Summary: This Study Guide will continue where we left off last week. What I mean is, we are going to just take what we learned last week and expand on it a little bit to get into more detail on how to configure the ConsoliDator+.

As with last week, following along with the recorded session is strongly recommended, and you can also follow along with the ConsoliDator+ Configuration Software and a ConsoliDator+ Demo Unit if you have one (if not, let's talk)!

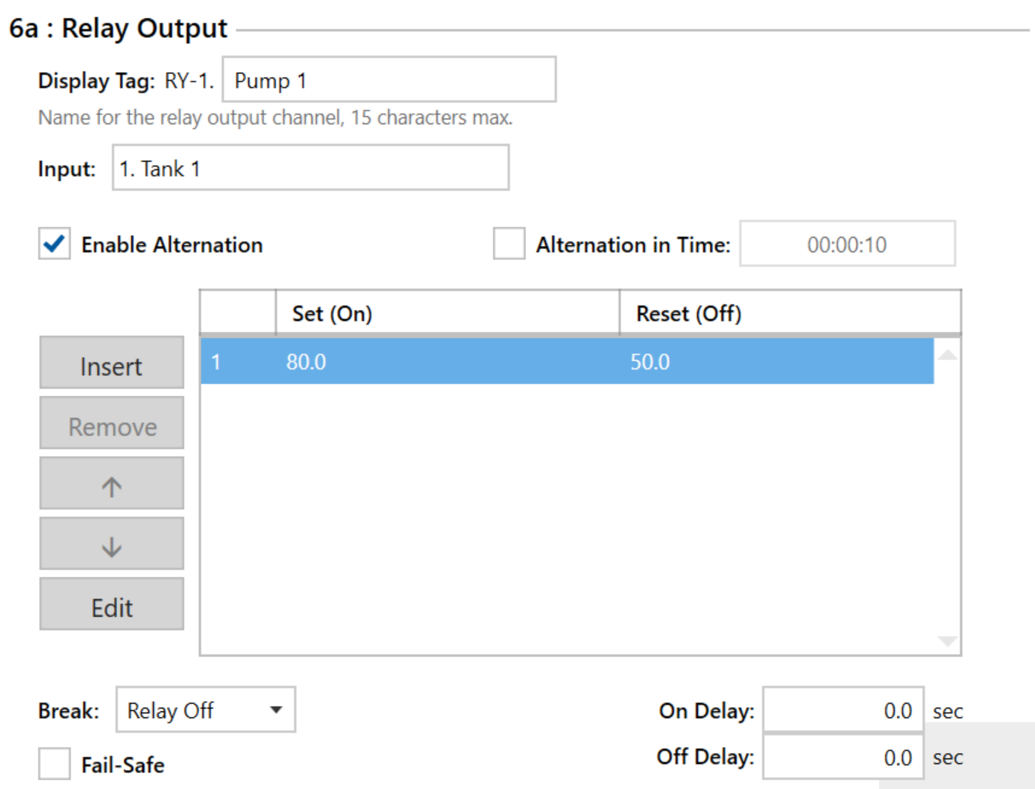
How to Configure Duplex Pump Controllers: One of the VERY first applications we took a look at in our first session was a screen that performed a "Duplex Pump Control" application. Another way to say this is, the ConsoliDator+ can act as a pump alternator!

This is usually done based off a level reading, so let's just assume we have a channel, "Tank 1" which is scaled from 0-100% capacity.

The first thing we need to do is identify which relay outputs on the unit will be connected to which pump. Once that is determined, label the relays so they are named, "Pump 1" and "Pump 2" or whatever you want to name them!

After we name the relays, go to the first relay and make sure the "Input" is our "Tank 1" channel so that we are programming the relay based on level conditions.

After that, you will notice a checkbox that says, "Enable Alternation". By default, this box is unchecked, so we need to activate the alternation. When you do that, the relay screen will appear like:



6a : Relay Output

Display Tag: RY-1. Pump 1

Name for the relay output channel, 15 characters max.

Input: 1. Tank 1

☑ Enable Alternation ☐ Alternation in Time: 00:00:10

	Set (On)	Reset (Off)
1	80.0	50.0

Break: Relay Off

☐ Fail-Safe

On Delay: 0.0 sec

Off Delay: 0.0 sec

After that, we want to put in our setpoint and reset point for when we want the pumps to turn on and off. In this case, we want a pump to turn on at 80% of capacity, and turn off once the tank has drained out to 50% capacity.

NOTE: *With how this is setup now, pump 1 will turn on at 80%, and run until the tank reads 50%, regardless of how long it takes. Then, the next time the tank gets to 80%, it will be pump 2 that kicks on and pumps until the tank reads 50%, regardless of how long that takes. And it will just keep going like that forever and ever.*

Now, there is another checkbox that appears which is labeled, “Alternation in Time”. With this feature enabled, this will allow the pumps to alternate based on pump run time OR level setpoint, whichever happens first.

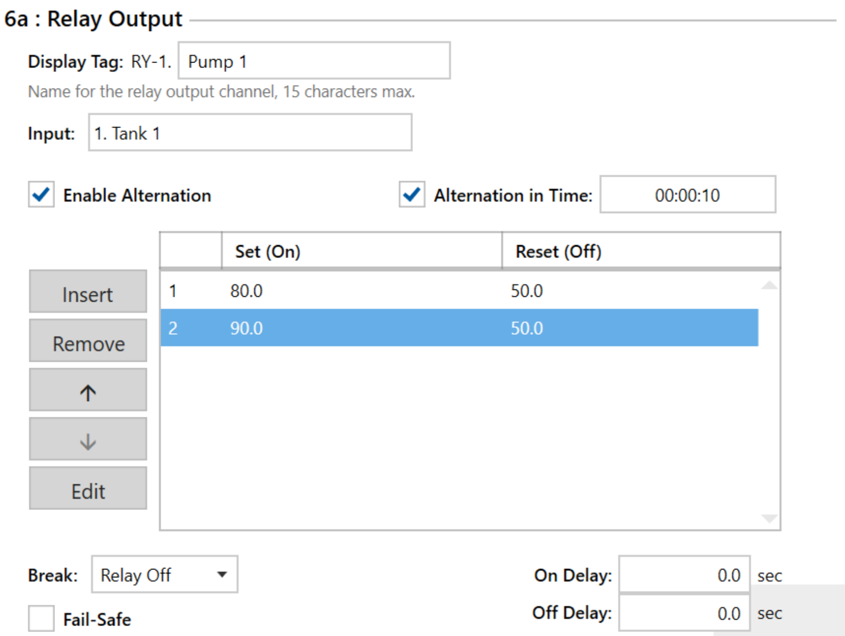
So, if we activate that checkbox, and put in 10 seconds (like it says in the picture) that means when pump 1 turns on, it will run until the level gets to 50% OR for 10 seconds, whichever occurs first. If pump 1 cannot drain the tank to 50% before the time elapses, then pump 1 will turn off, and pump 2 will take over. Pump 2 will follow the exact same pattern as pump 1. The cycle will continue until the tank remains at an acceptable level.

Most of the time with this type of setup, folks will program an even higher setpoint just in case one pump cannot drain quickly enough. Think of it this way…

The tank gets to 80% capacity and a pump comes on. But the water is coming in faster than the pump can remove it, so instead of going down, the level continues to rise. To prevent this from happening, we can program a secondary setpoint so that when reached, BOTH pumps will come on at the same time

Let’s pretend that our second setpoint will be at 90%. If the pump cannot drain out the tank quickly enough, then at 90% capacity, pump 2 will turn on to help out and share the load.

To do that, we want to click “Insert”, and then enter our set and reset point. The setpoint will be 90%, and the reset point can still be 50% so that both pumps are running together until the tank gets to 50%.



6a : Relay Output

Display Tag: RY-1. Pump 1

Name for the relay output channel, 15 characters max.

