

2102-447 Electronic Circuits Laboratory


การทดลองเรื่อง LabVIEW (LV)

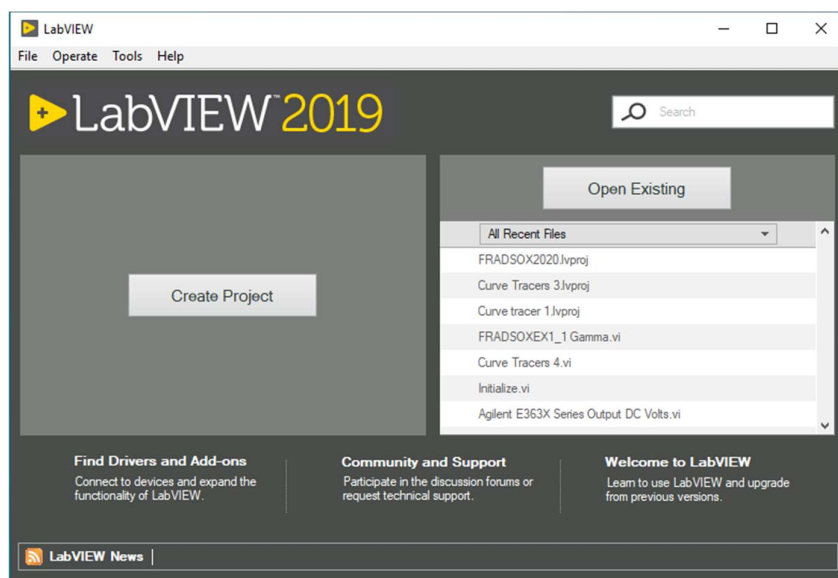
อาภรณ์ ชีรมงคลศรีมี

15 สิงหาคม 2562, 2 กรกฎาคม 2567

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) เป็นภาษาคำสั่งแบบภาพ (Graphical programming language) ที่ถูกออกแบบมาเพื่อใช้ในการงานด้านการวัดและทดสอบ โดยพื้นฐาน LabVIEW ใช้วิธีการลากวัตถุมาวางแทนการเขียนคำสั่งในการสร้างโปรแกรม วัตถุแต่ละชิ้นมีหน้าที่และวิธีใช้งานที่เทียบได้กับการเขียนภาษาคำสั่งแบบตัวอักษร เช่น คำสั่งลูป (Loop) คำสั่งเงื่อนไข (Condition) การจัดการกับตัวแปรอะเรย์ (Array manipulation) การจัดการกับตัวแปรสตริง (String manipulation) เป็นต้น การสร้างโปรแกรมด้วยวัตถุช่วยให้เห็นภาพโครงสร้างของโปรแกรมได้ดีกว่าการเขียนชุดคำสั่งด้วยตัวอักษร และมีข้อดีว่าการเขียนชุดคำสั่งแบบตัวอักษร คือ ช่วยประหยัดเวลาที่ต้องใช้ในการพิมพ์ทำให้เราสามารถเขียนโปรแกรมได้อย่างรวดเร็ว และแก้จุดบกพร่อง (Debug) ได้ง่าย นอกจากนี้ LabVIEW ยังถูกออกแบบมาสำหรับงานวัด จึงมีวัตถุที่ใช้สำหรับการติดต่อกับเครื่องมือวัด การควบคุมเครื่องมือ (Instrument control) การวิเคราะห์ข้อมูล (Data analysis) และการทำรายงาน (Report generation) เตรียมมาให้แล้ว

1. ทำความรู้จัก

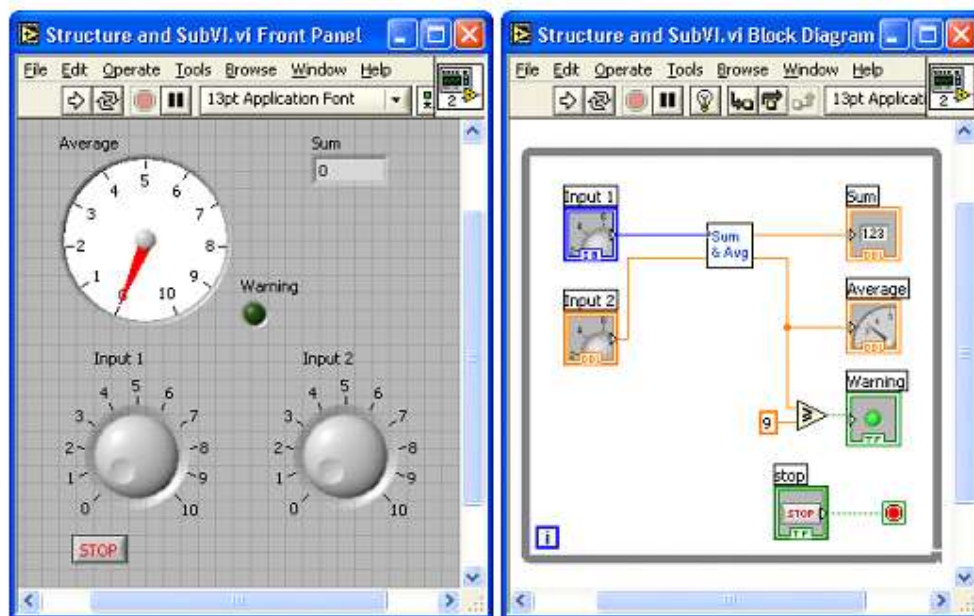
เปิดโปรแกรม LabVIEW ขึ้นมาโดยดับเบิลคลิกที่ไอคอน  LabVIEW ที่ใช้อยู่เป็น Teaching license แบบ Full development system (FDS) version 2019 หลังจากโปรแกรมเปิดขึ้นมาจะพบหน้าจอแรกของโปรแกรม ดังรูปที่ 1



รูปที่ 1 หน้าจอแรกของโปรแกรม LabVIEW

1.1 LabVIEW Environment

โปรแกรมที่พัฒนาด้วย LabVIEW เราจะเรียกว่า Virtual Instrument (VI) ไฟล์ที่บันทึกจะมีนามสกุลเป็น .vi ในแต่ละ VI จะประกอบส่วนของ Front panel และ Block diagram ดังแสดงในรูปที่ 2 ส่วนของ Front panel เป็นส่วนที่ติดต่อกับผู้ใช้งาน ส่วนนี้จะรับข้อมูลเข้าไปในโปรแกรม และ นำข้อมูลออกมาแสดงผลให้ผู้ใช้งาน ส่วนของ Block diagram เป็นหน้าต่างที่ใช้สำหรับเขียน source code เพื่อกำหนดการทำงานของ VI

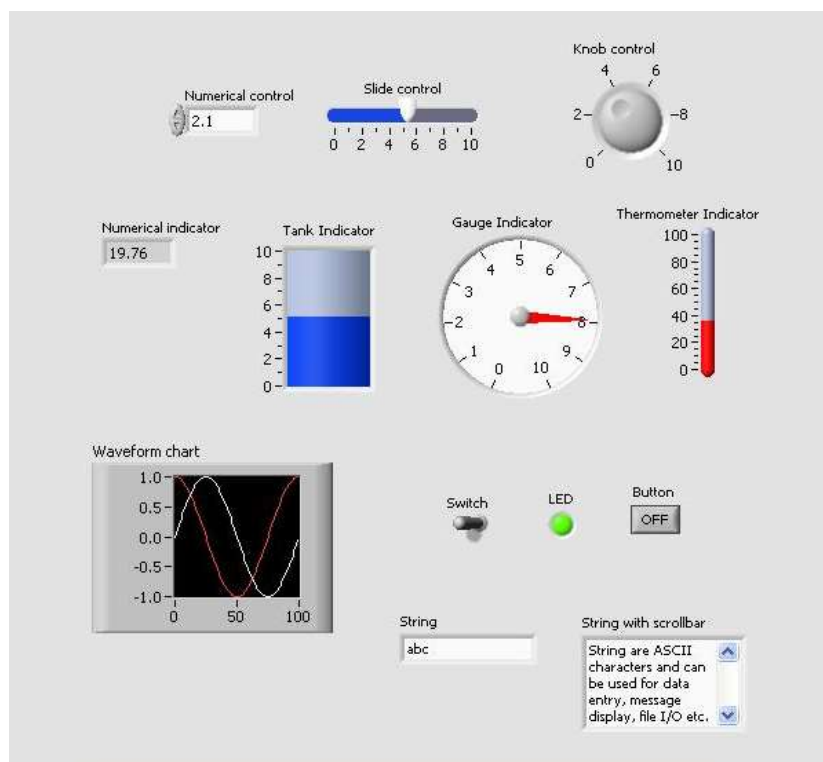


(ก)

