

Automation manuál pro 2N IP interkomy



Konfigurační manuál

Firmware: 2.29

Verze: 2.29

www.2n.cz

Společnost 2N TELEKOMUNIKACE a.s. je českým výrobcem a dodavatelem telekomunikační techniky.



K produktovým řadám, které společnost vyvíjí, patří GSM brány, pobočkové ústředny, dveřní a výtahové komunikátory. 2N TELEKOMUNIKACE a.s. se již několik let řadí mezi 100 nejlepších firem České republiky a již dvě desítky let symbolizuje stabilitu a prosperitu na trhu telekomunikačních technologií. V dnešní době společnost vyváží do více než 120 zemí světa a má exkluzivní distributory na všech kontinentech.



2N[®] je registrovaná ochranná známka společnosti 2N TELEKOMUNIKACE a.s. Jména výrobků a jakákoli jiná jména zde zmíněná jsou registrované ochranné známky a/nebo ochranné známky a/nebo značky chráněné příslušným zákonem.



Pro rychlé nalezení informací a zodpovězení dotazů týkajících se 2N produktů a služeb 2N TELEKOMUNIKACE spravuje databázi FAQ nejčastějších dotazů. Na www.faq.2n.cz naleznete informace týkající se nastavení produktů, návody na optimální použití a postupy „Co dělat, když...“.



Společnost 2N TELEKOMUNIKACE a.s. tímto prohlašuje, že zařízení 2N je ve shodě se základními požadavky a dalšími příslušnými ustanoveními směrnice 1999/5/ES. Plné znění prohlášení o shodě naleznete na CD-ROM (pokud je přiloženo) nebo na www.2n.cz.



Společnost 2N TELEKOMUNIKACE a.s. je vlastníkem certifikátu ISO 9001:2009. Všechny vývojové, výrobní a distribuční procesy společnosti jsou řízeny v souladu s touto normou a zaručují vysokou kvalitu, technickou úroveň a profesionalitu všech našich výrobků.

Obsah:

- 1. Termíny a piktogramy
- 2. Konfigurace Automation pro 2N® IP interkomy
- 3. Události (Events)
- 4. Akce (Action)
- 5. Podmínky (Condition)
- 6. Utilities
- 7. Dostupné digitální vstupy a výstupy
- 8. Příklady použití

1. Termíny a piktogramy

V manuálu jsou použity následující symboly a piktogramy:

Nebezpečí úrazu

- **Vždy dodržujte** tyto pokyny, abyste se vyhnuli nebezpečí úrazu.

Varování

- **Vždy dodržujte** tyto pokyny, abyste se vyvarovali poškození zařízení.

Upozornění

- **Důležité upozornění.** Nedodržení pokynů může vést k nesprávné funkci zařízení.

Tip

- **Užitečné informace** pro snazší a rychlejší používání nebo nastavení.

Poznámka

- Postupy a rady pro efektivní využití vlastností zařízení.

2. Konfigurace Automation pro 2N[®] IP interkomy

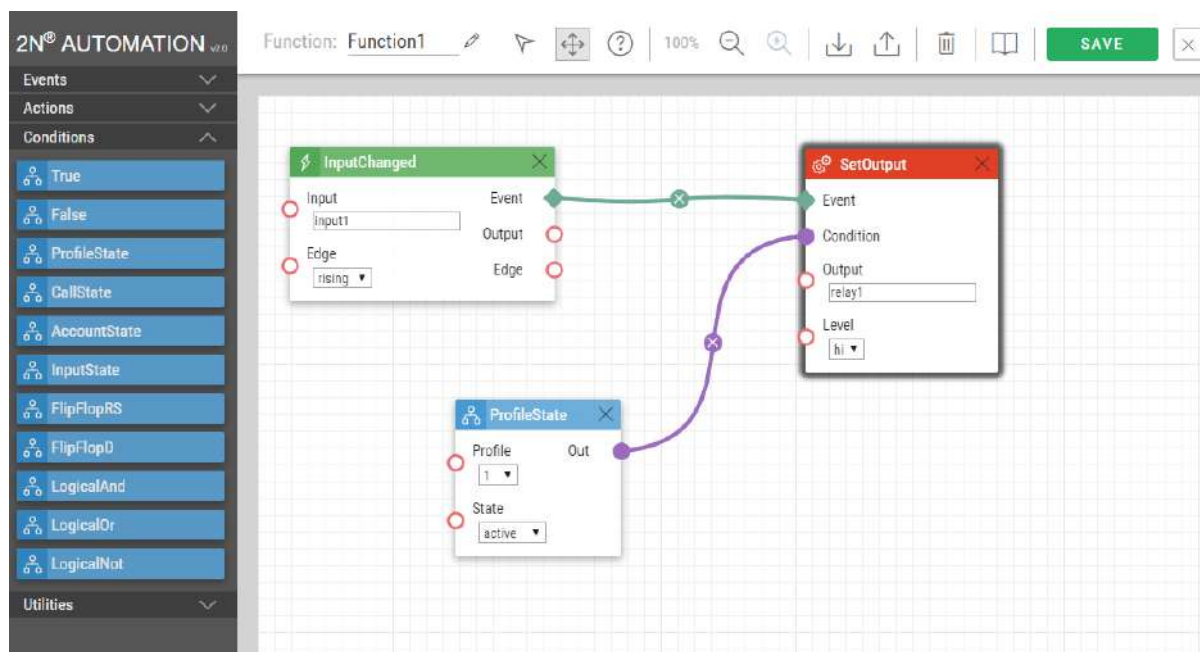
2N IP interkomy poskytují velmi flexibilní možnosti nastavení dle různorodých požadavků uživatele. Existují situace, kdy běžný rozsah nastavení (např. nastavení chování spínačů nebo volání) nedostačuje a pro tyto případy poskytují 2N IP interkomy speciální programovatelné rozhraní **Automation**. Typické použití **Automation** je v aplikacích, které vyžadují složitější propojení se systémy třetích stran.

Poznámka

- Funkcionalita **Automation** je dostupná pouze po vložení platného licenčního klíče pro licenci **Enhanced Integration** nebo **Gold**.

Některé modely interkomů 2N IP jsou vybaveny řadou digitálních vstupů a výstupů, z nichž většinu lze konfigurovat jako běžné spínače 2N IP interkomu (viz kap. Spínače). **Automation** umožňuje využít všechny tyto vstupy a výstupy a propojit je v různých kombinacích.

Automation umožňuje podle potřeby propojit události – **Event**, které v zařízení vznikají (např. stisk tlačítka, protažení RFID karty, změna stavu digitálního vstupu apod.) se specifickými akcemi – **Action** (např. sepnutí digitálního výstupu, přehrání uživatelského zvuku nebo volání apod.). Zároveň lze provádění akcí podmínit různými podmínkami – **Condition** (např. stavem časového profilu, stavem logického vstupu apod.).





Na obrázku výše je znázorněn příklad vzájemného propojení jednotlivých typů bloků – událostí (Events), akcí (Actions) a podmínek (Conditions). Obecně platí, že akce je vždy navázána na jednu konkrétní volitelnou událost a je vykonána, pokud je splněna konkrétní volitelná podmínka. Podmínka není povinná, a pokud není uvedena, akce je provedena vždy, když nastane jí přiřazená událost. **Automation** definuje množství událostí, akcí a podmínek, které lze podle potřeby parametrizovat. Jejich úplný seznam je uveden v následujících kapitolách.

Konkrétní případ propojení uvedený výše na obrázku lze interpretovat takto: Akce **SetOutput** (nastavení digitálního výstupu) se provede v případě vzniku události **InputChanged** (změna logického vstupu input1 z log 0 na log 1) za předpokladu podmínky **ProfileState** (aktivní profil č. 1).

Webové rozhraní **2N IP interkomů** umožňují jednoduchým způsobem konfigurovat propojení bloků na stránce Automatizace. Konfigurace na obrázku odpovídá výše uvedenému příkladu.

Ovládání Automation

Funkce - 2N IP interkom umožňuje vytvořit a propojit až 30 bloků na 5 nezávislých stránkách (nezáleží na tom, zda se jedná o události, akce nebo podmínky). Na událost nebo podmínku lze navázat více akcí. Můžete např. vytvořit 15 akcí a navázat je na 15 událostí nebo 29 akcí navázat na 1 událost.


- **Povoleno** - povoluje definovanou funkci
- **Jméno** - nastavené jméno funkce
- **Stav** - zobrazuje stav funkce Spuštěno/Zastaveno/Prázdné/Error
- **Akce** -  umožňuje nastavit funkci,  umožňuje smazat funkci

2N Helios IP Verso CZ | EN | DE | FR | IT | ES | RU Odhlásit

← Služby 

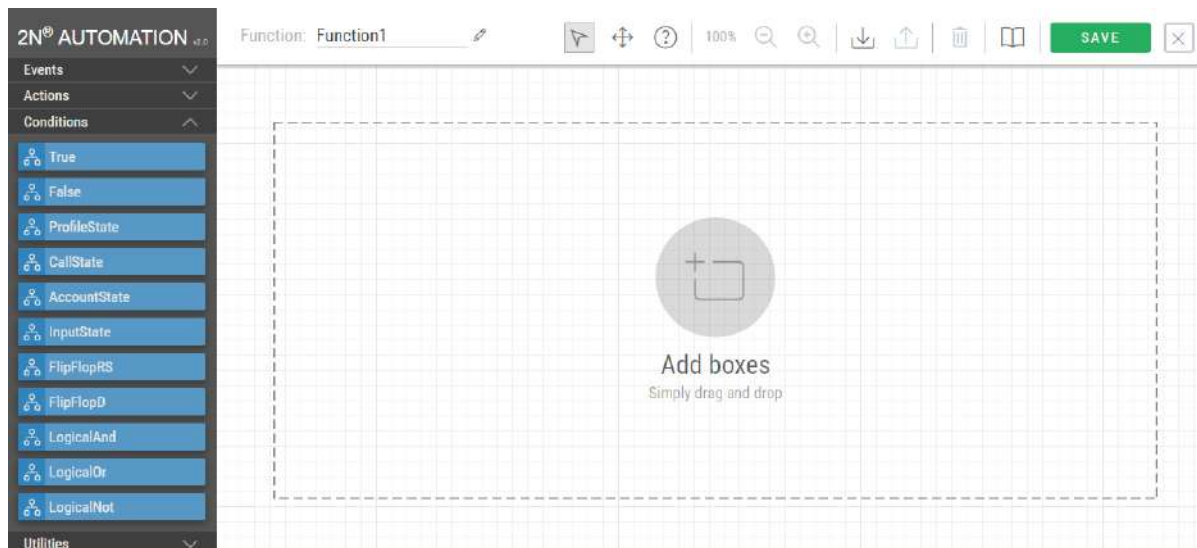
-  Telefon
-  Streamování
-  Orvif
-  E-Mail
-  Automatizace >
- HTTP API

Funkce ▾

POVOLENO	JMÉNO	STAV	AKCE
<input checked="" type="checkbox"/>	Function1	Spuštěno	 
<input checked="" type="checkbox"/>	Function2	Prázdné	 
<input checked="" type="checkbox"/>	Function3	Prázdné	 
<input checked="" type="checkbox"/>	Function4	Prázdné	 
<input checked="" type="checkbox"/>	Function5	Prázdné	 

Ovládací prvky Automation







Situace prázdné funkce Automation je zachycena na obrázku níže.



Sloupec funkčních bloků – obsahuje čtyři skupiny funkčních bloků (**Events, Actions, Conditions, Utilities**). Jednotlivé položky bloků lze přetažením myši umístit do pracovního pole.

Nástrojová lišta – obsahuje nástroje pro úpravy funkce

Pictogram	Popis
Function: <u>Function1</u> 	umožňuje nastavit jméno funkce
	režim úprav funkce
	režim přesunu bloků funkce
	režim nápovědy pro jednotlivé bloky. Po kliknutí na blok se zobrazí nápověda bloku.
68%	nastavené zvětšení

Pictogram	Popis
	oddálení/přiblížení
	import/export
	smazání bloku
	odkaz Automation manuál
	uložit
	ukončení editace funkce


Pracovní plocha slouží k umístění a propojení bloků.

Nastavení parametrů bloků

Ve **sloupci funkčních bloků** vyberte požadovanou událost (Event.xxx), akci (Action.xxx) nebo podmínku (Condition.xxx). U většiny bloků je potřeba uvést jeden nebo více parametrů. Všechny podporované parametry naleznete dále v textu, v kapitolách popisujících vlastnosti jednotlivých bloků. Hodnotu parametru bloku vyplňte do příslušného pole pod názvem parametru.

Změny se uplatní až po stisku tlačítka Save v pravém horním rohu stránky.

Pokud je zadání funkce v pořádku, po stisknutí tlačítka Save se zobrazí v zeleném poli informace o uložení. Pokud zadáte některý z parametrů nesprávně (neplatný název nebo hodnota parametru, případně není uveden povinný parametr bloku), po stisku tlačítka Save se zobrazí v červeném poli informace o chybném zadání. U chybně

zadaných hodnot se zobrazí  a název parametru je zvýrazněn červeně. **Automation** pracuje pouze tehdy, pokud všechny vytvořené bloky jsou správně nakonfigurovány. V opačném případě je funkce **Automation** ve stavu Error.

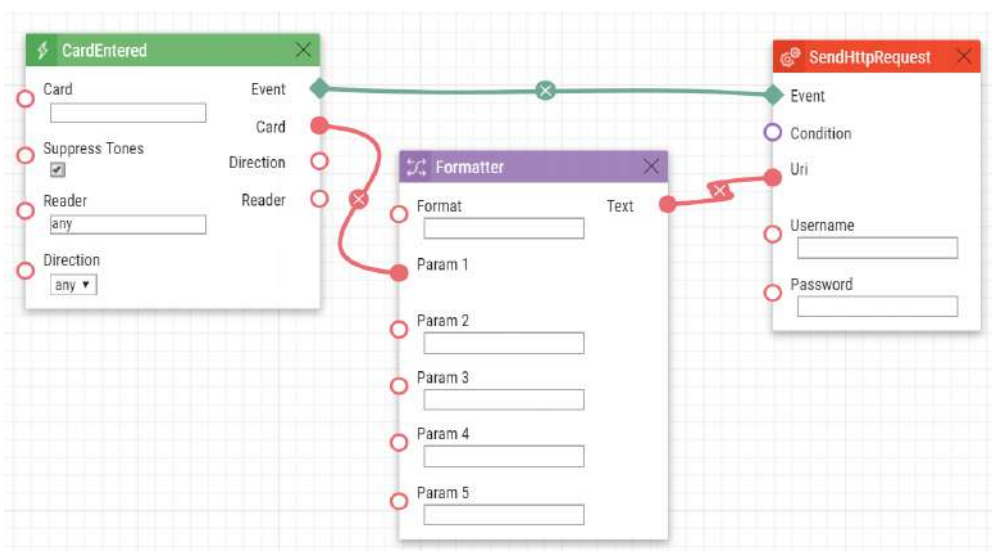
Většina bloků obsahuje parametry (např. Event, Condition, StartEvent apod.), které se odkazují na jiné bloky. Propojení jednotlivých bloků se provede kliknutím na výstup jednoho bloku a přetažením na vstup dalšího bloku.

Tip

- V názvech parametrů a hodnotách se nerozlišují velká a malá písmena.
- Některé parametry bloků jsou nepovinné. Pokud nepovinný parametr v definici bloku neuvedete, bude nastaven na implicitní hodnotu.

Použití výstupních parametrů

V některých případech je užitečné mezi jednotlivými bloky předávat doplňující informace – např. odeslat ID detekované karty pomocí HTTP příkazu jinému zařízení, parametry přijaté v HTTP příkazu použít pro nastavení parametrů navázané akce. K tomuto účelu slouží nástroj Formatter. Hodnoty výstupních parametrů jsou aktualizovány vždy v okamžiku vyvolání události. Hodnotu výstupního parametru lze využít v dalších blocích díky propojení.



Pro přenos výstupního parametru z Events do Formatteru je potřeba spojit příslušný výstupní parametr se vstupem Param1. Tento výstupní parametr je v poli Format dostupný pomocí adresace {číslo_výstupníparametr} .

- Příklad použití formateru pro přenos ID karty:
 - Format: **http://1.1.1.1./card={1}** }
 - Param1: připojeno na výstup Card bloku CardEntered
 - Text: připojeno na vstup Uri bloku SendHttpRequest

Každá událost definuje výstupní parametry **TimeStamp** a **Count**.

