ViaVai Access Control

Solution Overview

The Thinknx Access Control object permits to enhance the level of automation and security of the home/building where it is applied. It can be adapted to sectors where long term expirations are required such as service and industry sectors, but also applies to the hospitality sector where credentials are usually short term, and remote management is required.

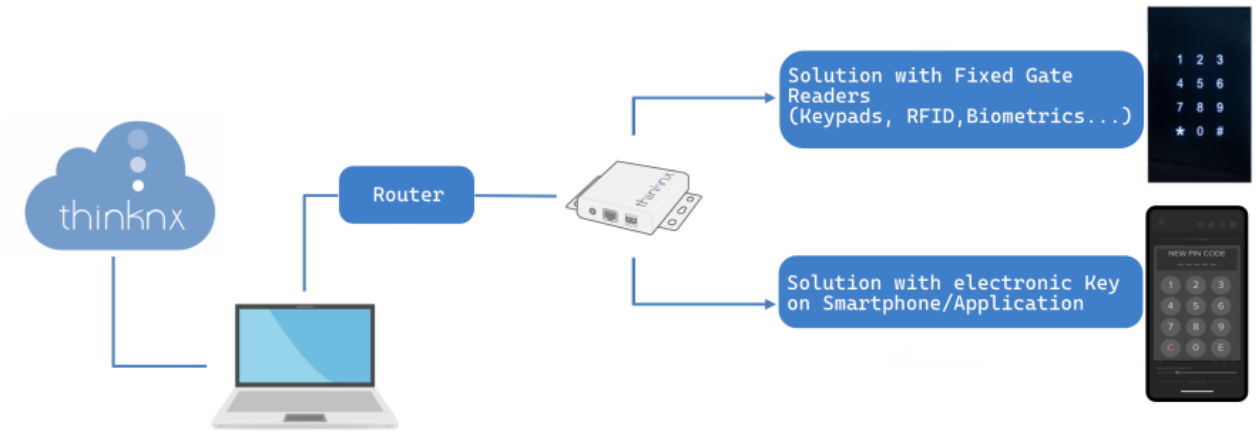


Figure 1: Access Control Usability

The Access Control object can communicate directly with the KNX system through Thinknx server, making the integration very easy and flexible. Any standard KNX keypad can be used as an access keypad, and its buttons as code entries. Once the code is inserted, it is possible to operate a lock or switch on a KNX actuator. In addition, communication with Wiegand technology is possible through the Thinknx-Wiegand adapter, making it possible to install a suitable RFID or biometric reader.

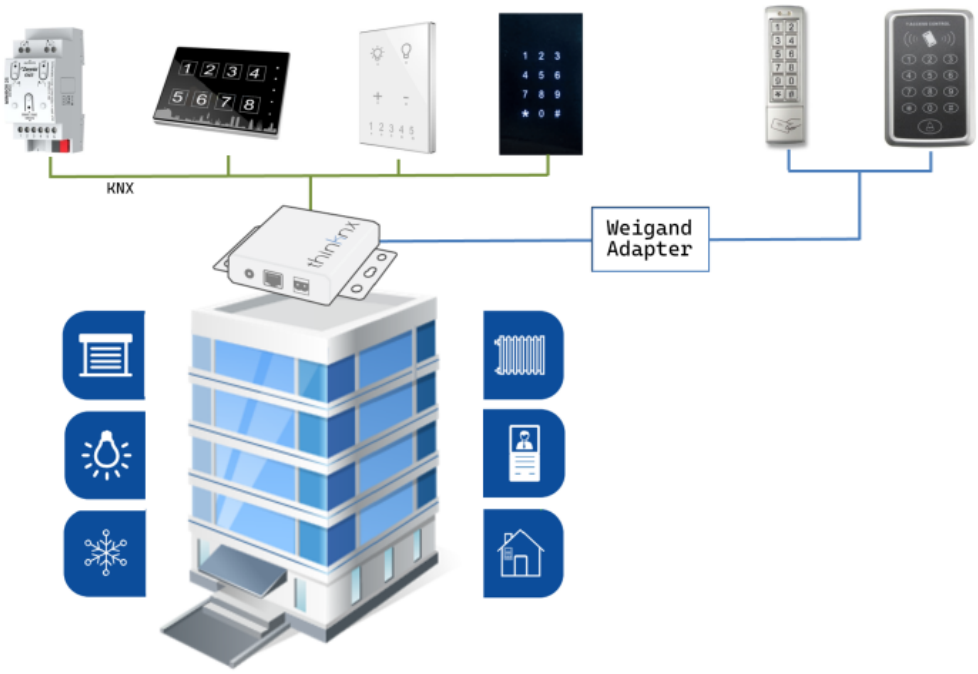


Figure 2: Access Control Diagram

Configuration

To facilitate the work of end users and adminitrators and still be able to cover complex situations, Thinknx Access Control is configured in two different phases and with two different tools:

- **Phase 1: Thinknx Configurator** The installer and system integrator will use Thinknx Configurator to completely define the topology of the system, creating and configuring readers, areas and roles, and creating interactions between KNX and other integrated systems.
- **Phase 2: Management** The manager, who is most likely available on site, will focus on the everyday management tasks such as creating users and associating them with predefined roles, create calendars and time-based restrictions, add or delete access codes, view logs and movements, control area occupancy and much more. The manager will not have access to the Thinknx Configurator project, but will use a dedicated web page or Thinknx application. Management

