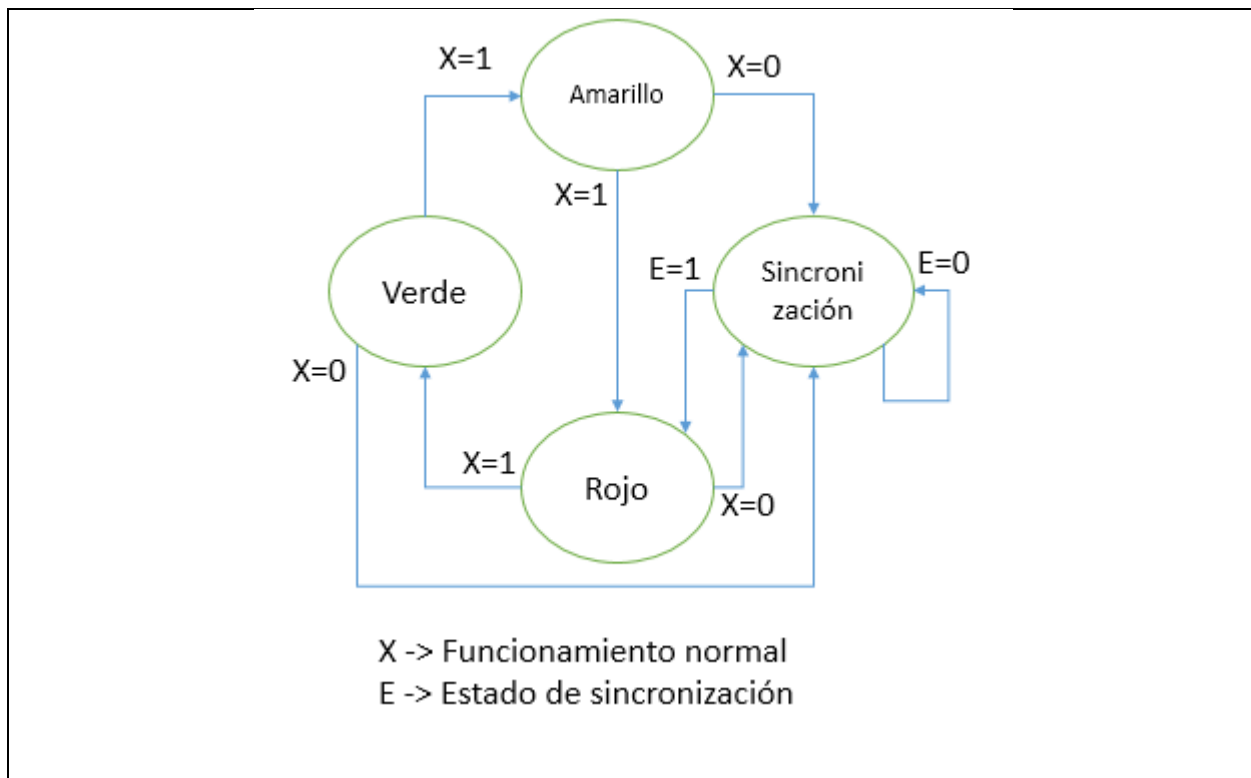


Fig. 1. Diagrama de transición de estados de los semáforos.

Para la sincronización se utilizó otra máquina de estados por separado como se muestra en la Fig. 2.