Výstupní parametr **TimeStamp** obsahuje zakódovaný datum a čas posledního vyvolání události ve formátu Unix Time (počet sekund od 00:00:00 1. 1. 1970).

Výstupní parametr **Count** obsahuje počet vyvolání události od spuštění zařízení nebo od poslední změny konfigurace bloků. Po každém vyvolání události se výstupní parametr zvyšuje o 1.

Další výstupní parametry jsou specifické pro konkrétní události a jsou popsány v následujících kapitolách.

 **Tip**

- V názvech výstupních parametrů se nerozlišují velká a malá písmena.

 **Upozornění**

- Výstupní parametry nelze použít v parametrech definujících vazby mezi bloky, tj. v parametrech Event, Condition apod.

 **Upozornění**

Za účelem dosažení plné funkčnosti a zaručených výkonů důrazně doporučujeme vždy již při instalaci ověřit aktuálnost používané verze produktu či zařízení. Zákazník tímto bere na vědomí, že produkt či zařízení může dosahovat zaručených výkonů a být plně funkční dle propozic výrobce pouze v případě, je-li používána nejnovější verze produktu či zařízení, která byla otestována na plnou interoperabilitu a která nebyla výrobcem označena jako nekompatibilní s určitými verzemi jiných produktů, a to pouze v souladu s pokyny, návodem či doporučením výrobce a pouze ve spojení s vyhovujícími produkty a zařízeními jiných výrobců. Nejnovější verze jsou dostupné na internetových stránkách https://www.2n.cz/cs_CZ/, popř. jednotlivá zařízení podle svých technických možností umožňují aktualizaci v konfiguračním rozhraní. Používá-li zákazník jinou než nejnovější verzi produktu či zařízení, popř. používá-li verzi, kterou výrobce označil za nekompatibilní s určitými verzemi jiných produktů, nebo používá-li zákazník produkt či zařízení v rozporu s pokyny, návodem či doporučením výrobce, nebo ve spojení s nevyhovujícími produkty či zařízeními jiných výrobců, je srozuměn s veškerými případnými omezeními funkčnosti takového produktu či zařízení a s důsledky s tím spojenými. Použitím jiné než nejnovější verze produktu či zařízení, popř. verze, kterou výrobce označil za nekompatibilní s určitými verzemi jiných produktů, nebo použitím produktu či zařízení v rozporu s pokyny, návodem či doporučením výrobce, popř. použitím s nevyhovujícími produkty či zařízeními jiných výrobců, zákazník souhlasí s tím, že společnost 2N TELEKOMUNIKACE a. s. není odpovědná za jakékoli omezení funkčnosti takového produktu ani za újmu související s takovým případným omezením funkčnosti.

3. Události (Events)

Automation definuje následující typy událostí:

- **AudioLoopTest** - proveden audio test
- **CallStateChanged** - změna stavu hovoru
- **CardEntered** - protažení RFID karty
- **CardHeld** - přidržení RFID karty
- **CodeEntered** - zadaný numerický kód
- **Delay** - definované zpoždění
- **DoorOpenTooLong** - otevřené dveře déle než stanovený limit
- **DtmfEntered** - detekován numerický kód v hovoru
- **DtmfPressed** - přijat DTMF kód v hovoru
- **FingerEntered** - přiložení prstu k biometrické čtečce
- **HttpTrigger** - přijat HTTP příkaz
- **InputChanged** - změna digitálního vstupu
- **KeyPressed** - stisk tlačítka
- **KeyReleased** - uvolnění tlačítka
- **LockdownStateChanged** - nouzové uzamčení
- **MobKeyEntered** - autorizace pomocí čtečky
- **MotionDetected** - detekován pohyb před kamerou
- **MulticastTrigger** - přijat příkaz pro více zařízení současně
- **OnvifVirtualOutputChanged** - příjem události z VMS
- **NoiseDetected** - hluk detekován mikrofonom
- **RegistrationStateChanged** - změna stavu registrace účtu SIP
- **Rebooted** - detekován start nebo restart zařízení
- **SilentAlarm** - aktivován tichý alarm
- **Time** - definovaný čas
- **Timer** - časovač periodické události
- **UnauthorizedDoorOpen** - neautorizované otevření dveří
- **UserAuthorized** - autorizace uživatele

- **OutputChanged** – změna výstupu

Detailní popis událostí, jejich vstupní parametry a použití je popsáno v následujícím textu.

AudioLoopTest

Blok **AudioLoopTest** definuje událost generovanou po provedení testu funkčnosti mikrofону a reproduktoru zařízení (Audio Test). Na základě výsledku testu lze provést další akce.

Vstupní parametry

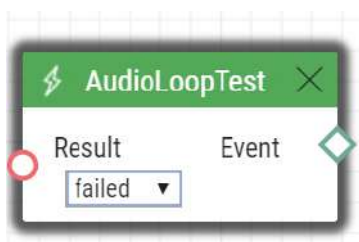
- **Result** - umožňuje specifikovat požadovaný výsledek audio testu.
 - Platné hodnoty:
 - **any** - událost je generována po každém provedení testu bez ohledu na výsledek
 - **passed** - událost je generovaná po provedení testu, pokud je výsledek pozitivní
 - **failed** - událost je generovaná po provedení testu, pokud je výsledek negativní
 - Parametr je nepovinný, implicitní hodnota je **failed**.

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.

Příklad

Událost generovaná po provedení audio testu, pokud je výsledek testu negativní (tj. mikrofón nebo reproduktor není funkční):



CallStateChanged

Blok **CallStateChanged** definuje událost generovanou při změně stavu hovoru (vyzvánění, spojení navázáno, hovor ukončen apod.)

Vstupní parametry

- **State** – definuje změnu stavu hovoru.
 - Platné hodnoty:
 - **ringing** – okamžik začátku vyzvánění
 - **connected** – okamžik úspěšného spojení hovoru
 - **terminated** – okamžik ukončení hovoru
- **Direction** – definuje směr hovoru.
 - Platné hodnoty:
 - **incoming** – příchozí hovory
 - **outgoing** – odchozí hovory
 - **any** – oba směry.
 - Parametr je nepovinný, implicitní hodnota je **any**.
- **Number** – definuje identifikátor (telefonní číslo, IP adresu), který se má porovnávat s identifikátorem volajícího, aby událost nastala. Je možné zadat více identifikátorů oddělených čárkou. Nevyplněná hodnota má stejný význam jako hodnota **any**.
 - Parametr je nepovinný, implicitní hodnota je **any**.

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **State** – zaznamenaný stav hovoru, který vyvolal tuto událost. Možné hodnoty odpovídají parametru State.
- **Direction** – zaznamenaný směr hovoru, který vyvolal tuto událost. Možné hodnoty jsou **incoming** nebo **outgoing**.
- **Uri** – výstup obsahující kompletní SIP uri protistrany.

Příklad

Událost generovaná při vyzvednutí hovoru z čísla 1234:



CardEntered

Blok **CardEntered** definuje událost generovanou při detekci (protažení) RFID karty se zadaným ID (pouze pro modely vybavených čtečkou RFID karet).

Vstupní parametry

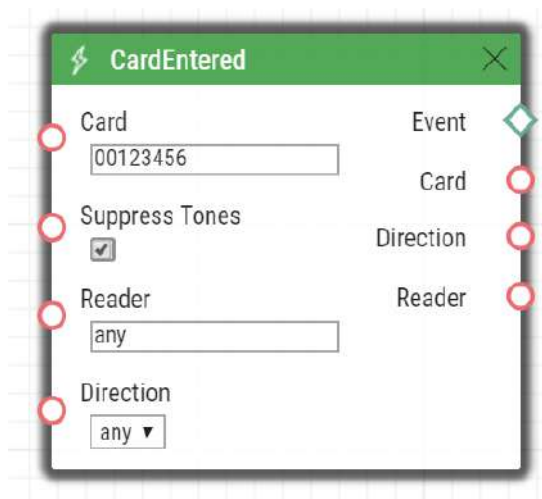
- **Card** – definuje ID RFID karty, viz kapitola Čtečka Karet v konfiguračním manuálu 2N IP interkomů.
 - Platné hodnoty:
 - **valid** – libovolná platná karta (uvedena v seznamu karet v interkomu)
 - **invalid** – libovolná neplatná karta
 - **any** – libovolná platná i neplatná karta
 - **< >** (**prázdná hodnota**) – událost nebude vyvolána
 - Mimo nabízených možností lze také vyplnit ID karty manuálně.
 - **SuppressTones** – umožňuje potlačit zvukovou signalizaci spojenou s detekcí neplatné karty. Parametr je nepovinný.
 - Platné hodnoty:
 - **zakázáno** – zvuky nejsou potlačeny
 - **povoleno** – zvuky jsou potlačeny (implicitní hodnota)
 - **Reader** – definuje použitou čtečku karet
 - Platné hodnoty
 - **internal_cardreader** – interní čtečka karet (2N[®] IP Vario, Force)
 - **external_cardreader** – externí čtečka karet (2N[®] IP Vario, Force)
 - **any** – libovolná čtečka
 - Manuálně lze také vyplnit název modulu konfigurovaný v parametru Jméno modulu v sekci Hardware / Rozšiřující moduly / menu použitého modulu (2N[®] IP Verso).
 - Parametr je nepovinný, implicitní hodnota je **any**
 - **Direction** – definuje směr průchodu
 - Platné hodnoty
 - **in** – vstupní čtečka
 - **out** – výstupní čtečka
 - **any** – nespecifikovaný směr
 - Parametr je nepovinný, implicitní hodnota je **any**.

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **Card** – zaznamenané ID detekované karty, která naposledy vyvolala tuto událost.
- **Direction** – směr průchodu, hodnota je nastavena u čtečky karet (**in**, **out**, **any**).
- **Reader** – jméno naposledy použité čtečky karet (**internal_cardreader** pro 2N[®] IP Vario a 2N[®] IP Force, **external_cardreader** pro externí čtečku karet, **<jméno_modulu>** pro 2N[®] IP Verso a 2N[®] Access Unit).

Příklad

Událost generovaná při protažení karty s ID 00123456:



CardHeld

Blok **CardHeld** definuje událost generovanou při přidržení RFID karty se zadaným ID (pouze pro modely vybavených čtečkou RFID karet). Událost se vygeneruje přidržením karty na 4 s.

Vstupní parametry

- **Card** – definuje ID RFID karty, viz kapitola Čtečka Karet v konfiguračním manuálu pro 2N IP interkomy.
 - Platné hodnoty:
 - **valid** – libovolná platná karta (uvedena v seznamu karet v interkomu)
 - **invalid** – libovolná neplatná karta
 - **any** – libovolná platná i neplatná karta
 - **< >** (**prázdná hodnota**) – událost nebude vyvolána
 - Mimo nabízených možností lze také vyplnit ID karty manuálně.
 - **SuppressTones** – umožňuje potlačit zvukovou signalizaci spojenou s detekcí neplatné karty. Parametr je nepovinný.
 - Platné hodnoty:
 - **zakázáno** – zvuky nejsou potlačeny
 - **povoleno** – zvuky jsou potlačeny (implicitní hodnota)
 - **Reader** – definuje použitou čtečku karet
 - Platné hodnoty
 - **internal_cardreader** – interní čtečka karet (2N[®] IP Vario, Force)
 - **external_cardreader** – externí čtečka karet (2N[®] IP Vario, Force)
 - **any** – libovolná čtečka
 - Manuálně lze také vyplnit název modulu konfigurovaný v parametru Jméno modulu v sekci Hardware / Rozšiřující moduly / menu použitého modulu (2N[®] IP Verso).
 - Parametr je nepovinný, implicitní hodnota je **any**
 - **Direction** – definuje směr průchodu
 - Platné hodnoty
 - **in** – vstupní čtečka
 - **out** – výstupní čtečka
 - **any** – nespecifikovaný směr
 - Parametr je nepovinný, implicitní hodnota je **any**

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.
- **Card** - zaznamenané ID detekované karty, která naposledy vyvolala tuto událost.
- **Direction** - směr průchodu, hodnota je nastavena u čtečky karet (**In**, **Out**, **Unspecified**).
- **Reader** - jméno naposledy použité čtečky karet (**internal_cardreader** pro 2N[®] IP Vario a 2N[®] IP Force, **external_cardreader** pro externí čtečku karet, <jméno_modulu> pro 2N[®] IP Verso a 2N[®] Access Unit).

Příklad

Událost generovaná při přidržení karty s ID 0012456:



CodeEntered

Blok **CodeEntered** definuje událost generovanou při zadání numerického kódu potvrzeného klávesou * (pouze pro modely s numerickou klávesnicí).

Vstupní parametry

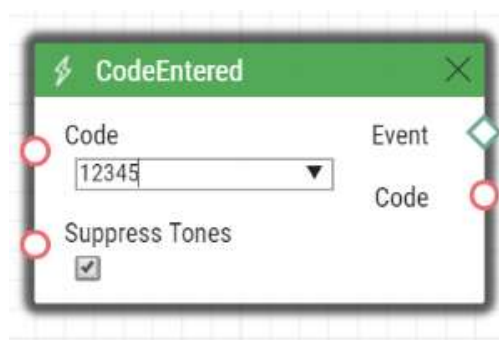
- **Code** - definuje numerický kód.
 - Platné hodnoty:
 - numerický kód - např. 12345
 - **valid** - libovolný platný kód
 - **invalid** - libovolný neplatný kód
 - **any** - libovolný platný i neplatný kód
 - **< >** (**prázdná hodnota**) - událost nebude vyvolána
- **SuppressTones** - umožňuje potlačit zvukovou signalizaci spojenou s přijetím neplatného numerického kódu. Parametr je nepovinný.
 - Platné hodnoty:
 - **zakázáno** - zvuky nejsou potlačeny
 - **povoleno** - zvuky jsou potlačeny (implicitní hodnota)

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.
- **Code** - zaznamenaný numerický kód, který naposledy vyvolal tuto událost.

Příklad

Událost vygenerovaná po zadání kódu 12345* na klávesnici



Delay

Blok **Delay** definuje událost generovanou na základě jiné definované události s definovaným zpožděním. Pomocí této události lze zpozdřit reakci na událost o definovaný časový interval (Delay).

Vstupní parametry

- **Start** - definuje událost, která odstartuje zpoždění.
- **Stop** - definuje událost, která zastaví zpoždění. Tento parametr je nepovinný.
- **Delay** - definuje délku zpoždění. Zadat lze pouze číselnou hodnotu, nikoliv hodnotu získanou z výstupního parametru jiného eventu.
- Příklad platných hodnot parametrů výše:
 - 10 - 10 sekund (jednotky není nutné uvádět)
 - 10 s - 10 sekund
 - 100 ms - 100 milisekund

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.

Příklad

Událost generovaná po 1s od vzniku události na řádku 1:



DoorOpenTooLong

Blok **DoorOpenTooLong** definuje událost generovanou v případě, že dveře zůstanou otevřeny déle než stanovený limit.

Vstupní parametry

- **State** – stav čidla dveří, který vygeneruje událost
 - Platné hodnoty:
 - **Start** – začátek události
 - **End** – konec události

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.

Příklad

Událost vygenerovaná při otevření dveří na dobu delší, než je nastavena.



DtmfEntered

Blok **DtmfEntered** definuje událost generovanou při zadání numerického kódu potvrzeného klávesou * pomocí DTMF v příchozím nebo odchozím hovoru.

Vstupní parametry

- **Code** – definuje numerický kód.
 - Platné hodnoty:
 - **numerický kód** – např. 12345
 - **< >** (prázdná hodnota) – událost nebude vyvolána
- **SuppressTones** – umožňuje potlačit zvukovou signalizaci spojenou s přijetím neplatného DTMF kódu. Parametr je nepovinný.
 - Platné hodnoty:
 - **zakázáno** – zvuky nejsou potlačeny
 - **povoleno** – zvuky jsou potlačeny (implicitní hodnota)

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **Code** – zaznamenaný přijatý numerický kód, který naposledy vyvolal tuto událost.

Příklad

Událost vygenerovaná při detekci DTMF kódu 12345*



DtmfPressed

Blok **DtmfPressed** definuje událost generovanou při přijetí definovaného DTMF kódu nebo libovolného DTMF kódu z definované skupiny. DTMF kódu jsou detekovány jak v příchozích, tak odchozích hovorech.