(ข)

รูปที่ 2 ตัวอย่างของ (ก) Front Panel และ (ข) Block diagram

LabVIEW Front panel: วัตถุที่วางบน Front panel สำหรับรับข้อมูล (Input) เรียกว่า Control วัตถุที่วางบน Front panel สำหรับแสดงข้อมูล (Output) เรียกว่า Indicator การวางวัตถุที่เป็น Control หรือ Indicator บน Front panel จะทำให้เกิดส่วนที่เรียกว่า Terminal บนหน้าต่างของ Block diagram ด้วย ชนิดของข้อมูล (Data types) ของทั้ง Control และ Indicator มีหลายรูปแบบดังในภาษาคอมพิวเตอร์ทั่วไป เช่น ตัวเลข (Numeric) สตริง (String) อะเรย์ (Array) บูลีน (Boolean) ฯ ดังแสดงในรูปที่ 3 การแบ่งประเภทของชนิดข้อมูลใน Block diagram จะใช้สีเป็นตัวช่วย ตัวอย่าง เช่น สีน้ำเงิน หมายถึง ข้อมูลที่เป็นจำนวนเต็ม (Integer) สีส้ม หมายถึง ข้อมูลที่เป็นจำนวนทศนิยม (Floating) สีชมพู หมายถึง ข้อมูลที่เป็นตัวอักษรหรือข้อความ (Character or Text) เป็นต้น



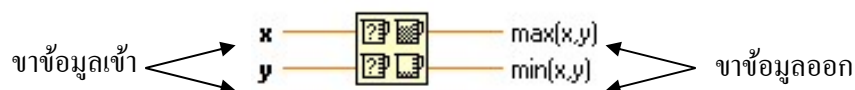
รูปที่ 3 ตัวอย่างของ Control และ Indicator ชนิดต่างๆ

LabVIEW Block diagram: Block diagram เป็นหน้าต่างที่บรรจุ “Source code” ของ VI โดยพื้นฐาน การเขียน code ใน LabVIEW จะเป็นลักษณะกราฟิก ใช้การลากสาย (Wire) เชื่อมระหว่าง Terminal กับ **วัตถุที่เป็นคำสั่งใช้จัดการข้อมูล ซึ่งเรียกว่า Node** ซึ่งก็คือ การนำข้อมูลจาก Front Panel ผ่านทาง Control มาจัดการใน Node ก่อนส่งออกมาผ่านทาง Indicator เพื่อมาแสดงผลบน Front panel

1.2 วัตถุใน LabVIEW

วัตถุแต่ละอันใน LabVIEW ไม่ว่าจะเป็น Control หรือ Indicator บน Front panel หรือ Node บน Block diagram มีคุณสมบัติ (Properties) ที่สามารถเข้าถึงได้ เมื่อใช้เมาส์ชี้ที่วัตถุแล้วคลิกขวาจะมี Pop up เมนูให้เลือก **เพื่อกำหนดคุณสมบัติ**

นอกจากนี้วัตถุทุกชนิดจะมีขา (Pin) ขาที่สำคัญแบ่งออกได้เป็น 2 ประเภท คือ ขาข้อมูลเข้า และขาข้อมูลออก วัตถุที่เป็น Control จะมี Pin ที่เป็นขาข้อมูลออก ส่วน Indicator จะมี Pin ที่เป็นขาข้อมูลเข้า ส่วน Node ที่ใช้จัดการข้อมูลจะมีทั้ง Pin ที่เป็นขาข้อมูลเข้าและขาข้อมูลออก **โดยพื้นฐานของ LabVIEW ขาข้อมูลเข้าจะปล่อยว่างไว้ไม่ได้ ต้องเชื่อมต่อกับขาข้อมูลออกของ Control หรือวัตถุอื่น มิฉะนั้นจะมี Error เกิดขึ้น** แต่ก็มีวัตถุหลายตัวใน LabVIEW ที่มีขาข้อมูลเข้า แต่ได้กำหนดค่าเริ่มต้น (Default) ไว้แล้ว กรณีไม่เชื่อมต่อกับขาข้อมูลเข้า LabVIEW จะใช้ค่าเริ่มต้นในการ Run VI

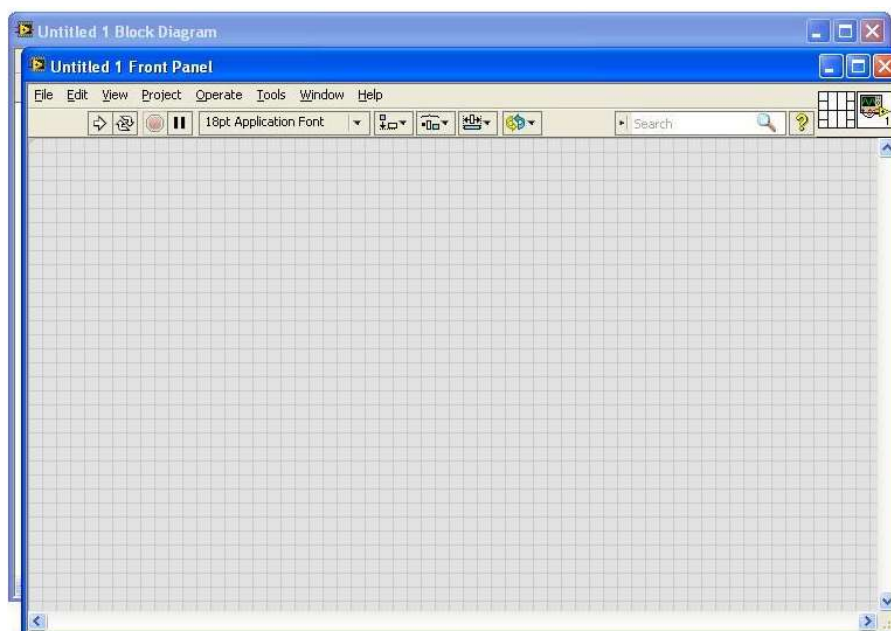


รูปที่ 4 ตัวอย่างของขาข้อมูลเข้าและขาข้อมูลออกของวัตถุใน LabVIEW

1.3 ตัวอย่างการสร้าง VI

เพื่อให้คุ้นเคยการเขียน VI ใน LabVIEW เราจะเริ่มต้นจากการเขียน VI ที่คูณค่าตัวเลข 2 ตัวจาก Control และนำผลที่ได้ไปแสดงบน Indicator

- 1) สร้าง VI ใหม่โดยเลือกที่เมนู File → New VI ของหน้าจอในรูปที่ 1 หรือ กด Ctrl-N จะปรากฏหน้าต่างของ Front panel และ Block diagram ขึ้น ซึ่งในแต่ละ VI จะต้องประกอบด้วย Front panel และ Block diagram เสมอ ดังแสดงในรูปที่ 5

