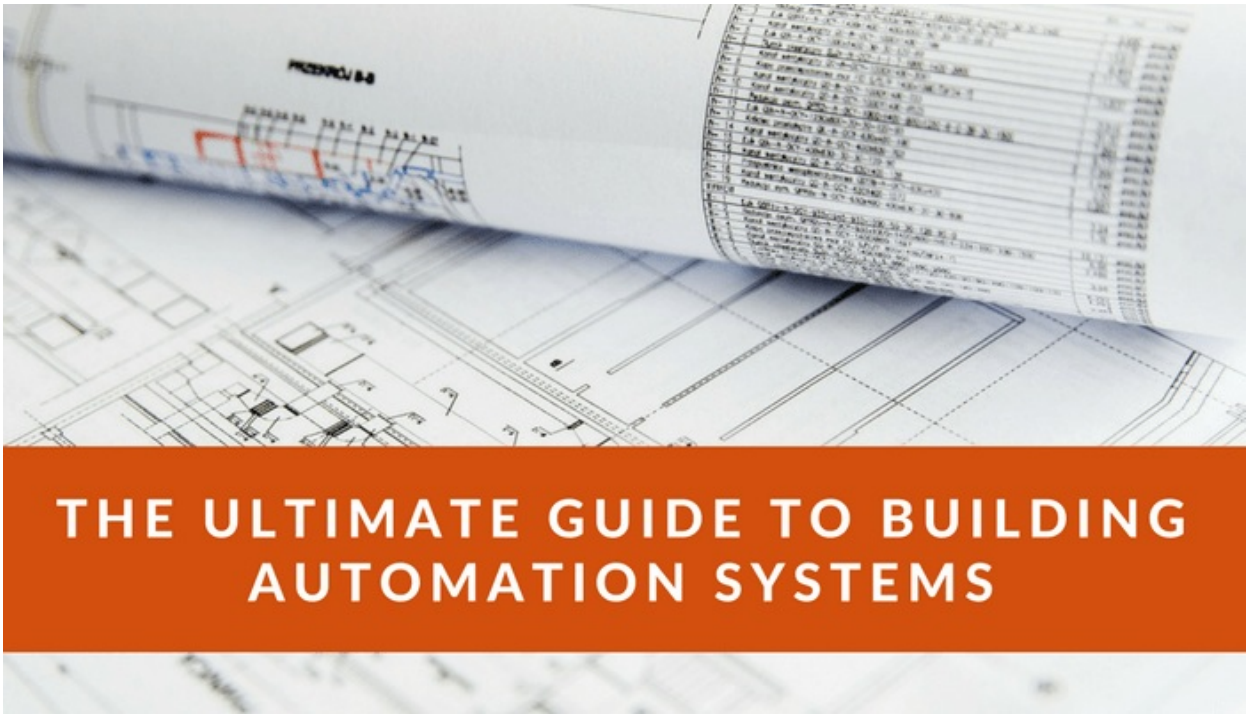
The Ultimate Guide to Building Automation Systems41 min read

buildingautomationmonthly.com/building-automation-system/

Phil Zito September 14, 2017



BEMS, BMS, BAS, EMCS, and this list goes on and on. Welcome the acronym-filled wasteland know as building automation.

You may be wondering, "What is a Building Automation System":

Even among "building automation" professionals there's a lack of consensus around what building automation is.

And this is exactly why I wrote this guide.

The purpose of this guide is to help you understand what building automation is.

No matter what your role is when you finish this guide **you will understand what a building automation system is and how it works**.

And that is where most "building automation 101" guides end.

But that is not where we end!

By the time you finish this guide you will understand:

- The real-world outcomes you can expect to achieve from a building automation system
- How the modern BAS systems came to be and why you should care
- A process for being able to look at BAS designs and intuitively understand them
- What the different pieces of a BAS system are used for

- The ins and outs of BAS software
- The stupidity of “cybersecurity fear mongering” and how to secure a BAS
- The secret to upgrading a BAS, without failing repeatedly
- Why trends, alarms, and reports are the most under-utilized feature of a BAS
- How to work with IT groups and why IT isn’t your enemy
- How to move into the world of BAS and how to develop your BAS teams (coming soon)

By now, you may be feeling overwhelmed. You may look at that list and say holy moley, that’s a ton of information.

And you know what, you’d be right. Building automation is a complex topic and this isn’t some 1,000-word guide designed to grab clicks and teach you nothing.

So here’s the deal:

I’m really good at simplifying complex topics, I like to think it’s my super power.

If you stick with me through this whole guide **you will leave with a massively increased knowledge of building automation**.

Sound good?

Let’s do this!

So what is a Building Automation System?

Great question right?

After all, that is the point of this guide.

When folks ask me what a building automation system is, I often ask them, what do you want it to be.

I know, that seems like a very evasive way to answer a question but think about it.

How many specifications have you read that called for a BEMS, BAS, BMS, EMS, EBMS, and my favorite the ABMAS (by the way that stands for Advanced Building Management Automation System)?

So what is a BAS?

Well in the simplest terms a building automation system, is a system that automates many of the tasks that are required to run HVAC. Yes, I know, why is it called a building automation system? Why don’t they just call it an HVAC control system?

Well, you want the truth?

You can sell a heck of a lot more Building Automation Systems then “HVAC control systems”.

In an ideal world, a real building automation system would control, HVAC, lights, access control, energy management, and much more.

But we live in the world of segmented contracting models where each manufacturer is contractually isolated from one another and you're lucky if you can get lighting and BAS to talk to one another.

Ok, ok, at this point some of you are getting antsy and want me to tell you more than a BAS controls HVAC.

So here you go.

A building automation system utilizes a control system to automate the control of various building systems (mainly HVAC). The BAS provides a user interface that allows the end user to adjust the control settings, view the system status, and detect any potential issues related to building system performance.

