

Array

Un array es una colección de datos todos ellos del mismo tipo. Puede tener una o más dimensiones y hasta 2 elementos por dimensión, según la memoria disponible. Un array puede ser de cualquier tipo excepto otro array, chart o graph. Se accede a cada elemento de un array mediante un índice, el cual es cero base, es decir, va de 0 a N-1 donde N es el número de elementos.

La creación de arrays de control o indicadores en el panel frontal se hace mediante la combinación del icono array shell de la paleta Array & Cluster del menú controls, con un objeto de datos, que puede ser numérico, booleano o string.

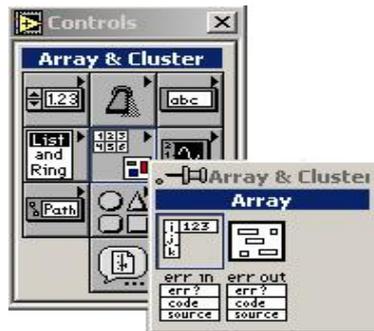


Figura 1.1

En la figura que sigue se presenta un array shell vacío. Para crear un array hemos de llevar un objeto dentro de la ventana de objetos o situarlo directamente usando el menú pop-up.



Figura 1.2

Un array de n-dimensiones necesita de n índices (cero-base) para localizar un elemento. En la figura siguiente hay un array de N columnas por M filas, conteniéndolo N veces M elementos.

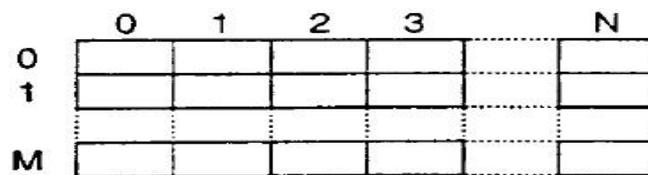


Figura 1.3

Se pueden añadir dimensiones a un array control o indicador de dos maneras: Desplegando el menú de index display (visualizador del índice) y a continuación, escogiendo la opción Add Dimensión (añadir dimensión) o situando el cursor sobre el index display y arrastrando hacia abajo tantas dimensiones como queramos.

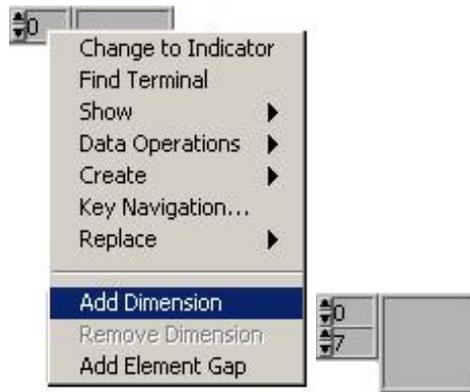


Figura 1.4

Podemos indexar e introducir elementos dentro de los arrays en bordes de las estructuras For Loop y White Loop de manera automática. Esto es lo que se llama auto-indexing (autoindexado). La figura inferior presenta un ejemplo de ello; cada iteración crea el siguiente elemento del array; una vez que se ha completado, el array pasa al indicador. Observar que el cable se hace más grueso al cambiar a array.

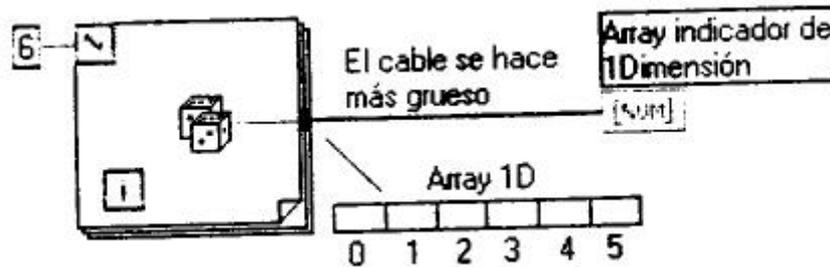


Figura 1.5

Para anular el auto-indexing hemos de desplegar el menú sobre el túnel de salida (el cuadrado negro) y escoger la opción Disable Indexing (deshabilitar indexado). Al desactivarlos, solo el último valor pasará a través del túnel.

