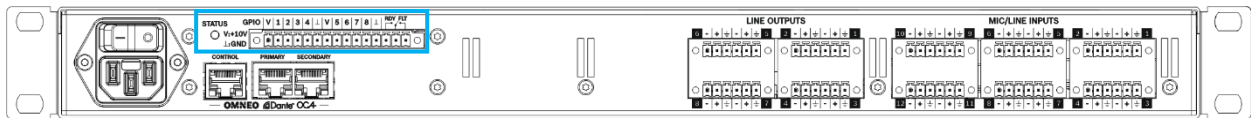


## Application Note

# Analoge Steuerung von IX-Verstärkern und MXE Matrix Mix Engines über den IX/MXE Control-Port (GPIOs)

**IX-Verstärker** und **MXE Matrix Mix Engines** sind mit einem **Control-Port** für Schnittstellen zu anderen Systemen, unter Verwendung analoger Steuerleitungen, ausgestattet.



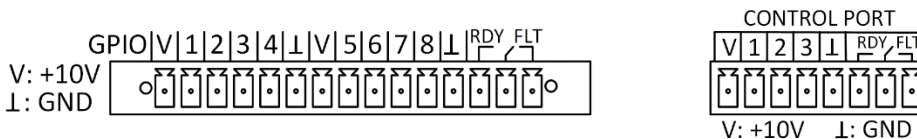
**Bild 1:** MXE-Rückansicht



**Bild 2:** IX-Rückansicht (8-Kanal-Modell abgebildet)

Der **Control-Port** (*GPIO* oder *CONTROL PORT*) befindet sich auf der **IX-** und **MXE-Rückseite**. Er bietet insgesamt **acht (MXE)** oder **drei (IX)** frei konfigurierbare **GPIOs** (General Purpose Inputs und Outputs), ein **Ready/Fault-Relais (RDY/FLT)** und +10 V (V) und Ground (⊥) Referenz-Pins.

Die **GPIOs** können in **SONICUE** entweder als **Analog In**, **Digital In**, oder **Digital Out** konfiguriert werden.



**Bild 3 und 4:** MXE (links) und IX (rechts) Control-Port-Detailansicht

## Anforderungen zur Verwendung der MXE Task Engine

**MXE Matrix Mix Engine** mit Firmwareversion 1.4.3119 (oder höher)

**SONICUE Sound System Software** 1.3.0 (oder höher) auf dem Computer installiert

## Anforderungen zur Verwendung der IX Task Engine

**IX-Verstärker** mit Firmwareversion 1.0.0 (oder höher)

**SONICUE Sound System Software** 1.5.0 (oder höher) auf dem Computer installiert

## TaskEngine Quick Start Guide

Jedes IX- und MXE-Modell beinhaltet eine leistungsstarke **Logikverarbeitung**, die unabhängig vom DSP für Audio ist. Die **TaskEngine** bietet alle Werkzeuge und eine umfangreiche Auswahl an Logiksteuerungs-Blöcken, die nötig sind, um umfassende integrierte Steuerungssysteme zu bauen. Sie bietet die **Verbindung** von verschiedener **SONICUE Hardware**, **PanelDesigner**-Userinterfaces und vielen **Drittanbieter**-Geräten.

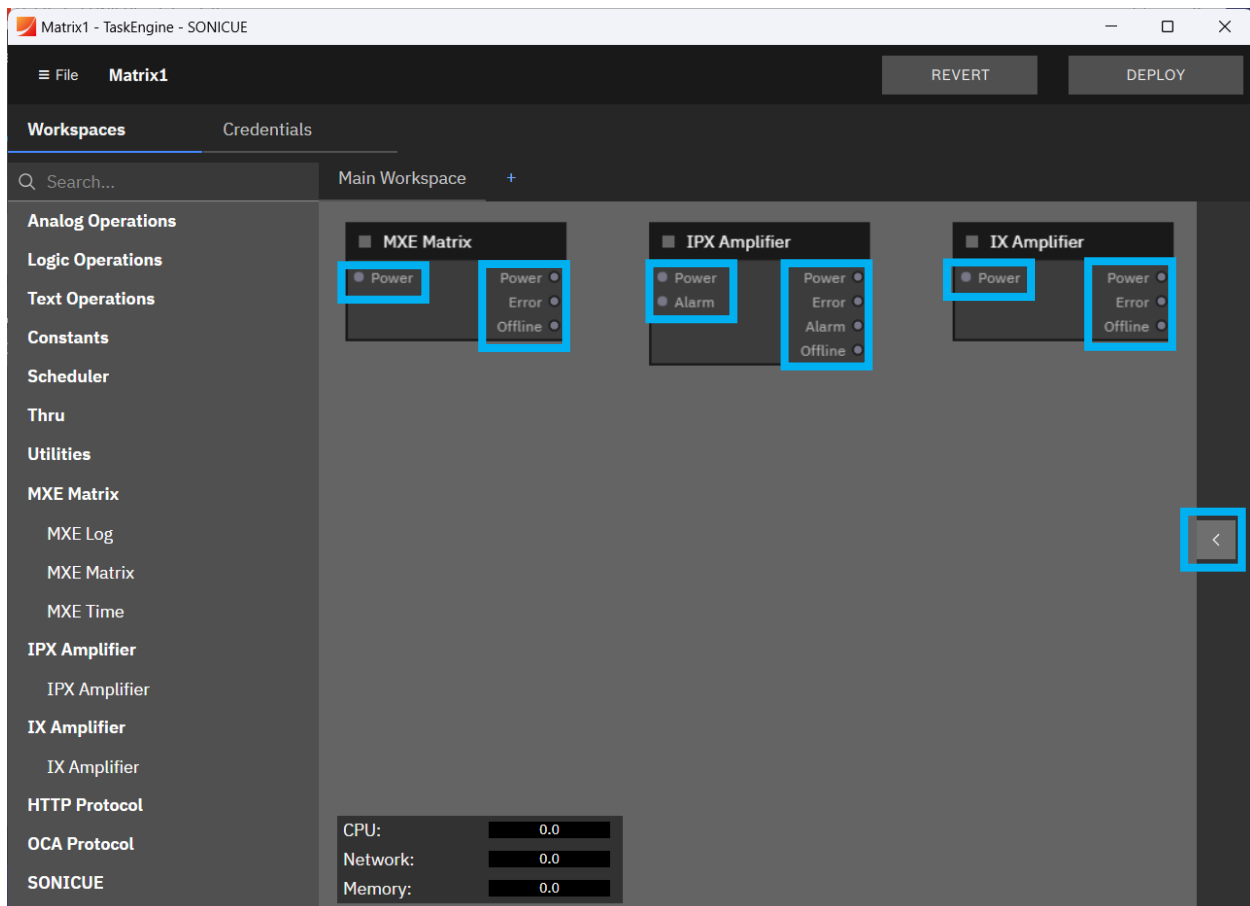
Die **IX-** und **MXE-TaskEngine** kann nicht nur **Parameter steuern**, die sich im Gerät selbst befinden, sondern auch **in anderen Geräten** auf demselben Netzwerk.