By the way, we will dive into control systems, user interfaces, and a lot more as we move through this guide.

A building automation system consists of four "layers".

These layers are:

- Server/Application Layer
- Supervisory Layer
- Field Controller Layer
- Input/Output Layer

Each layer of the building automation system serves a purpose and each layer builds upon the layer below it to provide more functionality and automation to the end user.

Outcomes that come from a BAS (marketing fluff land)

The reality is most BAS are the same. I can hear it now "Phil, that's not true, my BAS controller has whizbang, Wi-Fi, analytics, SQL features….".

Ok, maybe you do have that.

Give me $500,000 and a development team and I can replicate pretty much any feature you have in your BAS. That is not what makes or breaks a BAS manufacturer.

What really matters is the people and the processes. How you train and develop your people and how you execute your projects will allow you to outperform almost any technology (that is unless you're still installing modems and Windows 95).

Being that most BAS are the same, we can, or at least I can agree that the outcomes we will see from a BAS are largely the same.

Based on my experience working with tons of different BAS manufacturers across hundreds of projects I've discovered that the outcomes break down into four major areas.

These areas are:

- Life safety
- Uptime
- Energy savings
- Staff efficiency

So what do each of these outcomes mean?

Life Safety

Life safety is the ultimate purpose of any building system. At the end of the day if a system negatively impacts life safety then that system needs to be overhauled and fixed. **Life safety** quite simply is making sure that the health and well-being of building occupants are protected.

Uptime

Uptime is the amount of time that your BAS or the systems controlled by the BAS are up. When a system is down it's called **downtime**.

There are two types of downtime:

- Planned downtime
- Unplanned downtime

Planned downtime is ok, it's not ideal but it is necessary to perform maintenance. **Unplanned downtime** is BAD, this is when things are down because of failures or unplanned events.

Energy Savings

Energy savings, depending on where your building is this may or may not be a very important factor. The fact is energy savings, as an outcome ebbs and flows.

The outcome here is that the BAS will allow you to visualize and manage your building systems in such a way that you can create energy savings.

Staff efficiency

Finally, you have staff efficiency.

This is the ability of the staff to perform their day-to-day tasks in such a way that they are efficient and productive. Training has a huge factor in the success of staff efficiency. I address staff training in my article *How to create the ultimate project training plan*.

Control System vs a BAS

So you've heard me mention this term control system and you may be wondering "What is a control system and how is it different than a BAS?"

As I described above a building automation system, automates the functionality of a control system and provides a visualization component (think user interface and reporting).

These features allow building operators to know what is going on with their systems.

A control system is a subset of the automation system and is capable of operating independently of the building automation system.

I've actually seen a control system in a complex central plant use a time clock for scheduling and function completely independent of the building automation system.

Control systems exist to "control" the input/output and the field controller layer.

Depending on the type of control system you have you different device types.

However, there are some common "pieces" that you can expect to find:

Every control system, and ultimately almost every technology centric system, follows a pattern of Input => Process => Output. Control systems are no different.

In a control system, an input device provides a status or feedback signal to a "controller".

Depending on the control system type this could be a direct digital controller (these are the modern day BAS field controllers) or it could even be a simple pneumatic accumulator.

From here the "controller" will drive an output to perform a task.

It could be something as simple as turning a fan on when a wall switch is flipped. Or it could be as complex as controlling a wall of individually regulated fans (fan wall) based on the average of several different pressure sensors. Ultimately it doesn't matter, it all follows the pattern of:

Input => Process => Output

Pay attention because if you grasp what I say next you will be massively farther ahead in your knowledge of BAS than your peers.

It doesn't matter what type of control system you have.

You will be able to make sense of the system as long as you understand what type of inputs there are, what process they feed into (most likely a controller of sorts), and what output they connect to.

The era's of control systems and what they mean to you

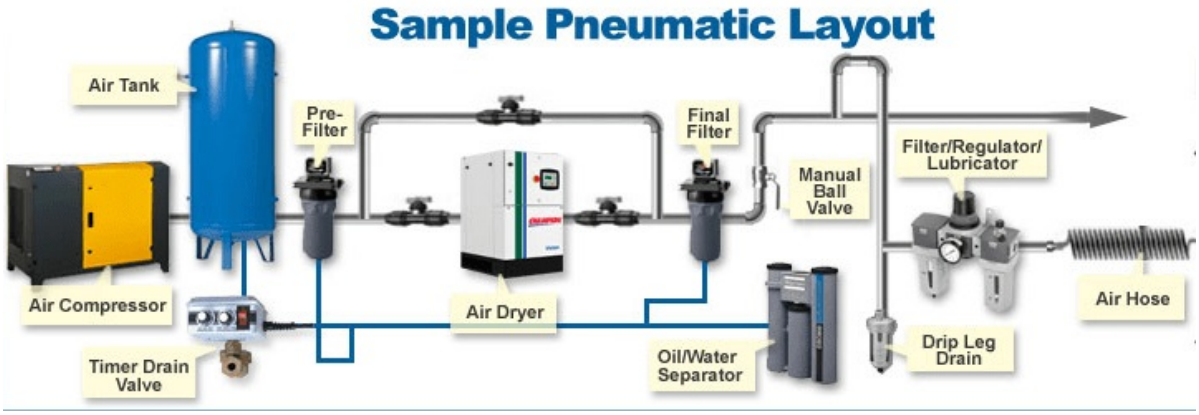
Ok, so what are the different flavors of control systems. So far I've named two of them, and in this section, I'm going to unpack the rest of them.

