

به نام خدا

نرم افزار LABVIEW

گردآوری

مهدی صالح پور، احمد آقا جانی

کلمات کلیدی

LABVIEW

چکیده

این مقاله آموزشی مختصری است از نرم افزار LABVIEW



۱. مقدمه

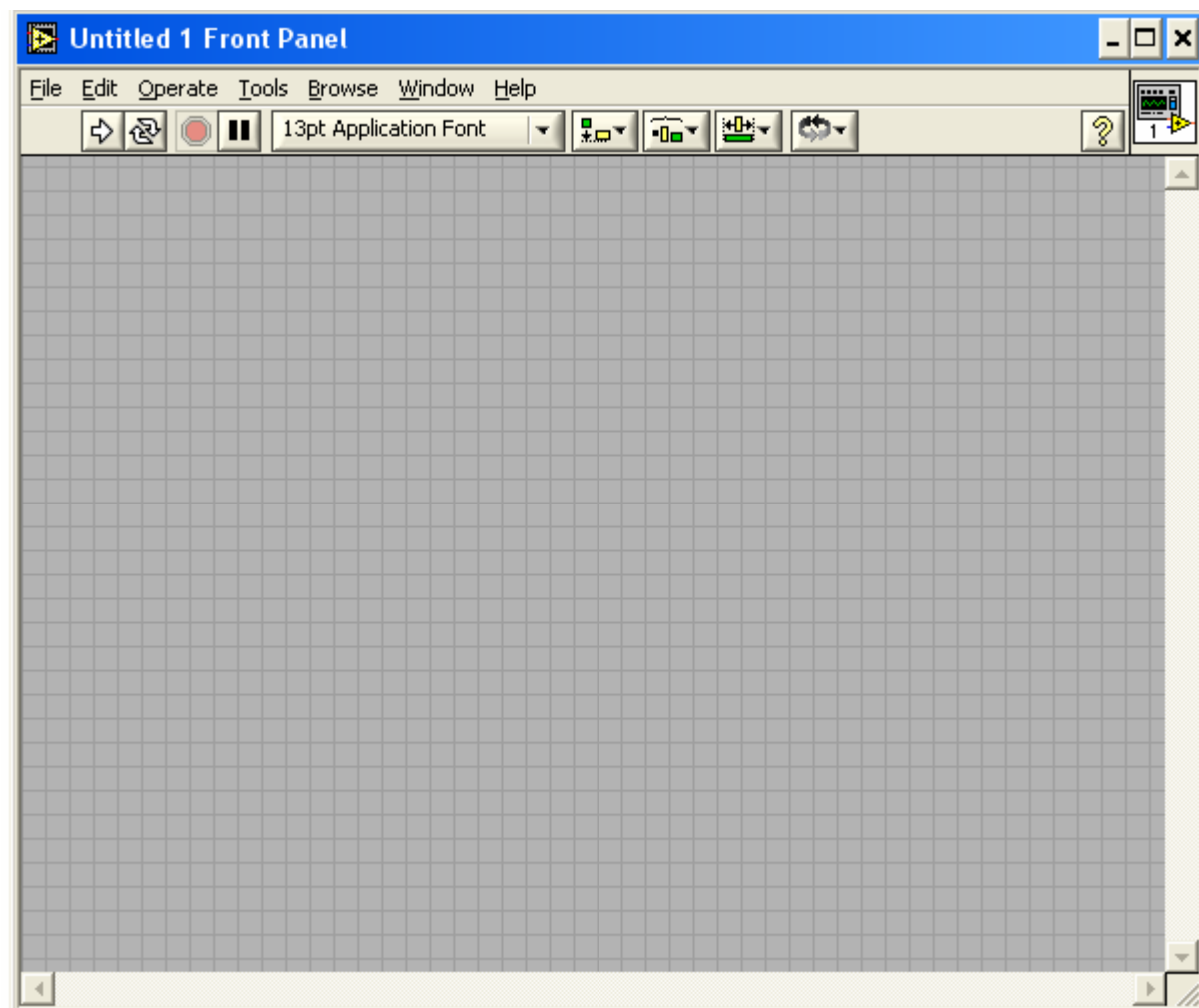
Labview مخفف عبارت Laboratory Virtual Instrument Engineering Workbench می باشد که یک زبان برنامه نویسی گرافیکی است، عملکرد نرم افزار Labview کاملاً از طبیعت ترتیبی متنی و زنجیره ای موجود در زبان های متنی و موسوم مجزا است و یک محیط گرافیکی را برای کاربر فراهم ساخته است. زبان برنامه نویسی این نرم افزار G نام دارد، به کمک این زبان وبا کمک Labview می توانید استعدادهای خود را براحتی شکوفا سازید زیرا نوشتن برنامه هایی که با استفاده از زبان های برنامه نویسی رایج دیگر در عرض چندین ماه نوشته می شود بوسیله ی این زبان در عرض چند ساعت نوشته می شود.