- MXE kann MXEs, IPX, IX, OCA-Geräte und Drittanbieterprodukte mit http API steuern.
- IX kann andere IXs und Drittanbieterprodukte mit http API steuern.

## Konfiguration von Input- und Output-Knoten

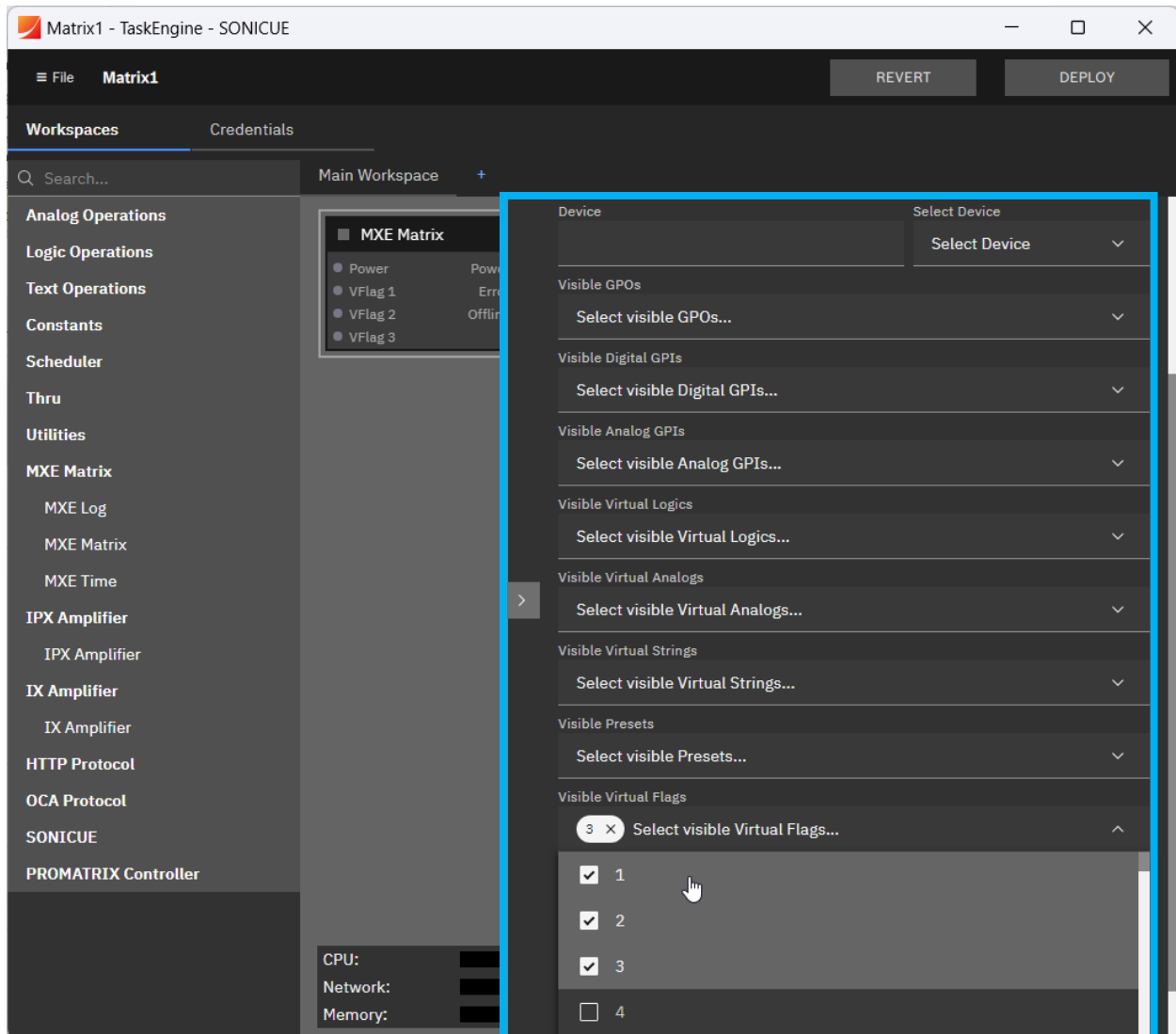
**MXE Matrix-**, **IPX Amplifier-** und **IX Amplifier-**Blöcke in der (MXE) **TaskEngine** haben per Default einen Input-Knoten zum Schalten von **Power** und Output-Knoten für **Power-**, **Error-** und **Offline-**Status.

**IPX-Verstärker** haben per Default **zusätzlich** Input- und Output-Knoten für das **Alarm**-Preset.



**Bild 5:** MXE-TaskEngine mit MXE Matrix-, IPX Amplifier- und IX Amplifier-Blöcken.

Durch die **Auswahl** eines Matrix- oder Verstärker-**Blocks** und das **Klicken** des -Buttons auf der rechten Seite des TaskEngine-Arbeitsblatts, wird der Eigenschaftenbereich geöffnet.



**Bild 6:** MXE-TaskEngine mit offenem Eigenschaftenbereich eines MXE Matrix-Blocks, *Visible Virtual Flags* 1-3 ausgewählt.