Input: 1. Tank 1

☑ Enable Alternation ☑ Alternation in Time: 00:00:10

	Set (On)	Reset (Off)
1	80.0	50.0
2	90.0	50.0

Break: Relay Off

□ Fail-Safe

On Delay: 0.0 sec

Off Delay: 0.0 sec

To recap, the way this is setup is that pump 1 will turn on at 80% capacity. It will run until the tank reaches 50%, or 10 seconds elapses.

If the time elapses before the reset point (50%) is reached, pump 1 will turn off, and pump 2 will take over.

If the reset point is reached BEFORE the time elapses, pump 1 will turn off.

Then, the NEXT time our tank reaches 80% capacity, it will be pump 2 that will come on and pump until the tank reaches 50% or 10 seconds elapses – whichever occurs first.

OR

If our tank reaches 80% capacity, pump one will turn on. But, if the level continues to rise and it reaches 90% capacity, pump 2 will kick on, and they will BOTH pump out the tank together until the reset point (50%) is reached (the elapsed pump run time does NOT affect anything if BOTH pumps are running).

But wait a second… I keep talking about 2 pumps, yet we only setup ONE relay?!

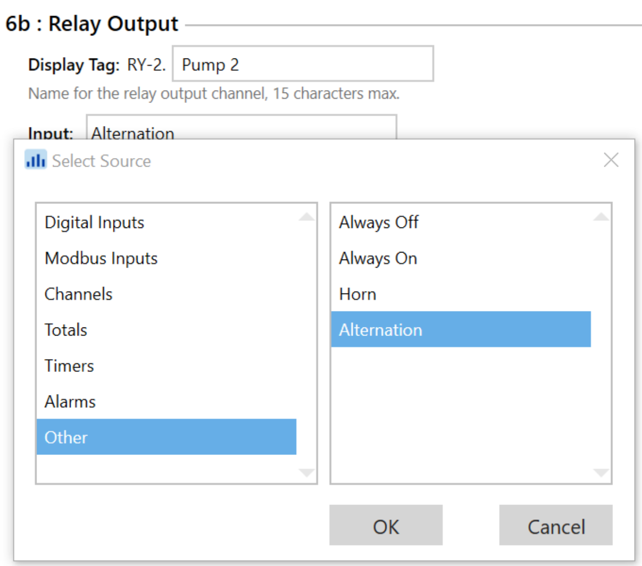
That's right! All the logic and control parameters are configured on the main relay, and the other relays that we will tie into this will just take direction from the main relay.

So, how exactly do we tie other relays into this scenario?

Well, click on your "Pump 2" relay so we can configure it.

Then, select the input. This part can seem a little tricky at first, but it's very simple.

The "Input" to this relay that we want is going to be in the "Other" category. The function we are looking for is "Alternation".



6b : Relay Output

Display Tag: RY-2. Pump 2

Name for the relay output channel, 15 characters max.

Input: Alternation

Once we select the “Alternation” function, we now have to tell the relay what it needs to alternate with! That only makes sense, right? Otherwise, the relay is being told to alternate with absolutely no context on which to base its alternation. It’s like a C Major triad without an E… without the major third, it just won’t have the same emotion to it, and it will sound bland to our ears.

Luckily for us, the ConsoliDator+ Configuration Software makes it EXTREMELY simple to do this!

When we select the right function, another box will appear that says, “Alternate With”.

Now, in this case, there is only going to be ONE option, and that option will be “Pump 1” (or, relay 1, but we named it Pump 1, remember?).

Now this new relay will take its instruction from that Pump 1 relay where we configured the set and reset points of everything!

To make it fancy you COULD create alarms so that the bar graph changes colors at certain points, but we’ve covered how to do that in previous sessions.

At this point, that’s how we setup a duplex pump controller! It’s as simple as that!

How to Configure TRIPLEX Pump Controllers: Since we can alternate between TWO pumps, what’s stopping us from alternating between three pumps?

Absolutely NOTHING is stopping us from doing that, and on the ConsoliDator+, it literally takes about 30 seconds to turn a duplex pump controller into a triplex pump controller.

Ready?!

It’s SUPER simple!!

Select your third relay, choose “Alternation” for its function, and then make sure it’s “Alternating With” Pump 1!

NOTE: *Remember that all the logic and instruction is coming from “Pump 1”. So, you may intuitively think this third pump should be “Alternating With” pump 2, but that’s not how it works out. If Pump 3 is alternating with pump 1, then it is also alternating with pump 2.*

Could you add a fourth or fifth pump to this line-up? Yes, you can add more than three pumps if you’d like, but I have yet to see that in my experience in the industry. Have you?

How to Configure Leak Detection: One of my favorite features of the ConsoliDator+ is the ability to act as a leak detector in a level application. You were able to see how that works on my demo unit during a previous session, but now I’m going to show you exactly how I configured that screen!

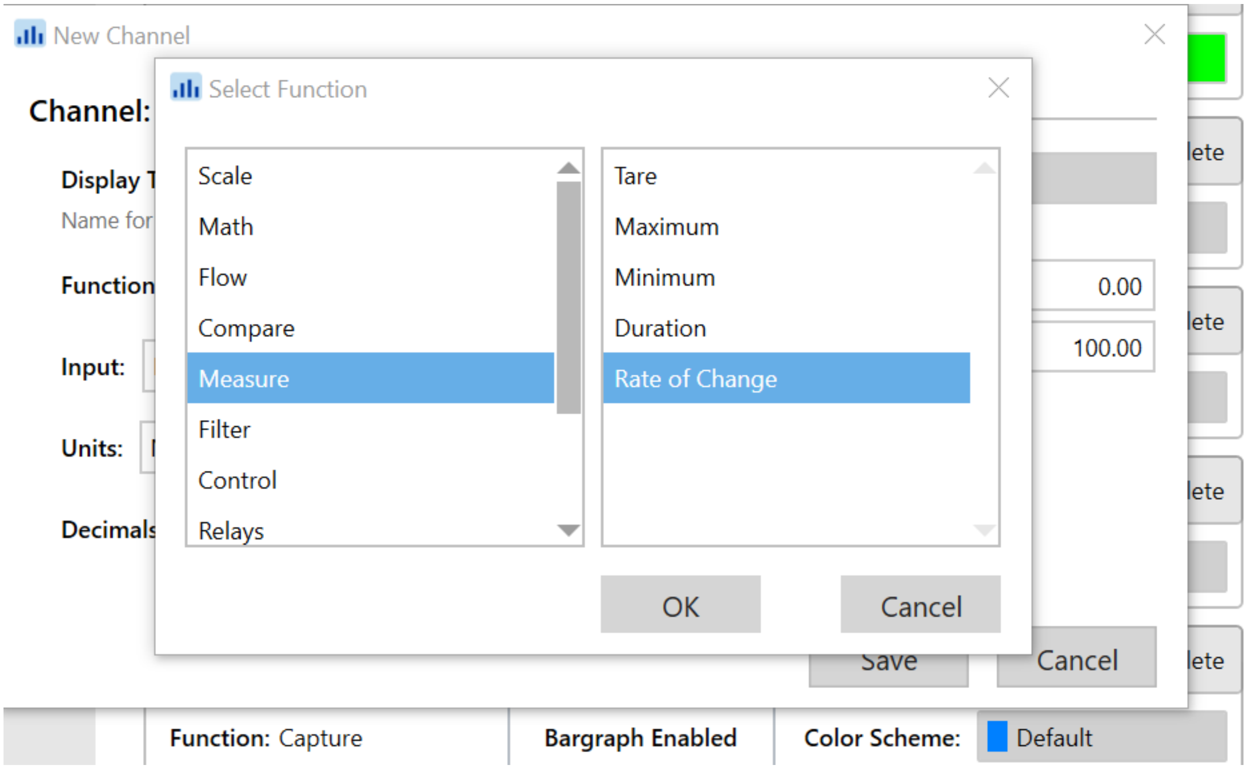
Let’s say we have a tank with a maximum capacity of 1,000 gallons. There is a valve at the bottom of the tank for emptying, and a pump to fill the tank when need be.