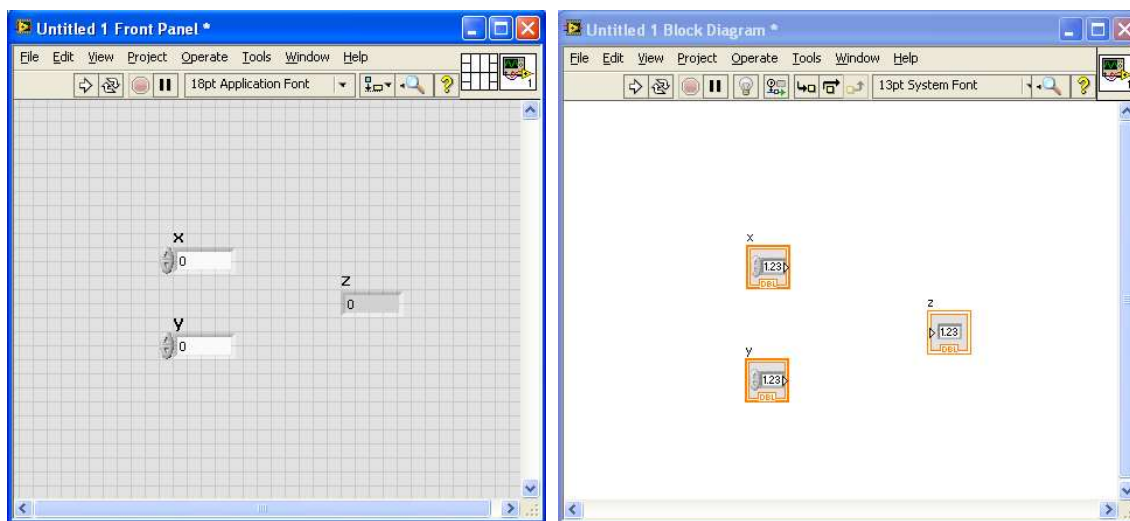


รูปที่ 5 หน้าต่างของ Front panel และ Block diagram

- 2) เราจะเริ่มจากการวาง Control ที่เป็นชนิด Numeric จำนวน 2 ตัว เพื่อใช้สำหรับข้อมูลที่เป็นตัวเลข และ Indicator ที่เป็นชนิด Numeric เหมือนกัน จำนวน 1 ตัว เพื่อแสดงผลของโปรแกรม คลิกขวาในพื้นที่ว่างของ Front panel จะปรากฏ Control palette, ตามด้วยการเลือก Control → Modern Numeric → Numeric Control เลือกวาง Numeric control ในตำแหน่งที่เหมาะสม ทำซ้ำอีกครั้งเพื่อวาง Numerical Control ตัวที่ 2 ถัดลงใช้เมาส์ดับเบิลคลิกที่ Numerical control ที่วางไว้จะไปที่

Terminal ของมันใน Block diagram การสลับระหว่างหน้าต่างของ Front panel และ Block diagram สามารถทำได้โดยการกด Ctrl-E

- 3) แก้ Label ของ Numeric และ Numeric 2 ให้เป็น x และ y ตามลำดับ ทำได้โดยการดับเบิลคลิกที่ตัว Label
- 4) ในทำนองเดียวกัน เราสามารถวาง Numerical indicator โดยการคลิกขวาเพื่อให้ Control Palette ปรากฏ เลือก Control → Modern Numeric → Numeric Indicator และแก้ Label ของ Numerical indicator เป็น z

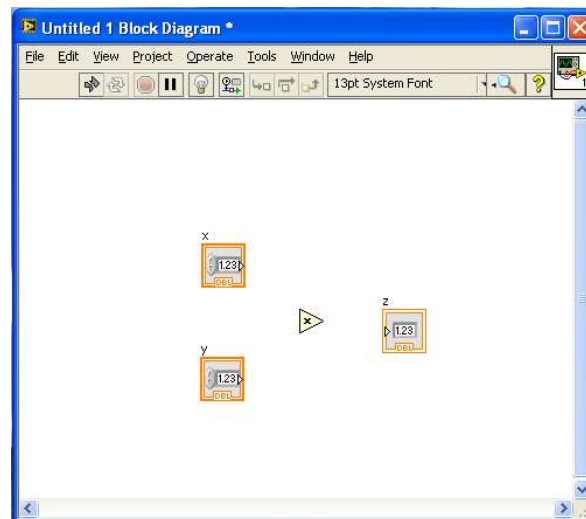


(ก)

(ข)

รูปที่ 6 (ก) Front panel และ (ข) Block diagram ของ VI คูณเลขหลังขั้นตอน 4

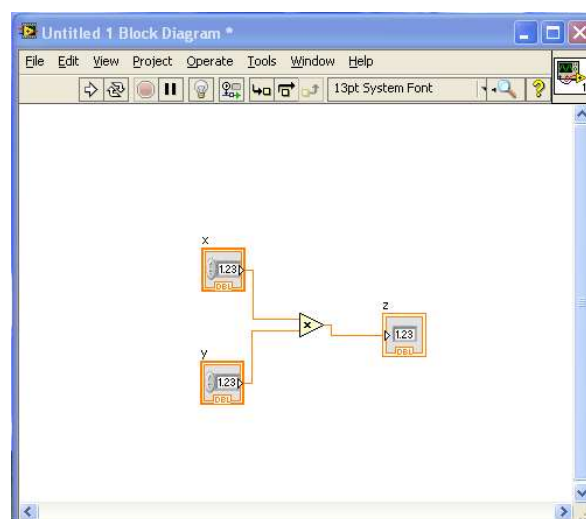
- 5) กด Ctrl-E เพื่อสลับไปยังหน้าต่างของ Block diagram จะเห็น Terminal ของ Numerical control และ Numerical indicator ที่ได้วางไว้ในหน้าต่างของ Front panel ปรากฏอยู่ ดังรูปที่ 6(ข)
- 6) คลิกขวาที่พื้นที่ว่างของหน้าต่างของ Block diagram จะปรากฏ Function palette ตามด้วยการเลือก Programming → Numeric → Multiply เลือกวาง Node ของ Multiply ในตำแหน่งที่เหมาะสม ดังรูปที่ 7




รูปที่ 7 Block diagram ของ VI หมายเลขหลังขั้นตอน 6)

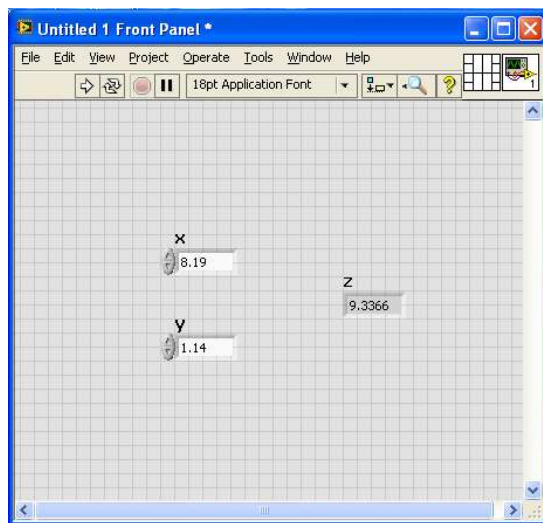
- 7) ขั้นตอนต่อไปเป็นการ wire สายระหว่าง Terminal นำเมาส์ไปชี้ที่โกลี Terminal หรือ Node จะปรากฏ Pin ให้เห็น นำเมาส์ไปชี้ที่ Pin ตัว Pointer จะเปลี่ยนรูปจากลูกศรเป็นรูปคล้ายหลอดด้วยโดยอัตโนมัติ กดคลิกซ้าย ลาก Wire จาก Pin เริ่มต้น ไปยัง Pin ที่ต้องการ กดคลิกซ้ายอีกครั้งเพื่อจบการ Wire ทำการ Wire ให้เสร็จดังรูปที่ 8 (ในกรณีปกติจะ Pointer ของเมาส์จะอยู่ในโหมดอัตโนมัติ ในกรณีที่นำ Pointer ไปชี้ที่ Node หรือ Terminal ไม่มีการเปลี่ยนแปลงใด ให้กด Ctrl+Shift+คลิกขวา จะปรากฏ Tools palette ขึ้น หรือ เรียกจากเมนู View → Tools palette)

หมายเหตุ ในกรณีที่ไม่ทราบว่า Node ที่อยู่บนหน้าต่างของ Block diagram ประกอบด้วย Pin อะไรบ้าง สามารถเรียก Context help ได้โดยการกด Ctrl-H