Für den **MXE Matrix**-Block können Knoten für Folgendes hinzugefügt werden:

- Bis zu **8\* Visible GPOs**
- Bis zu **8\* Visible Digital GPIs**
- Bis zu **8\* Visible Analog GPIs**
- Bis zu **200 Visible Virtual Logics**
- Bis zu **200 Visible Virtual Analogs**
- Bis zu **200 Visible Virtual Strings**
- Bis zu **60 User + 1 Factory Visible Presets**
- Bis zu **200 Visible Virtual Flags**

\*Die MXE hat 8 GPIOs, die als GPO, Digital GPI oder Analog GPI verwendet werden können.

Für den **IX Amplifier**-Block können Knoten für Folgendes hinzugefügt werden:

- Bis zu **3\* Visible GPOs**
- Bis zu **3\* Visible Digital GPIs**
- Bis zu **3\* Visible Analog GPIs**
- Bis zu **200 Visible Virtual Logics**
- Bis zu **200 Visible Virtual Analogs**
- Bis zu **200 Visible Virtual Strings**
- Bis zu **20 User + 1 Factory Visible Presets**
- Bis zu **10 Visible Virtual Flags**

\*IX hat 3 GPIOs, die als GPO, Digital GPI oder Analog GPI verwendet werden können.

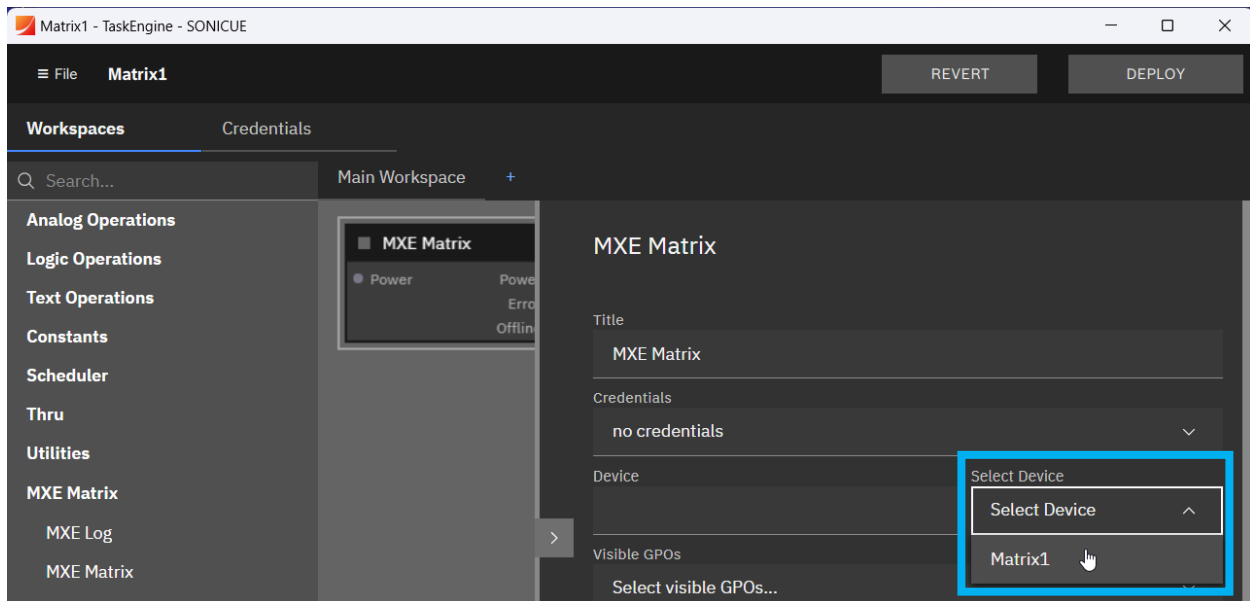
Für den **IPX Amplifier**-Block können Knoten für Folgendes hinzugefügt werden:

- Bis zu **3\* Visible GPOs**
- Bis zu **3\* Visible Digital GPIs**
- Bis zu **3\* Visible Analog GPIs**
- Bis zu **20 User + 1 Factory Visible Presets**

\*IPX hat 3 GPIOs, die als GPO, Digital GPI oder Analog GPI verwendet werden können.

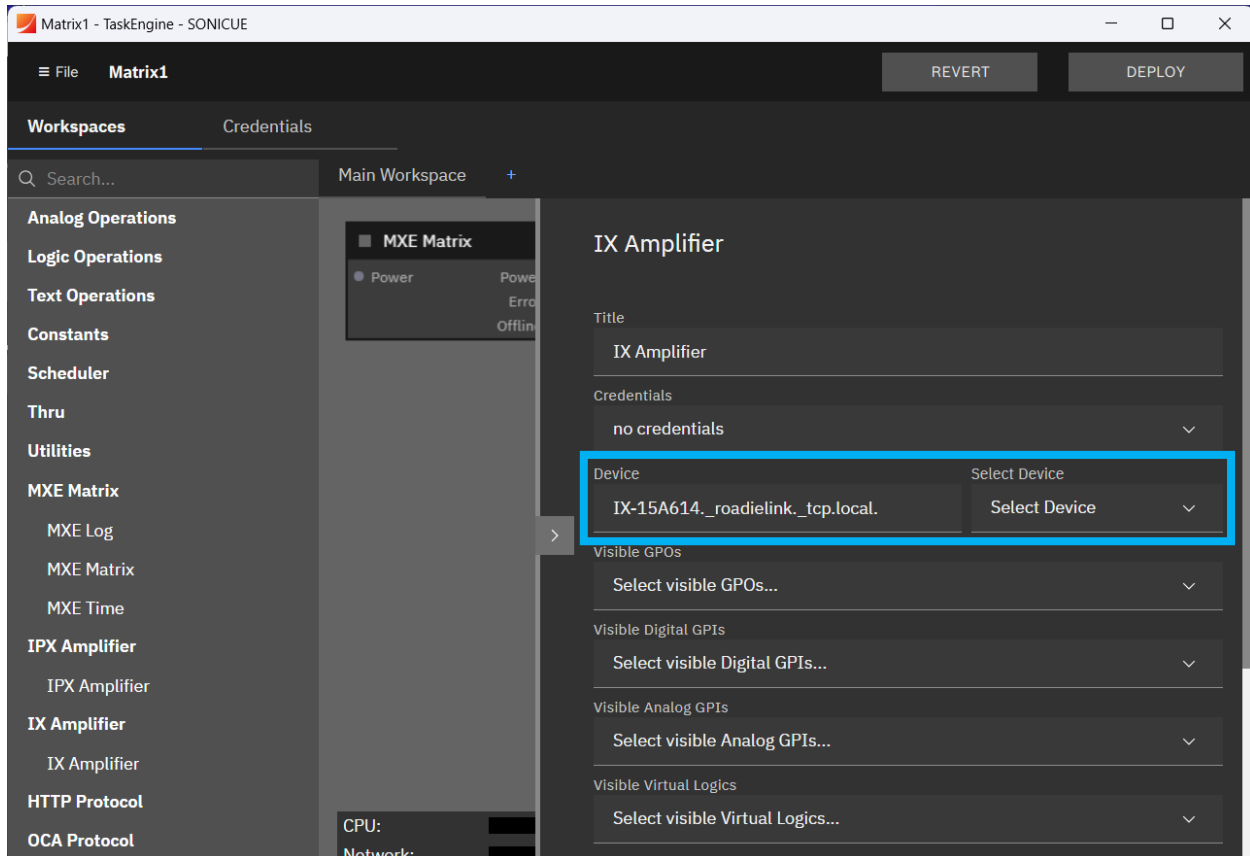
## Auswahl des korrekten Hardware-Geräts (Device)