Vstupní parametry

- **Key** - definuje DTMF kód (příp. skupinu DTMF kódů). V případě, že tento parametr není uveden, událost je vygenerovaná při detekci libovolného DTMF kódu (implicitní hodnota Any).
 - Platné hodnoty
 - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, *, #, A, B, C, D
 - **any** pro libovolný DTMF kód (implicitní hodnota).
 - Pokud chcete specifikovat skupinu kódů, oddělte hodnoty čárkou.
- **Direction** - definuje směr hovoru.
 - Platné hodnoty:
 - **incoming** - příchozí hovory
 - **outgoing** - odchozí hovory
 - **any** - oba směry
 - Parametr je nepovinný, implicitní hodnota je **any**.

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.
- **Pressed Key** - zaznamenaný přijatý DTMF kód, který naposledy vyvolal tuto událost. DTMF kód je uložen stejném formátu jako parametr Key.

Příklad

Událost vygenerovaná při detekci DTMF kódu #:



FingerEntered

Blok **FingerEntered** definuje událost generovanou při rozeznání známého otisku prstu na čtečce otisků prstů (pouze pro zařízení s biometrickou čtečkou).

Vstupní parametry

- **Fingerprint** – definuje platnost zadaného otisku prstu.
 - Platné hodnoty:
 - **valid** – otisk prstu patří uživateli
 - **invalid** – otisk prstu je neznámý
 - **any** – jakýkoli zadaný otisk prstu.
- **Finger** – definuje jeden ze dvou otisků prstů uživatele.
 - Platné hodnoty:
 - **any** – jakýkoli otisk prstu uživatele
 - **F1** – otisk byl při registraci otisků prstů označen "F1" kvůli automatizaci
 - **F2** – otisk byl při registraci otisků prstů označen "F2" kvůli automatizaci
- **Suppress Tones** – umožňuje potlačit zvukovou signalizaci spojenou s detekcí neplatného uživatele (otisku). Parametr je nepovinný.
 - Platné hodnoty:
 - **zakázáno** – zvuky nejsou potlačeny
 - **povoleno** – zvuky jsou potlačeny (implicitní hodnota).

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **User** – identifikace (UUID) uživatele, jehož otisk prstu vyvolal tuto událost.

Příklad

Událost generovaná při platném přiložení prstu k biometrické čtečce.



HttpTrigger

Blok HttpTrigger definuje událost generovanou při přijetí HTTP příkazu odeslaného na HTTP server interkomu. Po přijetí HTTP příkazu ve tvaru tvaru `http://ip_addr/enu/trigger/id`, dojde k vygenerování události, která má parameter id shodný s uvedeným za slovem 'trigger/' v http příkazu. Na tento požadavek intercom odpoví jednoduchou odpovědí (200 OK).

Vstupní parametry

- **Name** - definuje jednoznačný identifikátor pro HTTP příkaz. Může obsahovat znaky abecedy a číslice.

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.
- **Params** - obsahuje parametry odeslané v bloku Akce SendHttpRequest nebo v příkazu, který přijde na 2N IP interkom.

Událost HttpTrigger je vždy vyvolaná přijetím HTTP příkazu, který umožňuje nést seznam uživatelských parametrů, který je součástí URI příkazu.

`http://ip_adresa/enu/trigger/id?param1=value1¶m2=value2`

Seznam parametrů následuje za znakem ?. Každý parametr musí mít uveden název a hodnotu. Hodnota a název parametru jsou odděleny znakem =. Pokud seznam obsahuje více než jeden parametr, pak jsou jednotlivé vstupní parametry odděleny znakem &.

Jednotlivé vstupní parametry jsou při přijetí HTTP příkazu k dispozici jako výstupní parametry bloku HttpTrigger. Název výstupního parametru je shodný s názvem předaného parametru, tedy `$(line.param1)` a `$(line.param2)`.

Příklad

Událost generovaná při přijetí HTTP příkazu ve tvaru `http://ip_addr/enu/trigger/opendoor`:



InputChanged

Blok **InputChanged** definuje událost generovanou při změně logické úrovně na definovaném digitálním vstupu.

Vstupní parametry

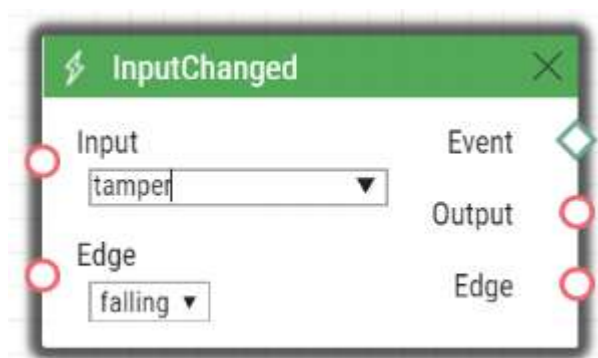
- **Input** – definuje logický vstup.
 - Platné hodnoty:
 - **tamper** – vstup tamper spínače
 - **input1** – digitální vstup 1
 - **input2** – digitální vstup 2
 - **cr_input1** – digitální vstup 1 na čtečce karet
 - **cr_input2** – digitální vstup 2 na čtečce karet
 - Seznam platných hodnot se může lišit pro různé modely interkomů **2N IP**, viz kap. **Dostupné digitální vstupy a výstupy**.
- **Edge** – definuje detekovanou změnu na digitálním vstupu.
 - Platné hodnoty:
 - **falling** – sestupná hrana, změna z log 1 na log 0
 - **rising** – vzestupná hrana, změna z log 0 na log 1
 - Parametr je nepovinný, implicitní hodnota je rising.

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **Output** – zaznamenané ID vstupu, jehož změna naposledy vyvolala tuto událost. Možné hodnoty odpovídají hodnotám parametru Input.
- **Edge** – zaznamenaná změna digitálního vstupu, která naposledy vyvolala tuto událost. Možné hodnoty jsou falling nebo rising.

Příklad

Událost generovaná při rozpojení tamper spínače (otevření zařízení):



KeyPressed

Blok **KeyPressed** definuje událost generovanou při stisku definovaného tlačítka nebo libovolného tlačítka z definované skupiny.

Vstupní parametry

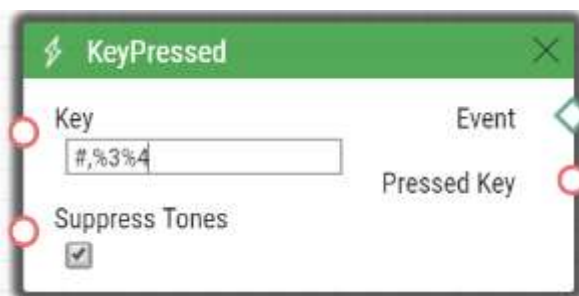
- **Key** – definuje tlačítko (příp. skupinu tlačítek). V případě, že tento parametr není uveden, událost je vygenerovaná při stisku libovolné klávesy (implicitní hodnota Any).
 - Platné hodnoty:
 - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, *, # – pro tlačítka na numerické klávesnici
 - %1, %2, ..., %999 – pro tlačítka rychlé volby
 - any – pro libovolné tlačítko (implicitní hodnota).
 - Pokud chcete specifikovat více tlačítek, oddělte hodnoty čárkou.
- **SuppressTones** – umožňuje potlačit zvukovou signalizaci spojenou se stiskem nenaprogramovaného tlačítka rychlé volby. Parametr je nepovinný.
 - Platné hodnoty:
 - zakázáno – Zvuky nejsou potlačeny
 - povoleno – Zvuk y jsou potlačeny (implicitní hodnota)

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **Pressed Key** – zaznamenaný kód stisknuté klávesy, která naposledy vyvolala tuto událost. Kód klávesy je uložen stejném formátu jako parametr Key.

Příklad

Událost vygenerovaná při stisku klávesy # a tlačítka rychlé volby 3 nebo 4:



KeyReleased

Blok **KeyReleased** definuje událost generovanou při uvolnění definovaného stisknutého tlačítka nebo libovolného tlačítka z definované skupiny.

Poznámka

- Model Vario: událost je generována při stisku klávesy, odpovídá tedy události KeyReleased

Vstupní parametry

- **Key** - definuje tlačítko (příp. skupinu tlačítek). V případě, že tento parametr není uveden, událost je vygenerovaná při uvolnění libovolné klávesy (implicitní hodnota Any).
 - Platné hodnoty:
 - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, *, # - pro tlačítka na numerické klávesnici
 - %1, %2, ..., %999 - pro tlačítka rychlé volby
 - any - pro libovolné tlačítko (implicitní hodnota).
 - Pokud chcete specifikovat více tlačítek, oddělte hodnoty čárkou.

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.
- **Released Key** - zaznamenaný kód uvolněné klávesy, která naposledy vyvolala tuto událost. Kód klávesy je uložen stejném formátu jako parametr Key.

Příklad

Událost vygenerovaná při uvolnění klávesy 1 a tlačítka rychlé volby 2:



LockdownStateChanged

Blok `LockdownStateChanged` definuje událost generovanou při aktivaci nouzového uzamčení.

Vstupní parametry

- **Door** – nastavuje parametr směru průchodu.
 - Platné hodnoty:
 - **any** – libovolný typ události
 - **entry** – vstup
 - **exit** – výstup
- **State** – stav čidla dveří, který vygeneruje událost.
 - Platné hodnoty:
 - **any** – libovolný typ události
 - **unlocked** – odemčený
 - **locked** – uzamčený

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **Door** – směr průchodu. Možné hodnoty jsou `entry` a `exit`.
- **State** – stav nouzového uzamčení, které vygeneruje událost. Možné hodnoty jsou `any`, `unlocked` a `locked`.

Příklad

Událost generovaná při aktivaci nouzového uzamčení.



MobKeyEntered

Blok **MobKeyEntered** definuje událost generovanou při přečtení známého mobilního klíče (Mobile Key) na Bluetooth čtečce (pouze pro zařízení s Bluetooth čtečkou).

Vstupní parametry

- **MobKey** – definuje platnost aktuálního Mobile Key.
 - Platné hodnoty:
 - **valid** – Mobile Key patří uživateli
 - **invalid** – Mobile Key je neznámý
 - **any** – jakýkoli zadaný Mobile Key.
- **Suppress Tones** – umožňuje potlačit zvukovou signalizaci spojenou s detekcí neplatného uživatele (Mobile Key). Parametr je nepovinný.
 - Platné hodnoty:
 - **zakázáno** – zvuky nejsou potlačeny
 - **povoleno** – zvuky jsou potlačeny (implicitní hodnota).

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **User** – identifikace (UUID) uživatele, jehož Mobile Key vyvolal tuto událost.

Příklad

Událost generovaná při platné autorizaci pomocí čtečky.



MotionDetected

Blok **MotionDetected** definuje událost generovanou v případě detekování pohybu. Pohyb může být detekován pouze interní kamerou. Vstupní parametry detekce pohybu jsou nastaveny v menu Hardware / Kamera / Interní kamera, sekce Nastavení detekce pohybu.

Vstupní parametry

- **State** – umožňuje nastavit, jestli bude událost generovat v okamžiku začátku nebo konce pohybu.
 - Platné hodnoty:
 - **start** – začátek detekce pohybu
 - **end** – konec detekce pohybu
 - Parametr je nepovinný, implicitní hodnotou je **start**.

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.

Příklad

Událost generovaná v případě, kdy začal být detekován pohyb.



MulticastTrigger

Blok **MulticastTrigger** definuje událost generovanou při přijetí příkazu odeslaného pomocí akce **SendMulticastRequest**. Příkaz tvoří zpráva odeslaná pomocí UDP protokolu na multicastovou adresu (235.255.255.250:4433) a tudíž může být přijata více zařízeními současně. Zpráva obsahuje identifikaci příkazu (parametr **Command**) a případné další Vstupní parametry. Zpráva může být zabezpečena pomocí hesla (parametr **Password**).

Vstupní parametry

- **Command** – definuje identifikátor příkazu. Pomocí tohoto parametru lze odlišit různé typy odesílaných příkazů. Blok **MulticastTrigger** reaguje na akci **SendMulticastRequest** pouze tehdy, pokud má uveden stejný identifikátor příkazu. Identifikátor může být libovolný text obsahující znaky A-Z, a-z a 0-9. V názvu příkazu se rozlišují velká a malá písmena.
- **CheckTime** – umožňuje zapnout režim kontroly času přijetí příkazu s časem uvedeným ve zprávě příkazu a tím zamezit možnosti útoku pomocí opakování již zpracované zprávy. Tato funkce vyžaduje synchronizovaný čas (pomocí NTP serveru) na všech zařízeních vysílajících a přijímajících příkazy.
 - Platné hodnoty:

zakázáno – čas zprávy není kontrolován

povoleno – čas zprávy je kontrolován (vyšší bezpečnost)

Parametr je nepovinný, implicitní hodnota je zakázána.

- **Password** – definuje heslo, pomocí kterého je příkaz zabezpečen proti neautorizovanému přístupu. Uvedené heslo se musí shodovat s heslem definovaným v akci **SendMulticastRequest**, na kterou má **MulticastTrigger** reagovat.

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **Params** – obsahuje parametry odeslané v bloku Akce **SendMulticastRequest**.

Událost **MulticastTrigger** je vždy vyvolaná přijetím hromadného příkazu, který umožňuje nést seznam uživatelských parametrů (parameter **Params** akce **MulticastRequest**). Každý z těchto parametrů je identifikován vlastním jménem, který si zvolí uživatel a tyto odeslané vstupní parametry jsou k dispozici jako výstupní parametry (s identickým jménem) v bloku **MulticastTrigger**.

Příklad: Byl přijat hromadný příkaz vyvolaný akcí MulticastRequest, u které byl uveden parametr Params="AAA=123". Událost MulticastTrigger, která tento příkaz zpracuje, bude mít automaticky nastavený výchozí parametr AAA na hodnotu 123. Na tento výchozí parametr je možné se odkázat v dalších navazujících blocích.

Příklad

Událost generovaná při přijetí hromadného příkazu opendoor:



OnvifVirtualOutputChanged

Blok OnvifVirtualOutputChanged slouží k přenosu události z VMS do 2N IP interkomu.

Vstupní parametry

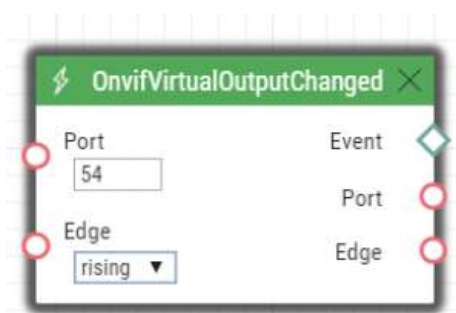
- **Port** – hodnota portu, který nastavuje VMS. Parametr má rozmezí 50–54.
- **Edge** – definuje detekovanou změnu na virtuálním vstupu.
 - Platné hodnoty:
 - **falling** – sestupná hrana, změna z log 1 na log 0
 - **rising** – vzestupná hrana, změna z log 0 na log 1
 - Parametr je nepovinný, implicitní hodnota je rising.

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **Port** – hodnota portu změněného z VMS. Možné hodnoty jsou 50–54.
- **Edge** – zaznamenaná změna virtuálního vstupu, která naposledy vyvolala tuto událost. Možné hodnoty jsou falling nebo rising.

Příklad

Událost generovaná v případě změny hodnoty virtuálního portu.



NoiseDetected

Blok **NoiseDetected** definuje událost generovanou v případě detekování hluku. Hluk může být detekován pouze interním mikrofonom. Vstupní parametry detekce hluku jsou nastaveny v menu Hardware / Audio, sekce Nastavení detekce hluku.

Vstupní parametry

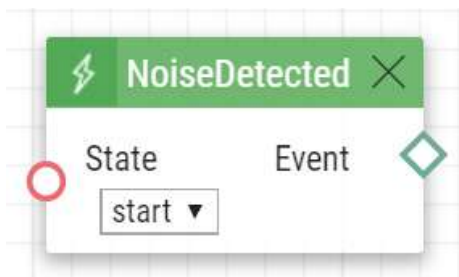
- **State** – umožňuje nastavit, jestli bude událost generovat v okamžiku začátku nebo konce hluku.
 - Platné hodnoty:
 - **start** – začátek detekce hluku
 - **end** – konec detekce hluku
 - Parametr je nepovinný, implicitní hodnota je 1.

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.

Příklad

Událost generovaná v případě, kdy začal být detekován hluk.



RegistrationStateChanged

Blok **RegistrationStateChanged** definuje událost generovanou při změně stavu registrace účtu SIP. Registrace SIP je nastavena v menu Služby / SIP 1 a SIP 2. Ke změně stavu registrace dochází po zapnutí interkomu a potom při změnách konfigurace nebo například při ztrátě spojení s registrarem.