The control systems you will commonly encounter are:

- Pneumatics
- Analog

- Electromechanical
- Digital
- DDC
- Future

Pneumatics



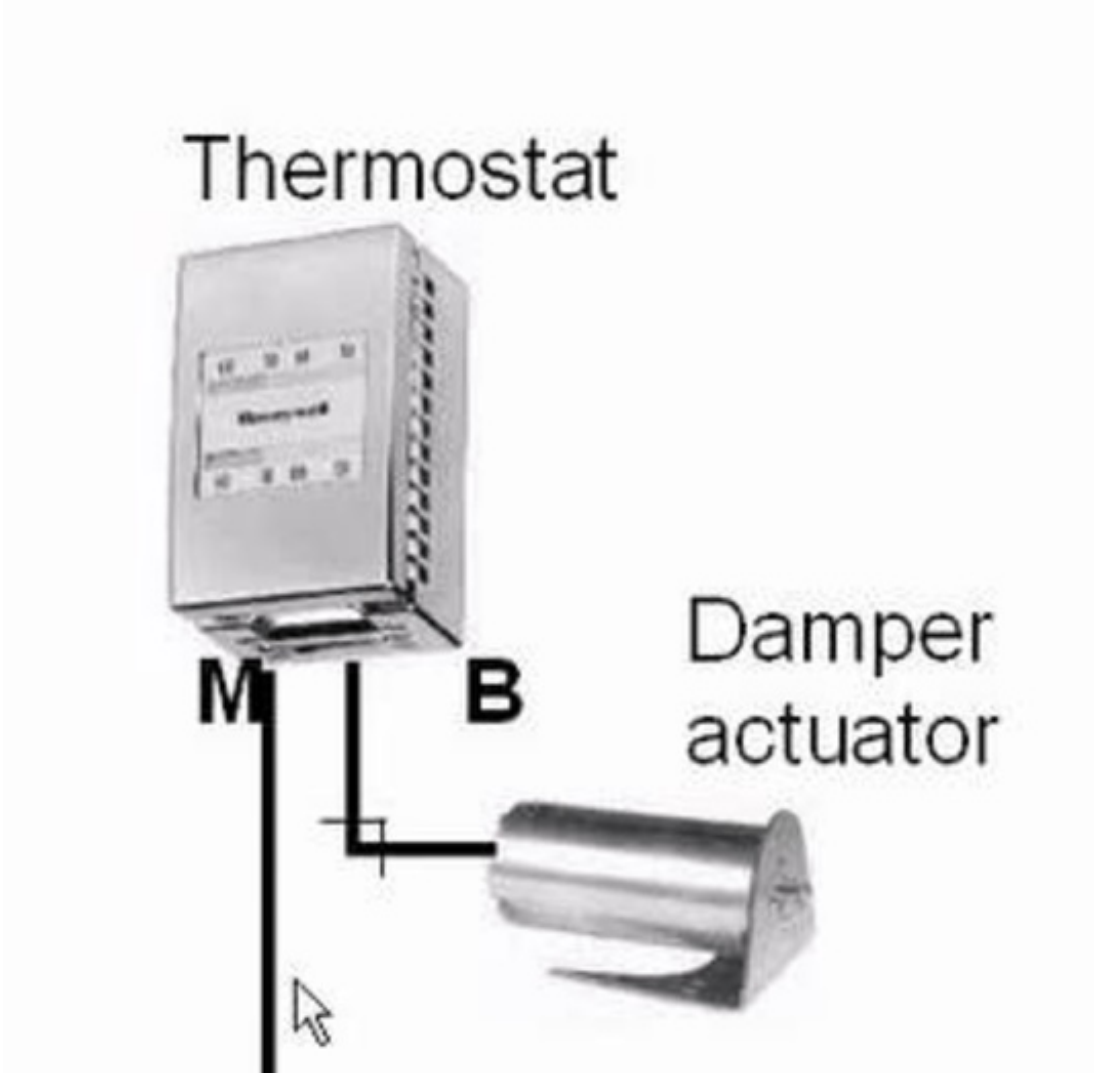
A picture of a pneumatic system

Pneumatics were one of the first original control systems. Pneumatics operate by compressing air which is then dried (to remove moisture) and sent down piping called **main lines**.

Along these main lines there are devices. These devices will consume airflow from the main line and regulate the pressurized air leaving them through a branch.

At the most basic level, the pressurized air from the main line will move through a sensing device like a thermostat. The thermostat will allow a certain amount of pressure out via its branch lines.

The **branch line** will act as a control signal to a device, like an actuator, and will regulate the amount of main line air that is entering the actuator. This is how the actuator is controlled.



A visual depiction of how a pneumatic thermostat controls an actuator

Analog

Analog systems are quite simple to describe you've probably used one today and not even been aware of it.

Have you ever changed the heat level on your toaster using a turn-knob? If so congratulations you've used an analog control system!

Analog control systems used to be quite prevalent but are slowly disappearing in favor of digital and electromechanical controls. You still see them in some situations mainly on ceiling mounted unit heaters and radiant heater coils that line the windows of buildings.

Analog systems work by injecting resistance upon a circuit.

This resistance then causes the control device (valve, relay, etc) to react. This is a really simple description but the reality is that these systems are really simple. Often analog systems are combined with electromechanical systems.

Electromechanical

Electromechanical control systems utilize mechanical changes to control their devices.

An example of this is The ceiling mounted unit heater described earlier. This unit heater will have a temperature element that reacts to changes in temperature.

Once the temperature changes enough the element will expand or contract to close or open a circuit. This will, in turn, cause the unit heater to turn on or off.

Digital

Now, this is something I know you've worked with!

You are using digital systems every day.

Digital systems are things like your microwave, smart thermostat, car radio. Essentially you have a microprocessor board that receives the signal from a button push or from some other action and then commands a corresponding output.

Digital systems, when networked form Direct Digital Control (DDC) systems.