Durch die **Auswahl** eines Matrix- oder Verstärker-**Blocks** und das **Klicken** des -Buttons auf der rechten Seite des TaskEngine-Arbeitsblatts, wird der Eigenschaftenbereich geöffnet.

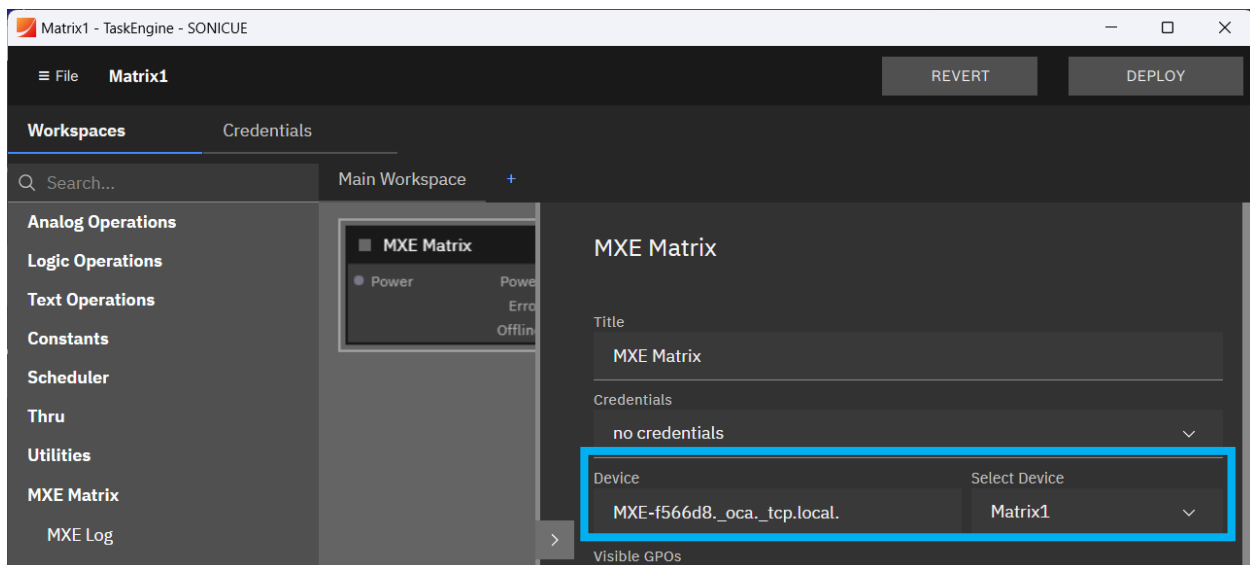


**Bild 7:** Device-Auswahl von Matrix1 für den MXE Matrix-Block über den Eigenschaftenbereich > Select Device.

Da sich **mehrere** MXE-Matrizen oder IX-/IPX-Verstärker auf dem Netzwerk befinden können, ist es elementar, **das korrekte** Hardware-**Device** mit **Select Device** auszuwählen. Dafür muss das Hardware-Gerät mit dem Netzwerk verbunden sein. Alternativ kann der korrekte Name manuell in das *Device*-Feld eingetragen werden, wenn das Gerät noch nicht mit dem Netzwerk verbunden ist.



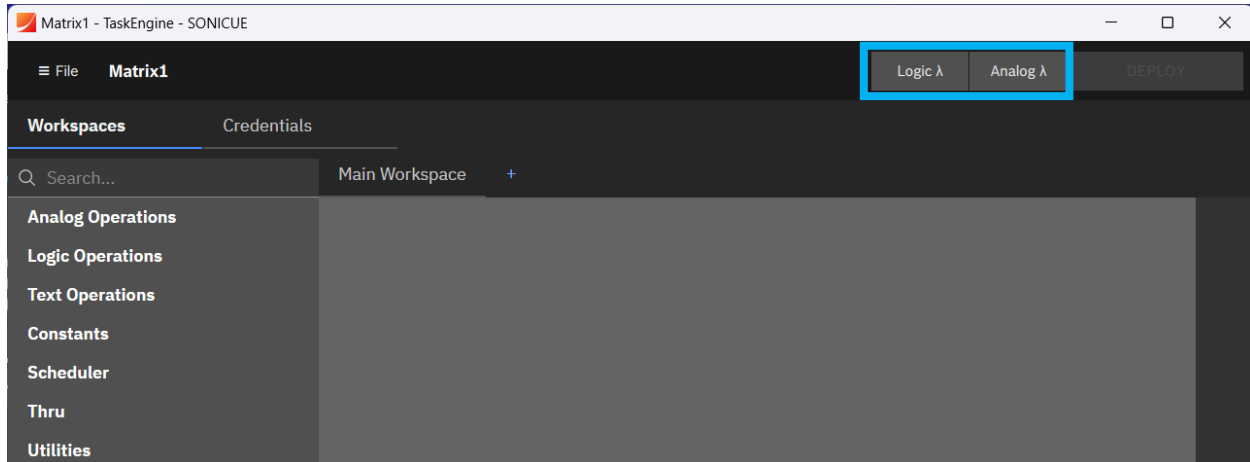
**Bild 8:** *Amp1* ist ausgewählt über *Select Device*, der Service-Name *IX-15A614* ist Teil des *Device*-Namens.