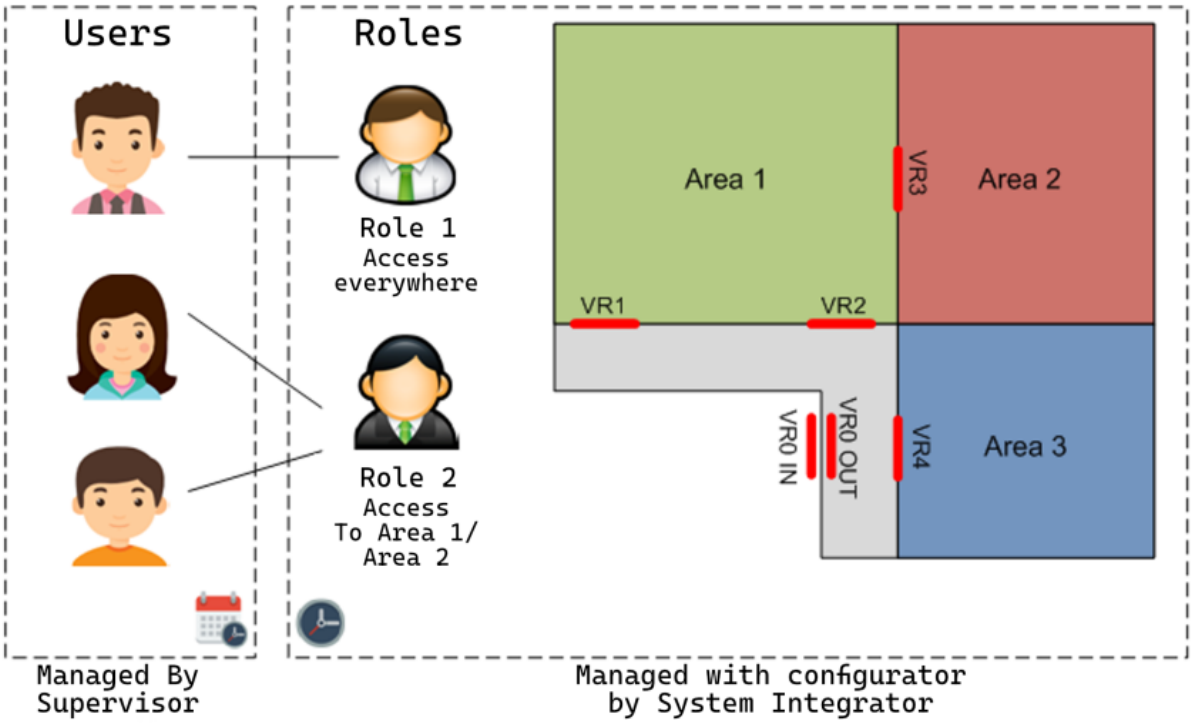


Figure 3: Access Control Structure

Some of the most important **features** available within the Access Control:

- Possibility to create Areas as parts of the building, limited by one or more gates with readers. Rights will be granted per area.
- Possibility to define for each area Entry Gates and Exit Gates, which can lead to fully count the number of persons inside the area, or for timekeeping operations.
- Possibility to create different Roles, with predefined privileges to access specific building areas. Any new user shall be assigned to a preconfigured role and will receive the same access rules as this role.
- Possibility to define automatic expirations for user authorizations.
- Automatic update of all gates when settings are modified.
- Possibility to log all movements and save them inside the database or send them automatically as reports.

Phase 1: Thinknx Configurator

In the Thinknx Configurator software, open the project and add the Access Control object in the System tab. The below parameters are available for this object:

Parameters

- **Label** Text to identify the object
- **Code length** Number of digits inside the access codes. All codes should have the same number of digits. If length of codes is 6, a valid code will be 123456.
- **Code numbers** Numbers contained inside the code, starting from 0. This depends on the number of buttons in keypads. If code numbers is 5, codes can contain the numbers 0,1,2,3,4.
- **Zero included** Permits to choose if zero (0) should be part of the code or not. If code

number is 3, and zero is included, the numbers used inside the code are 0,1,2. If zero is not included, the numbers are 1,2,3.

- **Confirm time (ms)** Maximum time between two consecutive keypress. Once the confirm time has been exceeded, the start of a new code will considered on keypress. This time is useful to start a new sequence if a mistake has been made when entering the code.
- **Guard wrong codes** if enabled, it is possible to monitor wrong codes attempts, and generate events once a threshold has been reached.
  - **Number of attempts** Maximum number of attempts with wrong codes allowed before triggering an event if tried within a certain guard time.
  - **attempts interval (s)** duration in seconds used to count the number of wrong code attempts before triggering the event. If the number of wrong attempts exceeds the number of attempts set previously during the attempt interval, then the event for wrong codes will be fired.

7. **Wallet Function** If Enabled, it will associated a wallet for each user and there will be commands to charge the user in case of any access control action.