DDC

DDC also known as **direct digital control** is the primary control system utilized today. When you talk to folks who have worked in the BAS industry for a long time they tend to define time periods by pre-DDC and post-DDC.

There are two big differences that DDC brought to the market:

The first capability that was introduced was direct digital control, I know obvious right?

Up until this point, several systems relied on analog inputs. These inputs were prone to calibration errors that could result in readings that were several degrees off. Imaging cooling a space down to 72 degrees when in actuality the space temperature was 66 degrees.

I've been to tons of buildings that are cold and humid due to subcooling, often times the problem is inaccurate sensor readings from poorly calibrated pneumatic systems.

While not perfect DDC did reduce the variables related to proper sensor readings. With DDC a facility operator simply needs to maintain the "offset" on his or her temperature sensor. This is a massive shift from having to maintain main trunk pressure, a temperature sensor, and branch pressure.

The second capability that was introduced by DDC was microprocessor control.

It's hard to imagine but less than 30 years ago control systems utilized dip switch settings and slot cards to create programs, and those were the premium control systems! The average user had to rely on a set of solenoids, relays, and timers to "drive" their control system.

With the advent of DDC systems, software programs could be written that would take the electro-mechanical relay funfest and convert it to software. This was huge!

Controls technicians and facility operators could now make changes to control sequences by simply changing the code. They no longer had to rewire circuits and install/remove relays.

The software-centric nature of DDC had another unforeseen benefit.

Since all the programs were “software” data could be quickly exchanged between controllers. These air handler controllers could share their valve position to chilled water plants allowing the chillers to reset their chilled water setpoint.

In many ways, this was the first Internet of Things.

From a technical perspective, DDC control systems have a CPU, known as a microprocessor and a series of digital inputs and outputs. They are typically powered by 24 volts Alternating Current (AC) but they can also be powered by direct current voltage.

DDC controllers will typically have some sort of communications trunk to facilitate communications between the field controllers and a centralized supervisory device. The main communication standard used by DDC controllers was and is RS-485 (which is a twisted pair 3 or 4 wire cable).

This “wired network” is daisy chained between controllers meaning it connects from one controller to another in a row.

However, there are newer communication designs that are being used for DDC controllers. The two most common communication designs are wireless, and hard-wired IP (which has 3 different design patterns itself; ring, bus, and daisy chain).

Diving even deeper, there are two main forms of wireless designs. These are 802.11 wireless (also known as Wi-Fi) and wireless mesh.

Wired IP networks can also use another technology called Power over Ethernet, also known as 802.3at provides around 25 to 30 watts of power over a traditional Cat 5E Ethernet cable. Cisco has a version of PoE called UPoE which provides up to 60 watts of power.

While these communication designs are new to DDC they are not new to the IT industry and the jury is still out as to which approach will win. Side note, I personally prefer wireless as it provides more flexibility and is a lower total cost, when you factor install, wiring, and switchgear.

Future….

What does the future hold, that is the multi-billion dollar question?

I often tell the folks I work with that I believe the future of controls is a world where 80% of projects are smart equipment and the rest are IoT devices like Arduino or Raspberry Pi boards with a common programming language.

Right now the profit margins are still high enough to justify creating multiple brands of controls but how long will that be the case?

We are getting to the point where there really isn't much more you can pack into a controller. Think about it, once the controller is wireless, which is my preferred approach, you will have freed yourself from all physical constraints except for power cabling.

I imagine inputs (thermostats, flow sensors, pressure sensors) and outputs (actuators, relays, etc) that have gotten low enough in cost and high enough in reliability to be wireless as well. Power will be locally sourced from the equipment or through batteries.

The end devices and field controllers will be smart enough to identify where they are and what system they should connect to. The BAS professionals job will switch from being focused on the physical installation of systems to being focused on IT systems like databases, analytics, and system integration.

Even that will eventually be replaced by self-learning systems that can sense the health of an entire building and adjust settings based on millions of variables.

Google is a great example of this technology in a very different field. Google processes 3 Trillion searches a year. 15% of these searches have never been seen before. Google has written programs powered by artificial intelligence that analyze millions of variables and determine the best search results to provide.

Is it unreasonable to think that we could take all of the knowledge we have around building systems into a computer program and that computer could analyze the performance of thousands of buildings to "learn" how to best control your building?

I definitely think that is a possibility. But don't worry, we will still need people to install and service this technology and these next generation "building automation programmers" will need to understand IT, programming, and other skills.

How far away is this future? It could be 50 years it could be 5 years. All it takes is someone like Elon Musk realizing that there is an industry ripe for change, choosing to enter the market.

A deeper look at a building automation architecture (from the top down)

Earlier in this guide, I discussed how there are four layers in a modern building automation system. In this section, we are going to take a much deeper look at each of these layers and how they function within a building automation system.

The layers

To recap there are four layers in a modern building automation system architecture those layers are:

- Server/Application Layer
- Supervisory Layer
- Field Controller Layer
- Input/Output Layer

Server/Application Layer

The server/ application layer serves to consolidate data from multiple different supervisory devices. It then delivers this data to the end user through the user interface (UI), often known as clients.

The server will also store trend, alarm, and schedule data in a database. This database can be used for reporting. The final thing the server can be used for is, is for serving up the API for the building automation system.

Supervisory Layer

The supervisory layer is where the supervisory devices sit. Supervisory devices are kind of like your home router. They collect all of the traffic from the field controllers and consolidate this traffic.

These devices serve to manage your communication trunks. Communication trunks allow your field controllers to connect to one another and allow your supervisory devices to collect information from the field controllers.