**Bild 9:** *Matrix1* ist ausgewählt über *Select Device*, der Service-Name *MXE-f566d8* ist Teil des *Device*-Namens.

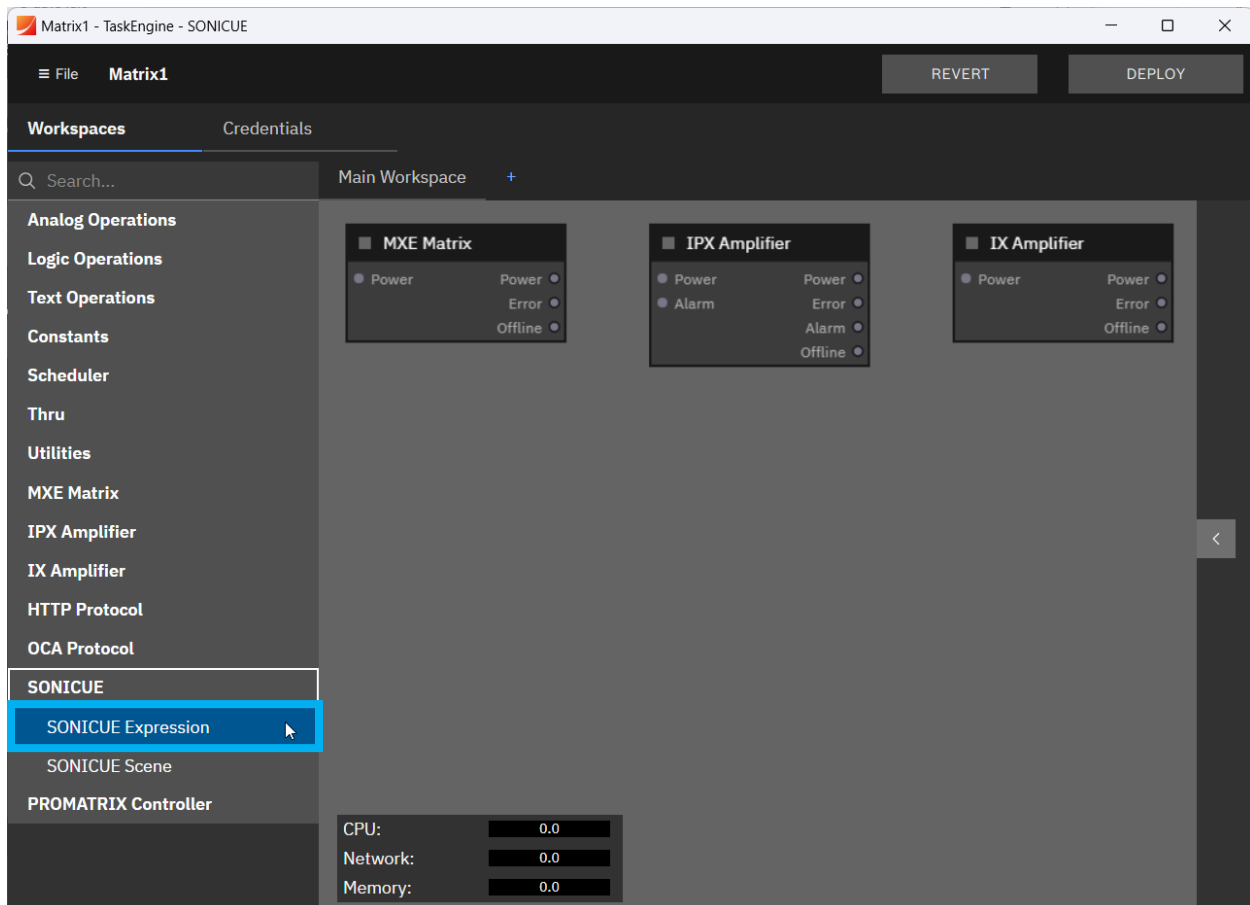
## Hinzufügen von SONICUE Expressions

In den **SONICUE**-Versionen **1.3.0...1.4.0** befinden sich die Buttons zum Hinzufügen von **Logic**- und **Analog**-Expressions in der **rechten oberen Ecke** des TaskEngine-Fensters.