Keypad prototypes

Keyboard prototypes allow defining the types of keyboards present in the system. To add and configure a prototype, click on the small button to the right to access the prototype creator window. For each added prototype, the following properties are available:

Parameters

- **Name** label of the prototype used.
- **Technology** technology used for the created protoype. The available options are:
  - Doory (Blumotix - BX-DOORY(BX-Q120W))
  - Generic KNX
    - Bit (DPT 1)
    - Unsigned 2 Bytes (DPT 7)
    - Unsigned 4 Bytes (DPT 12)
    - Unsigned Byte (DPT 5)

3. Virtual Keypad
4. Virtual Keypad over IP
5. Zennio IWAC
6. Generic Wiegand *coming soon*
7. Wiegand over network *coming soon*

3. If “Generic KNX” is selected
4. **KNX Data Type** Data type used on the KNX buttons to send the access code digits. Each button can be configured to send a 1bit object to a different KNX group, or send a certain value on a 1byte/2byte/4byte object corresponding to the digit on the same KNX group.
5. **Delete Key** if enabled, this option will delete the inserted sequence and start a new one before the “confirm time” expiration.
6. **Confirm tone** if enabled, the system will generate an event “confirm tone” in case of successful attempt.
7. If “DPT 1” is selected
8. **Keys** This parameter holds the keys collection used to enter a code. Clicking on the small right button will open the Keys Manager. From there, it is possible to configure every single key. Each key has a **name**, **key type** and a **group address**.

9. If DPT 5/7/12 is selected
10. **KNX group for keys** It holds the group address used on the keypad to sen the different code digits. A telegram with value 4 means that button 4 has been pressed on the keypad.

Keypads

Keyboards are the devices that send codes to the server for system control. They can be physical, using KNX or Wiegand, or virtual through the Thinknx application. **The number of keypads also determines the required license to use the service.** At least one keypad protocol must be created prior adding the keypads. Clicking on the small button to the right will open the Keypads Manager. Each added keypad will have the following parameters:

Parameters

- **Checker device** device label
- **Prototype** communication prototype used for this device. A prototype must be created in **Keypads Prototype** in order to view it in the list here.
    - If Doory(BX-DOORY(BX-Q120W)) is selected
        - **KNX group for code enabled** KNX group used by keypad to signal that the code is enabled (1 bit data type).
        - **KNX group for storing/deleting codes** KNX group to store and delete codes from the keypad (10 bytes data type).
        - **KNX group for erase request** KNX group to erase codes from the keypad (1 bit data type).
        - **Command for tone** command to execute to play tone.
        - **Command if fail** command to execute in case of failed entry attempt.
        - **Command if success** command to execute in case of successful entry.
    2. If a Generic KNX type is selected
        - **KNX physical address** physical address of the device.
        - **Command if fail** command to execute in case of failed entry attempt.
        - **Command if success** command to execute in case of successful entry.
    3. If Virtual key is selected
        - **Command if fail** command to execute in case of failed entry attempt.
        - **Command if success** command to execute in case of successful entry.
    4. If virtual key over ip is selected
        - Device IP Address: IP address of the checker device (keypad or any other access control device)
        - Device ID:Identifier of the device. Used in case there are multiple device connected to the same IP address. Can be any number associated to the checker device.
        - **Command if fail** command to execute in case of failed

entry attempt.
- **Command if success** command to execute in case of successful entry.

5. If Zennio IWAC is selected
   - **KNX Pin Log group** KNX group used by keypad to report pin events
   - **Delete all pins group** KNX group used by server to delete all PINS already configured on the keypad
   - **Delete PINS on startup** If Enabled, the PINS already memorized inside the keypad will be deleted and the access will be granted only using VIAVAI system.
   - **Command if fail** command to execute in case of failed entry attempt.
   - **Command if success** command to execute in case of successful entry.

Areas

It contains all the areas (rooms or parts of the building) of the project, which can be monitored by one or more device checker. It will be possible to set up flow access, codes for continuous unlocking or fixed locking of the area, to know how many people are inside, and if there has been a valid entry. By clicking on the small button to the right, the Area Manager is accessible. Each added area will have the following parameters:

Parameters

- **Name** label of the area.
- **Entrance doors** allows you to select the checkers that are used on site to enter this specific area. The checkers should be created first in the **Keypads** collection.
- **Dual flow** if disabled, the entrance checkers will be considered for both entrance and exit. If enabled, it means the area has different doors/checkers for entrance and exit. It permits to distinguish entrance and exit events and eventually count the people inside the area.
- **Exit doors** *only visible if* ***Dual flow*** *is enabled.* allows you to select the checkers that are used on site to exit this specific area. The checkers should be created first in the **Keypads** collection.
- **Count persons** *only visible if* ***Dual flow*** *is enabled.* If enabled, the system will count the number of persons inside the area based on the entrance and exit events.
- **Success enter command** Command to execute in case of successful entrance event.
- **Success exit command** Command to execute in case of successful exit event.
- **Nobody command** *only visit if* ***Count persons*** *is enabled.* Command to execute in case nobody is in the area.
- **Somebody command** *only visible if* ***Count persons*** *is*

enabled. Command to execute in case somebody enters the area first.
- **Active codes command** Command to execute whenever there is at least one active code for the area. If a new code is added for the area and there are not already valid codes, the command will be issued.
- **Inactive codes command** Command to execute whenever there is no active codes for the area. If a code is removed from the area or expires and there are not other valid codes, the command will be issued.
- **Consider permanent codes** If Enabled, when evaluating if the area
- **KNX group force closed** KNX group (1 bit DPT1) to force all doors/checkers to refuse any code.
- **KNX group force open** KNX group (1 bit DPT1) to force all doors/checkers to accept any code.
- **KNX group valid enter** KNX gorup (1bit DPT1) to signal a valid entrance into the area.
- **KNX group valide exit** KNX group (1bit DPT1) to signal a valid exit from the area.
- **KNX group someone inside** *only visit if **Count persons** is enabled.* KNX group (1bit DPT1) to signal somebody is inside the area.
- **KNX group nobody inside** *only visit if **Count persons** is enabled.* KNX group (1bit DPT1) to signal that nobody is inside the area.
- **KNX group n.persons** KNX group (2byte unsigned integer - DPT7) to signal the number of persons in the area.