Also at the top of the tank is an Eclipse 706 guided wave radar transmitter giving the ConsoliDator+ a nice 4-20 mA signal. That will be the ONLY feedback we are receiving from the tank.

So, how the hell will we detect a leak if the ONLY input is an analog signal from the level sensor?

Well, we are going to use our “Rate of Change” function on a channel. Basically, that channel is going to just monitor whichever channel you want, and report the rate at which the process changes. This will be expressed in unit/time (gallons/sec, liters/day etc.).

So, we have to create a NEW Channel with the “Rate of Change” function. This can be found in the “Measure” function category as seen below:



Function: Capture

Color Scheme: Default

Once you select your function, you choose which input this channel will monitor the rate of change. So, in this example, we want our 1,000 gallon tank channel to be the “Input” here. Select that channel as the input.

The last thing to do after that is to make sure the time base of the rate is in minutes, as the number will be rather large if you use seconds.

NOTE: *Since this is a RATE channel, you must make sure the units of this channel agree with the units of the tank channel. If you measure the tank in Feet, then we can’t measure the rate of change in gallons/time unless you were to perform a unit conversion. It is possible to do that, but that involves creating another channel, and it’s just much easier to make sure your units agree!*

Now that we have the channel programmed, we need to understand one simple thing before we continue.

As the tank’s volume increases, the “Rate of Change” will show a positive value. When the tank’s volume decreases, it will show a negative value.

NOTE: *It is possible to turn this into a channel with “Absolute Value” as the function so that we ONLY get positive values, but for the purpose of leak detection, we actually WANT negative values.*

So, what we need to do next is create a new “Alarm” so that we can actually KNOW a leak is occurring!

We already went over how to configure alarms in the last session, so we won’t go through the entire process. I will just show you how to configure an alarm for the “Rate of Change” channel since we will have to use negative values for our setpoint!

Creating the alarm is going to be the same as any alarm, so to review how to set everything up step-by-step, please review the Study Guide from "Week 8 - Intermediate Configurations".

Once you open the alarm creation window, enter the following values:

Type – Single Source

Input – The "Rate of Change" channel

Setpoint – Any negative value of volume/time that makes sense for your application

Reset – 0 volume/time

The reset point on this is actually critical. Think about what we are doing here.

We are wanting to alarm based on a RATE of change from a storage tank. If the tank loses volume, we get a negative rate. If the tank is filling, we get a positive rate.

So, the reason why we set the reset to "0 volume/time" is because the alarm will automatically go away the second the leak stops, or as soon as the tank starts increasing in volume. So, let's pretend we made our reset point a positive number. Let's say we made it (positive) 100 gallons/minute.

Well, that means the leak alarm would remain active until you start increasing the volume of the tank by 100 gallons/minute, and that may NOT happen when you first stop the leak. So, it just makes sense to use "0" as the reset value on a leak alarm – in this particular scenario.

Once you have your alarm programmed, the last step is to tie a relay output to that alarm, or not! The choice is totally up to the user, but most folks would appreciate some sort of audible or visual alarm if their tank is leaking, and the ConsoliDator+ can do that for them!

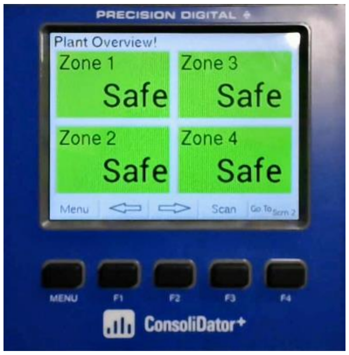
NOTE: *By utilizing the ability to make "Common Alarms", we can actually have leak detection on SEVERAL tanks, and still only have ONE relay output trigger if ANY of the tanks leak.*

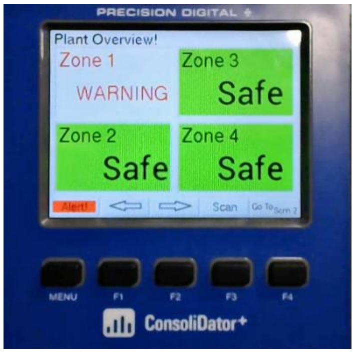
How to Configure "Annunciator Screens": During almost all of the sessions in the first half of the class, I would configure at least ONE "Annunciator Screen". Now, this term is just something I coined for the ConsoliDator+ and it does NOT necessarily mean it's only to be used as a traditional annunciator screen.

I call it the "Annunciator Screen" simply because that's what it LOOKS like, and it's easier to communicate that.

Basically, when I say "Annunciator" screen, I mean a screen with NO BARGRAPHS that is showing you discrete status. Typically, it's just an ON/OFF type of channel.

For example, you saw me use one of these during our "Gas Detection Applications" session and you can see the example slide below:





In the example slide, you notice that the channels on the screen are only ever showing two pieces of information… "WARNING" or "Safe". Because we only have those two options, and in this case the messages are triggered by an alarm setpoint, I call them "Annunciator Screens".

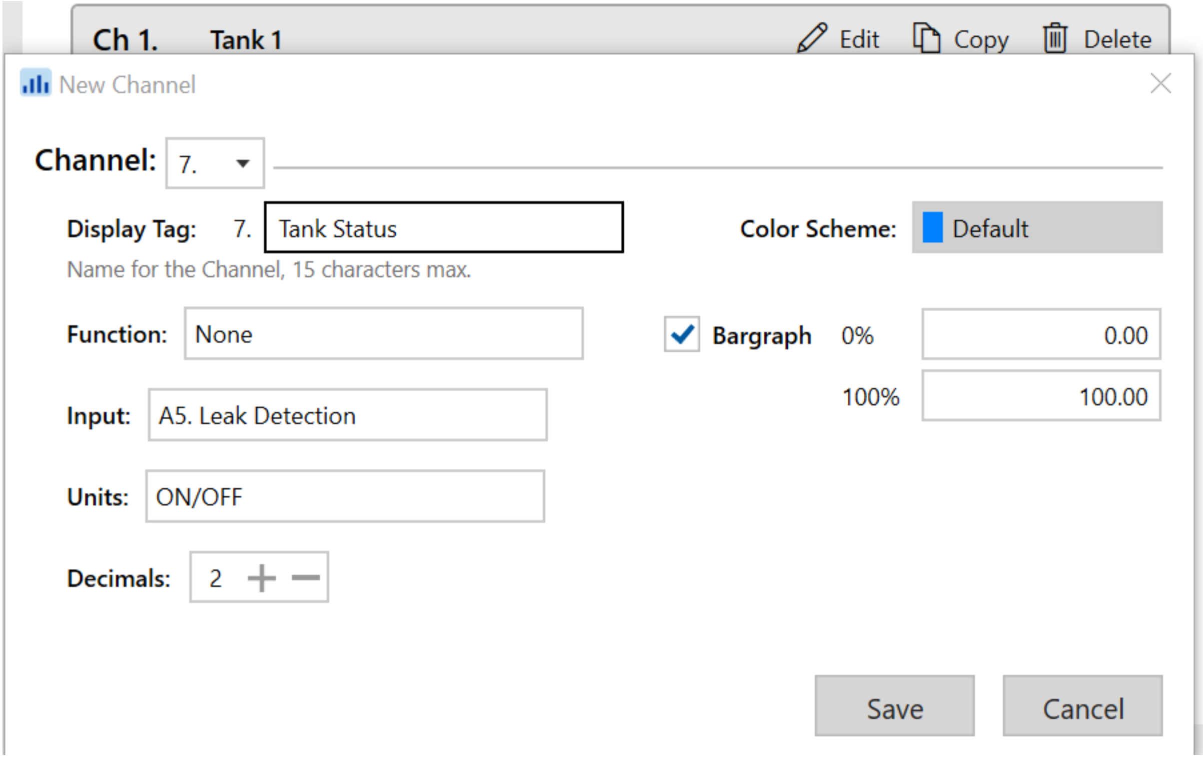
Hopefully that makes sense, and now that my "definition" is out of the way, let's take a look at how I set up these types of screens!