۲. نرم افزار labview چگونه عمل می کند؟

هر برنامه ای که در محیط labview نوشته می شود شامل سه بخش است .

۲-۱. صفحه ی *Front panel* :

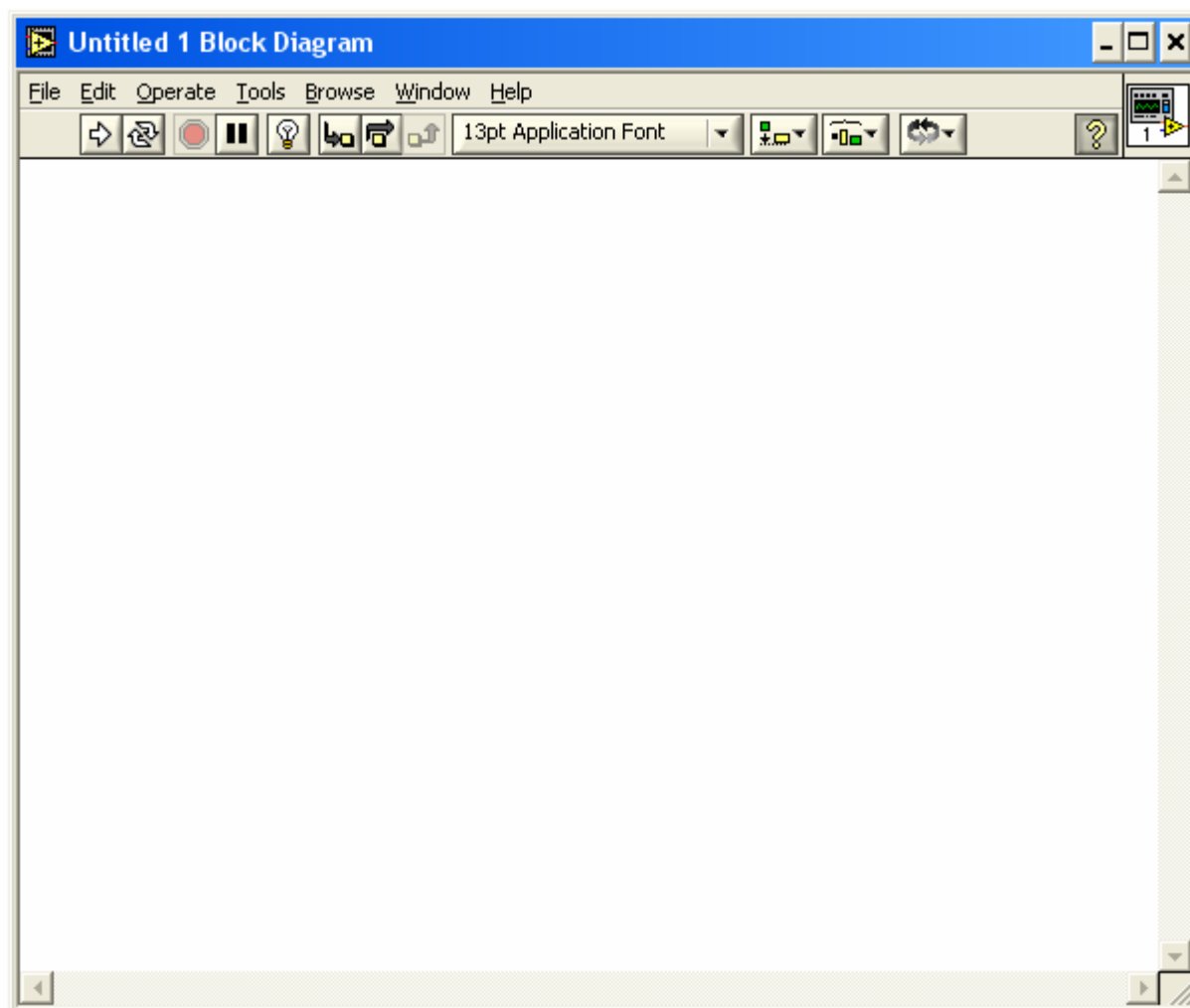
هنگامی که نرم افزار LABVIEW را اجرا کنید سه گزینه در صفحه ی اول ظاهر می شود با اجرای گزینه New صفحه ای دیگر باز می شود اگر در قسمت create new آیکن blank VI را انتخاب کنیم این صفحه باز می شود. روی این صفحه یک پانل قرار می گیرد که شامل دکمه ها و دیگر المان های ورودی است که در مجموع به آنها کنترل کننده می گوییم. در ادامه طرز استفاده از این المان ها را توضیح خواهیم داد.