Roles

This parameter holds the collection of roles created inside the project. Users on site can be assigned a role with predefined privileges to access specific building areas. Clicking on the small button to the right will open the Roles Manager. Each added role will have the following parameters:

Parameters

- **Name** Role name
- **Permitted areas** permits to select the allowed areas for the role created. The list will show areas created in the “Areas” parameter.
- **Permitted services** permits to select the allowed services for the role created. The list will show areas created in the “Wallet Services” parameter.

2N Integration

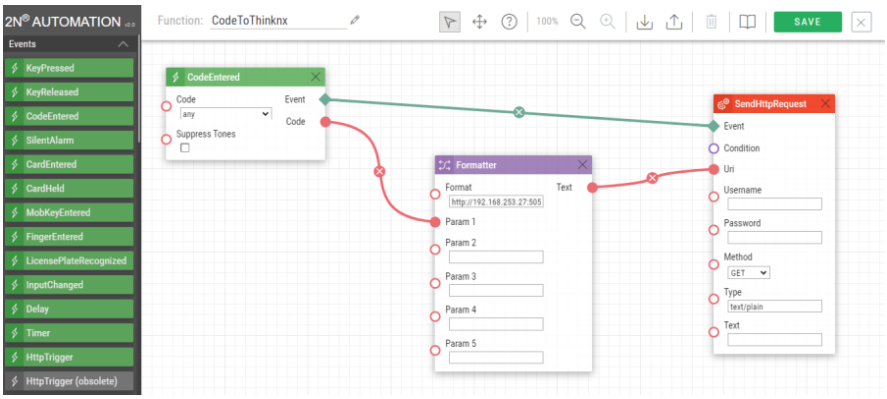
It is possible to send access codes directly to Thinknx from 2N

keypads to manage user access points through our system, following the configuration below.

On the 2n web page:

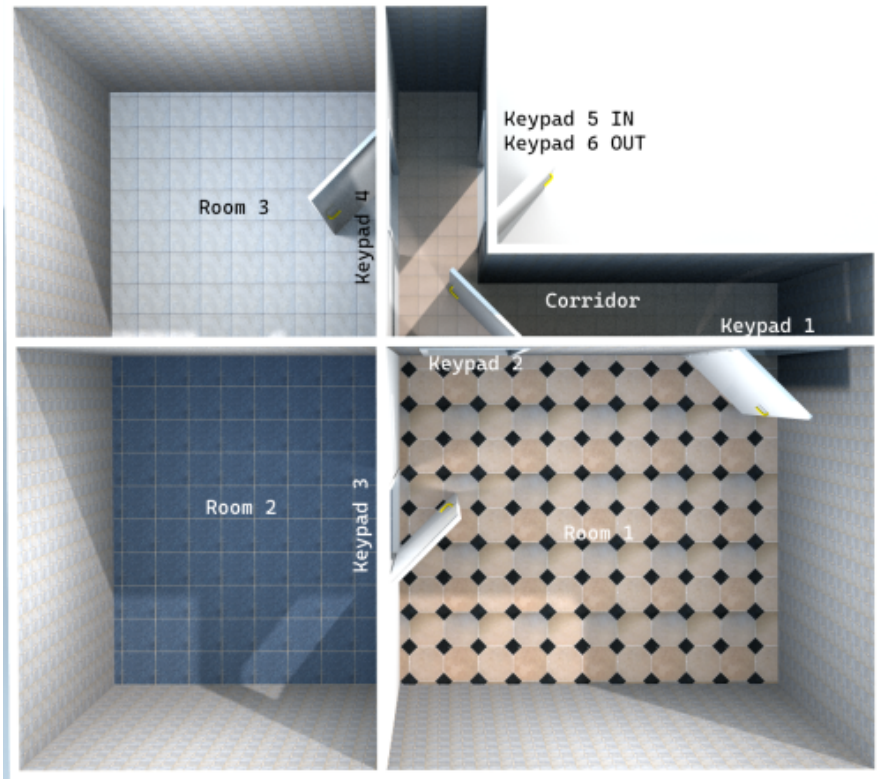
Configuration.

- Go To Automation
- Create a new automation
- Code Entered = Any
- Formatter
- Format: http://IPOFSERVER:5051/images/action.cgi?cmd=accessControlCodeEntered&code={1}
- Send HTTP Request
- Make link as picture.



File am to import in the Automation of 2N

Examples

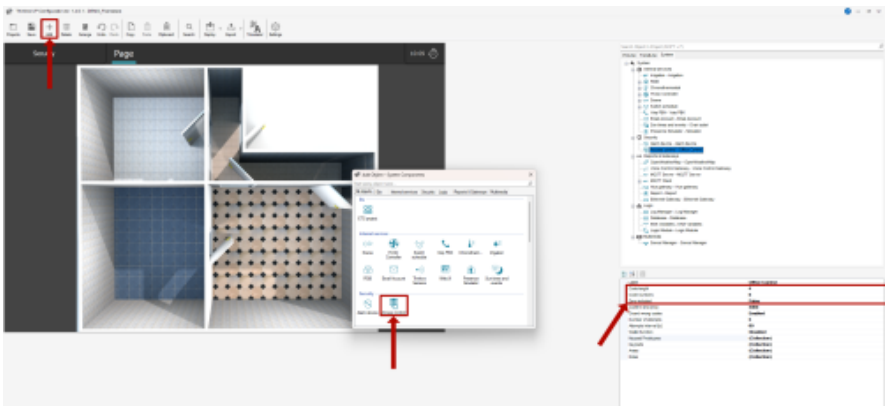


Starting from the image above, let's assume we have **Room 1** with a keypad (K1) to enter and a keypad (K2) to exit. **Room 2** has a virtual keypad (K3) accessible via an application. **Room 3** features a keypad (K4 - **2N access control Device**) that can be accessed through an IP address. There is also a **corridor** with two keypads (K5/K6) to monitor entry and exit.