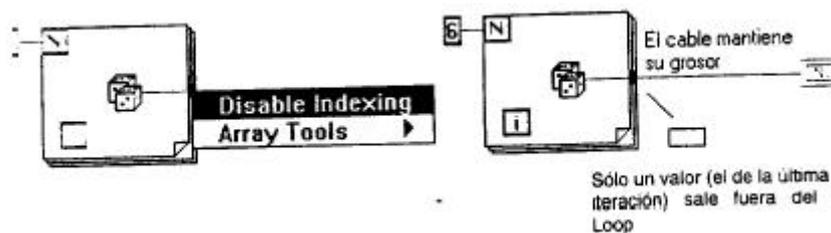


Figura 1.6

Si queremos introducir elementos en un array de dos dimensiones podríamos usar un For Loop dentro de otro For Loop. El interior crea los elementos de columna, mientras que el exterior crea los elementos de fila.

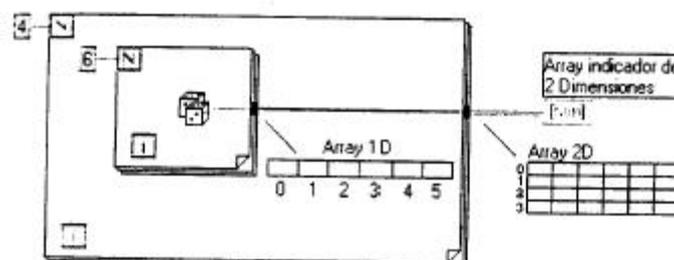


Figura 1.7

Si se habilita el auto-indexing sobre el túnel de entrada de un For Loop, LabVIEW toma el tamaño del array como el número de iteraciones; por tanto, no es necesario conectar ningún valor en N. Si lo hubiese, o diversos arrays tuvieran auto-indexing de entrada, se tomaría como número de iteraciones el de menor tamaño.

Las funciones aritméticas (sumar, restar, multiplicar, dividir, etc.) son polimórficas, es decir sus entradas pueden ser de diferentes tipos. Por ejemplo, podemos sumar un escalar con un array o dos arrays juntos.

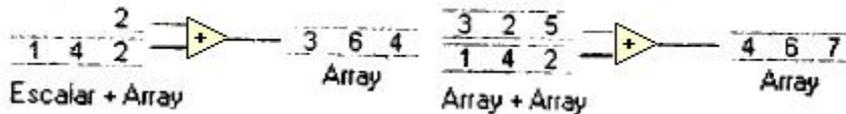


Figura 1.8

Funciones

LabVIEW tiene la paleta Array & Cluster (Array en la versión 4.0) del menú Functions un gran número de funciones para manipular arrays, Algunas de las más comunes se describen a continuación:

Array Size (tamaño del array): Da el numero de elementos del array.



Figura 1.9

Build Array (construir array): Concatena arrays o añade elementos extras a un array. La función aparece como cuando se pone en el diagrama de bloques. Podemos redimensionarla para incrementar el número de entradas.



Figura 1.10

Array Subset (Subarray de un array): Devuelve una parte de un array a partir de un índice y longitud determinados.



Figura 1.11

Index Array (indexar array): Accede a un elemento de un array.



Figura1.12

Esta función también se puede utilizar para separar una o más dimensiones de un array bidimensional y crear un subarray del original. Para ello añadir una nueva dimensión y seleccionar el comando Disable Indexing del menú pop-up del terminal de índice:

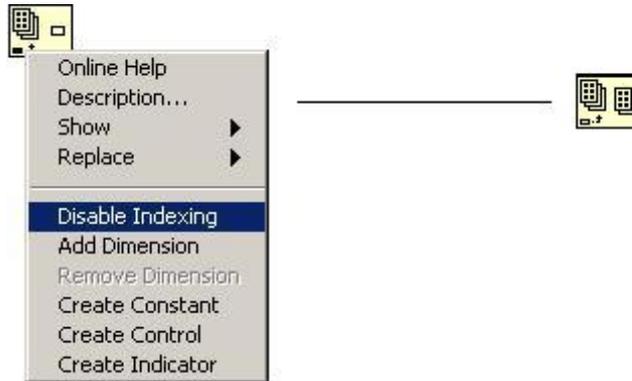


Figura 1.13

Podemos extraer subarrays de cualquier combinación de dimensiones. El siguiente ejemplo muestra como extraer una columna o fila de array a partir de un array de dos dimensiones (figura 1.14).