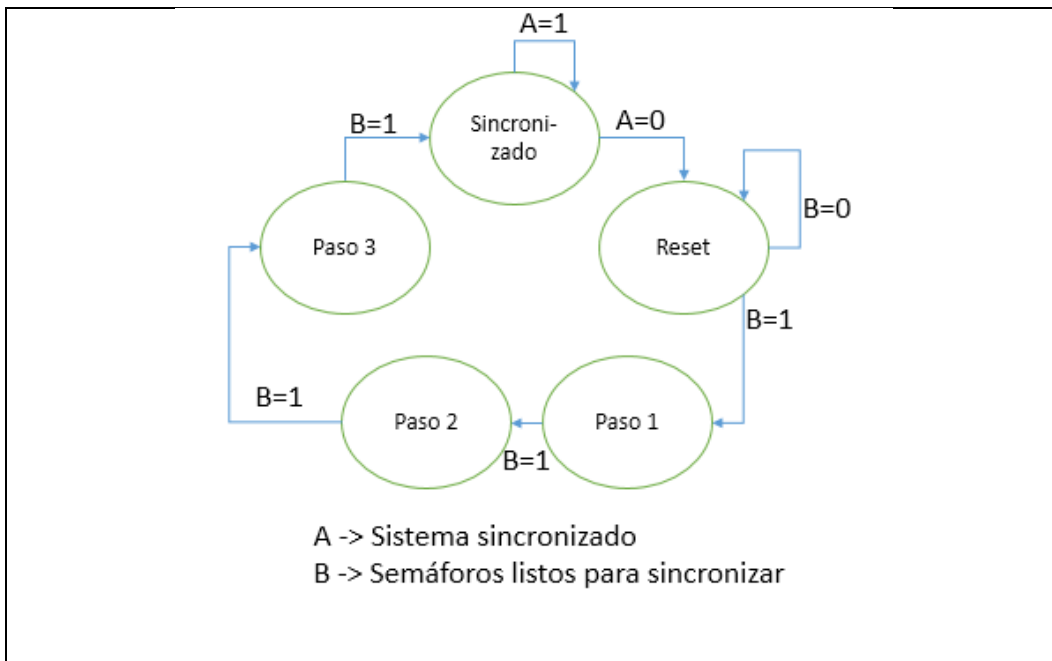


Fig. 2. Diagrama de transición de estados de la máquina de sincronización.

Para el monitoreo remoto del sistema se utilizaron variables compartidas para tener la oportunidad de acceder a ellas desde otra computadora o dispositivos móviles como teléfonos o una tableta electrónica.

En la Fig. 3, se muestran las variables compartidas, definidas en el proyecto.

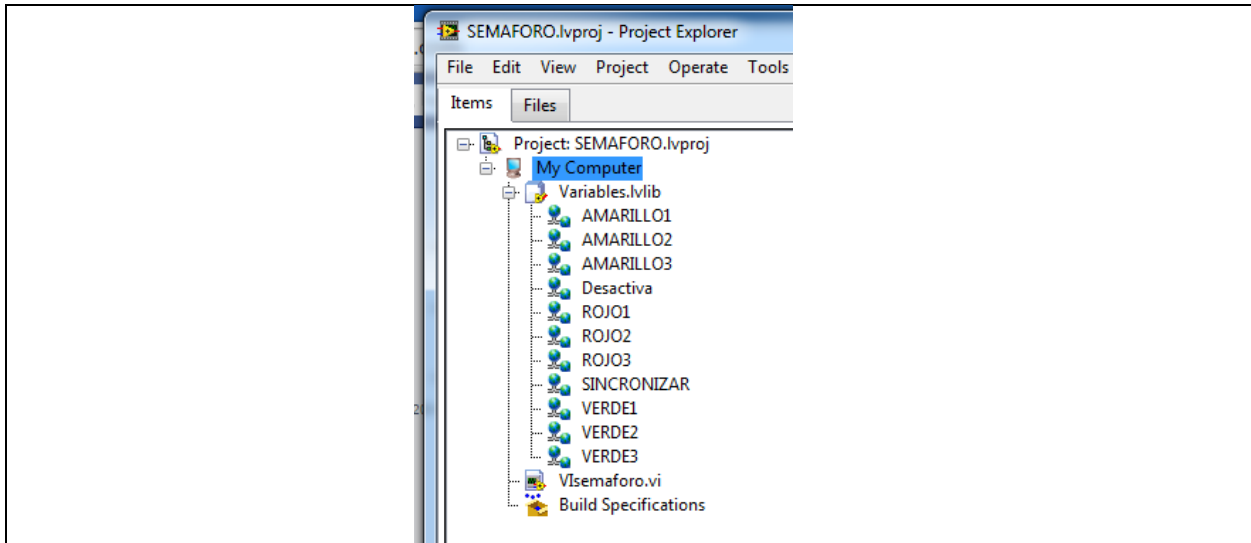


Fig. 3. Variables compartidas en la red.