In this setup, we will have various user levels with different access permissions to these rooms. Let's assume a system with a 4-digit code, 6 possible numbers, and without zero available.

Part 1 - Room 1

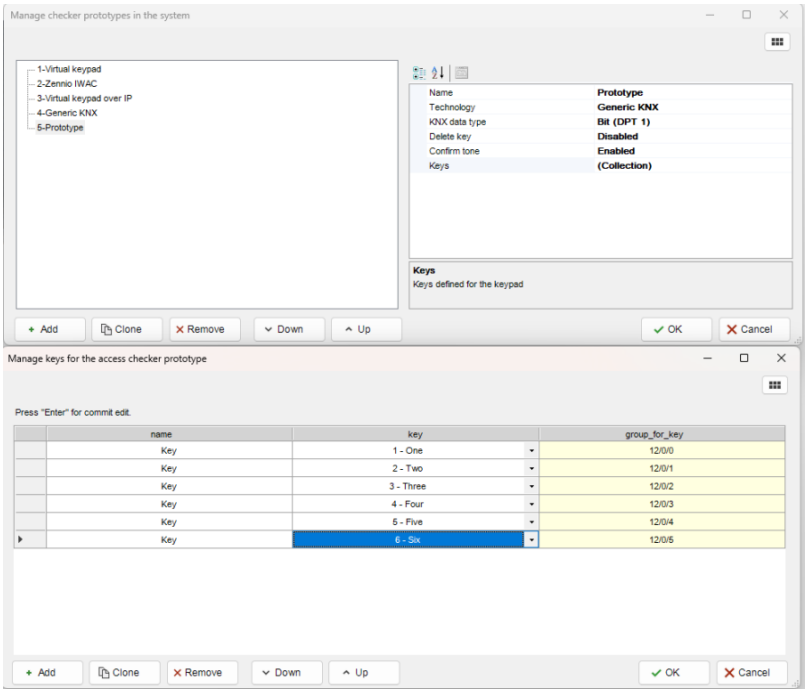
- Add Access Control to the system.
- Change the Code Number to 6.
- Set Zero Included to False.



Open Keypad Prototype.

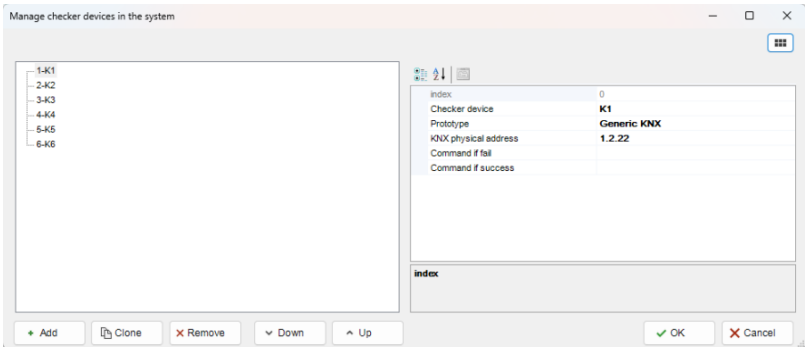
- Click on Add.
- Select Generic KNX.
    - Select Bit
    - Click on Keys
    - Click on Collection.

3. Create and associate KNX addresses for each number you want to use, in this case from 1 to 6.



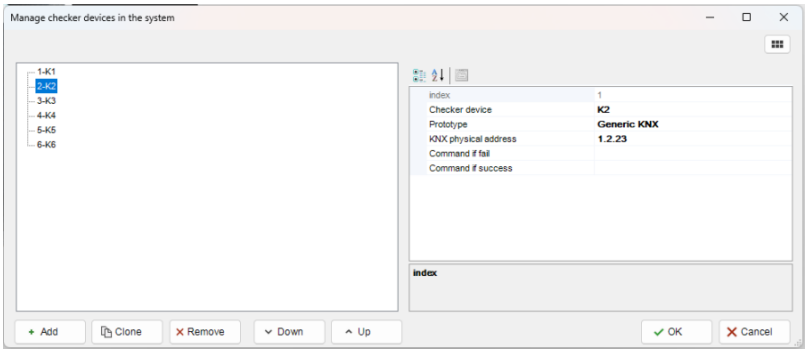
Open Keypads Section

- Create the First Checker Device
- Change it's name to K1
- Set the Prototype to Generic KNX
- Set the KNX address from which the server will monitor the incoming telegrams for the code.
- This means that the same KNX groups can be used for multiple control devices at different points in the project.