รูปที่ 8 หน้าต่างของ Block diagram ที่ Wire สายเรียบร้อยแล้ว

- 8) กด Ctrl-E เพื่อเปลี่ยนกลับมายังหน้าต่างของ Front panel ป้อนตัวเลขที่ต้องการใน Numerical Control x และ y สังเกตว่า Numerical indicator z ยังไม่มีการเปลี่ยนแปลงเนื่องจากยังไม่ทำการ run VI กดปุ่ม Run  บน Toolbar เพื่อผลลัพธ์ของ VI ดังรูปที่ 9



รูปที่ 9 หน้าต่างของ Front panel เมื่อ run VI แล้ว

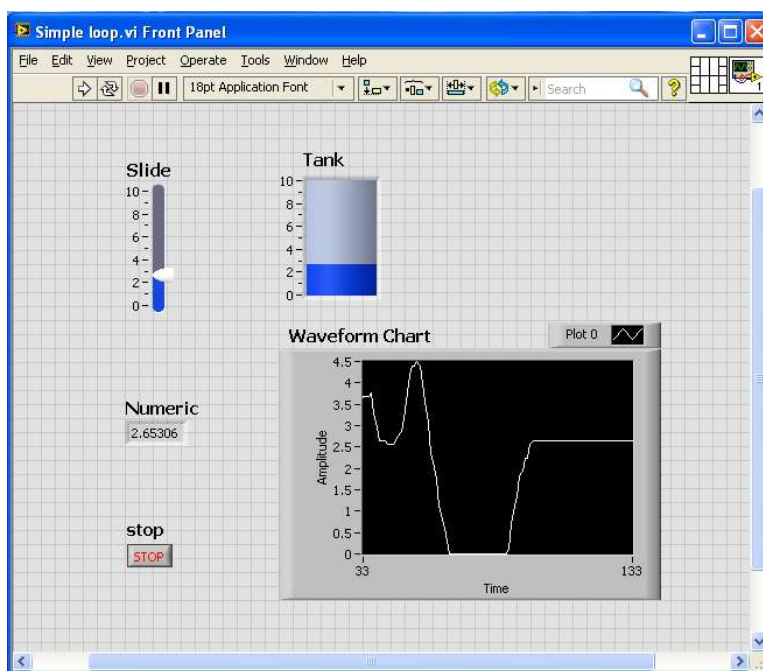
หมายเหตุ ในกรณีที่มี Error ของการเขียน VI จะไม่สามารถ run VI ได้ ปุ่มลูกศรบน Toolbar จะเป็นรูปลูกศรแตก ในกรณีนี้ให้กด Ctrl-L เพื่อเรียกดู Error lists หรือเรียกจากเมนู View → Error lists ก็ได้ จะปรากฏหน้าต่างของ Error lists ขึ้นเพื่อเป็นข้อแนะนำในการแก้ไข

2. การทดลอง

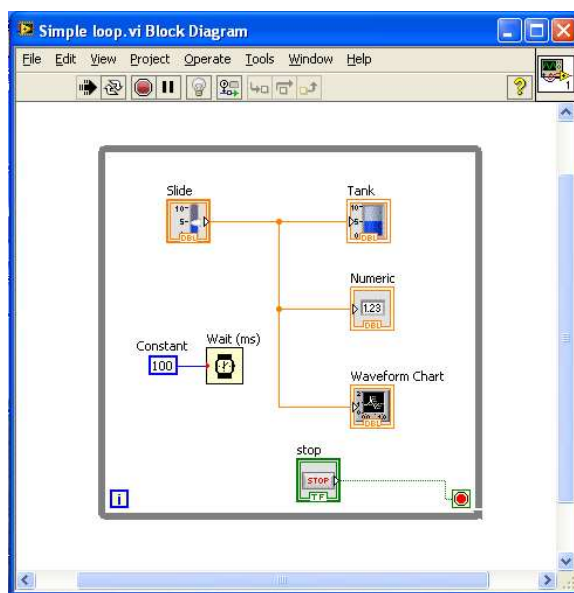
ในหัวข้อนี้จะเป็นการทดลองใช้งาน LabVIEW เขียน VI อย่างง่าย เพื่อให้เข้าใจถึงวิธีการใช้งานและตัวอย่างการประยุกต์ใช้งานการควบคุมเครื่องมือวัด

การทดลองที่ 1 การแสดงผลและการวอร์ปอย่างง่ายใน LabVIEW

ในการทดลองนี้ จะมีการวาง Numerical control ที่เป็นแบบ Slider และเชื่อมต่อ Numerical indicator ที่เป็นตัวเลข รูป Tank และ Chart โดยมีการใช้ While loop เพื่อให้ VI ทำงานต่อเนื่องและจะหยุดทำงานเมื่อมีการกดปุ่ม Stop



(ก)



รูปที่ 10 (ก) Front panel และ (ข) Block diagram ของ VI การแสดงผลและการวงรอบอย่างง่าย

- 1) เปิด VI ใหม่ โดยไปที่เมนู File → New VI
- 2) เขียน VI การแสดงผลและการวงรอบอย่างง่าย โดยเลือกวัตถุตามรายการต่อไปนี้

Front panel

1. Controls palette → Modern → Numeric → Vertical pointer slide
2. Controls palette → Modern → Numeric → Numeric indicator
3. Controls palette → Modern → Numeric → Tank

4. Controls palette → Modern → Graph → Chart
5. Controls palette → Modern → Boolean → Stop button

Block diagram


1. Functions palette → Programming → Timing → Wait
2. Functions palette → Programming → Numeric → Numeric constant

ใส่ตัวเลข 100

3. Functions palette → Programming → Structures → While loop

เมื่อเลือกแล้ว Pointer ของเมาส์จะเปลี่ยนไปให้คลิกซ้ายติกรอบให้ครอบวัตถุทั้งหมด

3) ลากสายเชื่อมโยงวัตถุตามรูปที่ 10

4) สลับไปที่หน้าต่างของ Front panel สั่งให้ VI ทำงานโดยการกดปุ่ม Run 

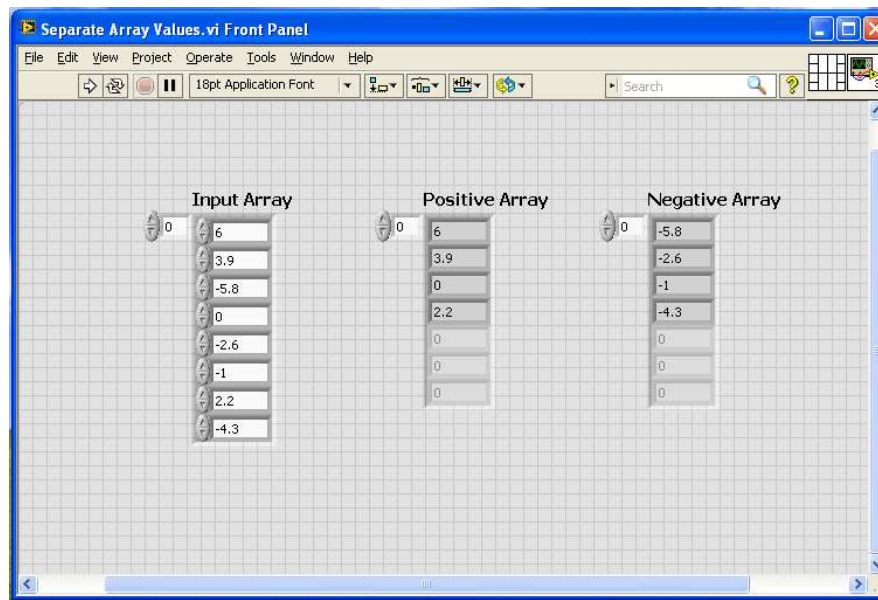
5) ลองปรับ Slider ดูสังเกตผลลัพธ์ที่ได้ว่า เป็นไปตามที่คาดหวังหรือไม่ เมื่อต้องการจบ VI ให้กด ปุ่ม