Some supervisory devices can also act as user interfaces for the BAS. Typical features that exist in the supervisory device are:

- User interfaces
- Trending, scheduling, alarming
- Global logic
- Communication Trunk management

Field Controller Layer

Field controllers look at data from inputs (temperature sensors, switches, etc) and then control outputs (actuators, relays, etc). BAS companies will use programming tools (usually developed by the BAS vendor) to program these field controllers.

The controller's programs will look at what the inputs are doing and then will control the outputs.

Input/Output Layer

The final piece of the puzzle is the input and output layer. This is where the sensors and control devices exist. There isn't a ton to add here except that you are starting to see IP-enabled sensors that use Ethernet or Wi-Fi for their communications.

These kinds of sensors will require a completely different approach and as of the time I wrote this guide, it's yet to be seen how all of this will shake out.

Making sense of the different pieces of controls

Ok, so now you have an understanding of the different layers that make up a building automation system and the difference between a control system and building automation system.

Now we are going to explore the physical pieces of the BAS.

From a physical perspective, the BAS consists of:

- Servers
- Supervisory Devices
- Field Buses
- Controllers
- Inputs
- Outputs

Servers

Servers are machines that collect and serve up the BAS data. These servers will either take the form of a desktop machine or a rack mounted server. These servers will run the BAS software and will connect to the network using network interface cards (NIC)

Desktop

Rack mounted server

Supervisory Devices

Supervisory devices can be either software or hardware based. Software supervisory devices are often known as soft-supervisors, where the supervisory software exists inside a server instead of a dedicated device, are becoming more common.

Soft-supervisors will utilize communication cards so that they can communicate with field buses.

Physical supervisory devices, where the supervisory device software is installed in a dedicated device, are still the most common devices. These devices will typically have an Ethernet NIC and a field trunk port (to connect field buses).

Field Buses

Field buses are the way building automation field controllers communicate back to supervisory devices. There are two prominent field buses right now. These are BACnet MS/TP and LON FT-10. These field buses connect field controllers back to the supervisory device using a daisy chain architecture.

If you're wondering what a daisy chain architecture looks like, just picture a set of Christmas lights. Each light is connected to the other light in a chain of lights. This is what modern field buses look like.

The supervisory device that connects these field controllers together will send messages across the field bus and will receive messages from the field controllers on the network.

Controllers

Controllers are potentially stand-alone devices that control systems. An example of a system would be an air handler unit or a central plant. These controllers are programmed using programming software.

This programming software is usually specific to each individual vendor and can only be used on their field controllers.

There are two main types of field controllers:

- Free programmable
- Application specific

Free programmable field controllers are able to be freely programmed. I know you're like "thanks, Phil that helps a lot…". Seriously though, back in the day, you couldn't configure a field controller. Nowadays you can log into a field controller and configure it to perform any control sequence you want it to.

On the other end of the spectrum, you have **application specific** field controllers. These controllers are specific to a single application. You cannot program these controllers you can only adjust preprogrammed settings.

Inputs and Outputs

I'm not going to spend a ton of time explaining what inputs and outputs (I/O) are as I'm

pretty sure you can figure this one out on your own.

Here's the down and dirty about I/O. You BAS controller will take signals from inputs (things like pressure or temperature sensors). Then the program inside the controller will decide to do something based on the value of these inputs.

Once that thing action is determined the BAS controller will command an output (actuator, relay, etc).

Pretty easy right.

Control Modes

At the end of the day building automation controllers exist to control outputs based on inputs. It really is that simple. To achieve that a BAS controller utilizes a variety of control modes.

Control modes are nothing more than a way of controlling outputs. And in the world of BAS there are 4 main control modes. Rather than making this post even longer then it already is I'm simply going to include a link to past articles that describe each of the four control modes The four control modes are:

- Binary, also known as on/off control
- Floating Control
- Sequenced Control
- Proportional Integral Derivative (PID) control

The trick to making sure your BAS provides excellent control is to make sure that you are matching the correct control mode to the output you are controlling. I cover that in each one of the articles I linked above.

The softer side of BAS software

Ok, I'll admit I'm not quite sure what the softer side of BAS means but it sure sounded good so let's go with it. We've already dug into the physical aspects of BAS now we're going to look at the software side of things.

In the world of BAS software breaks out into three main buckets:

- Databases
- Configuration software
- User Interfaces

Now you may be wondering why I am not addressing server software. The reason is that I've already covered that earlier in this guide. With that being said let's dive in.

Databases

Database software stores information. But they do soo much more than that. Whether you knew it or not databases are the lynchpin of your BAS.

How so you ask? Well, let me tell you the ways!

Databases store your configuration, schema, graphics, and so much more! All the bits and pieces that make your BAS your BAS are often stored in databases.

Databases collect invaluable information and store it for later use. Trends, alarms, schedules, setpoints, and more! They are all stored in databases.

And the really cool thing about this is that if you understand databases and the query languages that support them you can start to dig deep into your BAS to pull out past "trends" of your BAS performance.

I just so happened to write two very in-depth articles about how to do exactly that. You can read them here and here.

Configuration Software

There's a ton of different BAS manufacturers in the market and as a result, there's a ton of different types of BAS configuration software out there. But at the end of the day the software can be broken down into two buckets. Database configuration software and programming software.

Database configuration software is used mainly to configure servers and supervisory devices.

Warning… Uber geek moment. The majority of BAS are built using a three-tier software architecture (user interface, application, database). This is very similar to the MVC framework used by many modern web applications. Because of this the settings that determine the configuration of the BAS are kept in a database and are called up by the BAS application as required. Now, this has slightly changed with the introduction of HTML/5 user interfaces because those use a web server to render HTML files for the end user.

Ok, with that uber nerdy expedition over with. Let's talk about programming tools.

Programming software exists to allow the configuration of the field controllers. One of the biggest issues faced by BAS companies is that each company has its own programming tool. Because of this, only those with the programming tool can configure the controllers. This leads a lot of customers to feel as if they are stuck with the BAS company who provided the controls.