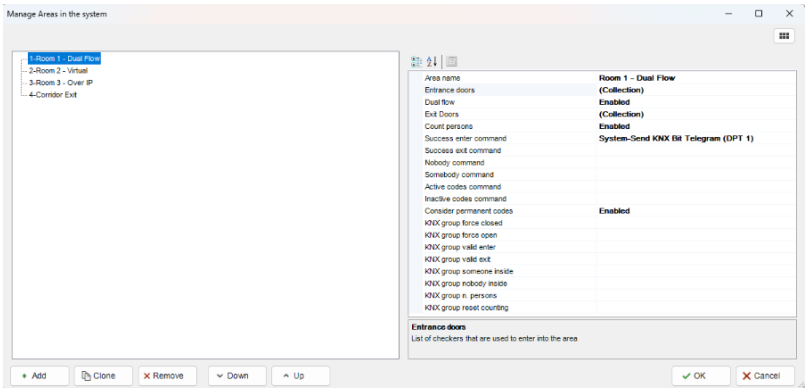
- Create the Second Checker Device
- Change it's name to K2
- Set the Prototype to Generic KNX
- Set the KNX address from which the server will

monitor the incoming telegrams for the code.
- This means that the same KNX groups can be used for multiple control devices at different points in the project.



Go to Area.

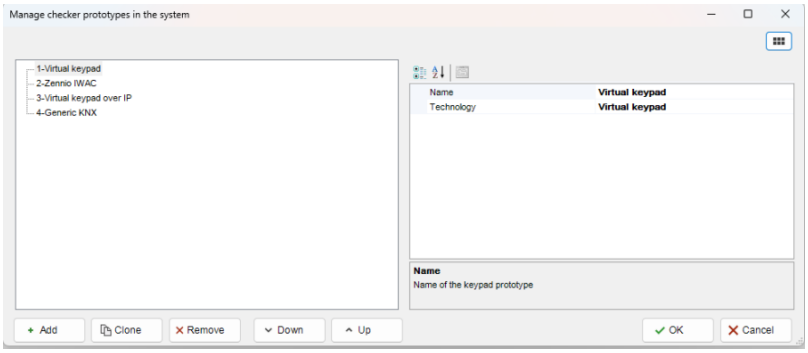
- Create a new Area
- Area Name to Room 1
- Under Entrance Doors, select K1.
- Enable Dual Flow
- Under Exit Doors, select K2
- In the Success Enter Command, set Send KNX Bit telegram.
- **This means that in case of the correct code, a 1-bit KNX telegram will be sent; otherwise, it will do nothing.**



Part 2 - Room 2

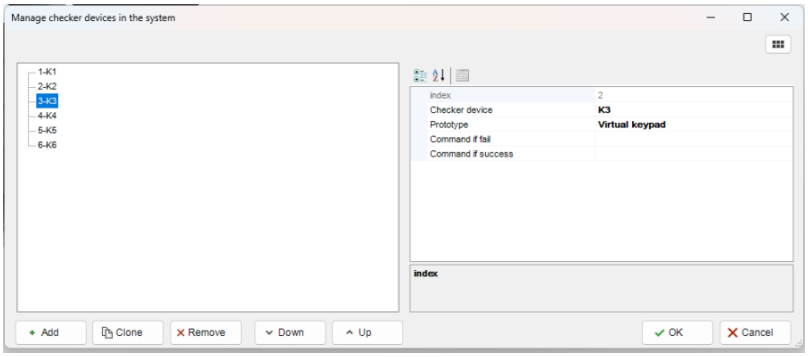
Open Keypad Prototype.

- Click on Add.
- Select Virtual Keypad



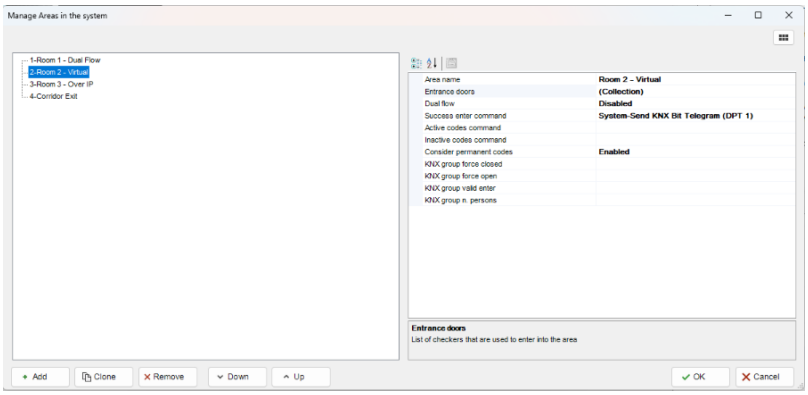
Open Keypads Section

- Create the Third Checker Device
- Change it's name to K3
- Set the Prototype to Virtual Keypad



Go to Area.

- Create a new Area
- Area Name to Room 2 - Virtual
- Under Entrance Door Select K3
- In the Success Enter Command, set Send KNX Bit telegram.
- **This means that in case of the correct code, a 1-bit KNX telegram will be sent; otherwise, it will do nothing.**