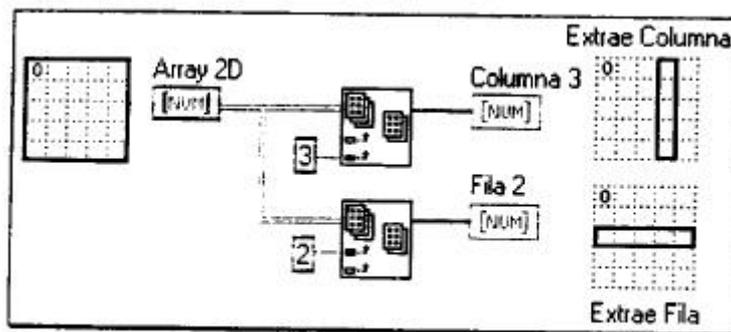
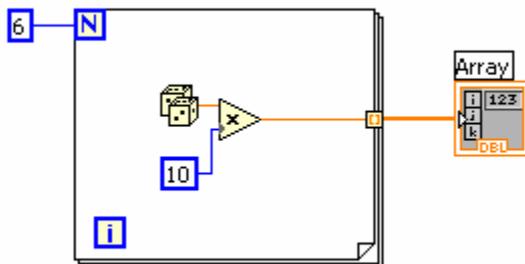


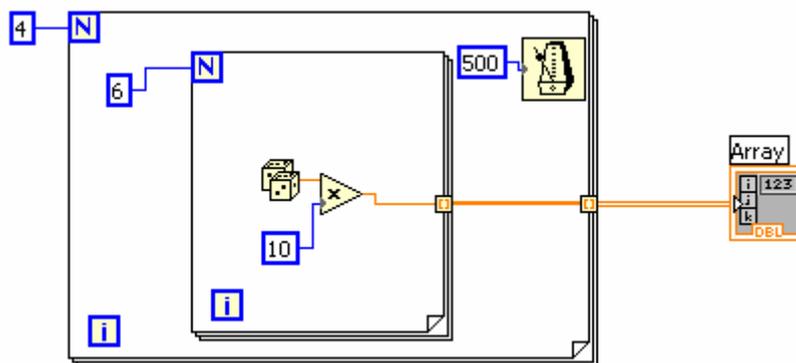
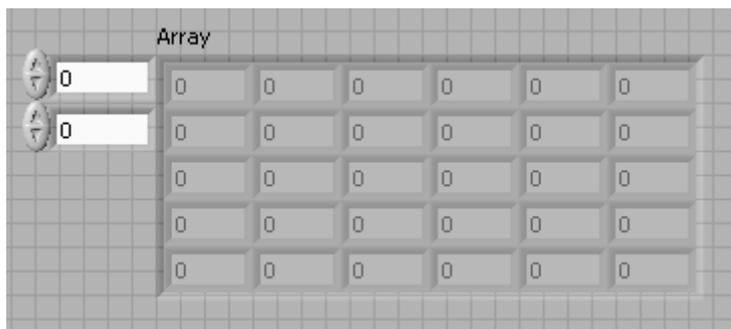
Figura 6.1 Extracción de una fila o columna del array

Figura 1.14

Ejemplo Array 1D



Ejemplo Array 2D



Cluster en LabVIEW

Un cluster en LabVIEW es una colección ordenada de uno o más elementos. A diferencia de los arrays, los clusters pueden contener cualquier combinación de tipos de datos. Se accede a sus elementos Unbundling (literalmente “desenvolviéndolos”) todos a la vez, en lugar de indexarlos uno a uno. Otra diferencia con los arrays es que los clusters tienen un tamaño fijo. Al igual que en el caso de los arrays, los clusters no pueden contener combinación de indicadores y controles.

Creamos un Cluster de indicadores o controles poniendo cualquier combinación de booleanos, strings, charts, graph, escalares, arrays o, incluso, otros clusters dentro de un cluster Shell, al cual se accede a través del menú Controles del panel frontal. Un cluster shell nuevo tiene un borde redimensional y una etiqueta opcional.

Cuando hacemos pop-up dentro del área vacía del cluster shell aparece el menú Controls. Podemos situar cualquier elemento de este menú o bien arrastrarlo desde cualquier punto del panel frontal. El cluster pasa a ser indicador o control dependiendo del primer elemento que situemos en su interior. Posteriormente podemos utilizar las opciones Change to Control (cambiar a control) o Change to Indicator (cambiar a indicador) para cambiar todos los elementos a la opción deseada.