Stop

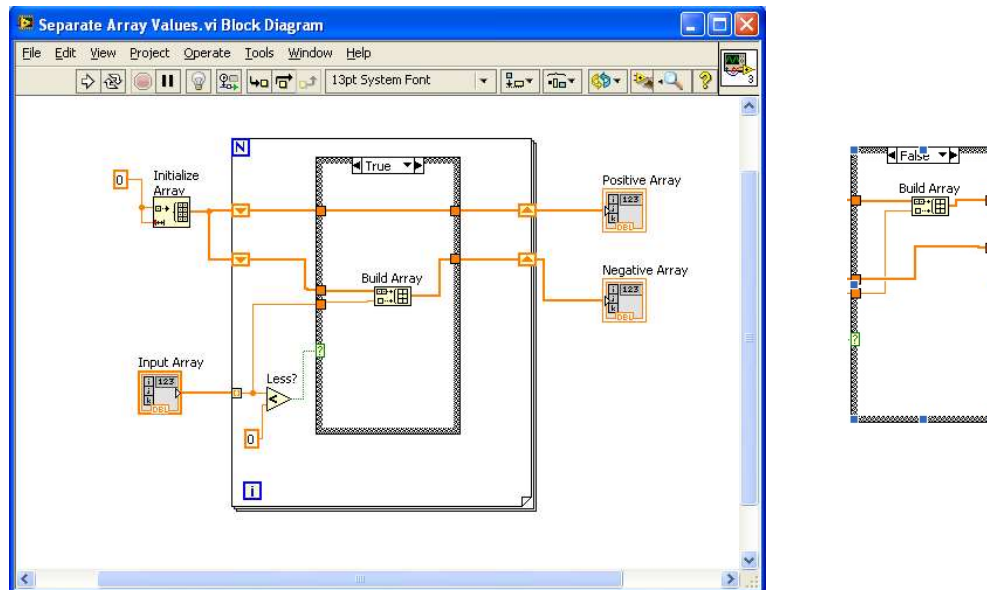
หมายเหตุ การทดลองนี้ใน While loop มีการใส่คำสั่ง Wait เข้าไป และกำหนดให้หรือเป็นเวลา 100 มิลลิวินาที ก่อนที่วนลูปรอบใหม่ ที่จำเป็นต้องทำเช่นนี้ เพราะถ้าไม่มีการรอระหว่างลูป CPU จะถูกเรียกใช้งานตลอดเวลา ทำให้ภาระงานของ CPU เป็น 100 % (สามารถตรวจสอบได้จาก Task manager ของ Windows)

การทดลองที่ 2 การแยกตัวเลขบวกและลบ

ในการทดลองนี้จะมีการสร้างอะเรย์ของตัวเลขทศนิยมที่เป็น Numerical control ให้ผู้ใช้ใส่ตัวเลขเข้าไป ทั้งเลขบวกและลบ VI ที่สร้างขึ้นจะแยกตัวเลขออกเป็น 2 กลุ่ม คือ เลขบวกและเลขลบ นำมาแสดงมาในอะเรย์ ของตัวเลขทศนิยม 2 อะเรย์ที่เป็น Numerical indicator VI นี้จะอ่านค่าจากอะเรย์ของ Numerical control เข้ามาทีละค่า แล้วเปรียบเทียบว่าน้อยกว่าศูนย์หรือไม่ ถ้าใช่ นำตัวเลขไปใส่ในอะเรย์ของเลขลบ ถ้าไม่ใช่ นำตัวเลขไปใส่ในอะเรย์ของเลขบวก ทำอย่างนี้เรื่อยไปจนครบทุกค่าของอะเรย์ของ Numerical control



(ก)



(ข)

รูปที่ 11 (ก) Front panel และ (ข) Block diagram ของ VI การแยกตัวเลขบวกและลบ

- 1) เปิด VI ใหม่ โดยไปที่เมนู File → New VI
- 2) เขียน VI การแยกตัวเลขบวกและลบโดยเลือกวัตถุตามรายการต่อไปนี้

Front panel

1. Controls palette → Modern → Array, matrix & cluster → Array

จะเห็นว่า วัตถุ Array มีช่องว่างอยู่ ซึ่งเหมือนกับกรณีที่เราอยากสร้างตัวแปรอะไรๆ แต่ยังไม่ได้กำหนดชนิดของตัวแปร

2. Controls palette → Modern → Numeric → Numeric control

นำเอา Numerical control ไปใส่ในช่องว่างของวัตถุ Array อะเรย์ที่ได้จะเป็นอะเรย์ของ Numerical control ที่สามารถรับค่าจากผู้ใช้ได้ แก้ไข Label ให้เป็น Input Array ตอนนี้อะเรย์จะแสดงค่าสมาชิกของอะเรย์เพียง 1 ค่าเท่านั้น นำ Pointer ของเมาส์ไปชี้ที่กึ่งกลางด้านล่าง Pointer จะเปลี่ยนเป็น ลูกศร 2 ทาง กดคลิกซ้ายเพื่อขยายการแสดงผลจำนวนสมาชิกของอะเรย์ตามต้องการ

3. Controls palette → Modern → Array, matrix & cluster → Array

4. Controls palette → Modern → Numeric → Numeric indicator

นำเอา Numerical indicator ไปใส่ในช่องว่างของวัตถุ Array แก้ไข Label ให้เป็น Positive Array ปรับขนาดของการแสดงผลจำนวนสมาชิกของอะเรย์ตามต้องการ

5. Controls palette → Modern → Array, matrix & cluster → Array

6. Controls palette → Modern → Numeric → Numeric indicator

นำเอา Numerical indicator ไปใส่ในช่องว่างของวัตถุ Array แก้ไข Label ให้เป็น Negative Array ปรับขนาดของการแสดงผลจำนวนสมาชิกของอะเรย์ตามต้องการ

Block diagram

1. Functions palette → Programming → Numeric → Numeric constant

2. Functions palette → Programming → Array → Initialize array

3. Functions palette → Programming → Structures → For loop

กดคลิกซ้ายให้พื้นที่ของ For loop มีขนาดตามต้องการ นำเมาส์ไปชี้ที่ด้านข้างของ For loop คลิกขวาจะปรากฏเมนูกำหนดคุณสมบัติของ For loop ให้เลือกที่ Shift register จะปรากฏกรอบสี่เหลี่ยม  ที่ด้านซ้ายและ  ด้านขวา สร้าง Shift register ซ้ำอีกครั้ง การสร้าง Shift register ก็คือการทำให้ Loop structure สามารถนำค่าตัวแปรในลูปก่อนหน้านั้นมาไว้ในลูปปัจจุบัน

4. Functions palette → Programming → Structures → Case structure

กดคลิกซ้ายให้พื้นที่ของ Case structure บรรจุอยู่ภายใน While loop ค่าเริ่มต้นของ Case structure ใน LabVIEW จะมีอยู่ 2 กรณี คือ True และ False การสลับระหว่าง True และ False ให้กดที่ Drop-down-lists ด้านบน หรือ ลูกศรด้านข้าง

5. เลือกให้ Case structure เป็น True เลือกวัตถุ Functions palette → Programming → Array → Build array

นำไปใส่ใน Case structure ปรับขนาดของ Build array ให้รับอินพุตได้ 2 Pin โดยนำ Pointer ของเมาส์ไปชี้ที่กึ่งกลางด้านล่าง Pointer จะเปลี่ยนเป็นลูกศร 2 ทาง กดคลิกซ้ายเพื่อปรับจำนวนอินพุต

6. เลือกให้ Case structure เป็น False เลือกวัตถุ Functions palette → Programming → Array → Build array

นำไปใส่ใน Case structure ปรับขนาดของ Build array ให้รับอินพุตได้ 2 Pin โดยนำ Pointer ของเมาส์ไปชี้ที่กึ่งกลางด้านล่าง Pointer จะเปลี่ยนเป็นลูกศร 2 ทาง กดคลิกซ้ายเพื่อปรับจำนวนอินพุต