Vstupní parametry

- Platné hodnoty:
 - 1 - účet číslo 1
 - 2 - účet číslo 2
 - Any - kterýkoli účet
- Parametr je nepovinný, implicitní hodnota je **Any**.
- **State** - stav registrace, který vygeneruje událost
 - Platné hodnoty:
 - **Unregistered** - interkom není registrován
 - **Registering** - probíhá registrace
 - **Registered** - interkom je registrován
 - **Unregistering** - probíhá odregistrace
 - **Any** - kterákoli ze změn stavu
- Parametr je nepovinný, implicitní hodnota je **Any**.

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.
- **Account** - volba účtu SIP, jehož události mají být sledovány.
- **State** - stav registrace, který vygeneruje událost

Příklad

Událost generovaná v případě, kdy došlo k odregistrování interkomu (například, když registrar neodpověděl na periodickou žádost o registraci):



Rebooted

Blok **Rebooted** definuje událost generovanou v případě, že došlo ke startu nebo restartu zařízení.

Vstupní parametry

Tento blok nemá žádné Vstupní parametry.

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **Time** – hodnota času, kdy došlo ke startu či restartu zařízení.
- **Reason** – hodnota obsahující důvod rebootu

Příklad

Událost vygenerovaná po startu zařízení.



SilentAlarm

Blok **SilentAlarm** definuje událost generovanou při spuštění tichého alarmu. Tichý alarm lze spustit zadáním kódu o 1 vyšším, než je některý uživatelský kód spínače. Pokud má tedy uživatel 1 kód spínače nastaven na 123, spustí se tichý alarm zadáním kódu 124.

Vstupní parametry

Tento blok nemá žádné Vstupní parametry.

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.

Příklad

Událost vygenerovaná po zadání kódu 112, přičemž některý z uživatelů má nastaven kód 111.



Time

Blok **Time** definuje událost generovanou každý den v zadaném čase. Pro omezení platnosti pouze na určité dny použijte u spouštěné akce podmínku Condition. ProfileState a do použitého časového profilu nastavte požadované dny.

Vstupní parametry

- **Time** - definuje čas, který spouští událost. Čas se zadává ve formátu hh:mm.

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.

Příklad

Událost generovaná každý den v 17:30.



Timer

Blok **Timer** definuje událost generovanou s definovaným zpožděním po jiné specifikované události s definovaným počtem opakování události. Pomocí této speciální události lze zpozdřit reakci na jinou událost o definovaný časový interval, nebo provést reakci několikrát po sobě.

Vstupní parametry

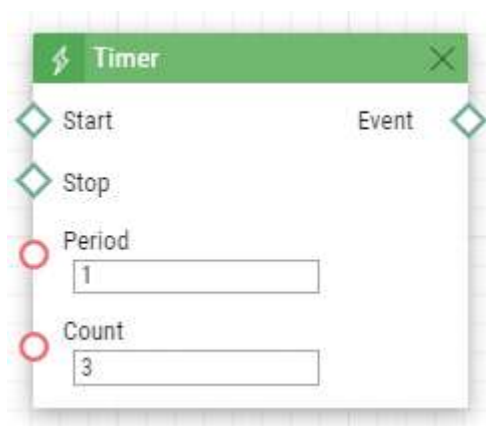
- **Start** - definuje událost, která odstartuje časovač (jedná se o číslo řádku v záložce Automation, na kterém je událost definovaná). Tento parametr je nepovinný. Pokud není uveden, časovač se spustí automaticky.
- **Stop** - definuje událost, která zastaví časovač (jedná se o číslo řádku v záložce Automation, na kterém je událost definovaná). Pokud nastane událost StopEvent, pak se časovač zastaví a znovu spuštěn bude pouze událostí StartEvent. Tento parametr je nepovinný.
- **Period** - definuje periodu časovače.
- Příklad platných hodnot parametrů výše:
 - **10** - 10 sekund (jednotky není nutné uvádět)
 - **10 s** - 10 sekund
 - **100 ms** - 100 milisekund
- Minimální perioda je **100ms**.
- **Count** - definuje počet opakování. Parametr je nepovinný a implicitní hodnota parametru je 0. V tomto případě není počet vygenerovaných událostí časovače omezen. V případě, že nastavíte hodnotu 1, časovač se chová jako zpoždění (Delay).

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.

Příklad

Událost generovaná 3x po 1 s od vzniku události na řádce 1:



UnauthorisedDoorOpen

Blok `UnauthorisedDoorOpen` definuje událost generovanou při neautorizovaném otevření dveří.

Vstupní parametry

- **State** – stav čidla dveří, který vygeneruje událost
 - Platné hodnoty:
 - **Start** – začátek události
 - **End** – konec události

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.

Příklad

Událost generovaná při neautorizovaném otevření dveří.



UserAuthorized

Blok **UserAuthorized** definuje událost generovanou při autorizaci uživatele libovolnou přístupovou metodou (kód, PIN, RFID, Bluetooth, otisk prstu).

Vstupní parametry

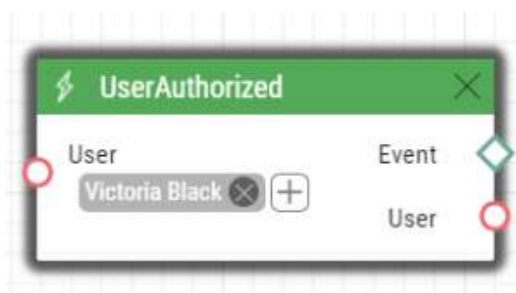
- **User** - určuje uživatele, případně seznam uživatelů. Pokud obsahuje prázdnou hodnotu (výchozí stav), pak na uživateli nezáleží.

Výstupní parametry

- **Event** - výstup pro vyvolání připojené Události nebo Akce.
- **User** - identifikace uživatele, která vyvolala tuto událost.

Příklad

Událost vygenerovaná autorizací uživatele Victoria Black.



Upozornění

- Parametr **User** je omezený celkovým počtem max. 10 uživatelů.

OutputChanged

Blok OutputChanged definuje událost generovanou při změně výstupu.

Vstupní parametry

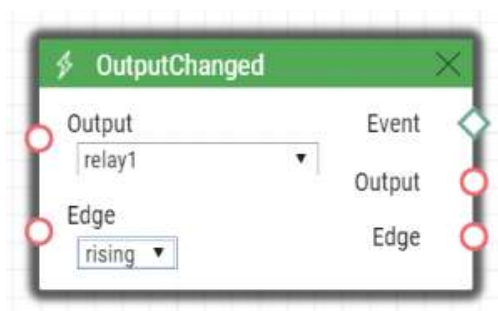
- **Output** – definuje logický vstup.
 - Platné hodnoty:
 - **relay1** – relé 1 na základní jednotce
 - **relay2** – relé 2 na základní jednotce
 - **output1** – výstup 1 na základní jednotce
 - **output2** – výstup 2 na základní jednotce
 - Seznam platných hodnot se může lišit pro různé modely interkomů **2N IP**, viz kap. **Dostupné digitální vstupy a výstupy**.
- **Edge** – definuje detekovanou změnu na výstupu.
 - Platné hodnoty:
 - **falling** – sestupná hrana, změna z log 1 na log 0
 - **rising** – vzestupná hrana, změna z log 0 na log 1
 - Parametr je nepovinný, implicitní hodnota je rising.

Výstupní parametry

- **Event** – výstup pro vyvolání připojené Události nebo Akce.
- **Output** – zaznamenané ID vstupu, jehož změna naposledy vyvolala tuto událost. Možné hodnoty odpovídají hodnotám parametru Input.
- **Edge** – zaznamenaná změna digitálního , která naposledy vyvolala tuto událost. Možné hodnoty jsou falling nebo rising.

Příklad

Událost vygenerovaná změnou výstupu relay 1.



4. Akce (Action)

Automation definuje následující typy akcí:

- **ActivateSwitch** - aktivace spínače
- **SetOutput** - nastavení stavu digitálního výstupu
- **BeginCall** - vytvoření odchozího hovoru
- **AnswerCall** - vyzvednutí příchozího hovoru
- **EndCall** - ukončení hovoru
- **SendHttpRequest** - odeslání HTTP příkazu
- **PlayUser** - přehrání uživatelského zvuku
- **StartMulticastSend** - spuštění vysílání audio streamu
- **StartMulticastRecv** - spuštění příjmu audio streamu
- **StopMulticastSend** - zastavení vysílání audio streamu
- **StopMulticastRecv** - zastavení příjmu audio streamu
- **SendMulticastRequest** - odeslání příkazu více zařízením současně
- **SetCameraInput** - nastavuje použitou kameru
- **ControlRtpStream** - řídí přehrávání RTP streamu
- **LogEvent** - pošle zachycený event na syslog server
- **SendDtmf** - pošle DTMF
- **SendEmail** - odesílá e-mail
- **SetOnvifVirtualInput** - virtuální vstup pro ONVIF
- **SendWiegandCode** - odešle kód na sběrnici Wiegand
- **UploadSnapshotToFtp** - nahraje snímek na FTP server
- **StartAutoUpdate** - stáhne aktuální firmware a konfigurace
- **OpenDoor** - světelná a zvuková signalizace přístupu pomocí čtečky karet

ActivateSwitch

Blok **ActivateSwitch** definuje akci pro sepnutí spínače interkomu nakonfigurovaného v záložkách Spínač 1-4. Činnost, která se provede při aktivaci spínače je zcela závislá na nastavení konkrétního spínače (může dojít k sepnutí digitálního výstup, odeslání HTTP požadavku apod.) Deaktivace (vypnutí) spínače je řízeno nastavením parametrů spínače.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Switch** – definuje aktivovaný spínač (1 až 4).
- **State** – definuje stav bistabilního spínače (parametr se neuplatní pro monostabilní režim spínače).
 - Platné hodnoty:
 - **on** – spínač je aktivován
 - **off** – spínač je deaktivován
 - **toggle** – spínač je přepnut
 - Parametr je nepovinný, implicitní hodnota je **on**.

Příklad

Aktivuje spínač 1 v případě vzniku události definované na řádku 2 a platnosti podmínky definované na řádku 3:



SetOutput

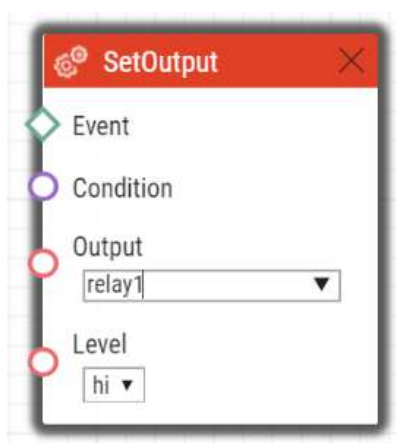
Blok **SetOutput** definuje akci pro nastavení výstupu interkomu do požadované úrovně.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Output** – definuje nastavovaný výstup.
 - Vzorové hodnoty:
 - **relay1** – relé 1 na základní jednotce
 - **relay2** – relé 2 na základní jednotce
 - **output1** – výstup 1 na základní jednotce
 - **output2** – výstup 2 na základní jednotce
 - Seznam platných hodnot se může lišit pro různé modely interkomů **2N IP**, viz kap. **Dostupné digitální vstupy a výstupy**.
- **Level** – požadovaný úroveň výstupu. Tento parametr je nepovinný.
 - Platné hodnoty:
 - **lo** – vypnutí výstupu
 - **hi** – zapnutí výstupu (implicitní hodnota)

Příklad

Aktivuje výstup Output1 v případě vzniku události definované na řádce 2:



BeginCall

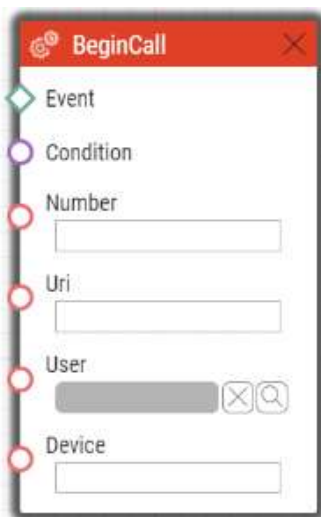
Blok **BeginCall** definuje akci pro vytvoření odchozího hovoru na definované telefonní číslo, SIP URI nebo uživatele v telefonním seznamu interkomu.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Number** – volané telefonní číslo (pokud je **2N IP** interkom registrován k ústředně)
- **Uri** – volané SIP URI ve tvaru **sip:user@domain**
- **User** – definuje uživatele, kterému bude voláno
- **Device** – volaná aplikace **2N[®] IP Mobile** ve tvaru **device:název_zařízení**
- Lze uvést pouze jeden z parametrů **Number**, **Uri**, **User** nebo **Device**.

Příklad

Uskuteční odchozí hovor v případě vzniku události č. 2:



The image shows a configuration window titled "BeginCall" with a red header bar. On the left side, there is a vertical list of parameters: "Event" (with a diamond icon), "Condition" (with a circle icon), "Number" (with a circle icon), "Uri" (with a circle icon), "User" (with a circle icon), and "Device" (with a circle icon). Each parameter has a corresponding input field. The "User" field is currently filled with a greyed-out text and has a search icon and a clear icon to its right. The "Number" and "Device" fields are empty.

AnswerCall

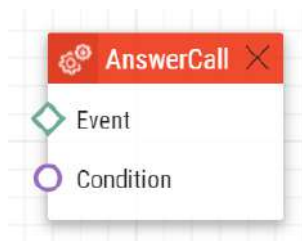
Blok **AnswerCall** definuje akci pro vyzvednutí příchozího hovoru. V případě, kdy na interkom nepřichází žádný hovor nebo příchozí hovor není ve stavu vyzvánění, akce neprovede žádnou činnost.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.

Příklad

Provede vyzvednutí hovoru v případě vzniku události č. 2:



EndCall

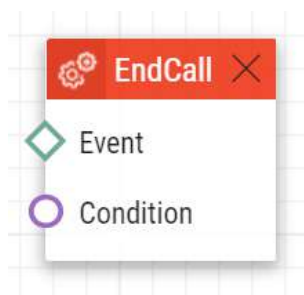
Blok **EndCall** definuje akci pro ukončení probíhajícího hovoru. V případě, kdy na interkom neprobíhá žádný hovor, akce neprovede žádnou činnost.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.

Příklad

Provede ukončení hovoru v případě vzniku události č. 2:



SendHttpRequest

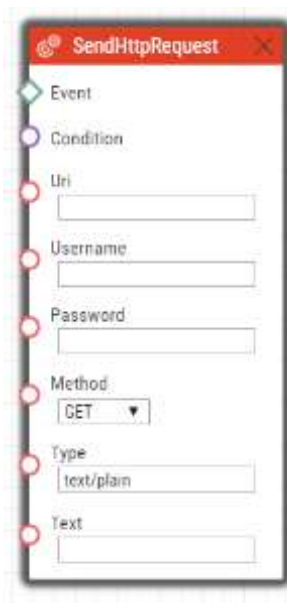
Blok **SendHttpRequest** definuje akci pro odeslání HTTP příkazu jinému zařízení v síti. Pomocí HTTP příkazu lze ovládat jiné zařízení v síti (IP relé, nahrávací systém, jiný interkom apod.)

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Uri** – standardní HTTP URI obsahující adresu cílového zařízení a volitelně cestu a příp. další Parametry. Maximální délka je 2048 bajtů.
- **Username** – definuje uživatelské jméno pro případ, že HTTP server vyžaduje autentizaci. Parametr je nepovinný – implicitní hodnota je "intercom".
- **Password** – definuje heslo v případě, že HTTP server vyžaduje autentizaci. Parametr je nepovinný.
- **Method** – definuje metodu požadavku HTTP, dostupné metody jsou: **GET**, **POST**, **PUT**, **DELETE**.
- **Type** – typ obsahu těla http požadavku. Povolené hodnoty: "application/json" a "text/plain". Týká se pouze platných metod **POST** a **PUT**.
- **Text** – textový obsah požadavku. Týká se pouze platných metod **POST** a **PUT**.