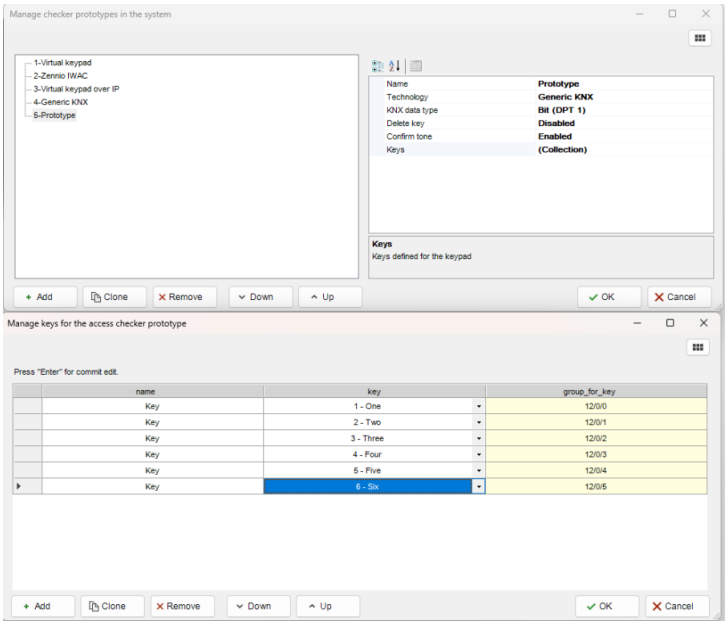


Part 3 - Room 3

Open Keypad Prototype.

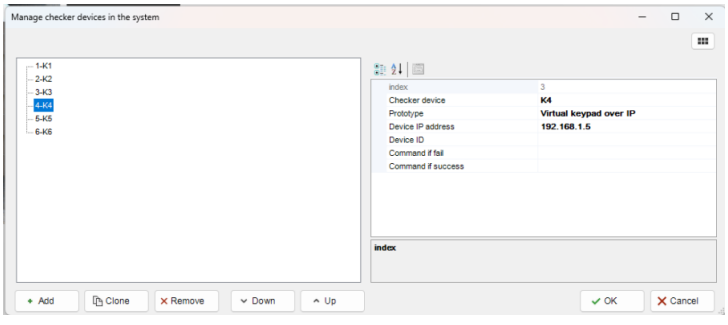
- Click on Add.
- Select Generic KNX.
  - Select Bit
  - Click on Keys
  - Click on Collection.

3. Create and associate KNX addresses for each number you want to use, in this case from 1 to 6.



Open Keypads Section

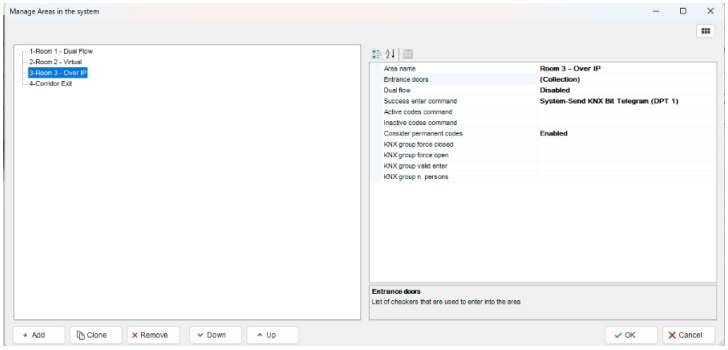
- Create the Fourth Checker Device
- Change it's name to K4
- Set the Prototype to Virtual Keypad
- Set the Device IP address
- Set the DEVICE ID



Go to Area.

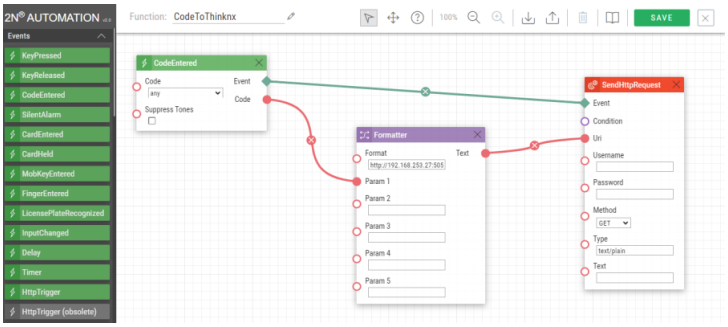
- Create a new Area
- Area Name to Room 3 - IP
- Under Entrance Door Select K4
- In the Success Enter Command, set Send KNX Bit telegram.

- **This means that in case of the correct code, a 1-bit KNX telegram will be sent; otherwise, it will do nothing.**



2N Configuration

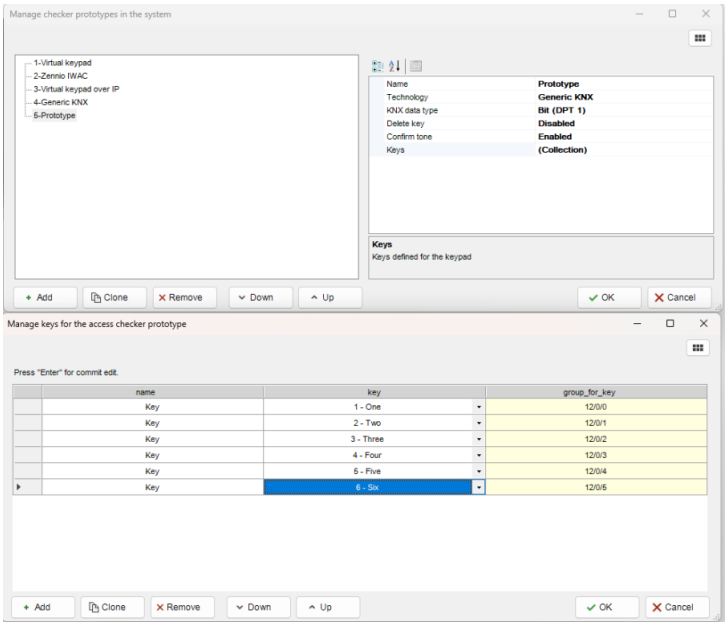
- Go To Automation
- Create a new automation
- Code Entered = Any
- Formatter
- Format: http://IPOFSERVER:5051/images/action.cgi?cmd=accessControlCodeEntered&code={1}
- Send HTTP Request
- Make link as picture.