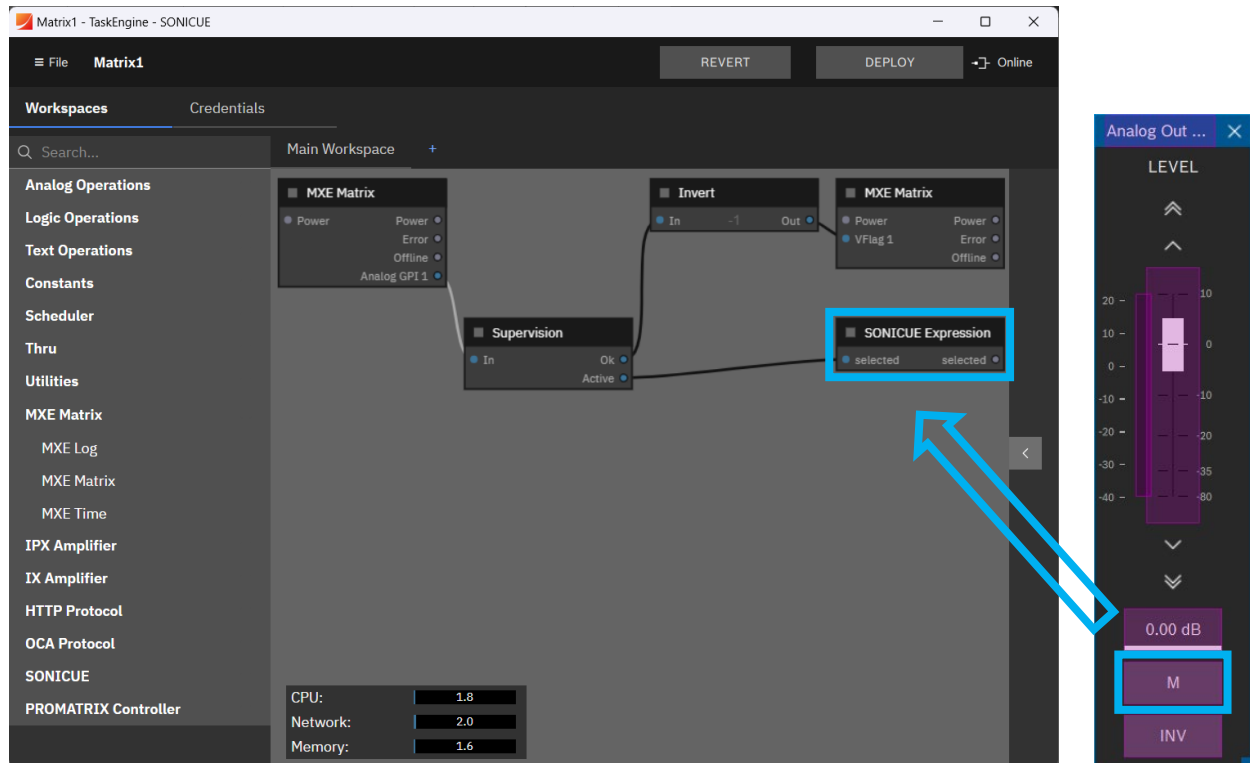
**Bild 10:** Logic λ- und Analog λ-Buttons zum Hinzufügen von SONICUE Expressions in der TaskEngine.

In den **SONICUE**-Versionen **1.5.0** und **höher**, können logische und analoge **SONICUE Expressions** aus dem **TaskEngine-Katalog**, **SONICUE**-Menü, hinzugefügt werden.



**Bild 11:** SONICUE-Menü mit SONICUE Expression.

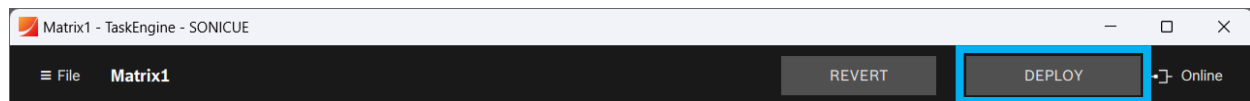
Alternativ ist es für **DSP-Parameter** möglich, diese über **Strg + Drag&Drop** direkt von einem **DSP-Flyout** einzufügen, selbst mit Mehrfachauswahl (= Gruppe).



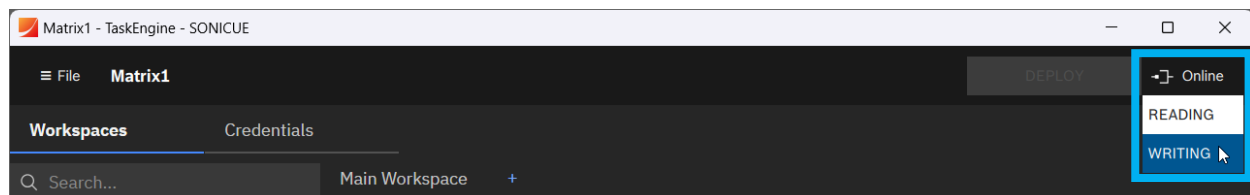
**Bild 12 und 13:** Hinzufügen einer *SONICUE Expression* zur TaskEngine über Drag&Drop vom *Output-Flyout*.

## Aktivieren (Deploy) der Logikkonfiguration

Der **wichtigste Schritt** nach dem **Bauen** oder **Ändern** einer **TaskEngine-Logik** ist stets die Logik mit **DEPLOY** in die Hardware zu schreiben, wenn man **ONLINE** mit dem Gerät ist. Beim schreibend Onlinegehen wird eine Logik automatisch aktiviert.



**Bild 14:** *DEPLOY*-Button "aktiv", wenn die TaskEngine-Konfiguration noch nicht aktiviert worden ist.



**Bild 15:** Schreibendes Onlinegehen (*WRITING*) aktiviert die TaskEngine-Logik automatisch.

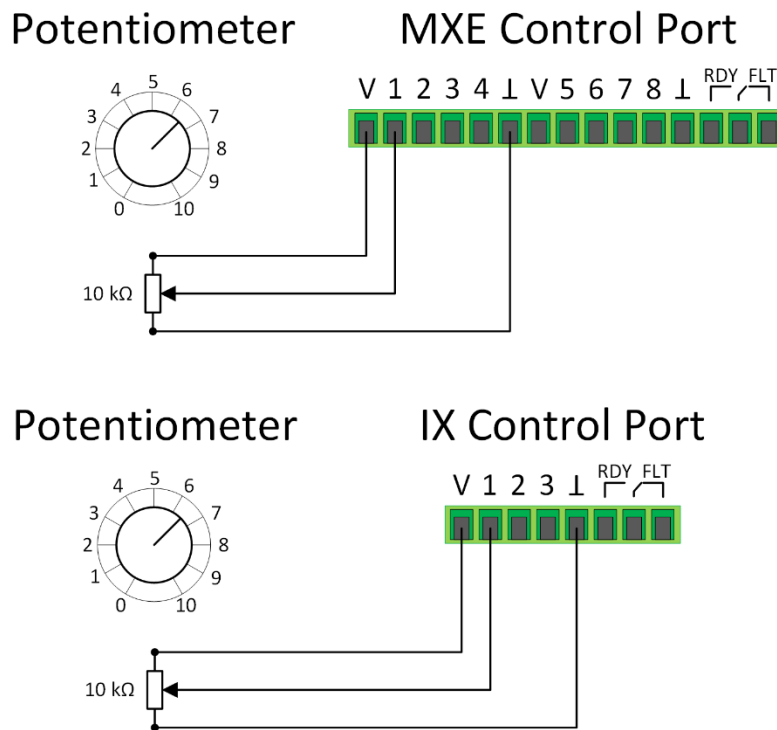


**Bild 16:** *DEPLOY*-Button "inaktiv", wenn die TaskEngine-Konfiguration bereits aktiviert worden ist.

## 1. Beispiel Analoge Lautstärkeeinstellung

Dieses Beispiel zeigt, wie ein **analoges Potentiometer**, verbunden mit dem **IX- oder MXE-Control-Port (GPIO)**, verwendet werden kann, um eine **DSP-Lautstärke** einzustellen.