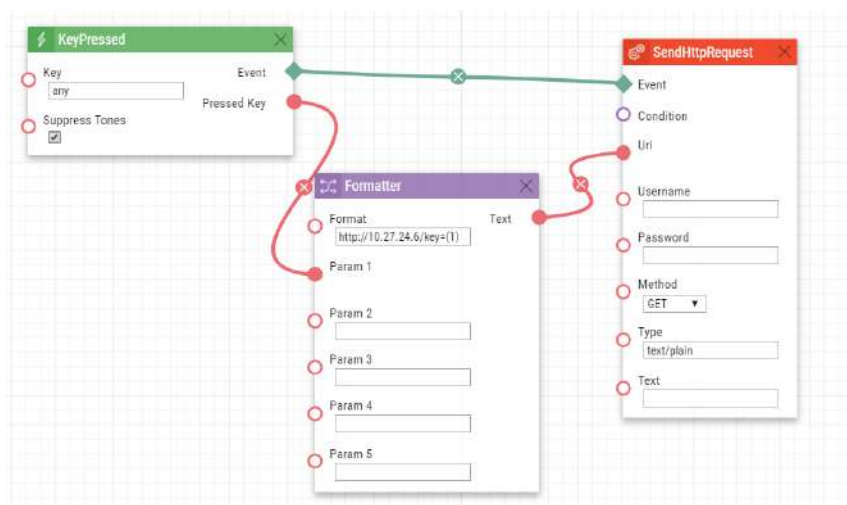
Příklad 1)

V případě vzniku připojené události odešle HTTP příkaz na zařízení s IP adresou 10.27.24.6:



Příklad 2)

V případě stisku jakékoliv klávesy se odešle její přesná identifikace na zařízení s IP adresou 10.27.24.6.



Upozornění

- Je podporována autentizace Basic i Digest, pro vyšší bezpečnost doporučujeme Digest.

Upozornění

- HTTP příkazy používají URL kódování. Při nastavení Automation podle příkladu 2 budou odesílané HTTP příkazy ve formátu:

Stisknuté tlačítko	Format ve Formatter	Odeslaný příkaz
Rychlá volba 1	<code>http://10.27.1.6/message={1}</code>	<code>http://10.27.1.6/message=%251</code> ("%" je zakódováno jako "%25")
Klávesnice 1	<code>http://10.27.1.6/message={1}</code>	<code>http://10.27.1.6/message=1</code>
Rychlá volba 1	<code>http://10.27.1.6/mess?age={1}</code>	<code>http://10.27.1.6/mess?age=%251</code>
Klávesnice 1	<code>http://10.27.1.6/mess?age={1}</code>	<code>http://10.27.1.6/mess?age=1</code>

Upozornění

- Parsování hodnot jednoho parametru oddělených čárkou

Jeden konkrétní parametr může být dále rozdělen čárkou na jednotlivé hodnoty. Jednotlivé hodnoty lze oddělit pomocí ''' (zpětný apostrof). A dále je možné použít \ jako escape znak pro zadání `.

P ř í k l a d y

`abc,def abc and bcd`

``abc,def` abc,bcd`

`abc\`def abc`def`

`abc\def abcdef`

`abc\`def abc\`def` (zpětné lomítko hned 2x za sebou)

- Platnost upozornění se znakem % zůstává.

PlayUserSound

Blok **PlayUserSound** definuje akci pro přehrání definovaného uživatelského zvuku.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Sound** – definuje přehrávaný zvuk
 - Platné hodnoty pro uživatelské zvuky:
 - **1-10** – číslo uživatelského zvuku
 - Platné hodnoty pro předdefinované zvuky (hvězdička odlišuje předdefinovaný zvuk od uživatelského):
 - ***1** – moderní vyzvánění
 - ***2** – velký gong
 - ***3** – štěkání psa
 - ***4** – lodní siréna
 - ***5** – decentní gong
- **Destination** – definuje, kde se přehrává uživatelský zvuk.
 - Platné hodnoty:
 - **speaker** – zvuk se přehrává na interkomu
 - **call** – zvuk se přehraje do hovoru
 - **multicast** – zvuk se přehraje přes multicastovou adresu
 - Parametr je nepovinný, implicitní hodnota je **speaker**.
- **Channel** – definuje číslo ovládaného kanálu (0-3).

Příklad

Přehraje uživatelský zvuk č 1 v případě vzniku události č. 2:

The image shows a configuration dialog box titled "PlayUserSound" with a red header bar containing a gear icon and a close button. The dialog is set against a light blue grid background. It features a vertical list of five options on the left side, each with a colored circular icon: a green diamond for "Event", a purple circle for "Condition", a red circle for "Sound", a red circle for "Destination", and a red circle for "Channel". The "Sound" field is an empty text input box. The "Destination" field is a dropdown menu currently showing "speaker". The "Channel" field is an empty text input box.

StartMulticastSend

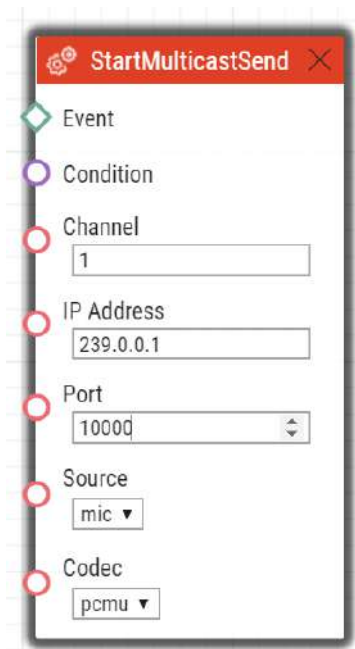
Blok **StartMulticastSend** definuje akci pro spuštění vysílání audio streamu na multicastovou IP adresu. Akcí lze ovládat až čtyři nezávislé vysílací kanály. Pro vysílání audio streamu je použit protokol RTP/UDP.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Channel** – definuje číslo ovládaného kanálu (0-3).
- **IP Address** – definuje multicastovou IP adresu audio streamu.
- **Port** – definuje UDP port, na který bude audio stream odesílán.
- **Source** – definuje zdroj zvuku
 - Platné hodnoty:
 - **mic** – zdrojem zvuku je mikrofon
 - **call** – zdrojem zvuku hovor
 - Parametr je nepovinný, implicitní hodnota je **mic**.
- **Codec** – definuje použitý audio kodek
 - Platné hodnoty
 - **pcmu** – kodek G.711 u-law
 - **pcma** – kodek G.711 A-law
 - **g729** – kodek G.729
 - **g722** – kodek G.722
 - **l16** – kodek L16, 16 kHz
 - Parametr je nepovinný, implicitní hodnota je **pcmu**.

Příklad

Spuštění odesílání audio streamu pomocí kanálu 1 na adresu 239.0.0.1:10000 v případě vzniku události č. 2:



StartMulticastRecv

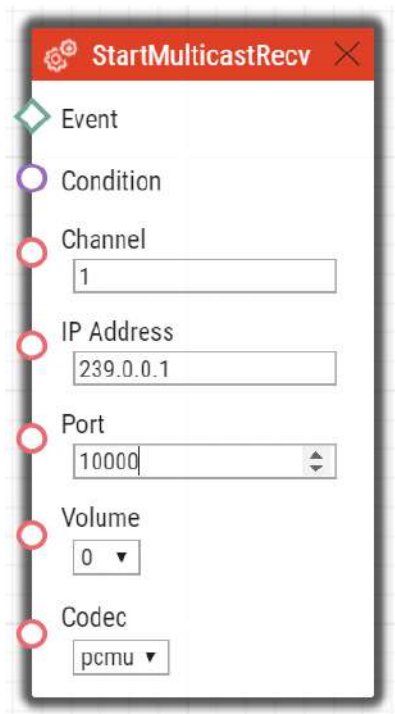
Blok **StartMulticastRecv** definuje akci spuštění příjmu audio streamu a jeho přehrávání. Akcí lze ovládat až čtyři nezávislé přijímací kanály. Pro příjem audio streamu je použit protokol RTP/UDP.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Channel** – definuje číslo ovládaného kanálu (0-3).
- **IP Address** – definuje multicastovou IP adresu audio streamu.
- **Port** – definuje UDP port, na který bude audio stream přijímán.
- **Volume** – definuje relativní hlasitost přehrávaného audio streamu v rozsahu -6 dB až +6 dB.
 - Platné hodnoty:
 - **-6** – minimální hlasitost
 - **0** – střední hlasitost (implicitní hodnota)
 - **6** – maximální hlasitost
 - Parametr je nepovinný, implicitní hodnota je **0**.
- **Codec** – definuje použitý audio kodek
 - Platné hodnoty
 - **pcmu** – kodek G.711 u-law
 - **pcma** – kodek G.711 A-law
 - **g729** – kodek G.729
 - **g722** – kodek G.722
 - **l16** – kodek L16, 16 kHz
 - Parametr je nepovinný, implicitní hodnota je **pcmu**.

Příklad

Spustí příjem audio streamu na multicastové IP adrese 239.0.0.1:10000 v kanálu 1 v případě vzniku události č. 2:



StopMulticastSend

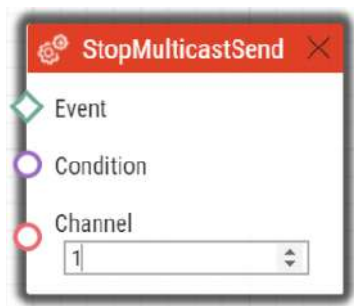
Blok **StopMulticastSend** definuje akci pro zastavení vysílání audio streamu na multicastovou IP adresu.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Channel** – definuje číslo ovládaného kanálu (0-3).

Příklad

Zastaví odesílání audio streamu v kanálu 1 v případě vzniku události č. 2:



StopMulticastRecv

Blok StopMulticastRecv definuje akci pro zastavení příjmu audio streamu na multicastovou IP adresu.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Channel** – definuje číslo ovládaného kanálu (0-3).

Příklad

Zastaví příjem audio streamu v kanálu 1 v případě vzniku události č. 2:



SendMulticastRequest

Blok **SendMulticastRequest** definuje akci pro odeslání uživatelského příkazu více zařízením současně. Příjem takto odeslaného příkazu lze zpracovat pomocí bloku **MulticastTrigger**. Příkaz tvoří zpráva odeslaná pomocí UDP protokolu na multicastovou adresu (235.255.255.250:4433) a tudíž může být přijata více zařízením současně. Zpráva obsahuje identifikaci příkazu (parametr **Command**) a případné parametry (parametr **Params**). Zpráva může být zabezpečena pomocí hesla (parametr **Password**). Tyto příkazy je doporučeno posílat s maximální intenzitou 1 příkaz za sekundu.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Command** – definuje identifikátor příkazu. Pomocí tohoto parametru lze odlišit různé typy odesílaných příkazů. Blok **MulticastTrigger** reaguje na akci **SendMulticastRequest** pouze tehdy, pokud má uveden stejný identifikátor příkazu. Identifikátor může být libovolný text obsahující znaky A-Z, a-z a 0-9.
- **Parameters** – definuje parametry příkazu, které budou odeslány v UDP zprávě. Lze uvést jeden nebo více parametrů oddělených čárkou. Jednotlivé parametry musí být ve formátu “název_parametru=hodnota_parametru”.
 - Příklad:
 - Parameters=“Address=192.168.1.1”, “Port=10000”
 - Takto odeslané parametry budou dostupné v události **HttpTrigger** reagující na tento příkaz jako výstupní parametry **Address** a **Port**. Lze je následně využít např. v akcích navázaných na **HttpTrigger**.
- **Password** – definuje heslo, pomocí kterého je příkaz zabezpečen proti neautorizovanému přístupu. Parametr je nepovinný. Pokud není heslo uvedeno, příkaz není zabezpečen. Heslo může být libovolný text obsahující znaky A-Z, a-z a 0-9.

Příklad

Odešle příkaz na opendoor na všechna zařízení v síti, která mají odpovídajícím způsobem nastaven blok Event.MulticastTrigger v případě vzniku události č. 2:



Upozornění

- Parsování hodnot jednoho parametru oddělených čárkou

Jeden konkrétní parametr může být dále rozdělen čárkou na jednotlivé hodnoty. Jednotlivé hodnoty lze oddělit pomocí ''' (zpětný apostrof). A dále je možné použít \ jako escape znak pro zadání `.

P ř í k l a d y

abc,def abc and bcd

`abc,def` abc,bcd

abc\`def abc`def

abc\def abcdef

abc\ \def abc\def (zpečtné lomítko hned 2x za sebou)

- Platnost upozornění se znakem % zůstává.

SetCameraInput

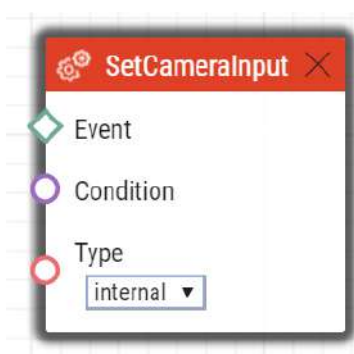
Blok **SetCameraInput** definuje akci umožňující přepínat mezi různými zdroji video signálu pro aktivní hovor. Tato akce umožňuje přepínat mezi zabudovanou kamerou a externí IP kamerou, případně mezi dvěma vstupy pro připojení analogové kamery k modelu 2N[®] IP Video Kit. Touto akcí nelze přepínat zdroj videa pro RTSP stream.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Type** – definuje typ zdroje video signálu. Změna během hovoru se projeví pouze pro tento hovor. Ostatní přijímače videa dostávají stále stejný kanál.
 - Platné hodnoty:
 - **internal** – interní kamera (resp. externí analogová připojená přímo k zařízení)
 - **external** – externí IP kamera
 - Parametr je nepovinný, implicitní hodnota je **internal**.

Příklad

Přepne zdroj video signálu na první vstup pro externí analogovou kameru v případě vzniku události č. 2:



ControlRtpStream

Blok **ControlRtpStream** definuje akci pro nastavení přehrávání RTP streamu. Řízeny jsou pouze hovorové streamy, streamy multicastu ovlivněny nejsou.

Parametry

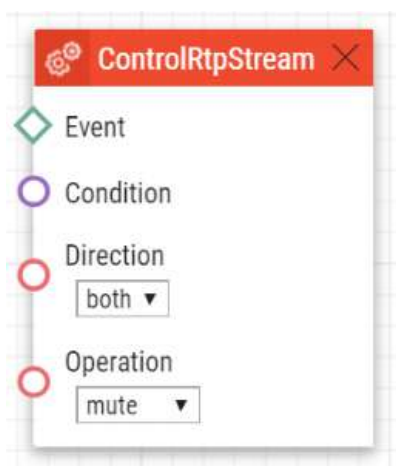
- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Direction** – definuje směr přehrávání RTP hovorového streamu.
 - Platné hodnoty:
 - **in** – příchozí stream na komunikátor
 - **out** – stream posílaný z komunikátoru
 - **both** – příchozí i odchozí stream
 - Parametr je nepovinný, implicitní hodnota je **both**.

Operation – definuje operaci provedenou se streamem

- Platné hodnoty:
 - **mute** – přehrávání streamu je zastaveno
 - **unmute** – přehrávání streamu je obnoveno.

Příklad

Vypne přehrávání streamů v obou směrech v případě vzniku události č. 2:



LogEvent

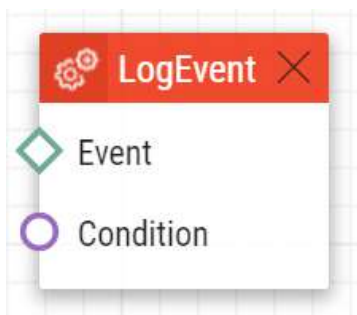
Blok **LogEvent** definuje akci, která pomocí syslogu pošle event, který akci vyvolal. Blok lze použít k ověření nastavení automation.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.

Příklad

Pomocí syslogu pošle zachycený event č. 2 (Event.CardEntered) v případě vzniku události č. 2.:



SendDTMF

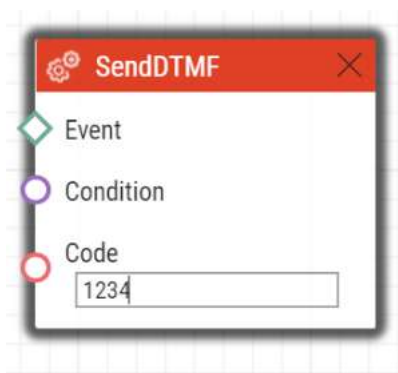
Blok `Action.SendDTMF` definuje akci, která pošle do aktivního hovoru DTMF kód.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Code** – definuje posílané DTMF značky
 - Platné hodnoty 0-9, A-D, F
 - Parametr je povinný

Příklad

Do aktivního hovoru pošle DTMF kód 1234



SendEmail

Blok **SendEmail** definuje akci pro odeslání emailu.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Sender** – definuje adresu odesílatele pro odchozí e-maily ze zařízení.
- **Subject** – definuje předmět odesílané e-mailové zprávy.
 - Do textu lze vkládat speciální zástupné symboly pro datum a čas, identifikaci zařízení. Tyto zástupné symboly budou před odesláním zprávy nahrazeny aktuální hodnotou. Viz popis parametru **Body**.
- **Body** – definuje obsah zprávy, kterou lze upravit obsah odesílané zprávy. V textu lze používat formátovací značky jazyka HTML. Do textu lze vkládat speciální zástupné symboly pro datum a čas, identifikaci zařízení. Tyto zástupné symboly budou před odesláním zprávy nahrazeny aktuální hodnotou. Viz následující tabulka zástupných symbolů:
 1. \$DateTime\$ – aktuální datum a čas
 2. \$DeviceName\$ – identifikace interkomu
- **Snapshots** – definuje počet snímků, které budou k e-mailu přiloženy <0, 5>.
 - Parametr je nepovinný, implicitní hodnota je 1.
- **Timespan** – časový rozsah snímků přiložených k emailu v sekundách. Každou vteřinu se uloží fotografie do paměti 2N IP interkomu. Délka paměti je 30 vteřin zpětně.
 - Parametr je nepovinný, implicitní hodnota je 1.