Para definir las variables compartidas se deben tomar en cuenta estas configuraciones principales: el nombre de la variable, el tipo de dato y el tipo de variable, como se muestra en la Fig. 4.

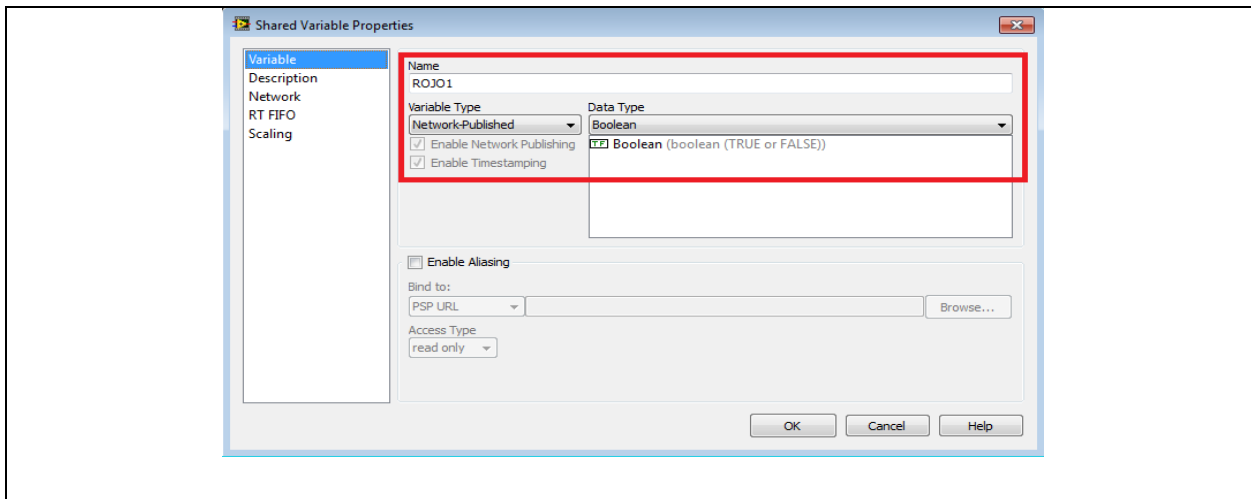


Fig. 4. Definición de variables compartidas.

Como ya se describió anteriormente, un programa (VI) en LabVIEW utiliza dos ventanas para su programación, el panel frontal y el diagrama de bloques. En el panel frontal se

insertaron indicadores booleanos para representar cada una de las lámparas del semáforo y controladores booleanos para los botones de control como se muestra en la Fig. 5.