Now as you can imagine there is a ton to know about programming a BAS. Because of that, I've spent a lot of time creating a vendor agnostic article that dives deep into the concept of programming a BAS. You can check it out by clicking here.

User interfaces

When it comes to user interfaces you have two real options. Those UI options are known as thick-client and thin-client. A thick-client is the traditional method that is used for connecting to building automation systems.

This is where you would either install an application or download an application that would run on your computer. The reason this is called a thick-client is that there is actually an application running our your laptop. The problem with thick-client applications was

The problem with thick-client applications was they were usually dependent on some form of software (e.g. Java). When you would upgrade the building automation system, the version of this software dependency would change and this would often break the thick-client (meaning it wouldn't work anymore).

To solve this problem the BAS world has largely shifted to using thin-clients.

Thin-clients, on the other hand, utilize web browsers to access the building automation system. The thought behind using web browsers like Google Chrome or Internet Explorer was that they would break the dependency on software like Java.

While this is true, they've introduced a new issue which is IT troubleshooting. Now instead of the BAS manufacturer having full control over their user interface, they are at the mercy of the web browser's code, which they may or may not understand.

Cybersecurity and building automation, (why people secretly think you're stupid if you say Target was a BAS hack)

Target had nothing to do with the BAS system. And IoT devices like IP cameras and baby-monitors are not even within the same continent as BAS devices.

There you go, I've just dispelled the two most common myths about cybersecurity.

Ah, if it was only that easy.

Here is what you need to know about cybersecurity:

Nothing is 100% secure

That's just the cold hard truth. Anyone who tells you their BAS is secure and unhackable, is full of it. Anything can be hacked, given enough time, money, and skill.

Cybersecurity is the process of identifying the cybersecurity risk that your system has and then implementing controls to mitigate that risk.

But what does that mean?

Well what happens, at least in the IT world, is that a professional assessor will "assess the IT systems" and identify potential vulnerabilities.

A **vulnerability** is a risk that can be exploited by an attacker.

Then a monetary impact is assigned to the vulnerability based on the likelihood of that vulnerability being exploited. From there the customer will select a set of controls to mitigate the vulnerabilities.

In the world of IT, the term **controls** describe steps that are taken to address the vulnerabilities. There are multiple types of controls but that is beyond the scope of this guide.

I tell you all of that so that you understand what the IT folks are talking about when they ask you questions around cybersecurity.

The trick to cyber security is being more secure than the other guys.

How can you do this?

Well, it's actually quite easy.

In my IT for BAS Professionals training program, I teach several actions that you can take to secure your BAS. I've included three of these tasks below.

If you do these things you will massively increase the security of your BAS.

These BAS securing tasks are:

- Have a unique username and password for each user
- Enforce password complexity
- Enable a firewall and close unused ports

Have a unique username and password for each user

I know, it's shocking to think that you would actually need to tell someone this. However, I've been to dozens of sites where the entire facility team uses the same username and password.

Not only does this put the BAS at risk from someone getting the "keys to the castle" but it also creates a problem with the users themselves. Because everyone uses the same username and password you have no idea as to who actually made any changes to the building automation system.

That's a double uh oh!

Enforce password complexity

I know how annoying it is to have to change your password all the time. I'm constantly having to change my password at work and gosh is it aggravating.

But the reality is, your username and password are the best defense you can have because they work even if someone has physical access to the BAS server, well most of the time…

I've linked to an article on creating complex passwords rather than boring you with a detailed explanation on how to create complex passwords.

Enable a firewall and close unused ports

Finally, we have the firewall. A firewall is a piece of software that allows and deny's network traffic from moving across the network. A firewall is like a security guard who decides which people get access into a building.

One of the most common security issues with a BAS is that it has a lot of software ports that are open to the world. But before I describe how to fix that issue, let's discuss what a port is.

Software needs to send certain types of traffic to other software. Ports allow the software to categorize and segment the data they are sending rather than just sending a huge blob of data.

When BAS folks install a BAS they tend to disable the firewall that way they don't have to worry about having the right ports open. This creates a huge hole in the network that attackers can exploit.

To avoid this, I teach that you should understand what ports your BAS needs open and only open up those ports on the Firewall. This is actually easier than it sounds.

You simply reach out to your manufacturer, ask them what ports to have open, and then you close down all the ports except for those ports.

If you do these three things you will massively increase the "securedness" of your BAS.

Upgrading a BAS, how to not screw it up

If you asked me what single task has the greatest likelihood to really mess up your BAS I'd say upgrading.

Hand's down upgrading a BAS can be the most tricky project you'll ever take on.

So how can you go about taking on a BAS upgrade project successfully?

Well, it just so happens I spend a quite a bit of time on upgrading a BAS in myBuilding Automation Systems A to Z training program. Here's a video from the program that goes through the "upgrade process".

In the video, above I took you through a lot of the things you need to think about when you are upgrading a BAS. I also briefly talked through my process for performing BAS upgrades.

Here is a list of the steps you should take when you are performing a BAS upgrade.

- Step 1: Decide on what day you will complete the project
- Step 2: Verify the job site systems and applications
- Step 3: Ensure you have the access you need
- Step 4: Identify the systems that will be affected
- Step 5: Decide on your upgrade strategy
- Step 6: Determine the people or groups that will be involved
- Step 7: Write out your upgrade plan
- Step 8: Put the systems in hand

- Step 9: Execute your plan
- Step 10: Document your changes
- Step 11: Verify proper operation
- Step 12: Check the opposite season control

As you can imagine each one of the steps has their own nuances. But the good news is that the steps I describe above are fairly intuitive. I shared these steps with you because I wanted you to have both eyes open when you decide to take on an upgrade project.