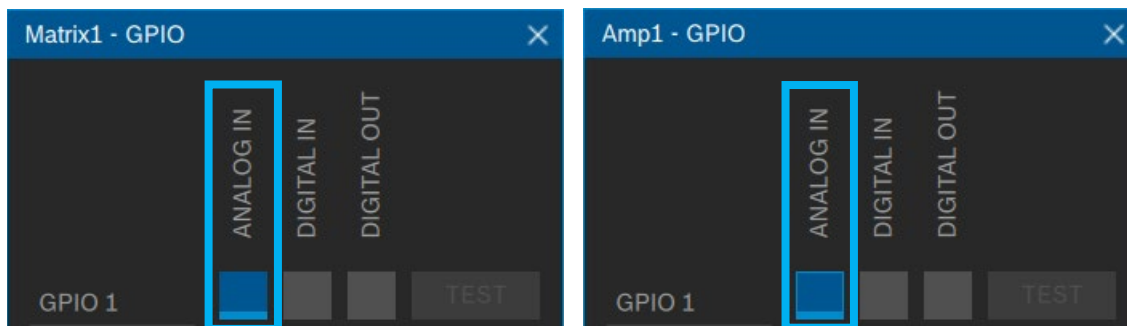
### Verdrahtungsdiagramm



**Bild 17:** Verbindung eines analogen 10 kΩ-Potentiometers mit dem MXE- oder IX-Control-Port.

### GPIO-Konfiguration

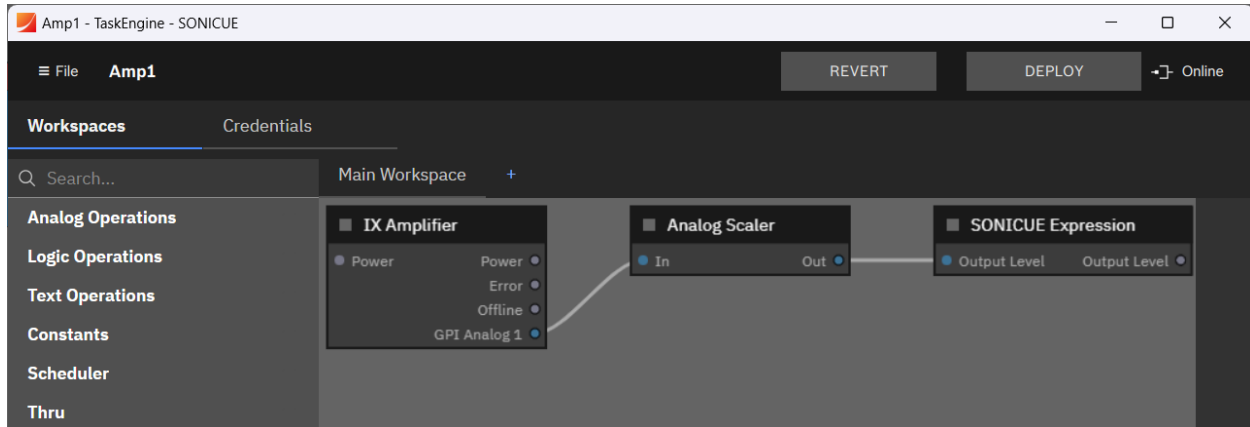
Stellen Sie sicher, dass in SONICUE unter **Setup>GPIO** der GPIO, der verwendet werden soll, korrekt ausgewählt ist. In diesem Beispiel muss **GPIO 1** als **ANALOG IN** konfiguriert sein.



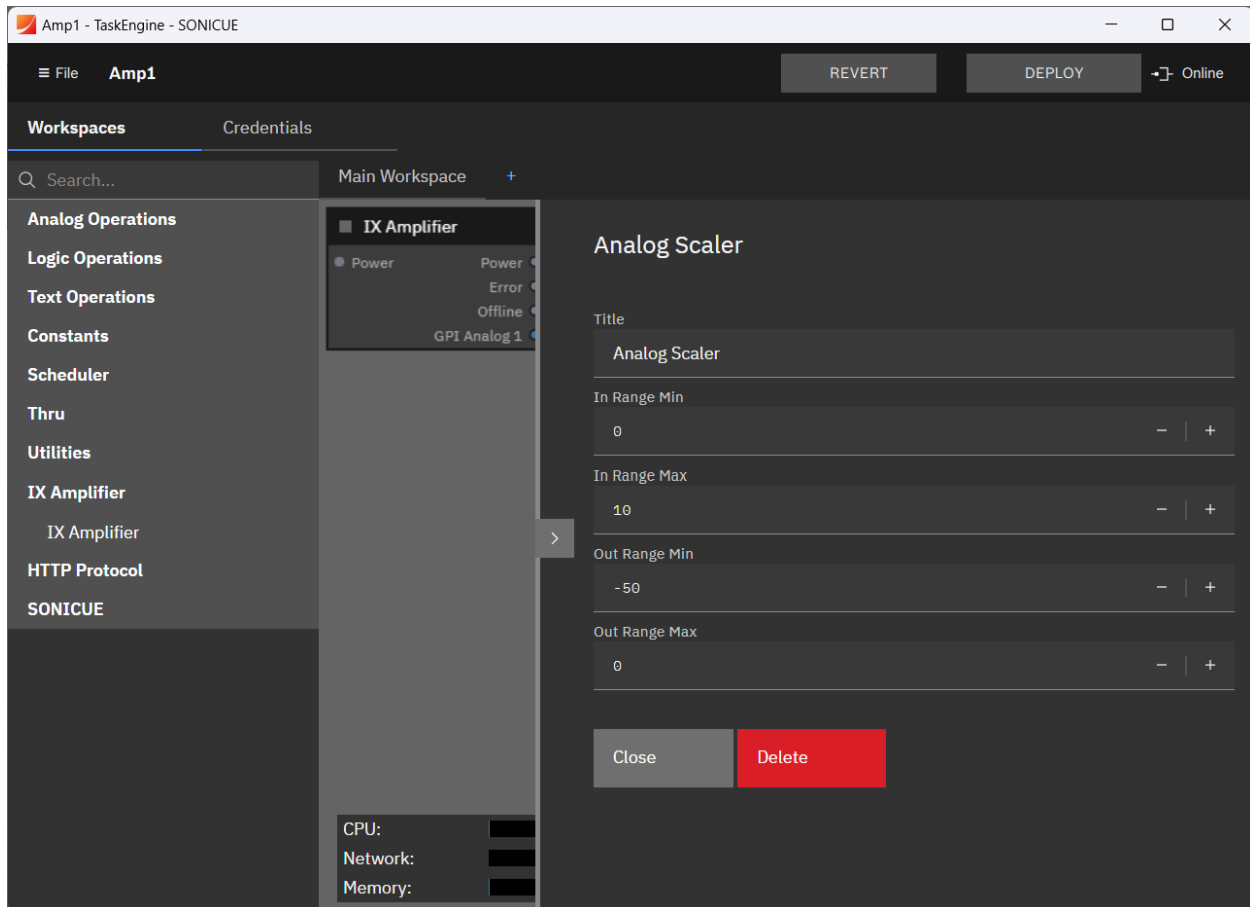
**Bild 18 und 19:** Setzen des *Matrix1* (links) oder *Amp1* (rechts) GPIO 1 als **ANALOG IN**.