۲-۲. صفحه ی Block Diagram :

این بخش شامل محل قرار گرفتن کدها و نمادهایی است که زیر بنای زبان برنامه نویسی گرافیکی <<G>> است.

صفحه ی نمودار بلوکی در حقیقت بخش اجرایی برنامه است. این صفحه شامل دستور، توابع، ثوابت و ساختارهاست. در این صفحه المان ها بوسیله سیم بهم متصل می شود. در صفحه اگر از منو windows گزینه ی show functions palette را انتخاب کنیم می توانیم دستور و توابع را مشاهده کرد در مورد دستور ها و... بعداً توضیح خواهیم داد.



۲-۳. آیکن و کانکتور:

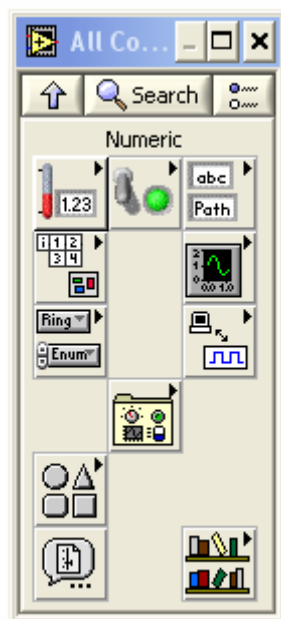
هرگاه بخواهیم از یک برنامه در برنامه دیگر استفاده کنیم در زبان های دیگر از، زیر برنامه استفاده می شود برای اینکار در Labview از کانکتور ویا آیکن استفاده می شود. آیکن یک نماد گرافیکی است که دارای تعدادی ورودی و خروجی است که تعداد ورودی و خروجی توسط خود برنامه نویس مشخص می شود. در ادامه یک مثال برای ایجاد کانکتور خواهیم زد.

یک مثال ساده برای چگونگی استفاده از صفحه ی پانل و بلوک دیاگرام:

می خواهیم جمع دو عدد را محاسبه کنیم.

در صفحه ی Front Panel از کنترلر ها گزینه Numeric واز آن numeric control, numeric indicator را انتخاب می کنیم، در صفحه ی block diagram نیز numeric واز آن گزینه ی add را انتخاب می کنیم و سپس مطابق شکل آن ها را به هم متصل می کنیم. بازدن Run برنامه را اجرا می کنیم.

۳. المان های کنترل کننده صفحه ی Front panel :



۳-۱. NUMERIC CONTROLS :

این پالت شامل المان هایی نشان دهنده اعداد، اعداد کنترلی، ترمومتر و... می باشد که مقدار آنها را می توان را برنامه نویس خود بدهد.

۳-۲. BUTTONS :

این پالت شامل المان هایی المان هایی همچون دکمه های ok, cancel و... می باشد.

۳-۳. TEXT CONTROLS :

این پالت شامل المان های file path control, Menu Ring, Text Ring, sting control می باشد که اگر با زبان های ویژوال آشنا باشید کار با آنها را به راحتی متوجه می شوید.

۳-۴. Numeric Indicator :

این پالت شامل همان المان های Numeric Controls می باشد با این تفاوت که مقادیر نشان دهنده خود را از برنامه می گیرند.

۳-۵. LEDS :

این پالت شامل دو شکل مختلف LED، Square LED و round LED می باشد.

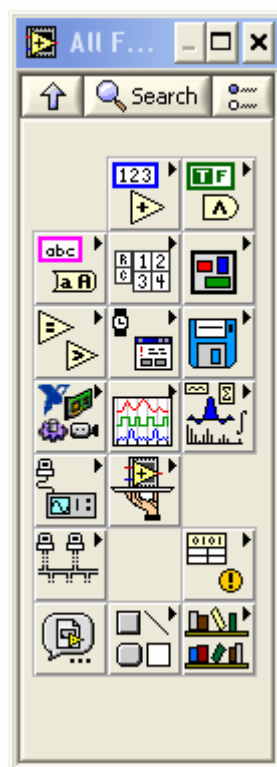
۳-۶. TEXT INDICATOR:

این پالت نیز همان المان های Text controls را دارد با این تفاوت که فقط نشان دهنده هستند نه گیرنده.