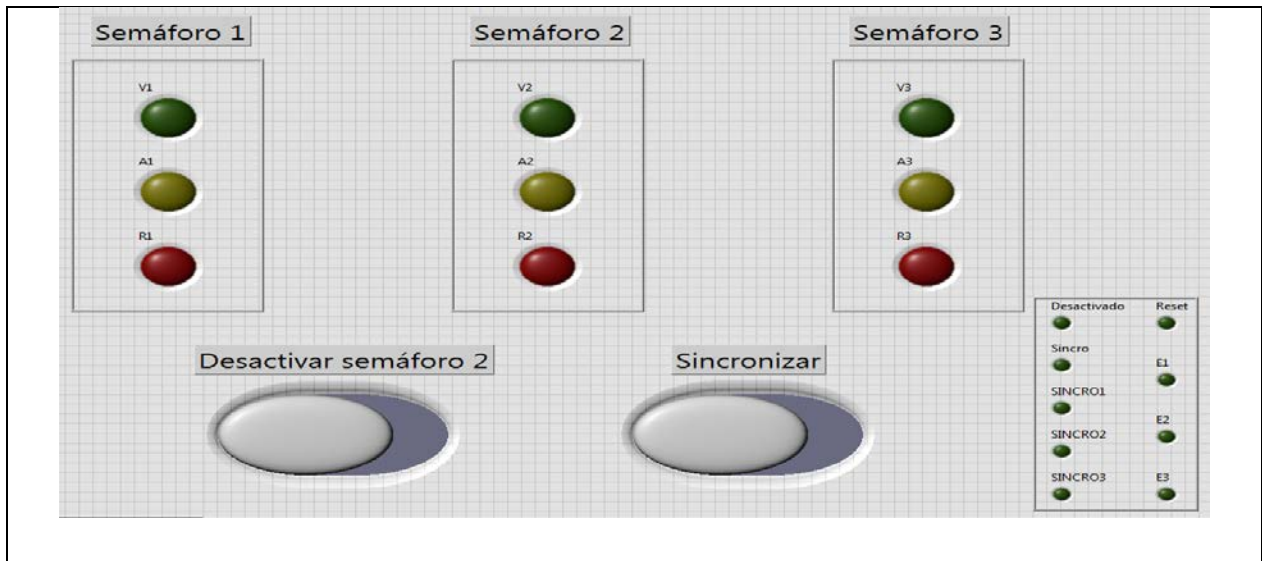


Fig. 5. Panel frontal del programa.

En la Fig. 5, también se pueden observar unos indicadores que sirvieron como registros auxiliares para la programación.

Posteriormente, se procedió a programar las 4 máquinas de estados, tres para los semáforos y una para la sincronización, además se agregó el bucle de monitoreo. En la Fig. 6, se muestra una vista general del diagrama de bloques.

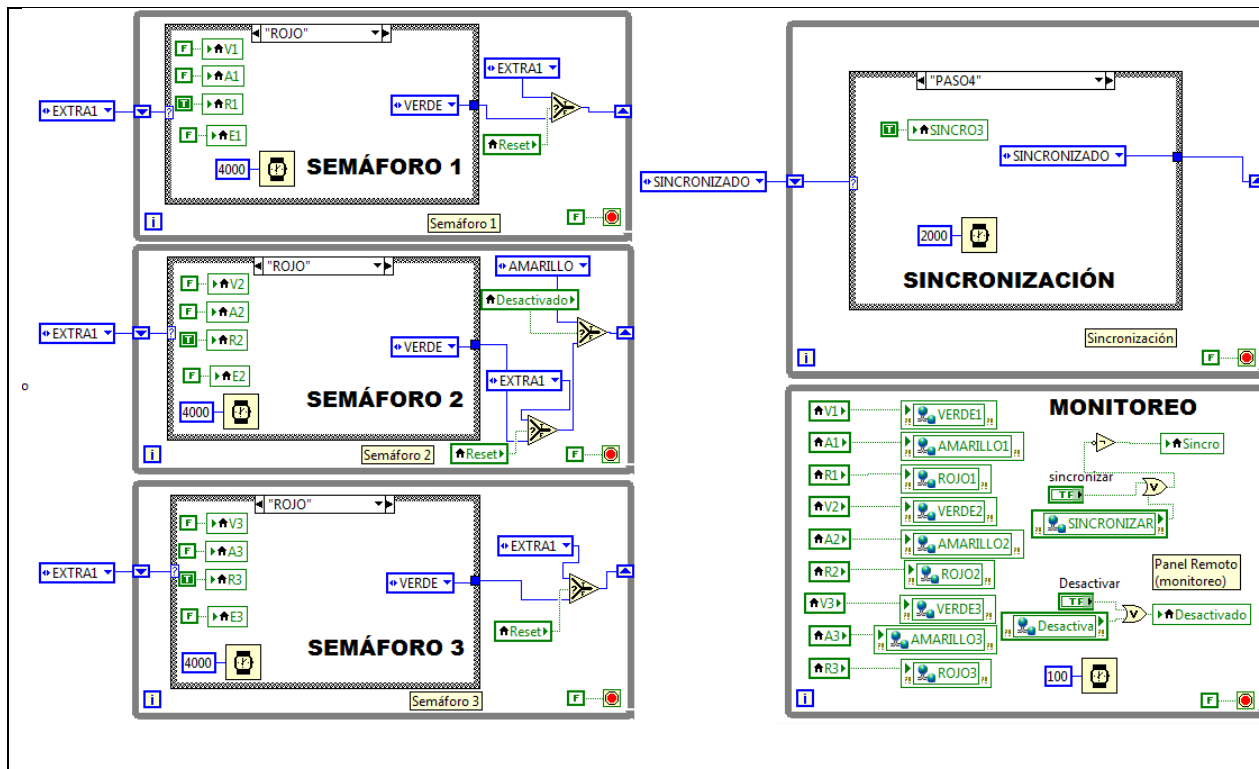


Fig. 6. Diagrama de bloques del programa.