7. Functions palette → Programming → Comparison → Less?

8. Functions palette → Programming → Numeric → Numeric constant

3) ลากสายเชื่อมโยงวัตถุตามรูปที่ 11

4) สลับไปที่หน้าต่างของ Front panel ใส่ตัวเลขใน Input array ตามต้องการให้ผสมระหว่างเลขบวกและเลขลบ

5) สั่งให้ VI ทำงาน โดยการกดปุ่ม Run 

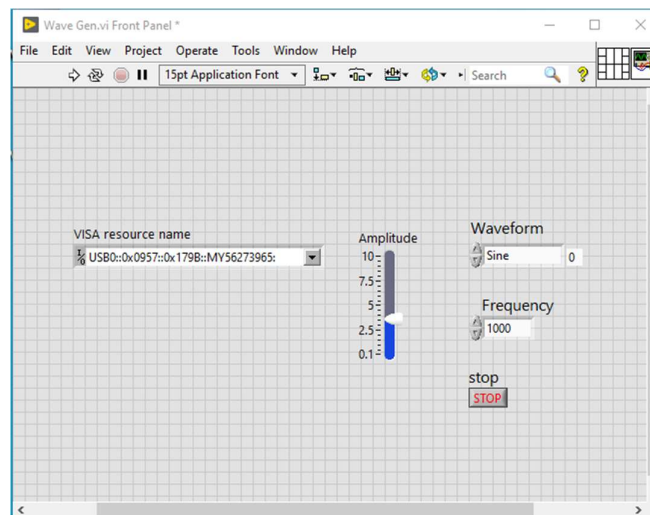
6) สังเกตผลลัพธ์ที่ได้ว่า เป็นไปตามที่คาดหวังหรือไม่

การทดลองที่ 3 การติดต่อกับอุปกรณ์ภายนอกผ่าน Instrument driver

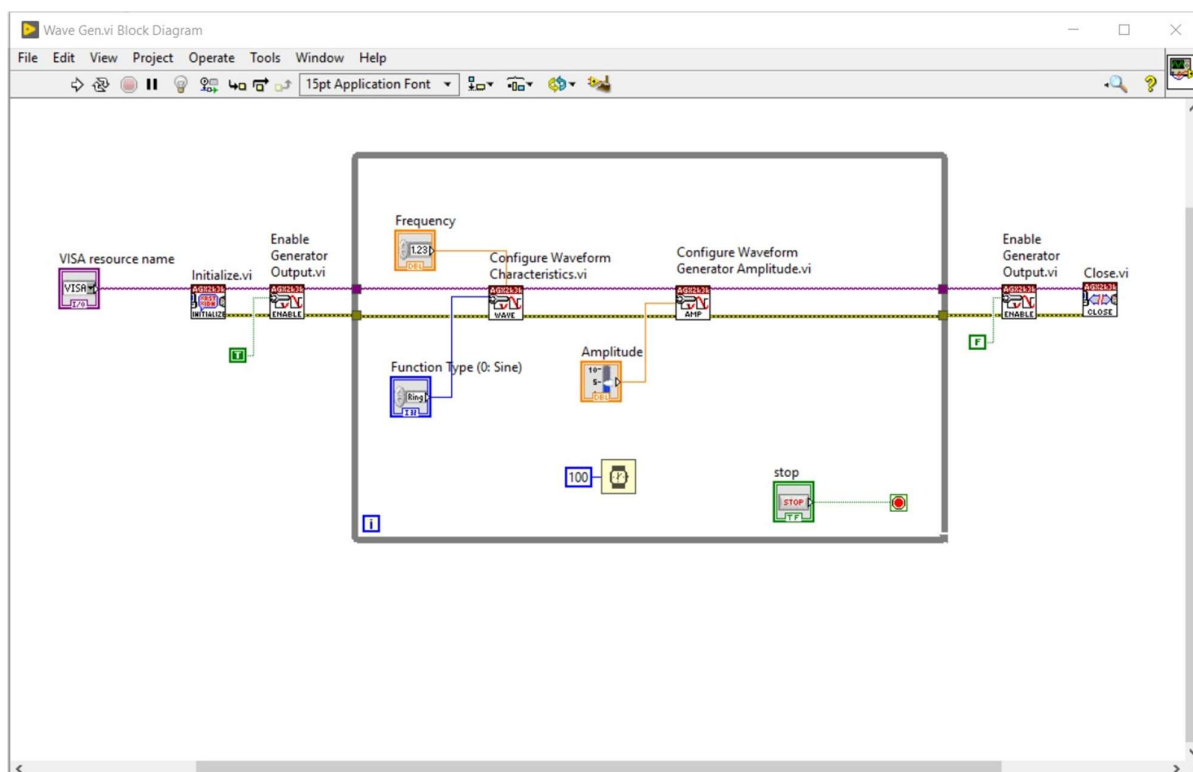
การทดลองนี้จะเป็นการสร้าง VI ที่มีความสามารถในการติดต่อกับเครื่องกำเนิดสัญญาณ WaveGen ของดิจิทัลออสซิลโลสโคป Keysight รุ่น InfiniiVision ที่อยู่ในห้องปฏิบัติการ เพื่อควบคุมรูปคลื่น ขนาด และความถี่ LabVIEW สามารถติดต่อกับอุปกรณ์ภายนอกผ่านการเชื่อมต่อของพอร์ตในรูปแบบต่างๆ เช่น พอร์ตอนุกรม (Serial port) พอร์ต GPIB (General Purpose Interface Bus) หรือ พอร์ต USB เป็นต้น สำหรับชุดทดลองในห้องปฏิบัติการพื้นฐานอิเล็กทรอนิกส์ เครื่องมือพื้นฐานทางไฟฟ้า ได้แก่ แหล่งกำเนิดไฟตรงเชื่อมต่อกับพอร์ต USB หรือ พอร์ตอนุกรม ดิจิทัลออสซิลโลสโคปที่มีฟังก์ชันการกำเนิดสัญญาณในตัว ได้เชื่อมต่อกับคอมพิวเตอร์ผ่านพอร์ต USB

หมายเหตุ แหล่งกำเนิดไฟตรงในห้องปฏิบัติการมี 2 รุ่น คือ Agilent รุ่น E3631A ที่สามารถเชื่อมต่อผ่านอนุกรม และ Keysight รุ่น E36311A ที่สามารถเชื่อมต่อผ่านพอร์ต USB

LabVIEW สามารถติดต่อกับอุปกรณ์ภายนอกผ่านวัตถุพื้นฐานที่มีอยู่ใน LabVIEW ซึ่งต้องทราบคำสั่งและขั้นตอนที่ใช้ในการสื่อสารของอุปกรณ์แต่ละชนิด (ศึกษาได้จากคู่มือของอุปกรณ์นั้น) อีกวิธีการหนึ่งที่สามารถทำได้ คือ การใช้ Instrument driver ซึ่งเป็น driver สำหรับเครื่องมือวัดนั้น โดยส่วนใหญ่ Instrument driver จะถูกพัฒนาโดยบริษัทเครื่องมือวัดให้สามารถใช้งานร่วมกับ LabVIEW โดย Instrument driver เป็นวัตถุที่ได้รวบรวมคำสั่งพื้นฐานของเครื่องมือวัดนั้นเข้าด้วยกัน ทำให้สามารถสั่งงานเครื่องมือวัดในระดับสูง เราจึงสามารถใช้หรือสั่งงานเครื่องมือวัดได้สะดวกและง่าย



(ก)



รูปที่ 12 (ก) Front panel และ (ข) Block diagram ของ VI เพื่อควบคุม WaveGen

- 1) เปิดเครื่องดิจิทัลออสซิลโลสโคป รอให้ออสซิลโลสโคปพร้อมใช้งาน
- 2) ใช้สายสัญญาณต่อ Output จาก WaveGen เข้ากับ CH1 ของออสซิลโลสโคป
- 3) เปิด VI ใหม่ โดยไปที่เมนู File → New VI
- 4) เขียน VI ตามรายการต่อไปนี้