Trends, alarms, and reports…How to take your building from oh crap, to oh yeah

So, contrary to popular belief you don't need to spend tons of money on an analytics solution. A building automation system is chock full of features that allow you to analyze the current and historic status of your BAS.

But the sad reality is so many building operators are not using their BAS to its full potential. The first step to taking your BAS to the next level of functionality is to level set on what these features are.

What are trends, alarms, and reports?

Ok pay attention, this section is important:

As I mentioned, a BAS has many features, and most of the features are left unused or misused by building operators. You've probably guessed by now that the often unused features of a BAS are Trends, Alarms, and Reports.

Trends

Trends are data points that are collected and stored for later recall. There are two main types of trends:

- Interval Trends
- Change of Value Trends

Interval Trends are trends that are collected at a predictable interval. Pretty simple right? The good news is the majority of trends are interval trends, easy to setup, reliable, but limited to the time slice you setup.

This means that if you are collecting trends at 15-minute intervals. there will be no record of if something rapidly changes between those 15 minutes. This is why I cringe most of the time when I see a specification that says every point shall be trended at "15-minute intervals".

Change of Value Trends record a value when the value of a point changes by a specific value. Hence the term change of value. These trends can be immensely useful when you are troubleshooting a specific issue or you are trying to measure a point that changes by very minute details.

However, you need to be careful when you use change of value trends as they can take up a lot of storage space if you set the change of value threshold to tight or you use to many change of value trends.

I wrote up three articles on how you can use trends to perform basic fault detection for three of the most common BAS issues:

- Simultaneous Heating and Cooling
- Out of Control Outdoor Air Economizer
- Over-cycling chillers

Alarms

Alarms are one of the most mismanaged functions of a BAS. Period! Oh, if I had a dollar for how many times I've visited facilities that had over 10,000 unacknowledged alarms. I remember this one customer I visited. Their central plant, yes the chillers! Where down for almost an hour before someone took notice. The reason why?

They had so many filter status and space temperature alarms coming in that the single chiller alarm got missed in the massive influx of alarms.

So if you are sitting on one of these ticking turd time bombs what can you do? How can you go and take a horribly implemented alarming strategy and turn it around?

Here are two steps you can take right now.

Step one: Do something crazy

I'm going to say something drastic but, here it goes.

Get rid of alarming. Yep, delete all your alarms. Poof gone.

Now, at the same time list out the systems that are absolutely critical. For most buildings (**hope you noticed that caveat**) your list should look something like this.

Chillers, boilers, air handlers, pumps, and maybe the temperatures in the networking/server rooms.

What it shouldn't look like is alarms on every friggin point. There's no reason to have sensors on filter statuses and common corridor space temperatures. I know that this flies in the face of the alarm everything strategy but as you'll see in a second there are things that are more effective than alarming. Alarming should be for failure conditions not hot or cold calls or clogged filters.

Step two: Create some standards!

I am a fan of standards, if you've been reading myblogs or listening to my podcast for any amount of time then you've heard me hammering this topic over and over. But what does that look like.

Here's an excerpt from my book *Building Automation Systems A to Z*, that discusses standards.

> Standards they make the world go round. Could you imagine if you went to Home Depot and each store called lights something different? Maybe one store feels like lights should be called glimmers, and another store wants to call them shiners. Imagine your confusion as you tried to communicate what you wanted to buy. BAS standards are the same way. A BAS standard, done right, can tell folks exactly how you want your BAS to work! – Pg 194-195 Building Automation Systems A to Z, 1st Ed.

When it comes to standards there are several different things you want to address. One of those "things", as you might guess, is alarms. But how? How does one create a standard around alarming?

Once again we turn to my book, *Building Automation Systems A to Z* to find the answer.

Alarm settings

Next, the alarm settings need to be determined for each of the points that have been defined for the VAV reheat unit. This is done by listing out the high and low settings for each alarm point.

Now one of the important things to note is that you only want to list alarm settings on points that can go out of range. Set points and outputs don't normally go out of range so you wouldn't set alarms on them.

Point Name	Point Description	Trend Interval	Alarm Hi	Alarm Low
ZN-T	Zone Temperature	300 Seconds	85°	55°
DA-T	Discharge Air Temperature	300 Seconds	160°	70°
ZNT-SP	Zone Temperature Set-point	300 Seconds	N/A	N/A
DPR-O	Damper Output Command	CoV (1% of range	N/A	N/A
CLGCFM-SP	Cooling CFM Set-point	CoV	N/A	N/A
MAXCLGCFM-SP	Max Cooling CFM Set-point	CoV	N/A	N/A
HTGCMF-SP	Heating CFM Set-point	CoV	N/A	N/A
HTG-O:	Heating Output	CoV 1% of range	N/A	N/A

Point Standard Matrix from Building Automation Systems A to Z, 1st Ed. Pg 198

So basically, you define the point list by the system, and then you apply the alarm thresholds based on if the system is critical.

But…. (side note, I sure use that lead in a lot don't I? I almost feel like a Sham-wow sales man, but wait, there's more!)

How do you determine if a system is critical?

Simple my friends, and for that, we once again turn to my book and take a look at my system criticality matrix. You'll find this on page 213.

Criticality Matrix	Work Stoppage	Regulatory Compliance	Life-Safety
Required (5)			
Significant (4)			
Moderate (3)			
Minor (2)			
Low (1)			
Overall Ranking			

Criticality matrix blank, from Building Automation Systems A to Z, 1st Ed. Pg 213.

Basically, you are going to rank the overall impact to three areas with a rank of 5 to 1. After you've ranked the three areas you sum up your ranking matrix and then use that to prioritize systems. You can see what this looks like in the image below.