En el diagrama de bloques se puede observar que los tres procesos de los semáforos son independientes, es decir no están conectados uno con otro. El proceso de sincronización es el responsable de iniciar cada semáforo a su debido tiempo.

Para el monitoreo remoto o creación de la interfaz hombre máquina (HMI) se utilizó una tableta electrónica modelo SAMSUNG Galaxy note 8.0 con la "app" NI DataDashboard la cual permite acceder a las variables compartidas del sistema.

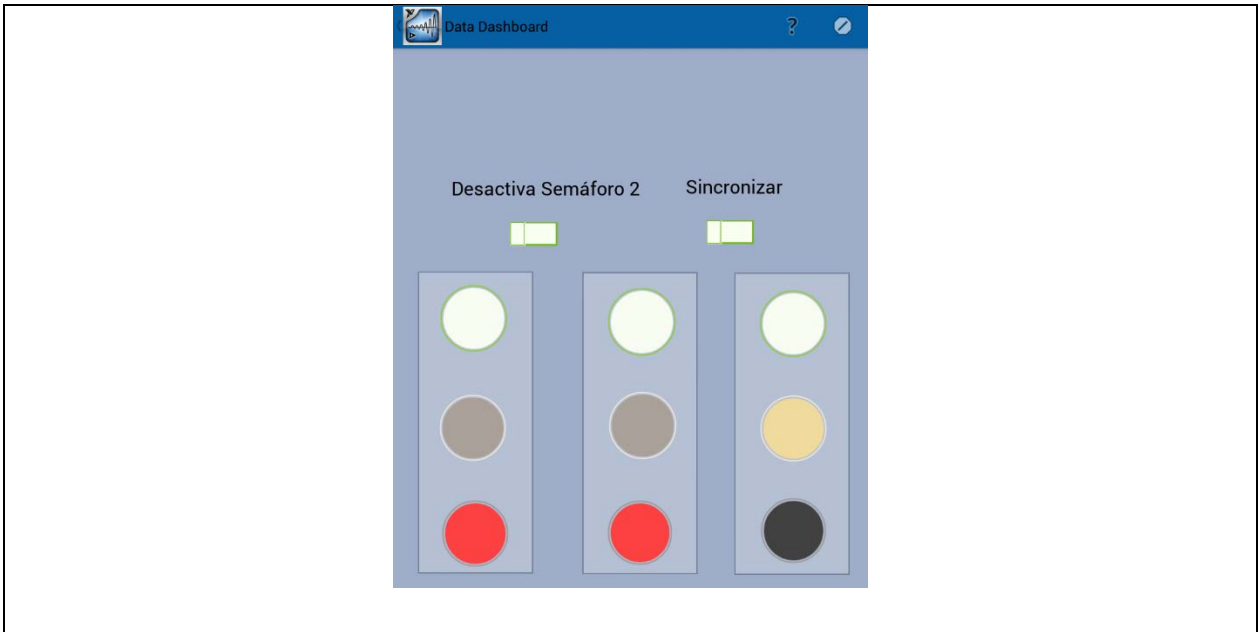


Fig. 7. HMI en tableta electrónica.

En la Fig. 7, se puede ver que en la HMI se usa la misma cantidad de lámparas para los semáforos y los dos botones que utiliza el programa original en LabVIEW.

3. Operación

El funcionamiento del sistema de semáforos se definió a partir de la tabla de estados de sincronización, (ver Tabla 1).