A continuación se presentan las funciones mas comunes para los clusters: Unbundle (separar): Descompone un cluster en sus elementos individuales.

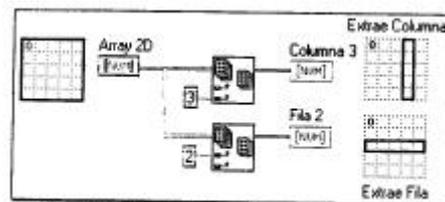


Figura 1.1

Bundle (unir). Une todas las entradas individuales en único cluster o cambia los valores de los componentes conectados.

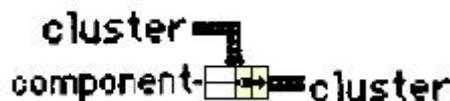


Figura 1.2

Unbundle by Name (separa por nombre): Devuelve los elementos del cluster cuyos nombres especificamos. Seleccionamos los elementos a los que queremos acceder haciendo pop-up sobre el nombre de los terminales de salida y seleccionando un nombre de la lista de elementos dentro del cluster.

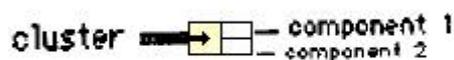


Figura 1.3

Bundle by Name (unir por nombre): Reemplaza componentes en un cluster ya existente. Siempre hemos de conectar las entradas del cluster. Si estamos creando un cluster como

indicador podemos conectar una variable local de ese indicador. Si el cluster funcionara como control en un SubVI, podemos hacer una copia de ese control sobre el panel frontal del VI y conectar los controles a la entrada el cluster.

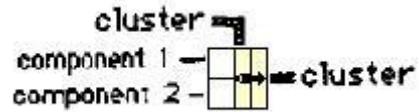
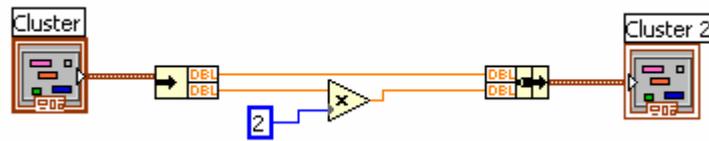
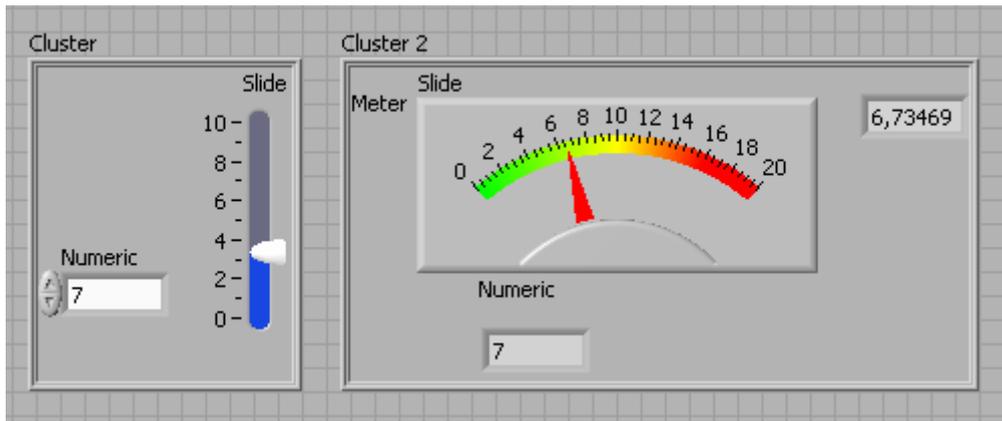


Figura 1.4

Ejemplo, operaciones con Clusters.



Controles e Indicadores de String (cadenas de caracteres)

En este apartado se describe como usar los controles e indicadores de string (cadenas de caracteres). Se puede acceder a estos objetos a través de la paleta String & Table del menú Controls.

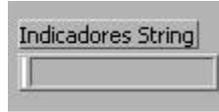


Figura 1.1

Un string es una colección de caracteres ASCII. No sólo se usan como mensajes de texto. En el control de instrumentos pasamos los datos numéricos como cadenas de caracteres o strings. A continuación convertimos esas cadenas en números. Así mismo se necesitan strings para almacenar datos numéricos en disco.