Front panel

1. Controls palette → Modern → I/O → VISA resource name

VISA resource name จะมี Drop-down-menu ให้เลือกเป็น USB0::xxxxx:INSTR (หมายถึง เครื่องมือที่ต่ออยู่กับพอร์ต USB หมายเลข 0 ในที่นี้ คือ ออสซิลโลสโคป)

2. Controls palette → Modern → Numeric → Vertical pointer slide

แก้ Label เป็น Amplitude กำหนด Scale ต่ำสุดเป็น 0.1 เนื่องจาก Waveform generator ที่ใช้ ให้ Output อยู่ในช่วง 0.1 – 10 Vpp

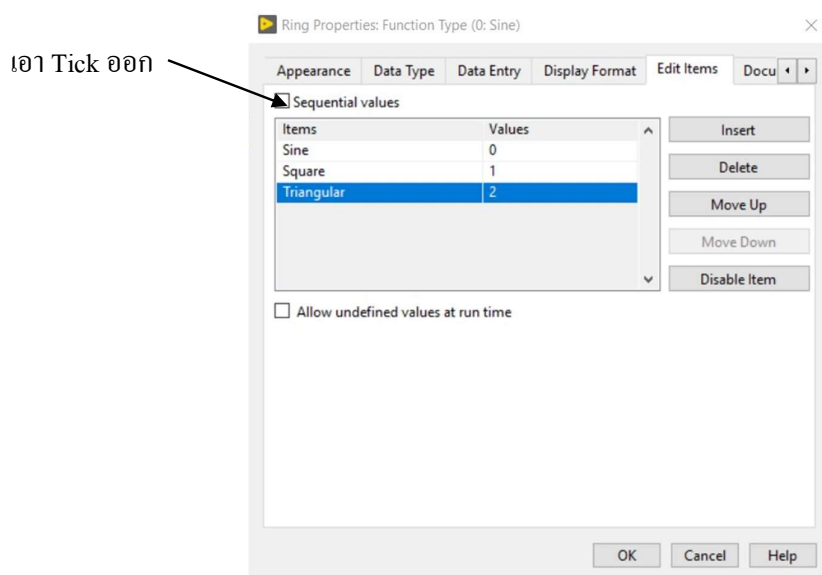
3. Controls palette → Modern → Ring & Enum → Text ring

แก้ Label เป็น Waveform function กดคลิกขวาเพื่อกำหนดคุณสมบัติ เลือกที่ Edit items ให้เอา Tick ที่ Sequential values ออก ใส่ค่าตามรูปที่ 13

4. Controls palette → Modern → Numeric → Numerical control

แก้ Label เป็น Frequency

5. Controls palette → Modern → Boolean → Stop button



รูปที่ 13 การกำหนดค่าสมาชิกของ Text ring

Block diagram

1. Functions palette → Instrument I/O → Instrument drivers → Agilent 2000 3000 X series → Initialize.vi

2. Functions palette → Instrument I/O → Instrument drivers → Agilent 2000 3000 X series → Configure → Waveform generator → Enable generator output.vi

3. Functions palette → Programming → Boolean → True constant

4. Functions palette → Programming → Structures → While loop

กำหนดพื้นที่ของ While loop ตามความเหมาะสม

5. Functions palette → Instrument I/O → Instrument drivers → Agilent 2000 3000 X series → Configure → Configure waveform characteristics.vi

6. Functions palette → Instrument I/O → Instrument drivers → Agilent 2000 3000 X series → Configure → Configure waveform amplitude.vi

7. Functions palette → Programming → Timing → Wait


8. Functions palette → Programming → Numeric → Numeric constant

9. Functions palette → Instrument I/O → Instrument drivers → Agilent 2000 3000 X series → Configure → Waveform generator → Enable generator output.vi

10. Functions palette → Programming → Boolean → False constant

11. Functions palette → Instrument I/O → Instrument drivers → Agilent 2000 3000 X series → Close.vi

2) ลากสายเชื่อมโยงวัตถุตามรูปที่ 12

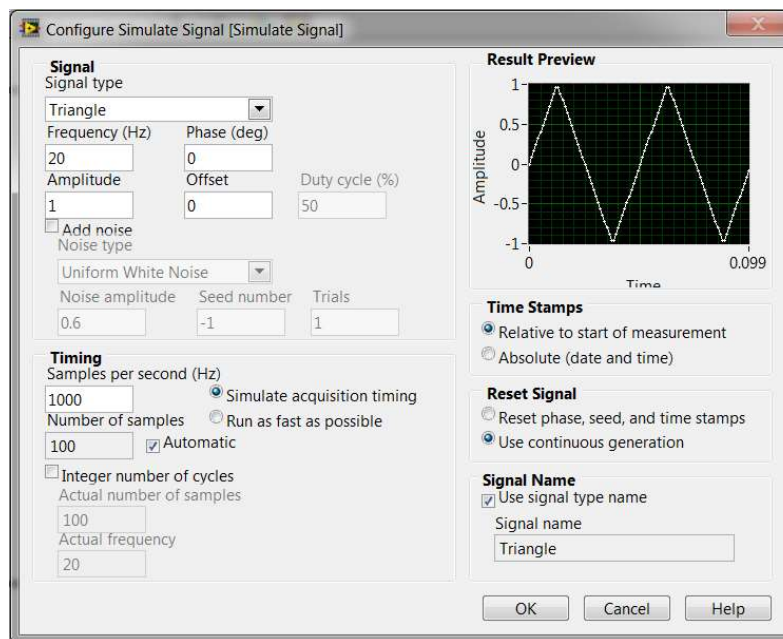
4) สลับไปที่หน้าต่างของ Front panel สั่งให้ VI ทำงานโดยการกดปุ่ม Run 

5) ลองปรับ Amplitude, Waveform function และ Frequency ดูสังเกตผลลัพธ์ที่ได้ว่า เราสามารถสั่งงาน Waveform generator เป็นไปตามที่คาดหวังหรือไม่ เมื่อต้องการจบ VI ให้กด ปุ่ม Stop

การทดลองที่ 4 แนะนำ Express VI

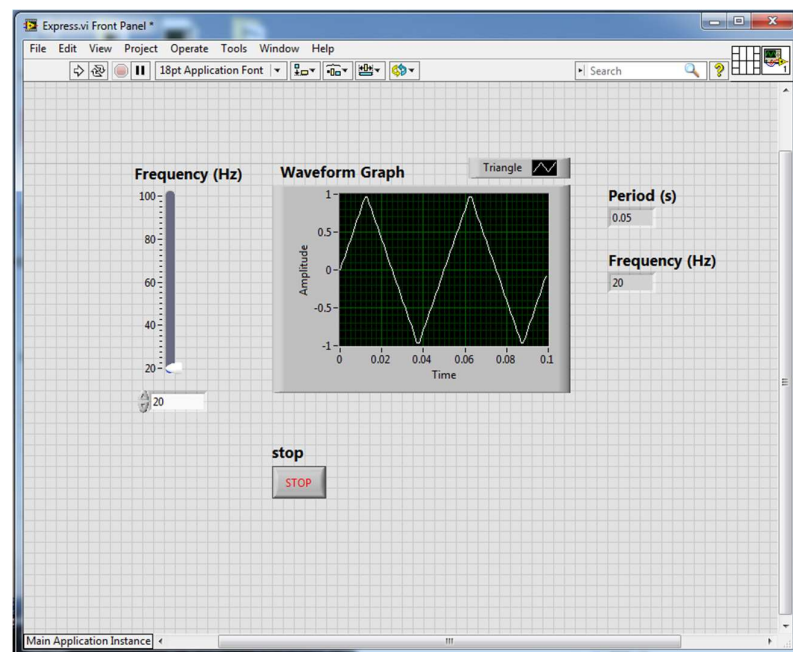
ดังได้กล่าวมาแล้วว่า โปรแกรมที่พัฒนาด้วย LabVIEW จะเรียกว่า VI อันที่จริงแล้ว วัตถุหรือ Node ที่เราวางในหน้า Block diagram ก็เป็น VI แบบหนึ่ง เพื่อความสะดวกในการใช้งานทาง National Instrument ได้สร้าง VI ขึ้นมาชนิดหนึ่ง ที่เรียกว่า Express VI Express VI เป็น VI ที่มี Dialog box เพื่อช่วยให้ความสะดวกและรวดเร็วแก่ผู้ใช้งานในการตั้งค่า หรือ config พารามิเตอร์ต่างๆ Express VI ได้ถูกสร้างขึ้นเพื่อทำฟังก์ชันที่ใช้งานบ่อย เช่น การหาขนาดหรือแอมพลิจูดของสัญญาณรายคาบ (Period signal) การหาความถี่ของสัญญาณรายคาบ เป็นต้น