For this example, let's actually make an "Annunciator Screen" based off our leak detection alarm that we just learned how to create. Basically, we will make a screen that simply shows a channel labeled something like, "Leak Status" and the message will switch between "LEAK" and "No Leak".

The first thing we need to do is to actually create a new "Channel". However, unlike the channels we have created in the past, the "Input" to this new channel is actually going to be the leak detection alarm we just created.

Also, unlike any other channel we have created, the "Function" is going to be "NONE".

It'll look something like this:



Channel: 7.

Display Tag: 7. Tank Status

Name for the Channel, 15 characters max.

Function: None

Input: A5. Leak Detection

Units: ON/OFF

Decimals: 2

Color Scheme: Default

☑ Bargraph 0% 0.00

100% 100.00

You will notice that the "Units" default to "ON/OFF". The reason this happens is because since the "Function" is none, the ConsoliDator+ knows that we are trying to setup an "Annunciator Screen" and they gives us the typically used, "ON/OFF" units.

That means that when our alarm is triggered, the channel message will say "ON", and when the alarm is not triggered, it will say, "OFF".

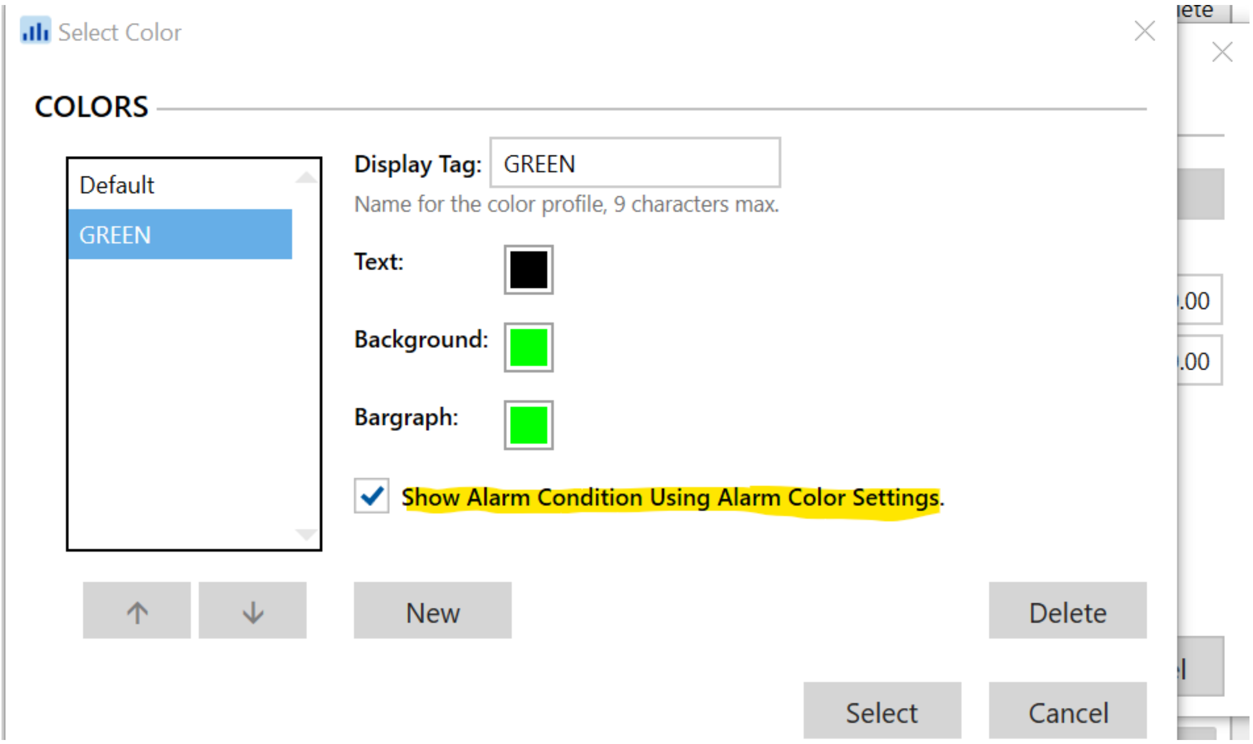
However, since we are allowed to create custom units, we can get pretty fancy with our channel messages (I call them "messages", but really, they're just units that are being displayed)!

So, you can either choose from the various “message” units we have on the device which can be found in the “Logic” category of the “units library”, or you can simply create custom units like I do with these. For example, “WARNING” is a pre-programmed unit, but I made a custom unit for the “Safe” message (as seen in the example slides on the previous page).

NOTE: *The “Decimals” on this type of channel are irrelevant since we are not displaying any numbers on this type of channel. You can just leave that alone.*

Just to spice things up a bit and make the screen itself look really nice, I typically like to use a custom color palette for this type of channel. For the most part, I use a GREEN background with black text. That means under normal operating conditions, the channel will be GREEN (again, as seen in the example slides in the previous section).

To really tie everything together, when you create (or choose) your GREEN color palette, be sure that the checkbox for alarm color changing is checked. See below for reference:



Select Color

COLORS

Default
GREEN

Display Tag: GREEN
Name for the color profile, 9 characters max.

Text: ⬛

Background: 🟩

Bargraph: 🟩

☑ Show Alarm Condition Using Alarm Color Settings.

Without that box checked, when our leak detection alarm triggers, the color of this channel will NOT change. Yes, the message will change as it should, but to really take advantage of the ConsoliDator+, having the channels change color on alarm is a really nice thing to add!

There is only ONE last step before this channel is completed!

Because this channel is meant to just show the status of a tank, as opposed to a continuous level reading, we should make sure the bargraph is removed from this channel by making sure the “Bargraph” checkbox is UNCHECKED (it will be checked by default, so be careful).

Now, after you save the channel, to take full advantage of this type of channel, you should also be sure to put these channels on a “Screen”. I typically don’t like to mix these types of channels with continuous readings, so I will usually make a completely separate screen for these types of channels!

Now, you can create multiple “Annunciator Channels” and have UP TO eight of them being displayed on a single screen. During our classes, I have only used a maximum of 4 objects per screen just to keep things looking simple, but just a reminder that EIGHT can fit on a single screen!

How to Configure AND/OR Alarms: One of the most flexible features on the ConsoliDator+ is the ability to configure “AND” and/or “OR” alarms (NAND and NOR also available).

If you recall during our “Gas Detection” session me mentioning the “common alarms”, the AND/OR alarms was actually what I was configuring. I simply called them “Common” alarms because it aligned with gas detection nomenclature.

Now, another thing to consider about these alarms before we move ahead any further is the fact that they don’t necessarily HAVE to be used for alarming!

What I mean is, we can use the “Alarms” to trigger certain actions and to use logic to tell the ConsoliDator+ what to do. In fact, it may be helpful for you to think of these as “EVENTS” as opposed to an alarm. Basically, when we create an alarm, we are writing a conditional statement!

For example:

“IF the level of this tank reaches X, THEN output Y”

When you think of ConsoliDator+ alarms in those terms, you’ll see that the possibilities are practically endless!

So, let’s take this just one step further. Let’s say we want to create an “AND” alarm. How would the conditional statement be written?

Example:

“IF the level of this tank reaches X AND the pressure drops to X2, THEN output Y”.

Meaning, this alarm will not trigger unless BOTH conditions are true. If ONLY the tank were to reach a certain point but the pressure didn’t move, then there would be no alarm… or, there would be no “EVENT” that happens.

Now, let’s take a quick look at how the conditional statement for an “OR” alarm would be written:

Example:

“IF the level of this tank reaches X OR the pressure drops to X2, THEN output Y”.

Meaning, this alarm will trigger if either the tank reaches a certain setpoint, or the pressure drops to a certain point. If only ONE of those is true, the alarm will still trigger. If BOTH of them are true, the alarm will still trigger.

With this new way of thinking, let’s take a look at how we actually “tell” the ConsoliDator+ how to respond to certain “EVENTS”.