۳-۷. GRAPH INDICATOR:

شامل waveform graph, waveform chart و Express xy graph می باشد.

۴. FUNCTION های صفحه ی بلوک دیاگرام



۴-۱. NUMERIC:

این پالت شامل المان هایی همچون زیر پالت های ضرب، جمع، تقسیم، جذر، round و.... است. برای استفاده از این زیر پالت ها فقط دو المانی که قرار است روی آن عمل انجام می شود به ورودی های زیرپالت مربوطه متصل کرد.

۴-۲. BOOLEON:

این پالت شامل المان هایی همچون زیر پالت های منطقی and, or, xor, nand, است.

۴-۳. **STRING**:

زیر پالت های مربوط به اعمال قابل انجام بروی string است.

۴-۴. **ARRAY**:

۴-۵. **CLUSTOR**:

۴-۶. **COMPARTION**:

شامل دستورات عملگری همچون انواع نا مساوی <=>، <=، >=، و دستور select و..... است.

۴-۷. **TIME&DIALOG**:

شامل دستورات wait unit Next ms Multiple, Tick count(ms), wait (ms) و.... می باشد که در ادامه بعضی از آنها توضیح داده شده است.

۵. کنترل و اجرای برنامه با استفاده از ساختارها:

ساختار ها از مهم ترین قسمت های برنامه نویسی Labview می باشد. این اجزا بر روند اجرای یک برنامه نظارت دارد، و آنها را فرمان دهی می کند. در این قسمت چهار ساختار مهم labview آشنا خواهیم شد.

۱-while loop:

۲-for loop:

۳-case structure:

۴-sequenc e structure:

۶. حلقه ها:

اگر تا به حال با زبان های دیگر برنامه نویسی کرده اید احتمالاً با دستورهایی همچون FOR,WHILE آشنا شدید.

در نرم افزار Labview دو ساختار for loop, while loop جهت تکرار بخشی از برنامه در نظر گرفته شده اند.

۶-۱. **For loop**:

یک حلقه for کدها و دستورات داخل چهارچوب حلقه را که به آنها Subdiagram گویند به تعداد دفعات کران اجرا می کند این عدد کران دار به ترمینال شمارشی حلقه نسبت داده می شود برای تنظیم مقدار

دهی به این ترمینال می توانید عددی از خارج از حلقه به ترمینال مذکور سیم کشی کنید . واضح است اگر عدد صفر به این ترمینال سیم کشی کنید حلقه اجرا نخواهد شد. در شکل زیر یک حلقه For به همراه ترمینال تکرار و شمارشی آن ملاحظه می کنیم .

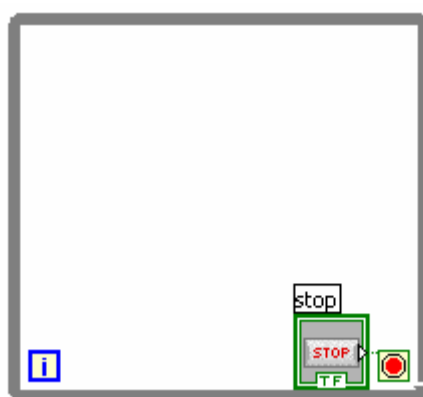
ترمینال تکرار تعداد حلقه هایی را که تا کنون تکمیل شده اند نشان می دهد بدین ترتیب که عدد صفر در خلال اولین تکرار حلقه در این ترمینال قرار می گیرد عدد یک در طول دومین تکرار حلقه و به همین ترتیب تا $n-1$. n عددی است که به ترمینال شمارشی حلقه سیم کشی است و به تعداد آن حلقه به اجرا در می آید .

عملکرد حلقه For معادل به کدهای فرضی زیر است .