En la siguiente figura puede verse un control y un indicador de String:



Figura 1.2

Se puede introducir o cambiar texto en un control string con las herramientas operating o labeling. El texto nuevo o cambiando no pasa al diagrama hasta que se pulsa Enter del teclado numérico, se hace clic en cualquier otra parte del panel o se pulsa el icono Enter de la barra de herramientas. Pulsando la tecla Enter del teclado alfanumérico se consigue un salto de línea en el string.

Para entrar una tabulación en el string se ha de seleccionar la opción/Codes del menú pop-up del string y escribir/t. En la tabla siguiente se listan todos los códigos de los caracteres no-imprimibles que podemos usar con los strings.

- \b Backspace
- \f Formfeed
- \n New line
- \r Return
- \t Tab
- \s Space
- \\ Backslash

Si necesitamos disminuir el espacio que ocupa un string en el panel frontal, podemos usar la opción Show Scrollbar (mostrar barra de desplazamiento) de su menú pop-up.

LabVIEW presenta un gran número de funciones para manipular strings. Estas funciones están disponibles desde la paleta String del menú Functions (figura 1.3)

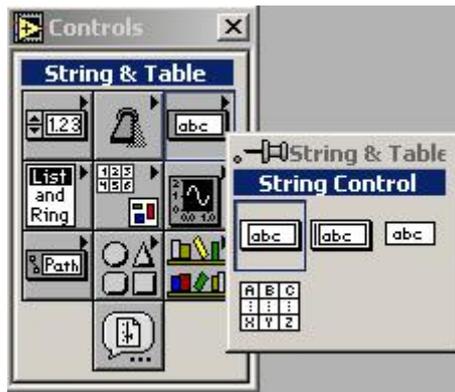
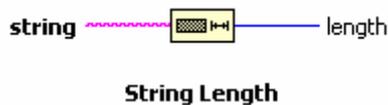


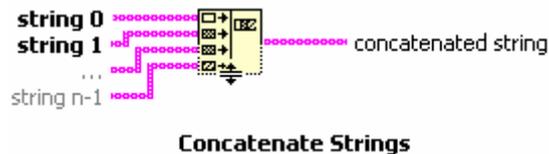
Figura 1.3

A continuación se describen algunas de las funciones más comunes:

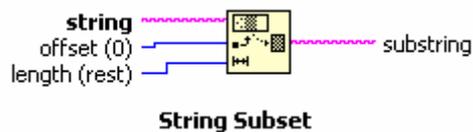
String Length (longitud de la cadena): Devuelve el número de caracteres (bytes) en la cadena, que puede ser un escalar, un array n dimensional o un cluster.



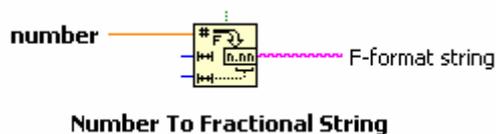
Concatenate Strings (unir cadenas): Concatena todos los strings de entrada en un único string de salida. String 0 y string 1 son los terminales de entrada por defecto. Se puede añadir tantos como sean necesarios.



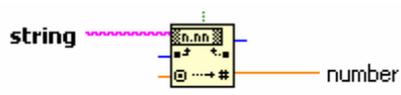
String Subset (subcadena de un string): Devuelve un substring del string original comenzando en el valor del offset y con la longitud determinada por length.



To Fractional (Paso a fraccional=: Convierte un número en un string de punto flotante con notación fraccional. El tamaño viene determinado por los parámetros width (ancho) y precisión.



From Exponential/Fract/Eng (Convierte desde los formatos exponencial/fraccional/ingeniería): Interpreta los caracteres 0 a 9, signo mas, signo menos, e, E y punto decimal dentro de una cadena y aparte del offset como un número en formato coma flotante en notación e ingeniería o formato exponencial o fraccional y devuelve ese número.



Fract/Exp String To Number

Ejemplo String

String	Numeric
Módulo de Comunicaciones Industriales-- ETI	45
String 2	
2º G.S.	
String 3	
Módulo de Comunicaciones Industriales-- ETI 2º G.S.	
String 4	
Comunicaciones	