	Semáforo 1	Semáforo 2	Semáforo 3
Paso 1	VERDE	ROJO	ROJO
Paso 2	VERDE	VERDE	ROJO
Paso 3	AMARILLO	VERDE	VERDE
Paso 4	ROJO	AMARILLO	VERDE
Paso 5	ROJO	ROJO	AMARILLO

Tabla 1. Tabla de estados de sincronización.

Al iniciar el programa, automáticamente los semáforos se sincronizarán y empezarán a comportarse como se muestra en la Tabla 1, por lo que para sacarlos de sincronía y observar el funcionamiento correcto de la programación se integró al sistema un botón para detener el semáforo central.

Si el semáforo central está detenido, este encenderá únicamente la lámpara amarilla y los semáforos 1 y 3 permanecerán funcionando normalmente. En la Fig. 8, se muestra un ejemplo.

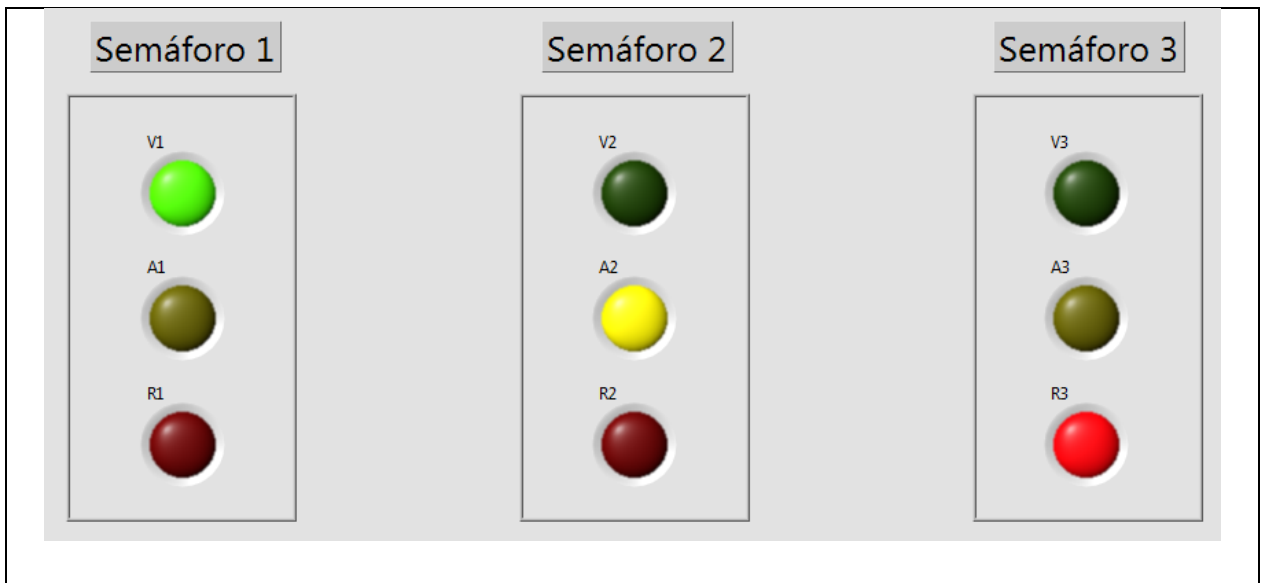


Fig. 8. Semáforo 2 detenido.

Al reactivar manualmente el funcionamiento del semáforo 2 ya no estará en sincronía como lo estuvo al principio por lo que con el botón de sincronizar, el sistema se reiniciará y tenderá a seguir el funcionamiento de la tabla de sincronización nuevamente.