For i=0 to N-1
Execute subdiagram

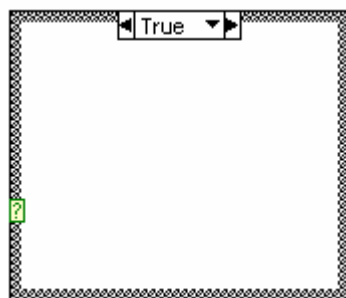
۲-۶. FOR WHILE

در حلقه While تا زمانی که مقدار جبری مربوط به حلقه به حالت False تغییر وضعیت نداده باشد دستورها و کدهای موجود در داخل چهارچوب اجرا می شوند این مقدار جبری از خارج از چهارچوب حلقه به ترمینال شرطی سیم کشی می شود . Labview مقدار موجود در ترمینال مذکور در پایان هر تکرار حلقه بررسی می کند و در صورتی که این مقدار در حالت True باشد حلقه یک بار دیگر تکرار می گردد مقدار جبری پیش فرض برای این ترمینال False می باشد . بنابراین در صورتی مقداری به سیم کشی نکرده باشید دستورات داخل چهارچوب حلقه فقط برای یک مرتبه به اجرا در می آیند ترمینال حلقه در While loop دقیقاً شبیه همان ترمینال در For loop رفتار می کند . در شکل زیر یک حلقه While به همراه دو ترمینال تکرار و شرطی آن نشان داده شده است . عملکرد حلقه While معادل با کدهای فرضی زیر است .



Do
Execute subdiagram (which sets condition)
While condition is True

۷. ساختار شرطی :



یک ساختار شرطی **روشی** است برای اجرای متون شرطی در محیط labview است. عملکرد این ساختار نظیر دستور "if-then-else" در سایر زبان های برنامه نویسی متداول متنی است. این ساختار در زیرپالت function >> structure قرار دارد.

همانگونه که از شکل پیدا است این ساختار از چند شرط یا حالت تشکیل شده است که در اصطلاح به هریک از آنها Case گوئیم. بسته به مقداری که به ترمینال انتخابی سیم کشی کرده ایم .تنها یکی از آنها قابل اجرا است.

اضافه نمودن CASE:

در منوی کرکره ای برروی دیواره ی CASE چند گزینه جهت افزودن، کپی کردن

۷-۱. محاسبه جذر یک عدد:

در این تمرین ساختار شرطی پنجره محاوره ای را مورد بررسی قرار می دهیم در این برنامه قصد داریم تا ریشه دوم یک عدد مثبت را محاسبه کنیم در صورتی که عدد ورودی منفی باشد یک پنجره محاوره ای باز می شود .

با انتخاب گزینه New یک برنامه جدیدی را باز می کنیم .

صفحه پانل این برنامه مطابق شکل انتخاب می کنیم .

عدد ورودی توسط المان Digital control که با برچسب Number مشخص شده است در برنامه اعمال می شود و المان Digital Indicator با برچسب Squart root value جذر عدد ورودی نشان می دهد بهتر است با انتخاب گزینه Format & precision از منوی کرکره های این المان تعداد ارقام اعشاری آن را افزایش دهد .

حال به صفحه نمودار بلوکی تغییر وضعیت می دهیم و دستورها و گره های نشان داده شده در شکل در این صفحه ایجاد می کنیم .

ساختار شرطی یا Case structure را از زیرپالت Function >> structure انتخاب نموده سپس آن را بر صفحه نمودار بلوکی قرار می دهیم نظیر آن چه که در مورد For loop & While loop عنوان شد پس از انتخاب این ساختار از زیر پالت مذکور نشانگر موس را در نقطه از صفحه نمودار بلوکی که قصد دارید گو

شه بالایی سمت چپ ساختار در آن محل قرار گیرد. کلیک کرده سپس موس را به صورت قطری حرکت داده و بدین ترتیب اندازه چهارچوب آن را تعیین کنیم.

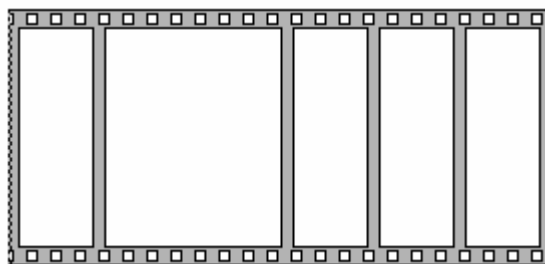
۷-۲. دستور SELECT:

دستور select در labview شبیه به دستور "if-then-else" در زبان های برنامه نویسی متداول متنی است. عملکرد این دستور با ساختار شرطی نیز مشابه است. دستور مذکور در زیر پالت Function >> comparison قرار دارد.

در صورتی که ورودی S در وضعیت true باشد این دستور مقدار اتصال یا فته به ورودی t را به خروجی اعمال می کند اگر ورودی s در وضعیت False باشد مقدار متصل شده به ورودی F به خروجی اعمال می شود با این تفاوت که در این حالت پنجره محاورهای باز نمی شود در شکل زیر برنامه محاسبه جذر یک عدد با این دستور آماده است.