หลังจากวางวัตถุที่เป็น Express VI ใน Block diagram แล้ว Dialog box ของ Express VI นั้น จะปรากฏขึ้นมาโดยอัตโนมัติ เพื่อให้ผู้ใช้ตั้งค่าพารามิเตอร์ต่างๆ หรือผู้ใช้สามารถกดดับเบิ้ลคลิกที่วัตถุ หรือ กดคลิกขวาแล้วเลือก properties เพื่อให้แสดง Dialog box ขึ้นมา ตัว Express VI จะมีไอคอนที่ใหญ่กว่าของ VI ปกติและจะมีพื้นหลังเป็นสีฟ้า ตัวอย่างของ Dialog box ของ Simulated signal ซึ่งเป็น Express VI แสดงในรูปที่ 14

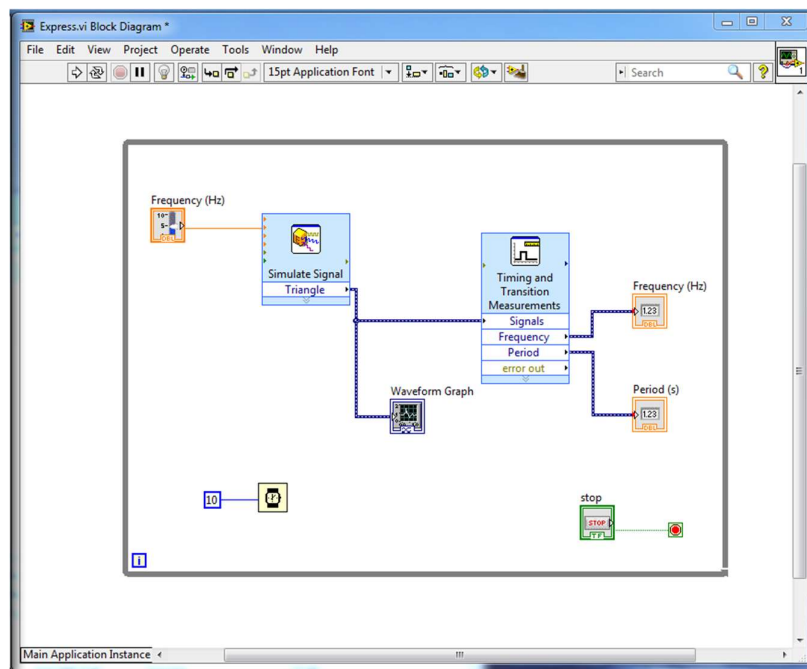


รูปที่ 14 ตัวอย่าง Dialog box ของ Simulated signal: Express VI ผู้ใช้สามารถตั้งค่าพารามิเตอร์ต่างๆ ได้อย่างสะดวก

ในการทดลองนี้ เราจะเขียน VI ที่ใช้วัตถุของ Simulate signal ซึ่งเป็น Express VI ในการสร้างสัญญาณจำลองที่เป็นรายคาบ และใช้ Timing and Transition Measurement ซึ่งก็เป็น Express VI ให้คำนวณหาค่าความถี่และคาบของสัญญาณที่ได้จำลองขึ้น



(ก)



(ข)

รูปที่ 15 (ก) Front panel และ (ข) Block diagram ของ VI เพื่อการจำลองสัญญาณและหาค่าความถี่และคาบเวลา

- 1) เปิด VI ใหม่ โดยไปที่เมนู File → New VI
- 2) เขียน VI โดยเลือกวัตถุตามรายการต่อไปนี้

Front panel

1. Controls palette → Modern → Numeric → Vertical pointer slide

กดคลิกขวา เลือก Visible display แล้วเลือก Digital display เพื่อแสดงตัวเลข วัตถุนี้ใช้เป็นตัวป้อนค่าความถี่ของสัญญาณที่จะจำลองขึ้น

2. Controls palette → Modern → Graph → Waveform graph

3. Controls palette → Modern → Numeric → Numeric indicator

แก้ไข Label ให้เป็น Frequency

4. Controls palette → Modern → Numeric → Numeric indicator

แก้ไข Label ให้เป็น Period

5. Controls palette → Modern → Boolean → Stop Button

Block diagram

1. Functions palette → Programming → Structures → While loop

กดคลิกซ้ายให้พื้นที่ของ While loop มีขนาดตามต้องการ นำวัตถุที่มีอยู่ใน Block diagram ทั้งหมดเข้าไปอยู่ใน While loop ที่มุมขวาล่างของ While loop จะมีปุ่มที่ใช้กำหนดเงื่อนไขของการหยุดวน loop ให้คลิกขวาที่ปุ่มนั้น เลือกเงื่อนไขของการหยุดเป็น Stop if True

2. Functions palette → Programming → Timing → Wait (ms)

3. Functions palette → Programming → Numeric → Numeric constant

แก้ไข ตัวเลขเดิมให้เป็นเลข 10 ลงไป

4. Functions palette → Express → Input → Simulate sig

เมื่อวางวัตถุลงบน Block diagram จะปรากฏ Dialog box ขึ้น ให้เลือก Signal Type เป็นสัญญาณที่เราชอบ เช่น Sine Triangle Square หรือ Sawtooth (ไม่เลือก DC) เลือกค่าความถี่ จำนวน Sample per second ตามชอบใจ กด ปุ่ม OK Dialog box จะปิดลง

5. Functions palette → Express → Signal analysis → Timing-Trans

เมื่อวางวัตถุลงบน Block diagram จะปรากฏ Dialog box ขึ้น ให้คลิกเลือกพารามิเตอร์ ที่อยากให้เห็น ในที่นี้เลือก Frequency และ Period นิติตอาจเลือกพารามิเตอร์อื่นเพิ่มเติมก็ได้ กด ปุ่ม OK Dialog box จะปิดลง

3) ลากสายเชื่อมโยงวัตถุตามรูปที่ 15 ข)

มีข้อสังเกตว่า สายเชื่อมโยงวัตถุมีหลากหลายสี แต่ละสีมีความหมาย เช่น เส้นสีเขียวแทนข้อมูล ชนิด Boolean เส้นสีส้มแทนข้อมูลชนิด Float ในบางกรณี LabVIEW ยอมให้มีการเชื่อมโยงสายระหว่างวัตถุที่ใช้ ข้อมูลต่างชนิดกัน อย่างเช่น ข้อมูลจากวัตถุจำพวก Express VI เป็น Dynamic data (ข้อมูลที่มีเฉพาะใน LabVIEW) สามารถเชื่อมต่อไปยังวัตถุ Numerical indicator ได้ แต่มีจุดแดงที่ Numerical indicator แสดงว่า เกิดการใช้ข้อมูล ต่างประเภทกัน

4) กลับไปที่หน้าต่างของ Front panel

5) สั่งให้ VI ทำงาน โดยการกดปุ่ม Run 

6) สังเกตผลลัพธ์ที่ได้ว่า เป็นไปตามที่คาดหวังหรือไม่ นิติตอาจลองเลือก Express VI ชนิดอื่นเพื่อหาค่า อื่นของสัญญาณที่ได้จำลองขึ้น เช่น ค่ายอดถึงยอด ค่าเฉลี่ย ค่า rms เป็นต้น