The matrix below has been filled out, for a central cooling plant at a hospital.

Criticality Matrix	Work Stoppage	Regulatory Compliance	Life-Safety
Required (5)	X	X	X
Significant (4)			
Moderate (3)			
Minor (2)			
Low (1)			
Overall Ranking	5	5	5

Criticality matrix filled in, from Building Automation Systems A to Z, 1st Ed. Pg 214

As you can see the matrix has been filled in for a central plant with a total ranking of 15. This would then be prioritized and any system that falls above a threshold determined by you would be a candidate for alarming.

Going and getting your alarming under control can be a huge undertaking. I've written a quick post to help you with this. You can read it right here.

Reports

Remember earlier when I said there were ways other than alarming to determine if things were out of whack. Well, my friends, reporting is that way.

- Want to know when any space has been greater or less than setpoint by more than four degrees for more than two hours? Reports can do that.
- Want to see all of the filters that have had a filter plugged status for more than 3 days? Reports can do that.
- Want to know all of the fans that have exceeded a run hour threshold by 10%? Reports can do that as well.

Reports should be your main go to strategy when you are trying to avoid nuisance alarms. As a matter of fact, reports can be a great strategy for prioritizing maintenance efforts. Yet in so many of the sites I've visited reports are seldom if ever used. Why is that?

In my experience, it's because we have trained ourselves and our teams to believe that the only way to catch a problem is with alarms. Add to this that very few sites prioritize their systems like I described earlier and you have a recipe for everything being critical. And we all know we have to use alarms for critical things…

That is why before starting on your trend/alarm/report journey you absolutely must identify what equipment is critical and what equipment is not.

Working with IT and the Zen of networked buildings

Back in 2007 I first started in building automation IT was something that was on the fringes. Fast forward 10 years and IT is involved in almost every aspect of building automation. From servers to IP-enabled sensors, if you are hoping to avoid IT you're out luck.

When it comes to IT there are two areas that seem to challenge BAS folks. The first area is interacting with IT and the second area is all of the "technical" mumbo jumbo.

Fortunately for you, I've got some great stuff on both.

How to interact with IT

About 2 months ago when I was launching my online IT training program I rolled out a webinar to help folks solve the number one problem most BAS folks talk about. Dealing with IT. Even though the webinar has long since passed, I recorded the three strategies I taught to the webinar attendees. Here's a summary of the recordings:

Here's a summary of the recordings:

- In the first recording, I teach you how an IT group is structured, this is critical to making sure you are talking to the right person
- In the second recording, I teach you my secret question that will help you get what you need from IT
- In the third recording, I discuss exactly what you need to learn so you're not spending years learning things you don't need to know

As I mentioned I recorded these videos and you can check them out below.

Video 1: How IT groups are organized

Video 2: The Secret Question

Video 3: What you need to learn about IT

Resources for you to learn IT

I've been on a bit of a mission lately to help folks learn IT. The world we exist in right now is only becoming more and more dependent on technology. That world is bleeding into the BAS world more and more every day. That's why a large amount of the content I've produced has focused on IT. But sometimes it can be hard to sort through all of the great content I've created. That's why I've taken the time to segment out all of the content in some easy to access links below:

Networking

- Network Fundamentals
- The basics of subnetting
- An overview of the TCP/IP Stack
- The OSI Model
- The basics of IP Addressing

Servers

Servers run your BAS and ultimately run the Internet. You're actually using a server right now to access my WordPress site that is hosted on a cloud based server.

- The basics of Web Servers
- The fundamentals of servers

Databases

Databases are one of the core technologies that power building automation systems. Yet so few people seem to understand how they work.

- SQL Overview
- SQL Commands for your BAS

CyberSecurity

There are so many misunderstandings when it comes to cybersecurity. And it doesn't help that companies are putting out "expert" opinions that couldn't be anything further from the truth.

- The truth about recent hacks
- How a hack happens

API's and Integration-

Systems integration is near and dear to my heart. It's one of the three things that helped catapult my career. It also seems to be one of the topics that confuses BAS professionals the most.

- The non-techie guide to API's
- My complete guide to systems integration

Now I realize that this was and is a ton of information and a lot of it requires you to have the initiative and discipline to create a learning plan and perform a ton of self-study. If you'd like to shortcut that whole process and learn exactly what you need to know about IT in days vs. years then be sure to check out my self-paced online IT training program. It will literally save you years of studying and thousands in travel costs.

How to enter the world of BAS and develop the people you have

So you're excited. You've learned more from this single guide then you've learned at any training you've ever attended (that is unless you've attended my training programs). And now you want to either enter the world of BAS or get your team up to speed.

But how?

Well, just so happens I've put together a ton of stuff on this exact topic.

Becoming a BAS professional

In order to enter into the field of BAS you need to have one of the three skills (HVAC, electrical, or IT). Yes, that even applies to sales folks and project managers. But what exactly do you need to know? And what can you expect when you go to your first interview? Well, my friends, you are in luck because I answered those exact questions in past articles and episodes of my podcast. Check them out below:

- You're new to BAS, here's what to do
- 7 things you must know when you start your career in BAS
- You can have a career in BAS, and here's how
- 7 tips to pass your BAS interview
- You've got a job in BAS now what?

Developing your team and hiring talent

Man, if there is one thing I know, it's that good BAS folks are hard to come by. There's just not a lot of good programs out there to develop talent. That's the whole reason why I started this blog and created my training programs.

So how can you get started on developing your team?

[Coming soon]

Conclusion

So there you have it, you just read the most complete guide to building automation systems you will find anywhere on the Internet. I'm sure you have questions after reading this so drop down in the comments section and ask away, I'd love to hear from you.

By the way, if you found my free stuff valuable. Just imagine how good my premium training programs are. Check them out by clicking here.