 **Tip**

Doporučujeme volit hodnoty parametru Timespan a Snapshot tak, aby podíl TimeSpan/Snapshots bylo celé číslo.

Příklad: Timespan = 8

Snapshot = 5

Zobrazí se nejaktuálnější fotografie a každá další s dvouvrteřinovým rozpětím do minulosti.

Pokud je časové rozpětí menší než počet dostupných fotografií, použijí se některé fotografie opakovaně.

- **Width** – definuje parametr nastavující šířku rozlišení přiloženého obrázku z kamery. Šířka snímku musí odpovídat jedné z podporovaných variant rozlišení interkomu.
 - Parametr je nepovinný, implicitní hodnota je **640**.
- **Height** – definuje parametr nastavující výšku rozlišení přiloženého obrázku z kamery. Výška snímku musí odpovídat jedné z podporovaných variant rozlišení interkomu.
 - Parametr je nepovinný, implicitní hodnota je **480**.
- **User** – definuje jednoho nebo více uživatelů, kterým se pošle e-mail. Zadaní musí být alespoň jeden z parametrů User nebo E-mail.
- **E-mail** – definuje emailovou adresu, na kterou se pošle e-mail. V případě potřeby lze zadat více e-mailových adres oddělených čárkou v uvozovkách.
 - Platné hodnoty:
 - `uživatel@doménové_jméno`
 - `uživatel@ip_adresa`
 - `uživatel@doménové_jméno, uživatel@ip_adresa`

 **Tip**

- Parametr **User** má přednost před parametrem **E-mail**.

Příklad

Pošle e-mail na e-mailovou adresu nastavenou u uživatele Jana:



The image shows a configuration window titled "SendEmail" with a red header bar. The window contains a vertical list of settings, each with a colored circular icon to its left:

- Event**: Represented by a green diamond icon.
- Condition**: Represented by a purple circle icon.
- Sender**: Represented by a red circle icon; the text box contains "intercomip@test.com".
- Subject**: Represented by a red circle icon; the text box contains "E-mail from Intercom".
- Body**: Represented by a red circle icon; the text box is empty.
- Snapshots**: Represented by a red circle icon; the text box contains "1".
- Timespan**: Represented by a red circle icon; the text box contains "1".
- Width**: Represented by a red circle icon; the text box contains "640".
- Height**: Represented by a red circle icon; the text box contains "480".
- User**: Represented by a red circle icon; the text box contains "Jana" with a close button (X) and an add button (+).
- Email**: Represented by a red circle icon; the text box is empty.

Upozornění

- Parametr User je omezený celkovým počtem max. 10 uživatelů.

SetOnvifVirtualInput

Blok **SetOnvifVirtualInput** definuje akci pro poslání změny virtuální vstupu pomocí protokolu ONVIF. Pro odzkoušení je možné použít například program ONVIF Device Manager (verze 2.2.250).

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Port** – definuje číslo virtuálního portu,

- Platné hodnoty:
 - **0-10** – číslo portu

Level – definuje vstupní úroveň

- Platné hodnoty:
 - **hi** – nastavuje logickou hodnotu true
 - **lo** – nastavuje logickou hodnotu false.
- Parametr je nepovinný, implicitní hodnota je **hi** (true).

Příklad

Pošle pomocí eventů v ONVIF informaci o tom, že port číslo 8 změnil svou hodnotu na úroveň logická 1 v případě vzniku události č. 2:



Na ONVIF je posláno:

- InputToken: onvif_port_08
- LogicalState:true

SendWiegandCode

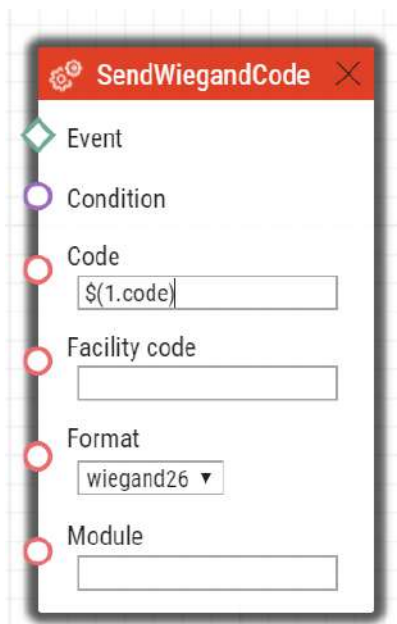
Blok **SendWiegandCode** definuje akci pro posílání zadaného kódu na sběrnici Wiegand.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Code** – definuje kód, který se posílá přes sběrnici Wiegand. Pokud je zadaný kód delší, než odpovídá zvolenému formátu, tak jsou oříznuty nejvyšší bity kódu.
 - Platné hodnoty
 - Dekadické číslo
 - Parametr je povinný
- **Fac ility code** – fa cility kód zařízení. Nastavení se uplatní pouze pro formát „wiegand26”.
 - Platné hodnoty
 - Dekadické číslo v rozmezí **0-255**
 - Parametr je nepovinný, pokud není parametr nastaven, tak není použit.
- **Format** – definuje formát zprávy na sběrnici Wiegand
 - Platné hodnoty
 - **wiegand26** – 26 bitů
 - **wiegand32** – 32 bitů
 - **wiegand37** – 37 bitů
 - Parametr je nepovinný, implicitní hodnota je **wiegand26**
- **Module** – definuje modul, přes který je kód odeslán.
 - Platné hodnoty
 - Jméno modulu nastavené v menu Hardware / Rozšiřující moduly / Modul Wiegand, parametr Jméno modulu
 - Parametr je povinný pro model **2N[®] Verso** a **2N Access Unit**, ostatních modelů se netýká

Příklad

Pošle na sběrnici Wiegand kód zadaný v události 1. pomocí Event.CodeEntered.



SendWiegandCode

- Event
- Condition
- Code:
- Facility code:
- Format:
- Module:

Pro 2N[®] Verso a 2N Access Unit pak druhý řádek vypadá následovně:



SendWiegandCode

- Event
- Condition
- Code:
- Facility code:
- Format:
- Module:

UploadSnapshotToFtp

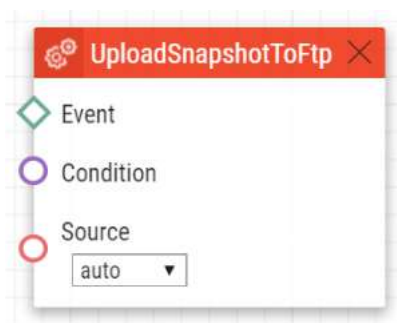
Blok **UploadSnapshotToFtp** definuje akci pro nahrání snímku z kamery na FTP server. Parametry FTP serveru a snímku jsou nastaveny v menu Služby / Streamování / FTP.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Source** – definuje zdroj videa, ze kterého se má nahrát snímek na FTP server.
 - Platné hodnoty:
 - **auto** – zdroj videa je vybrán podle nastavení **Hardware / Kamera / Společné nastavení / Výchozí zdroj videa**
 - **internal** – interní kamera
 - **external** – externí kamery.
 - Parametr je nepovinný, implicitní hodnota je **auto**.

Příklad

Uloží fotku z kamery na FTP server v případě vzniku události č. 2:.

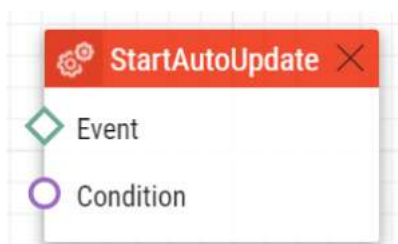


StartAutoUpdate

Blok **StartAutoUpdate** definuje akci pro spuštění automatické aktualizace firmware a konfigurace. Parametry aktualizace jsou nastaveny v menu Systém / Aktualizace.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.



OpenDoor

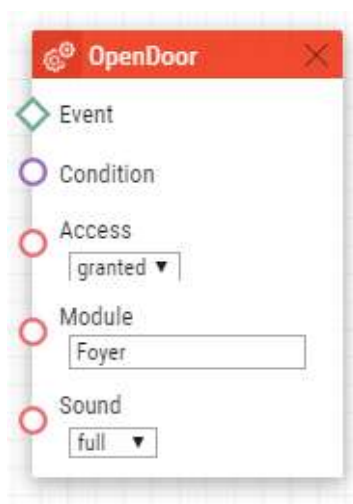
Blok **OpenDoor** definuje akci pro světelnou a zvukovou signalizaci při platném nebo neplatném přístupu pomocí modulu čtečky karet **2N[®] IP Verso**.

Parametry

- **Event** – definuje událost, která spustí tuto akci.
- **Condition** – definuje podmínku, která musí být splněna, aby akce byla provedena. Tento parametr je nepovinný.
- **Access** – signalizuje platnost přístupu
 - granted – platný
 - denied – neplatný
- **Module** – definuje název modulu čtečky karet
- **Sound** – definuje zvukovou signalizaci
 - Platné hodnoty:
 - **full** – zvuková signalizace pro platný i neplatný přístup
 - **none** – bez zvukové signalizace

Příklad

Platný přístup u čtečky ve foyer bude signalizován světelným a zvukovým signálem.



Poznámka

Ostatní příklady nastavení

- Access granted / Sound - none = svítí zeleně, přehraje zvuk dle nastavení spínače v HW
- Access denied / Sound - none = svítí červeně, nepřehraje žádný zvuk

Upozornění

- Moduly čteček karet ve spojení s blokem DoorOpen musí být vyloučeny z kontroly přístupu nastavením v Hardware / Rozšiřující moduly. U daného modelu nastavením parametru Dveře / Nepoužito.
- Moduly, které nemají možnost konfigurace jména, je možné adresovat pomocí ext <pozice_modulu> , např. "ext3".

5. Podmínky (Condition)

Automation definuje následující typy podmínek:

- **True** - vždy pravdivá podmínka
- **False** - vždy nepravdivá podmínka
- **ProfileState** - stav časového profilu
- **CallState** - stav probíhajícího hovoru
- **AccountState** - stav registrace k SIP účtu
- **InputState** - stav digitálního vstupu
- **FlipFlopRS** - klopný obvod typu RS
- **FlipFlopD** - klopný obvod typu D
- **LogicalAnd** - logický součin podmínek
- **LogicalOr** - logický součet podmínek
- **LogicalNot** - negace podmínky
- **LockdownState** - nouzové uzamčení
- **OnvifVirtualOutputState** - stav virtuálního portu nastaveného z VMS
- **OutputState** - stav výstupu

Detailní popis podmínek, jejich parametry a použití jsou popsány v následujícím textu.

True

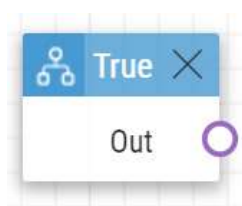
Blok True definuje podmínku, která je vždy splněna.

Výstupní parametry

- Out

Příklad

Podmínka je vždy splněna:



False

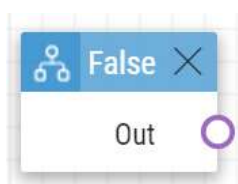
Blok False definuje podmínku, která není nikdy splněna.

Výstupní parametry

- Out

Příklad

Podmínka je vždy nesplněna:



ProfileState

Blok **ProfileState** definuje podmínku, která je splněna v případě aktivního (příp. neaktivního) časového profilu.

Vstupní parametry

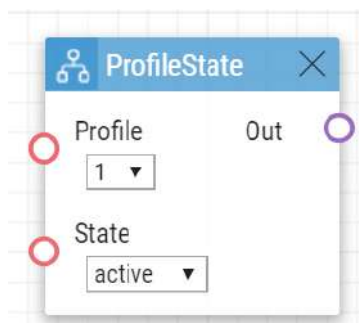
- **Profile** – číslo časového profilu (1-20 podle modelu interkomu).
- **State** – požadovaný stav profilu. Tento parametr je nepovinný.
 - Platné hodnoty:
 - **active** – profil je aktivní (implicitní hodnota)
 - **inactive** – profil je neaktivní

Výstupní parametry

- **Out**

Příklad

Podmínka splněna v případě neaktivního časového profilu č. 1:



CallState

Blok CallState je splněn v případě definovaného stavu probíhajícího hovoru.

Vstupní parametry

- **State** – definuje stav hovoru.
 - Platné hodnoty:
 - **idle** – hovor neprobíhá
 - **connecting** – hovor se spojuje (pouze odchozí hovory)
 - **ringing** – probíhá vyzvánění
 - **connected** – hovor je spojen
- **Direction** – definuje směr hovoru.
 - Platné hodnoty:
 - **incoming** – příchozí hovory
 - **outgoing** – odchozí hovory
 - **any** – oba směry
 - Parametr je nepovinný, implicitní hodnota je **any**.

Výstupní parametry

- **Out**

Příklad

Podmínka splněna v případě neaktivního hovoru:



AccountState

Blok AccountState je splněn v případě úspěšné registrace k SIP účtu.

Vstupní parametry

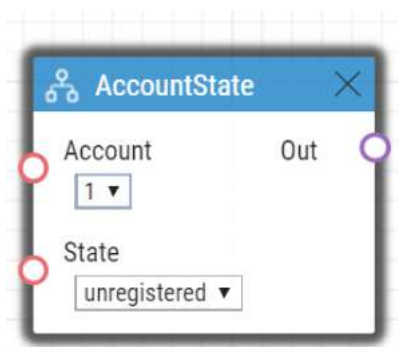
- **Account** – definuje použitý SIP účet.
 - Platné hodnoty:
 - 1 – účet 1
 - 2 – účet 2
 - Parametr je nepovinný, implicitní hodnota je 1.
- **State** – Definuje stav registrace.
 - Platné hodnoty:
 - **registered** – účet je zaregistrován
 - **unregistered** – účet není zaregistrován
 - Parametr je nepovinný, implicitní hodnota je **registered**.

Výstupní parametry

- Out

Příklad

Podmínka splněna v případě nezaregistrovaného 1. účtu:



InputState

Blok **InputState** definuje podmínku, která je splněna v případě připojení definované logické úrovně na definovaný digitální vstup.

Vstupní parametry

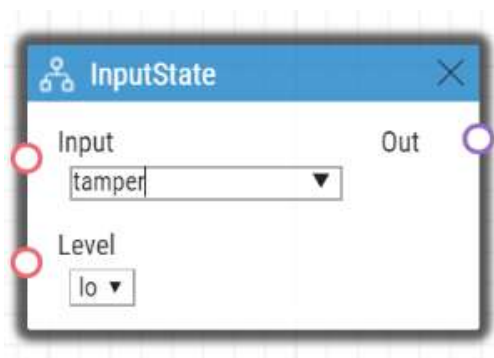
- **Input** – definuje digitální vstup.
 - Platné hodnoty:
 - **tamper** – tamper spínač
 - **input1** – digitální vstup 1
 - **input2** – digitální vstup 2
 - **cr_input1** – digitální vstup 1 na čtečce karet
 - **cr_input2** – digitální vstup 2 na čtečce karet
 - Seznam platných hodnot se může lišit pro různé modely interkomů **2N IP**, viz kap. Dostupné digitální vstupy a výstupy.
- **Level** – Definuje požadovanou úroveň na digitálním vstupu. Parametr je nepovinný.
 - Platné hodnoty:
 - **lo** – logická 0
 - **hi** – logická 1 (implicitní hodnota)

Výstupní parametry

- **Out**

Příklad

Podmínka splněna v případě sepnutého spínače tamperu (zařízení není otevřeno):



FlipFlopRS

Blok **FlipFlopRS** je jednobitová paměťová buňka (výstupní parametr), která mění svůj stav na 1 příp. 0 při vzniku definovaných událostí. Výstup bloku FlipFlopRS lze použít jako podmínku řídicí další akce ve složitějších případech použití **Automation**. Jedná se o simulaci klopného obvodu typu R-S.