۷-۳. Sequence Structure:

تعیین نمودن اجرای یک برنامه توسط آرایش دادن المانهای آن در ترتیب ها و مراحل مشخص را در اصطلاح (جریان کنترل) گویند. این اصل، جز جدا نشدنی زبانی برنامهنویسی متنی نظیر gc basic است. زیرا سطرها برنامهم به همان ترتیبی که در برنامه قرار دارند به اجرا در می آیند. در lab view برای اجرای این اصل از ساختار ترتیبی یا sequence structure استفاده می شود. در این ساختار هر یک از مراحل را که باید به ترتیب به اجرا در آیند، یک frame گویند. در این ساختار ترتیبی، ابتدا frame صفر به اجرا در آمده، سپس frame1، frame2 و... همانند یک frame فیلم اجرا می شود، این ساختار در زیر پالت function >> structure قرار دارد. هنگامی که برای اولین بار این ساختار را از زیر پالت structure انتخاب می کنید و آن را روی صفحه نمودار بلوکی قرار می دهید تنها دارای یک frame است. و در بالای این ساختار هیچ عدد یا پیکانی که نشانگر شماره این فریم و وجود فریم های دیگر باشد دیده نمی شود برای ایجاد فریم های جدید منوی کرکرهای را بر روی دیواره ساختار باز نموده، یکی از دو گزینه add frame after یا add frame before را انتخاب کنیم. همانگونه که از نام این ساختار استفاده می شود، از ساختارهای ترتیبی جهت کنترل ترتیب اجرای گره هایی استفاده می شود که برای اجرا شدن به داده های گره های دیگر وابسته نیستند، ترتیب اجرای گره ها در داخل هر فریم توسط وابستگی به یکدیگر تعیین می شود. برخلاف ساختارهای شرطی هر case بایک خروجی داشته باشد در ساختارهای ترتیبی توئل های خروجی تنها می توانند یک منبع داشته باشند. این خروجی می تواند از هر یک از فریم ها نشئت گرفته باشد. اما به خاطر داشته باشید در صورت انجام کامل ساختار داده ها به خارج از آنجا انتقال می یابد و نه در زمانی که فریم های مجزا به پایان می رسد. در ضمن داده مربوط به توئل ورودی برای تمام فریم ها معتبر است.



۸. زمان بندی در برنامه:

در برخی موارد لازم است تا زمان اجرای کنترل کنید. برای انجام این عمل می توان از گزینه های `wait(ms)`، `wait unit Next ms multiple`، `tick count(ms)` استفاده نمود. گزینه های مذکور در زیر پالت `Functions>>Time&Dialog` قرار دارند. در تمامی گزینه های مذکور تاخیر بر حسب میلی ثانیه تعیین می شود. به کار گیری گزینه `wait(ms)` سبب می شود که اجرای برنامه با تاخیر انجام گیرد.



دستور `wait unit Next ms Multiple` باعث می شود Labview تا زمانی که کلاک داخلی مساوی یا بزرگتر از مقدار ورودی `Millisecond multiple` نشده است قبل از اجرا نمودن برنامه منتظر بماند. این دستور جهت اجرا نمودن حلقه ها در فواصل زمانی مشخص و همچنین همزمان کردن فعالیت ها بسیار مفید و کاربردی است این دو دستور با یکدیگر مشابهند اما یکسان نمی باشد. به عنوان مثال دستور `wait unit Next ms Multiple` بسته به مقدار کلاک در هنگام شروع ممکن است کمتر از زمان مشخص شده در اولین حلقه منتظر بماند.



دستور `Tick Count` مقدار کلاک داخلی سیستم عامل رایانه را بر حسب میلی ثانیه به دست می دهد.



۹. گره ی فرمولی:

گره ی فرمولی شامل یک چهارچوب است که قابلیت تغییر اندازه دارد و با استفاده از آن می توانید فرمولهای ریاضی را در صفحه نمودار بلوکی وارد نمایید به کار گیری این سلختار در مواردی که نیاز به حل یک معادله پیچیده داشته باشید بسیار مفید و کاربردی است به عنوان مثال معادله بسیار ساده $y =$

x^2+x+1 را در نظر بگیرید در صورتی که قصد داشته باشید این تساوی را از دستورهای معمول و متداول محاسباتی در Labview محاسبه کنید دنبال نمودن آن در صفحه نمودار بلوکی اندکی مشکل تر از حالتی است که این معادله با استفاده از معادله های ریاضی وارد می کنید در شکل زیر استفاده از آن در این مورد نشان داده شده است .