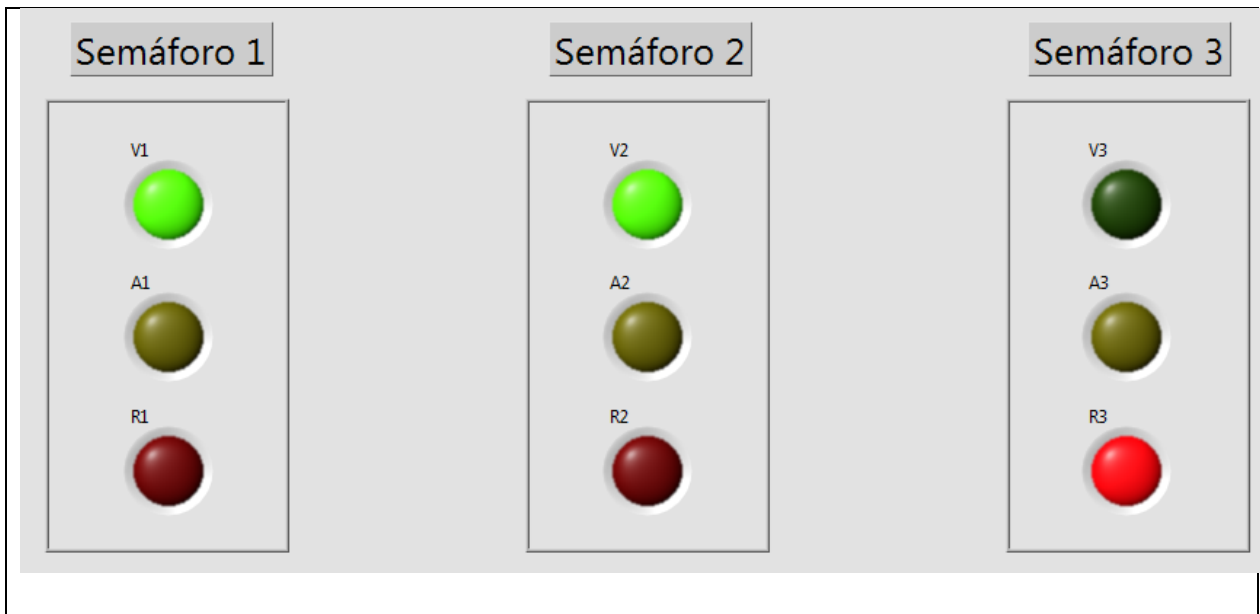


Fig. 9. Semáforos sincronizados.

En la Fig. 9, se muestran los semáforos después de apretar el botón de sincronizar, esta ya es una configuración válida para la Tabla 1.

4. Resultados

El resultado de este trabajo es una estructura de programación capaz de aplicarse en un conjunto de subsistemas secuenciales que necesiten una integración sincronizada para realizar un proceso.

La implementación de interfaces hombre máquina (HMI) inalámbricas de bajo costo con alta robustez, fácil implementación y confiabilidad para el monitoreo de procesos y control de procesos.

Se comprobó que esta estructura implementada en LabVIEW es confiable y fácil de modificar, según se requiera.

Si usted quiere conocer más acerca del proceso de sincronización propuesto, puede descargar los archivos del proyecto a través de la siguiente liga: <https://www.dropbox.com/s/i3z9m3z15ofwgf0/SEMAFORO.zip?dl=0>

5. Conclusiones

Esta estructura de programación ayudará a simplificar los problemas de sincronización sin tener que modificar los subprocesos directamente.

La estructura de programación propuesta facilitara la monitorización de las variables y los elementos de control al utilizar la herramienta de variables compartidas de LabVIEW.

Aun cuando existen muchos métodos de programación en distintos lenguajes y entornos, esta estructura ofrece facilidad en la comunicación de paneles remotos y al mismo tiempo llevar un orden estructurado para la resolución de problemas.

La propuesta de programación descrita en este artículo fue empleada por los autores en un proceso industrial donde se necesitaba la sincronización de varios subsistemas con secuencias diferentes entre ellos.

6. Referencias

- [1] KAIRIYAMA, J. C.; FERNÁNDEZ, G. A.; KURTZ, V. H. Sistema de automatización, supervisión y control del Aprovechamiento Hidroeléctrico Arroyo El Tigre. La máquina de estado como herramienta de diseño.
- [2] VIZCAÍNO, José Rafael Lajara; SEBASTIÁ, José Pelegrí. LabView: entorno gráfico de programación. Marcombo, 2011.

- [3] <http://www.ni.com/tutorial/13757/es/>, Introducción al panel de control de datos "Data Dashboard" para LabVIEW. Septiembre, 2015.