Part 4 - Corridor

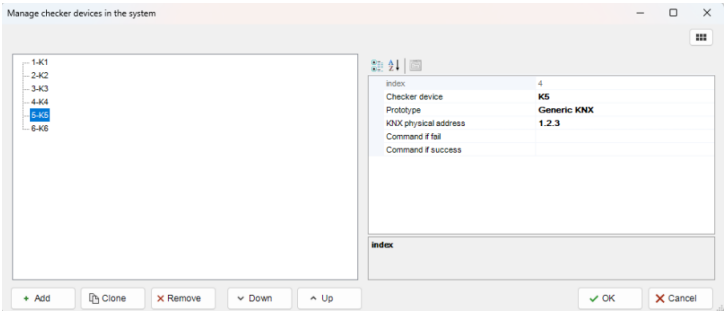
Open Keypad Prototype.

- Click on Add.
- Select Virtual Keypad over IP

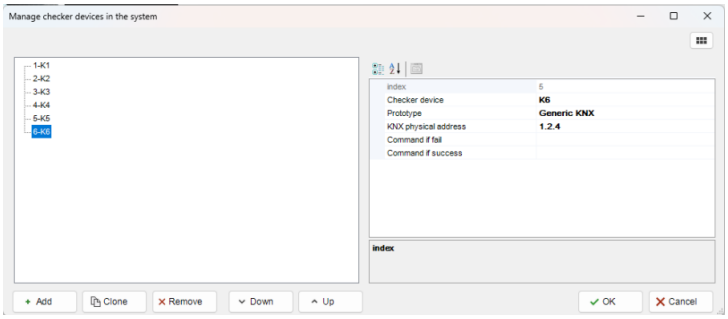


Open Keypads Section

- Create the Fifth Checker Device
- Change it's name to K5
- Set the Prototype to Generic KNX
- Set the KNX address from which the server will monitor the incoming telegrams for the code.
- This means that the same KNX groups can be used for multiple control devices at different points in the project.

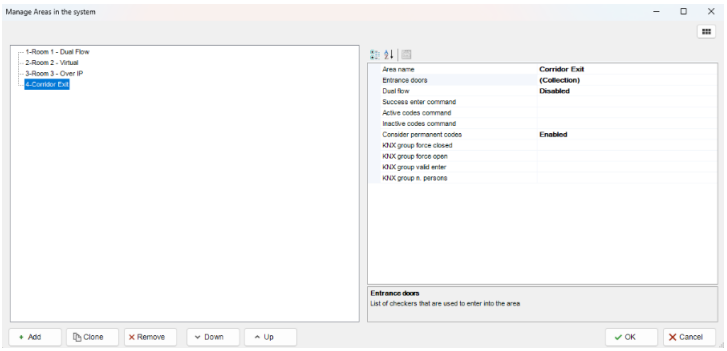


- Create the Sixth Checker Device
- Change it's name to K6
- Set the Prototype to Generic KNX
- Set the KNX address from which the server will monitor the incoming telegrams for the code.
- This means that the same KNX groups can be used for multiple control devices at different points in the project.



Go to Area.

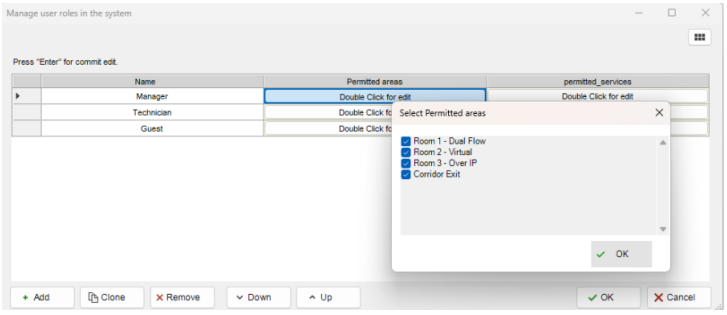
- Create a new Area
- Area Name to Corridor
- In the Success Enter Command, set Send KNX Bit telegram.
- **This means that in case of the correct code, a 1-bit KNX telegram will be sent; otherwise, it will do nothing.**



Part 5 - Create Roles

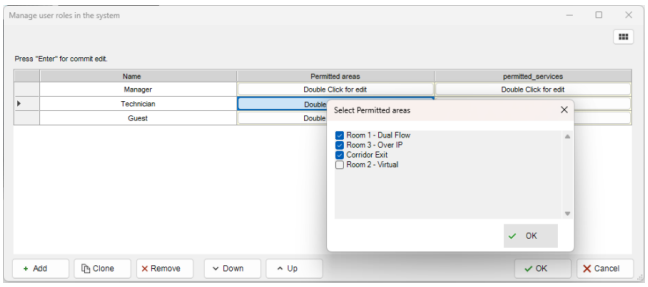
Go to Roles

- Create a new Roles
- Change it's name to Manager
- Enable all the permitted areas
- This Role will be able to enter every Area



- Create a New Roles
- Change it's name to Technician
- Enable Room 1 - Room 3 - Corridor
- This Role will be able to enter Area 1, 3 and

corridor



- Create a New Roles
- Change it's name to Guest
- Enable Room 1 - Corridor
- This Role will be able to enter Room 1 and Corridor