The first thing that needs to be done to program these types of alarms is to actually have two (or more) alarms already created. Basically, these “AND/OR” alarms are meant to combine multiple alarms that have been configured.

You can think of it this way…

A “Single Source” alarm is just ONE conditional statement (IF this, THEN that).

Combining these statements with "AND/OR" commands just helps us easily communicate our logical thinking into terms that the machine can understand. We are simply combining conditional statements to have much more flexibility over the control logic.

So, for this example, let's say we are monitoring TWO tanks in a plant. We will measure them in 0-100% capacity to keep things simple, and we want to know when either of those tanks reaches 80% capacity.

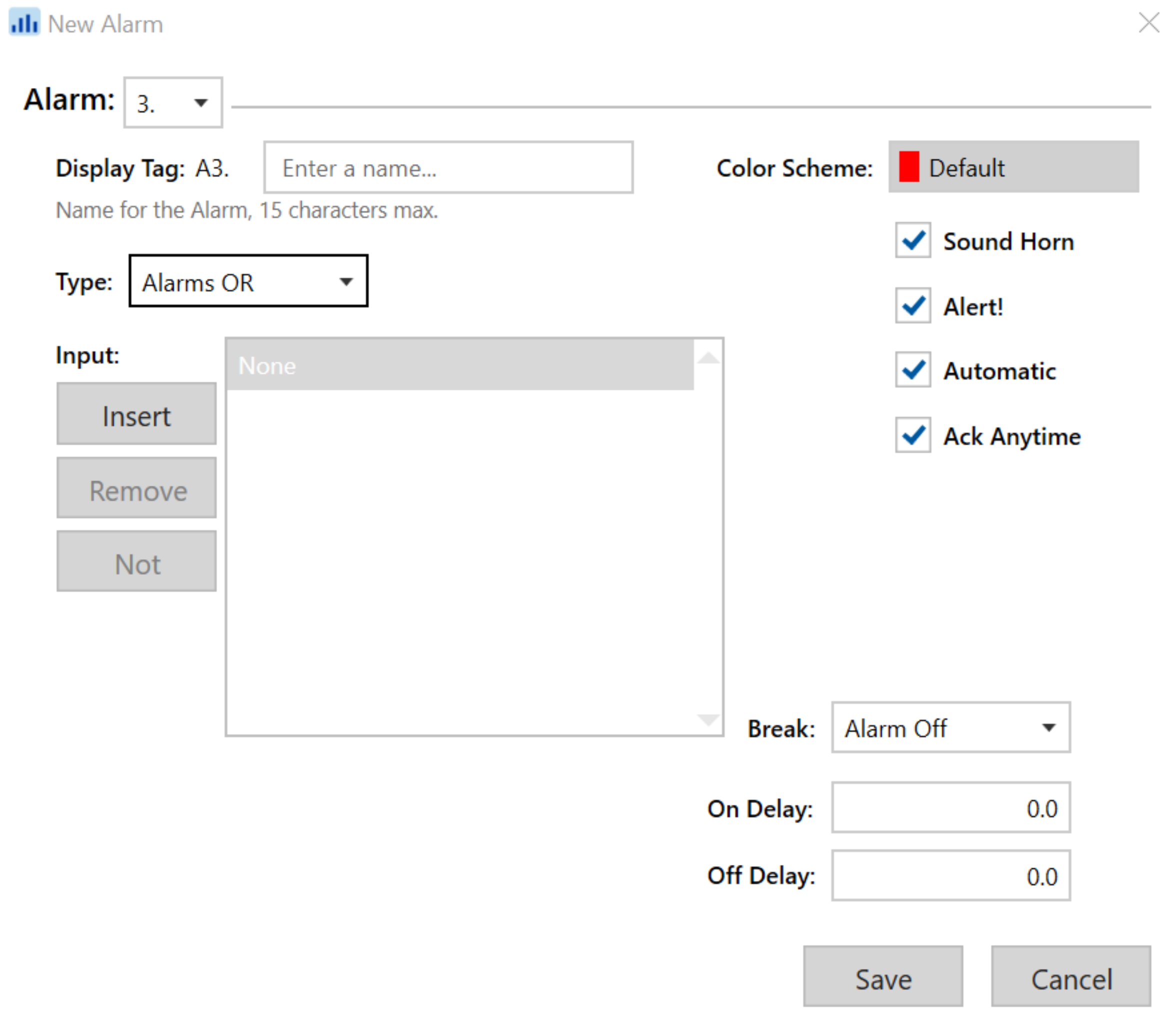
For this example, it will not matter WHICH tank gets to 80%, we just want to know if EITHER of them have.

The first thing to do is create a normal "Single Source" alarm for each tank channel. So, you can make the setpoint 80%, and make the reset point whatever you want. I usually choose 50% just for simplicity's sake!

They do NOT need to be added to any screen for them to still work. These objects CAN work in the "background".

NOTE: *For a reminder on how to create alarms, please refer to the Study Guide for "Week 8 – Intermediate Configuration".*

Once you have created both of the "Single Source" alarms, now it's time to create a NEW alarm object, except we are going to want the "Type" to be, "Alarms OR". Once you select that type, you will notice the screen changes a bit from what a typical Single Source alarm looks like.



Alarm: 3.

Display Tag: A3. Enter a name...

Name for the Alarm, 15 characters max.

Color Scheme: 🟥 Default

☑ Sound Horn

☑ Alert!