Vstupní parametry

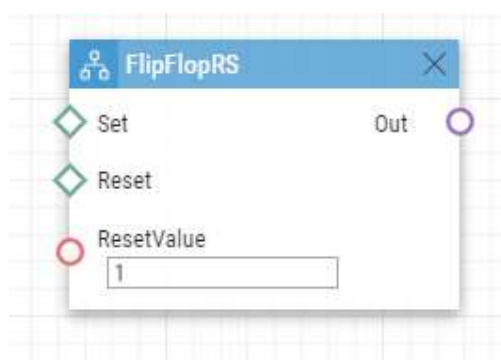
- **Set** – definuje událost, která nastaví podmínku do stavu splněno (1).
- **Reset** – definuje událost, která nastaví podmínku do stavu nesplněno (0).
- **ResetValue** – výchozí hodnota podmínky po restartu zařízení. Parametr není povinný.
- Platné hodnoty parametrů:
 - 0 – podmínka je nesplněná (implicitní hodnota)
 - 1 – podmínka je splněná

Výstupní parametry

- Out

Příklad

Podmínka bude splněna při vzniku události 1 a nesplněna při vzniku události 2:



FlipFlopD

Blok **FlipFlopD** je jednobitová paměťová buňka (výstupní parametr), která si zaznamená stav jiné podmínky v okamžiku vzniku definované události pro pozdější použití. Výstup bloku FlipFlopD lze použít jako podmínku řídicí další akce ve složitějších případech použití **Automation**. Jedná se o simulaci klopného obvodu typu D.

Vstupní parametry

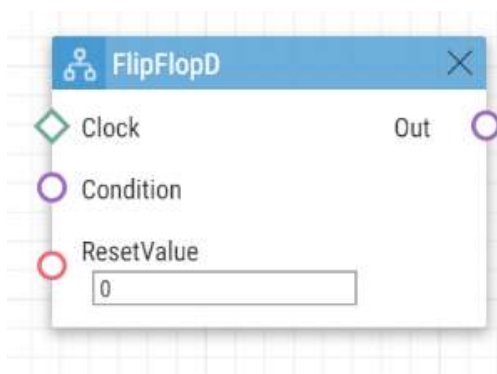
- **Clock** - definuje událost, při které je zaznamenán aktuální stav podmínky Condition.
- **Condition** - definuje podmínku, která je zaznamenána při vzniku události ClockEvent.
- **ResetValue** - výchozí hodnota podmínky po restartu zařízení. Parametr není povinný.
- Platné hodnoty parametrů:
 - 0 - podmínka je nesplněná (implicitní hodnota)
 - 1 - podmínka je splněná

Výstupní parametry

- Out

Příklad

Stav podmínky bude shodný se stavem podmínky 2 v okamžiku vzniku události 1:



LogicalAnd

Blok **LogicalAnd** umožňuje vytvářet složitější zřetězení různých podmínek. Blok je splněn v případě splnění všech podmínek z definované skupiny.

Vstupní parametry

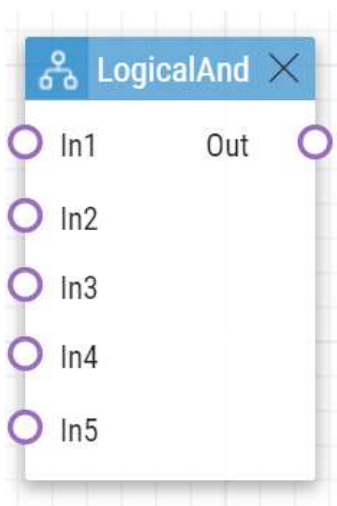
- In1 – Definuje podmínku, která má být splněna.
- In2 – Definuje podmínku, která má být splněna.
- In3 – Definuje podmínku, která má být splněna.
- In4 – Definuje podmínku, která má být splněna.
- In5 – Definuje podmínku, která má být splněna.

Výstupní parametry

- Out

Příklad

Podmínka splněna v případě současného splnění podmínek 1, 2 a 3:



LogicalOr

Blok **LogicalOr** umožňuje vytvářet složitější zřetězení různých podmínek. Tento blok je splněn v případě splnění alespoň jedné podmínky z definované skupiny.

Vstupní parametry

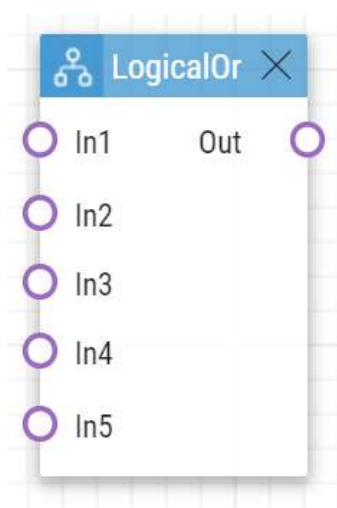
- In1 – Definuje podmínku, která má být splněna.
- In2 – Definuje podmínku, která má být splněna.
- In3 – Definuje podmínku, která má být splněna.
- In4 – Definuje podmínku, která má být splněna.
- In5 – Definuje podmínku, která má být splněna.

Výstupní parametry

- Out

Příklad

Podmínka splněna v případě současného splnění alespoň jedné z podmínek 1, 2 nebo 3:



LogicalNot

Blok **LogicalNot** definuje podmínku, která je splněna v případě nesplnění jiné definované podmínky.

Vstupní parametry

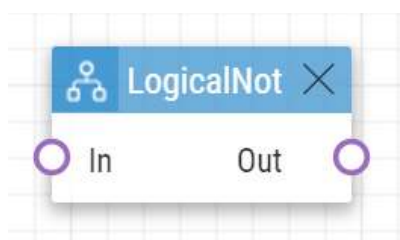
- In - Definuje podmínku, která má být splněna.

Výstupní parametry

- Out

Příklad

Podmínka splněna v případě nesplnění podmínky 1:



LockdownState

Blok LockDownState definuje podmínku, která je splněna při aktivním nouzovém uzamčení.

Vstupní parametry

- Door - definuje podmínku, která má být splněna. Platné hodnoty jsou entry a exit.

Výstupní parametry

- State



OnvifVirtualOutputState

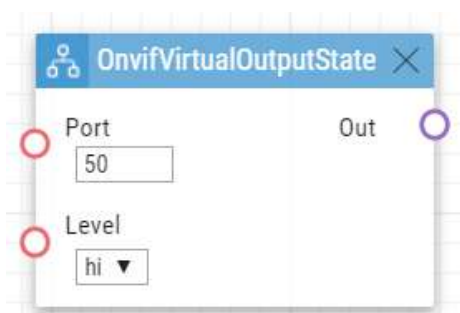
Blok `OnvifVirtualOutputState` definuje podmínku, která je splněna při nastaveném stavu virtuálního portu.

Vstupní parametry

- **Port** – definuje sledovaný port. Platné hodnoty jsou 50–54.
- **Level** – definuje logickou hodnotu sledovaného portu. Platné hodnoty jsou high a low.

Výstupní parametry

- **Out**



OutputState

OutputState definuje podmínku, která odpovídá logickému stavu daného výstupu.

Vstupní parametry

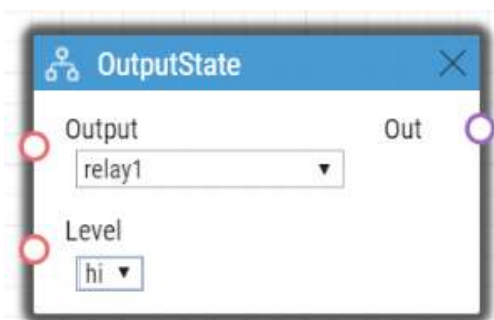
- **Output** – definuje digitální vstup.
 - Platné hodnoty:
 - **relay1** – relé 1 na základní jednotce
 - **relay2** – relé 2 na základní jednotce
 - **output1** – výstup 1 na základní jednotce
 - **output2** – výstup 2 na základní jednotce
 - Seznam platných hodnot se může lišit pro různé modely interkomů **2N IP**, viz kap. Dostupné digitální vstupy a výstupy.
- **Level** – Definuje požadovanou úroveň na digitálním vstupu. Parametr je nepovinný.
 - Platné hodnoty:
 - **lo** – logická 0
 - **hi** – logická 1 (implicitní hodnota)

Výstupní parametry

- **Out**

Příklad

podmínka splněna v případě aktivního relay 1



6. Utilities

Automation definuje následující typy událostí:

Formatter

Blok **Formatter** umožňuje použít výstupní parametry z bloku Events v blocích Actions, dále umožňuje formátovat vstup do parametrů v blocích Actions.

Parametry

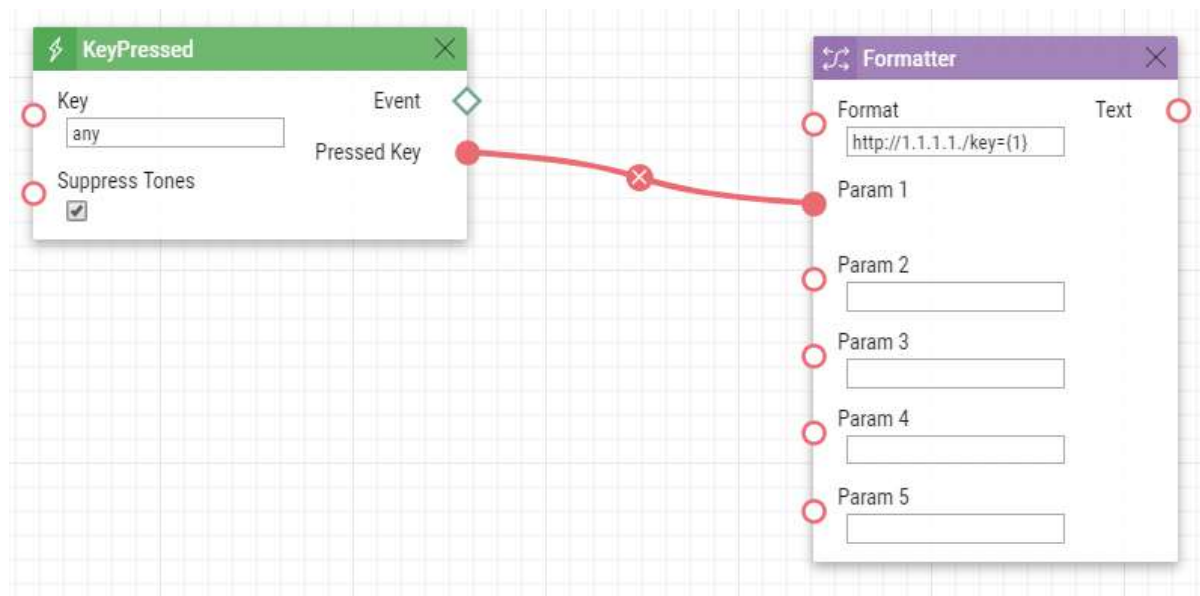
- **Format** – definuje text, který je následně zaslán na výstup do výstupního parametru Text. Pro vložení hodnoty výstupního parametru Param 1 se použijí složené závorky {}.
 - Příklad použití
 - **http://1.1.1.1/key={1}** (uloží text s hodnotou výstupního parametru z Param 1)
- **Param 1** – umožňuje nastavit výstupní parametr pro použití v parametru Format. Obvykle je připojeno k výstupnímu parametru v blocích Events.
- **Param 2** – umožňuje nastavit výstupní parametr pro použití v parametru Format. Obvykle je připojeno k výstupnímu parametru v blocích Events.
- **Param 3** – umožňuje nastavit výstupní parametr pro použití v parametru Format. Obvykle je připojeno k výstupnímu parametru v blocích Events.
- **Param 4** – umožňuje nastavit výstupní parametr pro použití v parametru Format. Obvykle je připojeno k výstupnímu parametru v blocích Events.
- **Param 5** – umožňuje nastavit výstupní parametr pro použití v parametru Format. Obvykle je připojeno k výstupnímu parametru v blocích Events.

Výstupní parametry

- **Text** – výsledný text v poli Format s uplatněním případných nastavených výstupních parametrů. V parametru Param 1.

Příklad

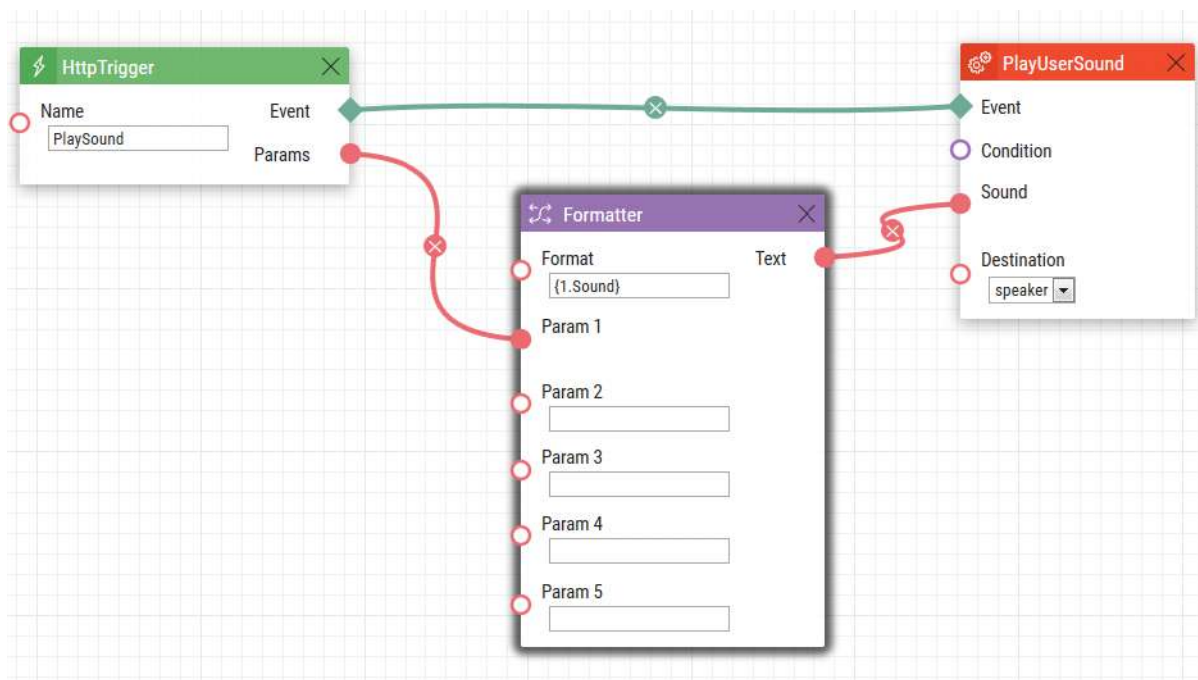
Připraví HTTP příkaz obsahující hodnotu stisklé klávesy v bloku KeyPressed.



Příklad

Použije hodnotu parametru přijatého pomocí HTTP příkazu jako proměnou do akce.

Po odeslání příkazu na zařízení ve tvaru `http://<device_address>/enu/trigger/id?param1=value1¶m2=value2` jsou hodnoty proměnných dostupné pomocí příkazu `{param_input.param_name}`.



Odeslaný HTTP příkaz: `http://10.27.24.15/enu/trigger/PlaySound?Sound=3`

Format: `{1.Sound}`

Přehraný uživatelský zvuk: 3

7. Dostupné digitální vstupy a výstupy

V této kapitole jsou popsány dostupné digitální vstupy a výstupy na jednotlivých modelech interkomů 2N IP.