اما با استفاده از گره فرمولی می توان به راحتی چنین معادلات را حل کرد. برای این منظور کافی است فرمول مورد نظر را در داخل چهارچوب ساختار گره ی فرمولی وارد نمایید . برای تعیین متغیرهای ورودی یا خروجی کافی است یکی از دو گزینه Add input یا Add output را از منوی کرکره ای بر روی چهار چوب ساختار گره ی فرمولی انتخاب کنید . سپس نام متغیر ها را وارد نمایید . در انتهای هر فرمول حتماً باید از کارکتور ؛ استفاده شود . گره ی فرمولی در زیر پالت Functions >> structure قرار دارد.

ترسیم منحنی $y=\sin(x)$:

در این تمرین قصد داریم تا معادله $y=\sin(x)$ را با استفاده از ساختار گره فرمولی اعمال نموده منحنی آن را رسم کنیم .

یک برنامه جدید باز کنید سپس گزینه Waveform graphe را زیر پالت graph >> Control را انتخاب کنید و آن را بر روی صفحه پانل قرار دهید و سپس آن را با برجسب Graph نام گذاری کنید . منحنی معادله $y = \sin(x)$ بر روی صفحه این گراف به نمایش در می آید .

صفحه نمودار بلوکی برنامه را مطابق شکل زیر ایجاد کنید . به کمک Formula node (>> Function structures) می توانید فرمول های ریاضی را مستقیماً در صفحه نمودار بلوکی وارد کنید . جهت ایجاد متغیر ورودی ، منوی کرکره ای را بر روی دیواره ساختار گره فرمولی باز کنید و گزینه Add input را انتخاب نمایید . برتی ایجاد متغیر خروجی نیز به طریق مشابه گزینه Add output را از همان منو انتخاب کنید . نام این متغیر ها باید دقیقاً با متغیر های انتخاب شده در فرمول هماهنگی داشته باشد . عدد ثابت π در زیر پالت Additional Numeric Constant >> numeric >> Functions قرار دارد . در خلال هر یک از تکرار های حلقه ترمینال تکرار در عدد ثابت $(\pi/10)$ ضرب می شود . نتیجه حاصل ضرب این دو مقدار به گره فرمولی سیم کشی می شود و مقدار سینوس آن را محاسبه می کند سپس نتیجه را در یک آرایه در دیواره For loop ذخیره می شود . آرایه مذکور پس از اتمام اجرای حلقه For ترسیم می شود .

حال به صفحه پانل باز گردید و برنامه را اجرا کنید . توجه داشته باشید که در این تمرین به جای به کار گیری گره فرمولی می توانی از تابع Sine در زیر پالت Trigonometric >> numeric >> Function قرار دارد نیز استفاده کنید . اما این نکته را نیز به خاطر داشته باشید که در محیط Labview برای هر فرمول مورد نیاز شما یک دستور از پیش نوشته شده وجود ندارد .

```

For i=0 to 99
X=i*( $\pi$ /10)
Y=sin(x)
Array[i]=y
Next i
Graph (array)

```

۱۰-۱. آرایه

۱۰-۲. کلاستر :

این ساختار در نرم افزار lab view متناظر با record و struct در زبانهای برنامه نویسی c و pascal است. برای درک بهتر مفهوم کلاستر آن را به صورت دسته ایسیم در یک کابل مخا برای در نظر بگیرد. هر یک از انتقال داده های کلاستری از یک رشته سیم نشان داده می شود.

۱۱. گراف و نمودار:

۱۱-۱. نمودار:

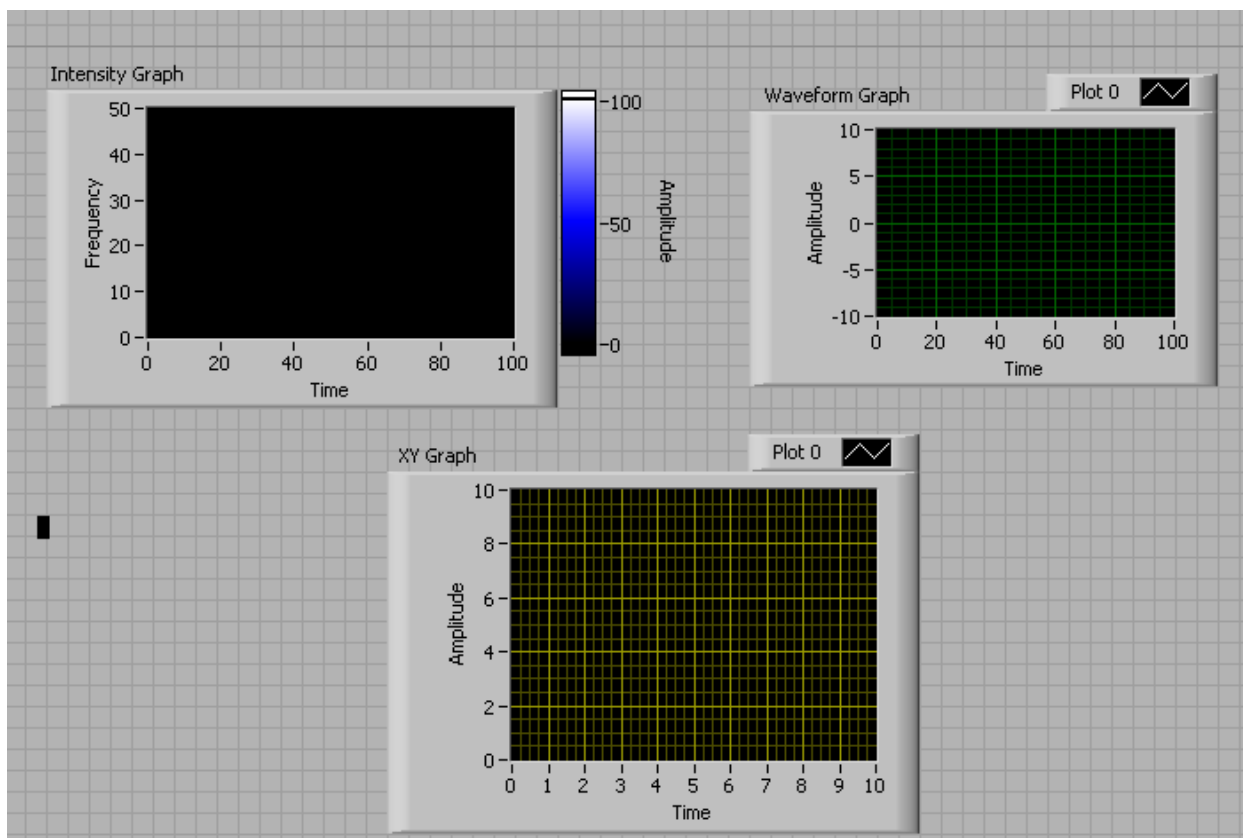
به بیان ساده نمایش دادن مقادیر y در مقابل مقادیر x را نمودار گویند. در اکثر نمودارها مقدار داده ها و نتایج بدست آمده بر روی محور y به نمایش در می آیند. حال آن که مقادیر x برای نشان دادن زمان به کار می روند.

۱۱-۱-۱. مدهای مختلف نمایش نمودار :

در نمودار شکل موج جهت به روز رسانی داده ها از سه مد مختلف استفاده می شود. این حالت عبارتند از `sweep chart mode`, `strip chart mode`, `scop chart mode`. برای تعویض مدها میز کور منوی `data operation s>>update` را زبزن منوی `waveform chart` باز کنید و مد دلخواه را زبزن منوی `strip chart` یک صفحه نمایش دهنده لغزشی است. مدهای `sweep scop` دارای صفحات نمایش دنباله کننده شبیه به یک اسیلوسکوپ هستند.

۱۱-۲. گراف:

بر خلاف نمودار که داده ها مجدداً به همراه داده های قبلی بر روی صفحه نمایش ترسیم می کنند. در گراف آرایه ای از پیش تولید شده یک جا رسم می گردند. به عبارت دیگر قابلیت به روز رسانی داده ها در گراف وجود ندارد. برای افزایش قابلیت انعطاف و کاربرد بیشتری در این المان ها در نرم افزار `labview` سه نوع گراف تعبیه شده است. این گراف ها عبارتند از `waveform graph`، `xy graph`، `intensity graph` هستند.



هر دو نوع گراف waveform و xy در زیر پالت graph >> controls قرار دارند. گراف شکل موج یا waveform graph برای ترسیم تابع یک به یک استفاده می شود. xy graph یک گراف همه منظوره در مختصات دکارتی می باشد. و برای ترسیم داده هایی که به صورت پراکنده قرار می گیرند یا مبدا زمانی متفاوتی دارند ایده آل است.

از این نوع گراف همچنین برای رسم داده هایی نظیر اشکال دایره ای که در آنها به هر مقدار x چند مقدار y اختصاص می یابد، استفاده می گردد. هماهنگی که ذکر شد شکل ظاهری این دو نوع گراف یکسان می باشد. اما نوع داده های ورودی آنها متفاوت است. بنابراین در هنگام کار با آنها مراقب باشید.