☑ Automatic

☑ Ack Anytime

Type: Alarms OR

Input: None

Break: Alarm Off

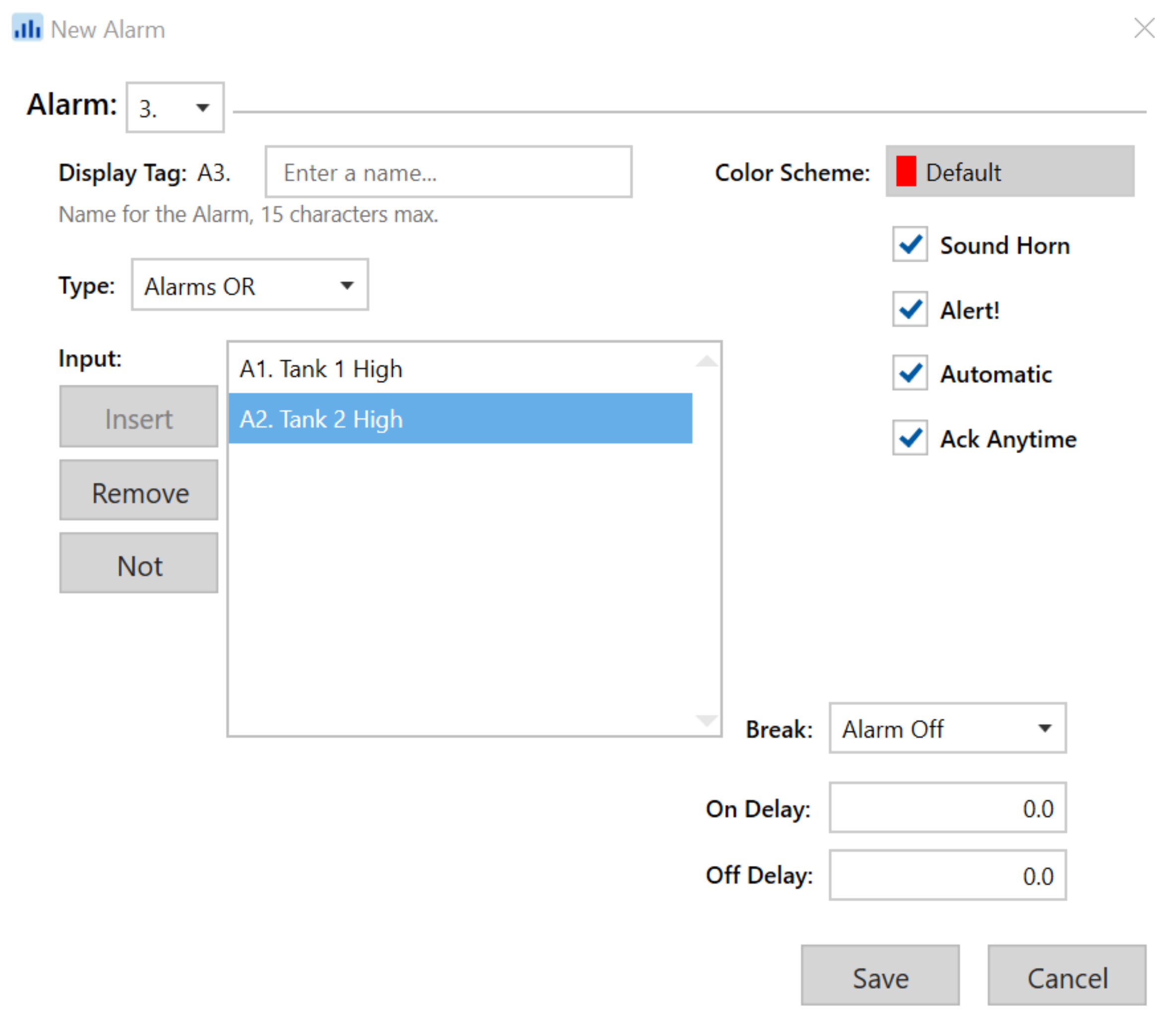
On Delay: 0.0

Off Delay: 0.0

Once you have selected the correct "Type", you can now start "Inserting" the other alarms that you would like to have on this alarm. In this example, we want the high alarms for our two tanks to be on this alarm.

You can simply press, "Insert" and look through the list of compatible alarms. You will notice, unlike other objects, ONLY alarms can be inserted into this "OR" alarm.

Once you select the two single source alarms you want to tie together, the screen will look like this:



New Alarm

Alarm: 3.

Display Tag: A3. Enter a name...

Name for the Alarm, 15 characters max.

Color Scheme: 🟥 Default

☑ Sound Horn

☑ Alert!

☑ Automatic

☑ Ack Anytime

Type: Alarms OR

Input:
- A1. Tank 1 High
- A2. Tank 2 High

Break: Alarm Off

On Delay: 0.0

Off Delay: 0.0

There are a couple things to notice here. For starters, you will notice that we are not entering any setpoints or reset points. That is because that was done at the “Single Source” alarm level. Remember, this “Alarm” is simply a way for us to combine our “Conditional Statements” to perform complex control!

The other thing to notice is a button which says, “NOT”.

Although this operation could fill up an entire section on its own, we can briefly review what this really clever little button does for us!

NOTE: *To truly understand this, I would highly suggest doing a bit of learning “Gate Logic”, because that is where this type of logic originates.*

For this brief overview, you can think of that “NOT” button as inverting our logic, or, adding another condition to our “Conditional Statement”.

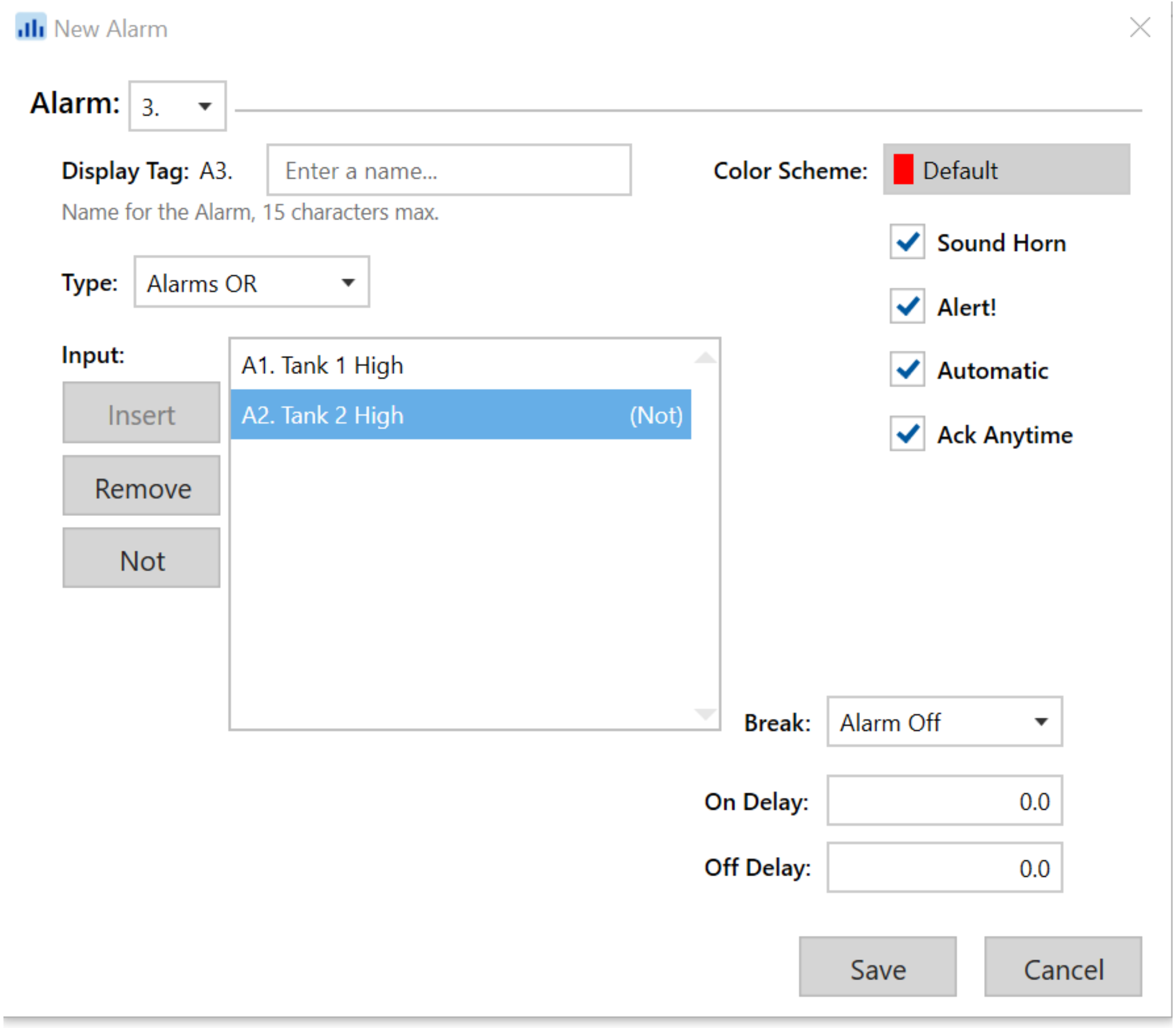
What in the world am I going on about here?

Let’s review what our “Conditional Statement” is saying with that OR alarm we just made.

“IF tank 1 OR tank 2 reach 80% capacity, THEN trigger the alarm”.

Okay, now let’s see how our conditional statement will change if we make “Tank 2 High” a “NOT”.

First, see what that looks like below:



New Alarm

Alarm: 3.

Display Tag: A3. Enter a name...

Name for the Alarm, 15 characters max.

Color Scheme: 🟥 Default

☑ Sound Horn

☑ Alert!

☑ Automatic

☑ Ack Anytime

Type: Alarms OR

Input:

A1. Tank 1 High

A2. Tank 2 High (Not)

Break: Alarm Off

On Delay: 0.0

Off Delay: 0.0

You will notice that “Tank 2 High” now has a “(Not)” next to it. This indicates that the user has selected to change the conditional statement with regard to that specific alarm.

Now our “Conditional Statement” says something like:

“IF Tank 1 is at or above 80% capacity BUT Tank 2 is at 80% or below, THEN trigger the alarm”

Or you could think of it this way.

The “Single Source” alarms can be considered as “TRUE” or “FALSE”. If the alarm is triggered, we can call it “TRUE”, and when the alarm is not triggered, we can call that “FALSE”.

In order for this “OR” alarm to work, “Tank 1 High” MUST be true, and “Tank 2 High” MUST be false. If and when those conditions are met, our “OR” alarm will trigger.