- 2N[®] IP Vario
- 2N[®] IP Force/Safety
- 2N[®] IP Audio/Video Kit
- 2N[®] IP Verso/LTE Verso
- 2N[®] IP Solo
- 2N[®] SIP Audio Converter
- 2N Access Unit
- 2N[®] IP Base

2N[®] IP Vario

Výstupy

- **relay1** – reléový výstup na základní jednotce
- **relay2** – reléový výstup na přídatném spínači (pokud je instalován)
- **cr_relay1** – reléový výstup 1 na čtečce karet (pokud je instalována)
- **cr_relay2** – reléový výstup 2 na čtečce karet (pokud je instalována)
- **led_secured** – červená signalizační led pod jmenovkami (pouze modely 9137xxxU bez displeje)

Vstupy

- **cr_input1** – digitální vstup 1 na čtečce karet (pokud je instalována)
- **cr_input2** – digitální vstup 2 na čtečce karet (pokud je instalována)

2N[®] IP Force/Safety

Výstupy

- **relay1** – reléový výstup na základní jednotce
- **output1** – aktivní digitální výstup na základní jednotce (pouze verze desek 555v3 a vyšší, desky 555v2 mají aktivní digitální výstup spojený s výstupem relay1)
- **relay2** – reléový výstup na přídatném spínači (pokud je instalován)
- **output2** – aktivní digitální výstup na přídatném spínači (pokud je instalován)
- **cr_relay1** – reléový výstup na čtečce karet (pokud je instalována)
- **cr_output1** – aktivní digitální výstup na čtečce karet (pokud je instalována)
- **led_secured** – červená signalizační led na čtečce karet (pokud je čtečka instalována)
- **led_ringing** – oranžová led signalizující vyzvánění (pouze modely s piktogramy)
- **led_connected** – modrá led signalizující spojený hovor (pouze modely s piktogramy)
- **led_door** – zelená led signalizující otevření dveří (pouze modely s piktogramy)
- **led_key1** – podsvětlení prvního tlačítka na Safety
- **led_key2** – podsvětlení druhého tlačítka na Safety
- **led_key3** – podsvětlení třetího tlačítka na Safety

Vstupy

- **tamper** – tamper spínač (pokud je instalován)
- **cr_input1** – digitální vstup 1 na čtečce karet (pokud je instalována)
- **cr_input2** – digitální vstup 2 na čtečce karet (pokud je instalována)
- **input2** – digitální vstup 2 na přídatném spínači (pokud je nainstalován)

2N[®] IP Audio/Video Kit

Výstupy

- relay1 – reléový výstup
- output1 – digitální výstup 1
- output2 – digitální výstup 2
- led1 – výstup pro řízení LED 1
- led2 – výstup pro řízení LED 2
- led3 – výstup pro řízení LED 3

Vstupy

- input1 – digitální vstup 1
- input2 – digitální vstup 2

2N[®] IP Verso/LTE Verso

Základní jednotka

Výstupy

- output1 – digitální výstup 12 V
- relay1 – reléový výstup
- led_secured – červená signalizační led na panelu

Vstupy

- input1 – digitální vstup na základní jednotce

Modul I/O

Vstupy / výstupy jsou adresovány <jméno_modulu>.<jméno_vstupu/výstupu>, např. modul5.relay1.

Jméno modulu se nastavuje v menu Hardware / Rozšiřující moduly, parametr jméno modulu.

Výstupy

- relay1 – reléový výstup 1
- relay2 – reléový výstup 2

Vstupy

- input1 – digitální vstup 1
- input2 – digitální vstup 2
- tamper – tamper spínač (pokud je instalován)

Modul Wiegand

Vstup je adresován <jméno_modulu>.<jméno_vstupu>, např. modul2.tamper

Jméno modulu se nastavuje v menu Hardware / Rozšiřující moduly, parametr jméno modulu.

Výstupy

- output1 – výstup LED OUT

Vstupy

- **input1** - vstup LED IN
- **tamper** - tamper spínač (pokud je instalován)

2N[®] IP Solo

Výstupy

- **output1** – digitální výstup 12 V
- **relay1** – reléový výstup
- **led_secured** – červená signalizační led na panelu

Vstupy

- **input1** – digitální vstup na základní jednotce
- **tamper** – tamper spínač

2N[®] SIP Audio Converter

Výstupy

- relay1 – reléový výstup

Vstupy

- Nejsou k dispozici.
- Lze použít Event.KeyPressed: Key=%1 pro události vygenerované na vstupu LOGIC IN

2N Access Unit

Základní jednotka

Výstupy

- output1 – digitální výstup 12 V
- relay1 – reléový výstup
- led_secured – červená signalizační led na panelu

Vstupy

- input1 – digitální vstup č. 1 na základní jednotce
- input2 – digitální vstup č. 2 na základní jednotce
- input3 – digitální vstup č. 3 na základní jednotce
- tamper – tamper spínač

Modul I/O

Vstupy / výstupy jsou adresovány <jméno_modulu>.<jméno_vstupu/výstupu>, např. modul5.relay1.

Jméno modulu se nastavuje v menu Hardware / Rozšiřující moduly, parametr jméno modulu.

Výstupy

- relay1 – reléový výstup 1
- relay2 – reléový výstup 2

Vstupy

- input1 – digitální vstup 1
- input2 – digitální vstup 2
- tamper – tamper spínač (pokud je instalován)

Modul Wiegand

Vstup je adresován <jméno_modulu>.<jméno_vstupu>, např. modul2.tamper

Jméno modulu se nastavuje v menu Hardware / Rozšiřující moduly, parametr jméno modulu.

Výstupy

- output1 – výstup LED OUT

Vstupy

- input1 – vstup LED IN
- tamper – tamper spínač (pokud je instalován)

2N[®] IP Base

Výstupy

- **output1** – digitální výstup 12 V
- **relay1** – reléový výstup
- **led_secured** – červená signalizační led na panelu

Vstupy

- **input1** – digitální vstup na základní jednotce
- **tamper** – tamper spína

8. Příklady použití

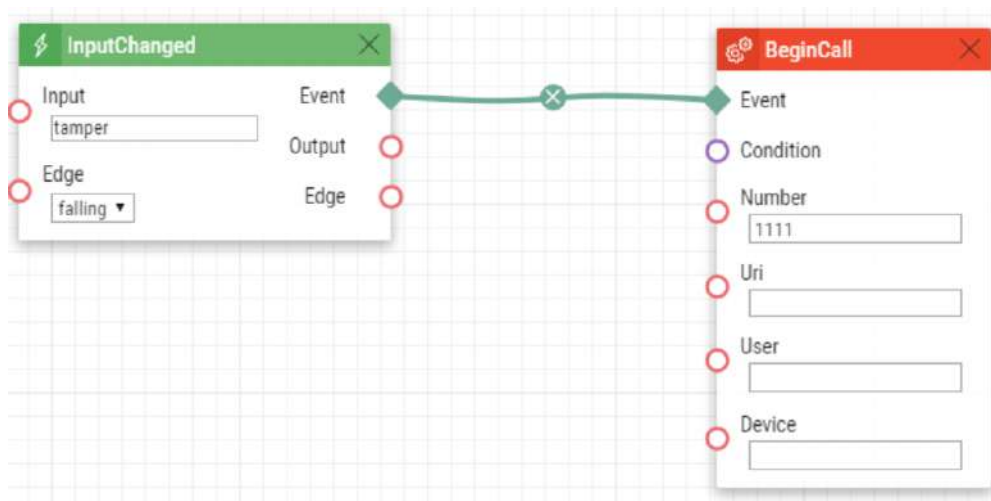
Volání na dispečink při neoprávněném otevření zařízení

Zadání

Po rozepnutí tamper spínače (otevření zařízení) volat na zvolené telefonní číslo.

Blokové schéma

V okamžiku náběžné hrany na vstupu tamper (1: Event.InputChanged) se vyvolá akce volání na zvolené telefonní číslo (2: Action.BeginCall).



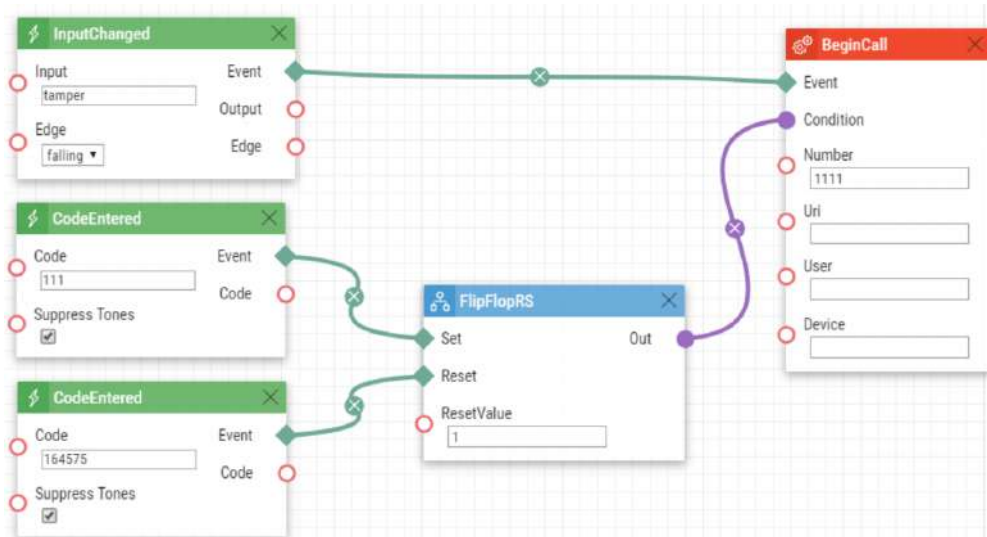
Volání na dispečink při neoprávněném otevření zařízení s možností blokování servisním kódem

Zadání

Po rozeptnutí tamper spínače (otevření zařízení) volat na telefonní zvolené telefonní číslo. Umožnit blokování a opětovné povolení poplachu numerickým kódem zadaným na klávesnici interkomu.

Blokové schéma

V okamžiku náběžné hrany na vstupu tamper (1: Event.InputChanged) se volá na zvolené telefonní číslo (5: Action.BeginCall) v případě splnění podmínky. Podmínka (4: Condition.FlipFlopRS) je platná po restartu interkomu nebo po zadání zvoleného kódu (2: Condition.CodeEntered) na numerické klávesnici. Podmínka bude neplatná po zadání jiného zvoleného kódu (3: Condition.CodeEntered).



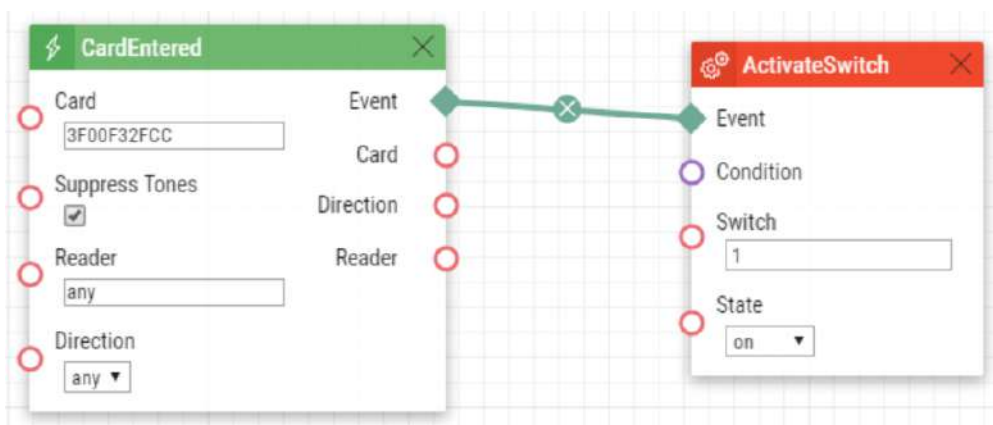
Otevření dveří po protažení RFID karty

Zadání

Po přiložení konkrétní RFID karty aktivovat spínač dveřního kontaktu.

Blokové schéma

V okamžiku přiložení RFID karty se zadaným ID (1: Event.CardEntered) dojde k aktivaci spínače 1 (2: Action.ActivateSwitch).



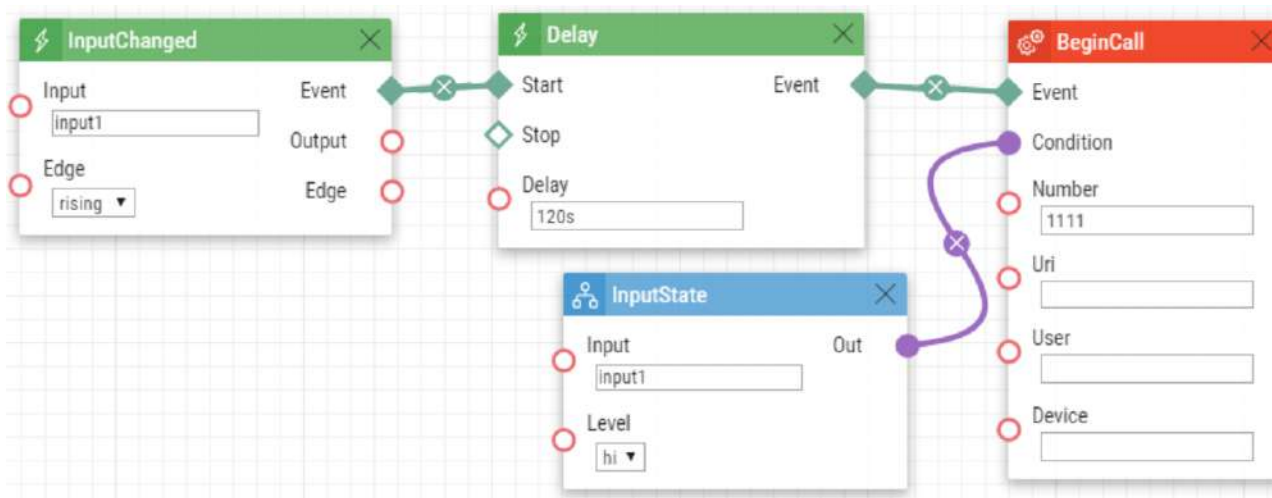
Poplach (volání na dispečink) při otevření dveří delším než 2 minuty

Zadání

V případě, že jsou dveře otevřeny po dobu delší než 2 minuty volat na dispečink. Příklad předpokládá, že na vstup Input1 je připojen kontakt, který signalizuje otevření dveří.

Blokové schéma

Po otevření dveří - náběžné hrana na signálu Input1 (1: Event.InputChanged) se se zpožděním 120 s (2: Event.Delay) se zavolá na zvolené telefonní číslo (4: Action.BeginCall). Volání se provede jen tehdy, pokud po uplynutí 120 s jsou dveře stále otevřeny (3: Condition.InputState).



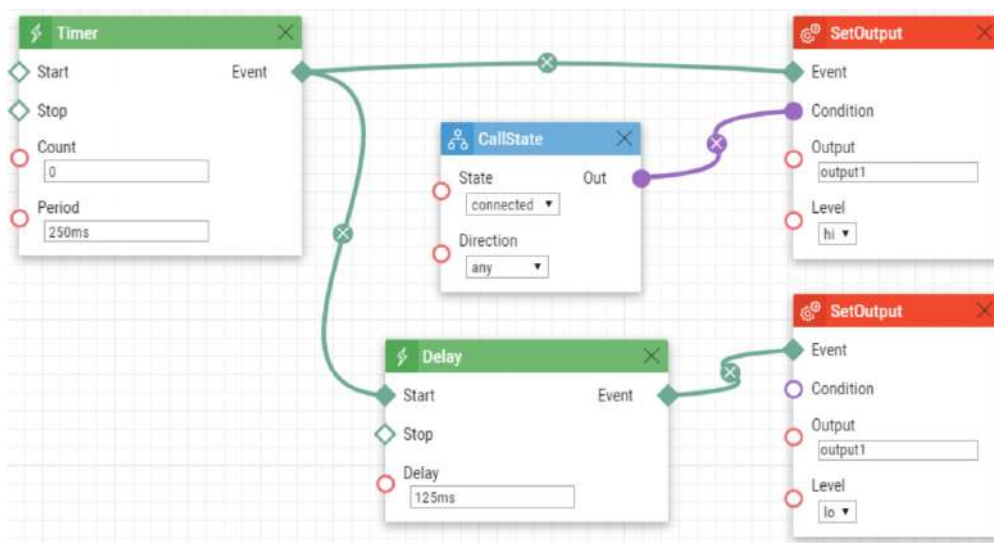
Blikání s LED v průběhu hovoru / Blikání v průběhu odemčení elektrického zámku dveří

Zadání

V průběhu aktivního hovoru blikat s LED.

Blokové schéma

Funkce blikání je řešena pomocí periodického časovače (1: Event.Timer) a zpoždění (2: Event.Delay). Tyto dva bloky definují periodu (250 ms) a střídu signálu - resp. dobu rozsvícení LED (125 ms). Na tyto dvě události jsou navázány akce pro rozsvícení (4: Action.SetOutput) a zhasnutí led (5: Action.SetOutput). Akce pro rozsvícení led je podmíněna probíhajícím hovorem (3: Condition.CallState).





An Axis company

2N TELEKOMUNIKACE a.s.

Modřanská 621, 143 01 Prague 4, Czech Republic

Phone: +420 261 301 500, Fax: +420 261 301 599

E-mail: sales@2n.cz

Web: www.2n.cz

v2.29