If BOTH of those alarms are “TRUE”, then the “OR” alarm will NOT trigger.

This concept can be a little tricky to explain and understand, and if you have any questions, please feel free to reach out.

Just by going over this, hopefully you can see just how flexible this device can be. And, not only that, but think of the kind of logic statements we can make with this device.

For example, what if you had TWO “OR” alarms configured?

Well, you can actually turn those two "OR" alarms into ANOTHER "OR" alarm, or you can combine them into an "AND" alarm.

The possibilities are almost endless.

If you wanted to create an "AND" alarm, you can follow the exact same steps! The only thing that really changes with the "AND" alarms is how our "Conditional Statement" is written.

For example, if we chose an "AND" alarm for our "Tank 1 High" and "Tank 2 High" our conditional statement would read like:

"IF Tank 1 AND Tank 2 are at or above 80% capacity, THEN trigger alarm.

In other words, BOTH of the "Single Source" alarms that are tied to this "AND" alarm MUST be true in order for the "AND" alarm to trigger.

How to Configure Open Channel Flow: Open channel flow has always been a popular application for Precision Digital products, except the ConsoliDator+ just takes us to the next level with its flexibility. If you have ever setup one of our ProVu totalizers for open channel flow, you'll know that there were some limitations.

For example, it was only capable of handling a couple different open channel flow tools (weir/flume) and it was a bit tricky to program… but we don't have to get into that here.

The ConsoliDator+ can handle a wide range of open channel flow applications, and I have personally configured a few of these devices for this exact application! In fact, a couple of those people are in this class with you!

There are a couple ways to approach this type of configuration, and it doesn't matter in which order you choose to do it. However, for this example, we are going to follow the order in which I personally configure these types of channels.

The very first thing we want to do here is setup a "Channel" for our "Head Height".

NOTE: *This is not a lesson on how open channel flow works, but I'd be more than happy to help explain it to you during a one-on-one.*

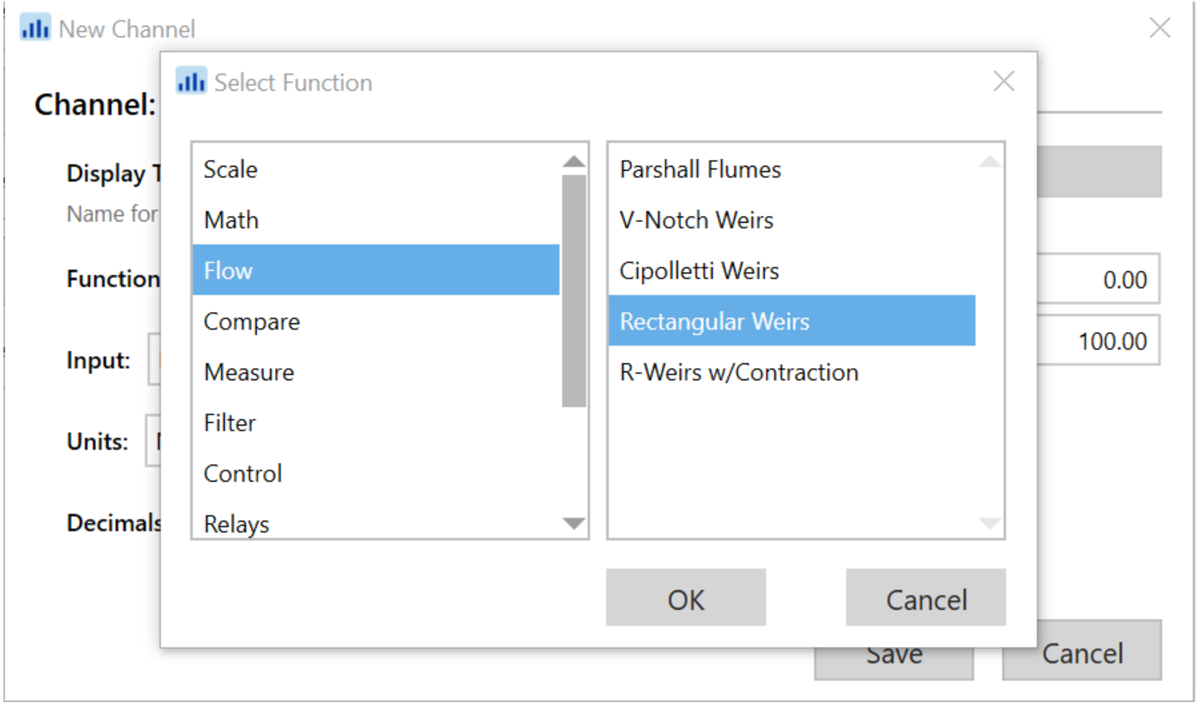
Another thing to keep in mind here is to make sure our units agree. Even if YOU measure your weir in inches, we need to make sure we scale our head height and width of the weir in FEET (this is to ensure the math works out properly. Trust me, I learned this the hard way!).

So, to create a "Head Height" channel, you simply just need to create a "Scale – 2 pnt linear" channel and simply scale the 4-20 mA signal for 0-X feet. That's it! The first part is done!

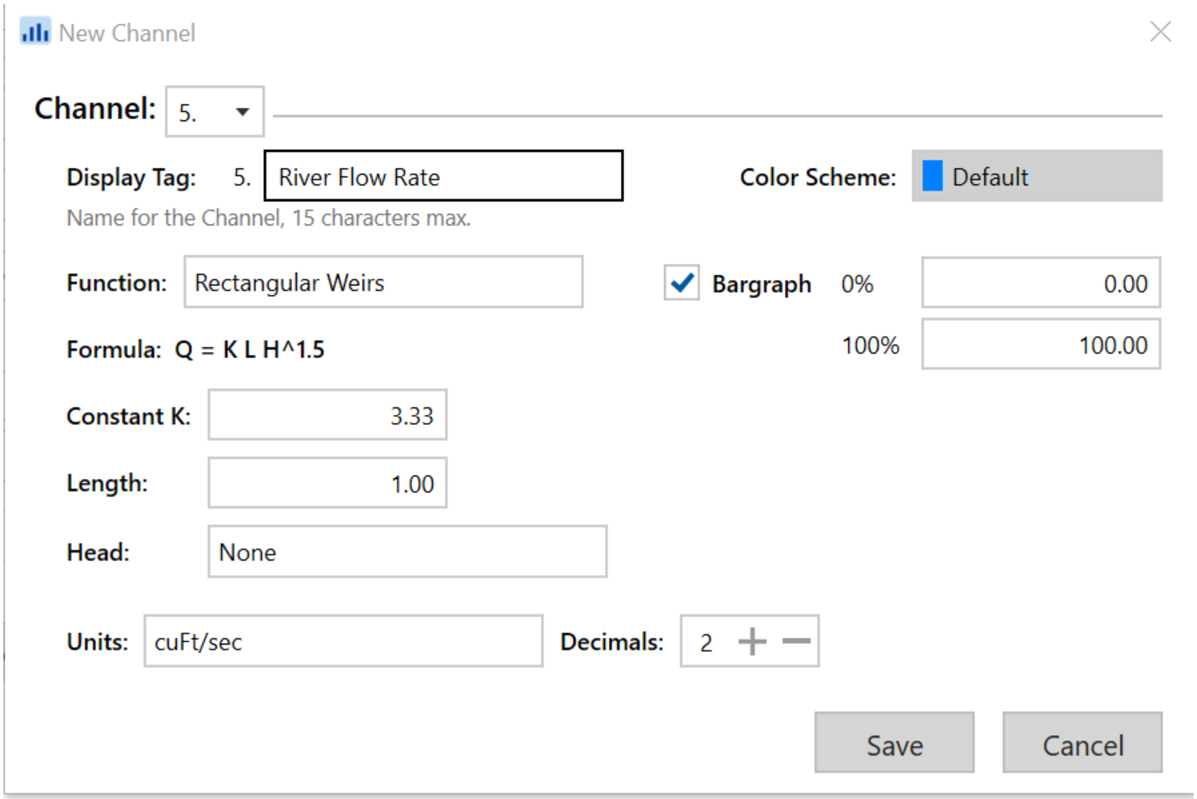
The next thing we want to do is actually create a channel where we can convert the signal from our level sensor into a flow rate! So, let's create another new channel!

However, when you create this channel, we are going to want to choose the "Flow" category in the list of functions, and then find which weir or flume you have at your facility.

For this example, we are just going to choose "Rectangular Weirs" (with no end contractions).



Once we select that, there will be a few other settings which can be adjusted, but it's VERY important to pay close attention to which ones you change. Making a mistake here can easily give you weird numbers that won't make any sense.



Channel: 5.

Display Tag: 5. River Flow Rate

Name for the Channel, 15 characters max.

Color Scheme: 🟦 Default

Function: Rectangular Weirs

☑ Bargraph 0% 0.00

100% 100.00

Formula: Q = K L H^1.5

Constant K: 3.33

Length: 1.00

Head: None

Units: cuFt/sec

Decimals: 2 + −

The FIRST thing I want you to notice here is that the "Units" is automatically filled out with "Cubic Feet per Second". For now, just leave that alone so I can explain why this happens.

The "Constant K" field is basically a "constant" number which is derived from the formula and it is completely dependent on the units in which you want to measure your flow rate.

If you look through your “ISCO Open Channel Flow Handbook” you will notice that the “3.33” number is used if we want to measure our rate in “Cubic Feet per Second”.

However, in my experience, folks typically want to measure open channel flow in millions of gallons per day, or “Megagallons” per day. Again, if you refer to the ISCO handbook, it’ll tell you that the constant for that measurement is “2.152”.

So, let’s assume for this exercise we want to measure our rate in “Megagal” per day. Now, logic may dictate that you would want to change that “Constant K” to “2.152” but that is NOT what we should do.

I repeat, that is NOT what we do here.

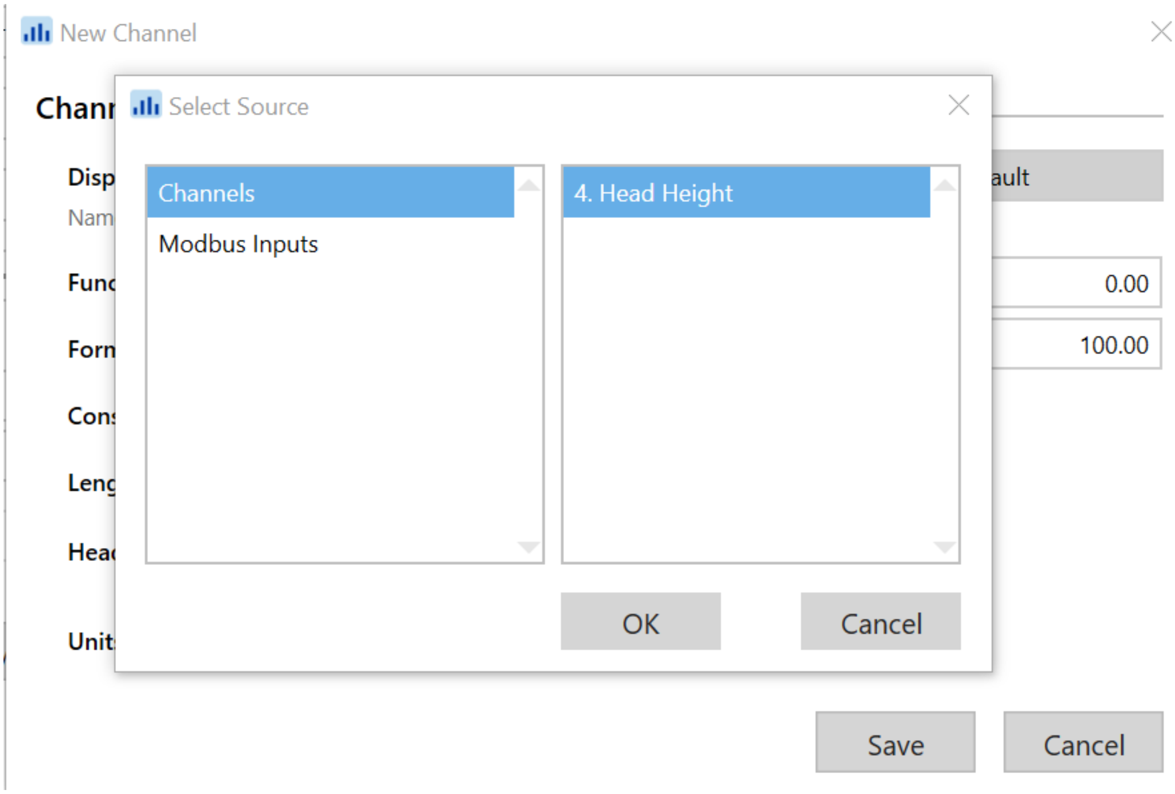
So, we are not going to touch that “Constant K” number, even though the software allows you to!

The next section to fill out is the “Length” of your weir (technically I believe it’s the width that we are concerned with to make sure we have a known geometry, but it’s labeled as “length” in the software).

Again, it is very important to make sure our units agree, remember? We scaled our “Head Height” in Feet, so we want to make sure the “Length” of our weir is also in Feet. For simplicity’s sake, let’s just pretend we have a 3-foot-wide weir.

The next thing to fill out is our “Head”. Remember, the “Head” is a completely separate channel that we made, so when you click on that box, you will be able to select one of any compatible channels available.

NOTE: *Because our units MUST match to make the math work out, the software will ONLY allow you to choose a channel with a unit in the “Distance” category (Feet, Inches, cm, etc.)*



Once you have selected the “Head Height” channel, there is only ONE last step to do, if applicable.

Remember how I said that we want to read our flow rate in Megagal per day? Well, the “Units” on this channel is still in “Cubic Feet per Second” so we need to change that to the unit we want!

NOTE: *“Megagal” is NOT a standard unit and will require you to create a “Custom Unit”. To review how to make a custom unit, please refer to the Study Guide for “Week 8 – Intermediate Configuration”.*

Once you change the unit to whichever unit you want, the ConsoliDator+ is going to do the unit conversion for you automatically. THAT is why we don’t want to touch that “Constant K” number. Basically, the ConsoliDator+ will calculate everything in terms of “Cubic Feet per Second” and then just does a simple unit conversion when you change the “Units”.

After you have that step done, you have just successfully configured an open channel flow rate…channel!

If you have been following along and trying to configure this channel, with a maximum Head Height of 2 Feet, a Weir Length of 3 Feet measuring in “Megagal” - at 2 Feet of Head Height, you should be flowing around 15.3 Megagal/per day!

Obviously people want to be able to totalize off this flow rate, and since you already know how to create a totalizer channel, you simply just create a new “Total” and make sure this open channel flow rate is the “Input” to that totalizer!

I know that was a lot of reading, but hopefully this all makes sense to you now! Of course, there are MANY other things the ConsoliDator+ is capable of doing, and I could go on for days and days with the different features and advantages.

However, the point of these last few sessions were not to teach you EVERY little thing about the ConsoliDator+ and how to program it. Rather, these last few sessions were designed to give you enough of a foundation to be able to go out, experiment, and figure out the rest on your own.

You now know more than 98% of the world’s population in regard to programming this device. Do you know EVERY possible thing? No, but I promise you that you now have the ability to go out and figure out how to make this device do what you need.

In fact, that’s how I learned this device! I didn’t have an instruction manual when this device was released. I didn’t have someone teach me anything. I quite literally asked the boss to borrow one of the two prototypes we had three years ago (a year before its release) and I sat down on a Friday night and poked around with it (sounds like a real FUN Friday night, huh?!).

If I can do it, YOU can too!

Sincerely,

“Professor” Devin Gates
Cell: (508) 683-9034
Email: dgates@predig.com