## TaskEngine-Programmierung

In der folgenden TaskEngine-Struktur wird **GPI Analog 1** einer IX verwendet, um einen DSP-Level (*SONICUE Expression*) einzustellen. Der **Analog Scaler**-Block **übersetzt** der Steuer-Spannungsbereich (0...10 V) in den DSP-Lautstärkebereich (z.B. -50...0 dB) der mit dem analogen Potentiometer einstellbar sein soll.



**Bild 20:** IX-TaskEngine-Konfiguration mit *Analog Scaler*-Block.

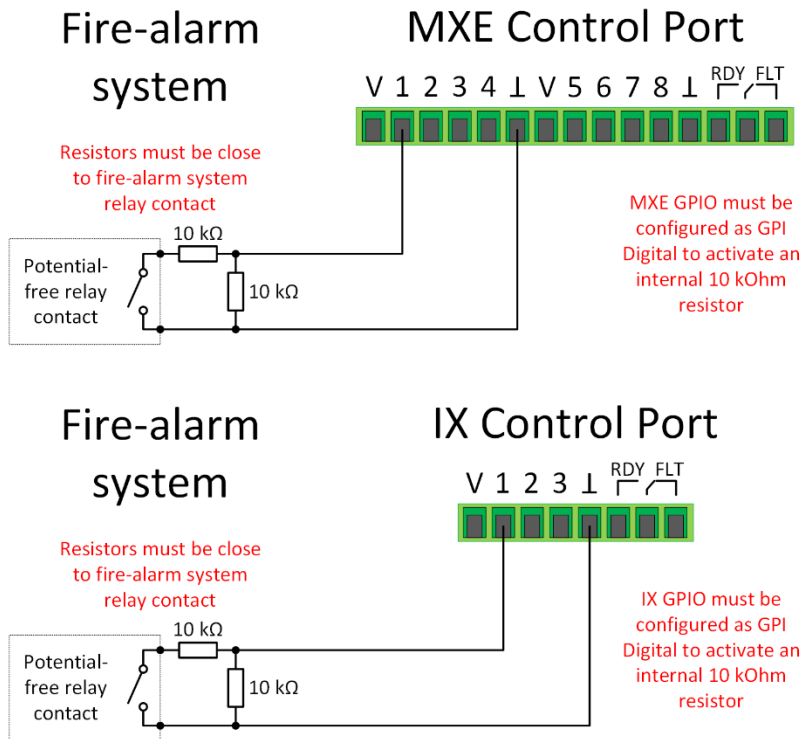


**Bild 21:** Eigenschaftenbereich des *Analog Scaler*-Blocks offen, mit typischen Werten für Zonen-Level (-50...0 dB).

## 2. Beispiel Brandmelde-Mute

Dieses Beispiel zeigt, wie ein **potenzialfreier Relaiskontakt**, verbunden mit dem **IX- oder MXE-Control-Port (GPIO)**, verwendet werden kann, um ein **DSP-Mute** zu aktivieren. Die Steuerung wird mit zwei 10 kOhm Widerständen überwacht.

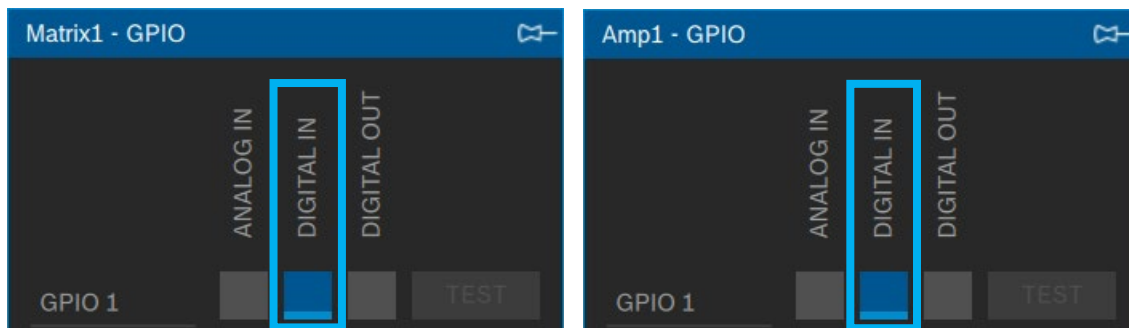
### Verdrahtungsdiagramm



**Bild 22:** Verbindung eines potenzialfreien Kontaktes einer Brandmeldeanlage mit dem MXE- oder IX-Control-Port.

### GPIO-Konfiguration

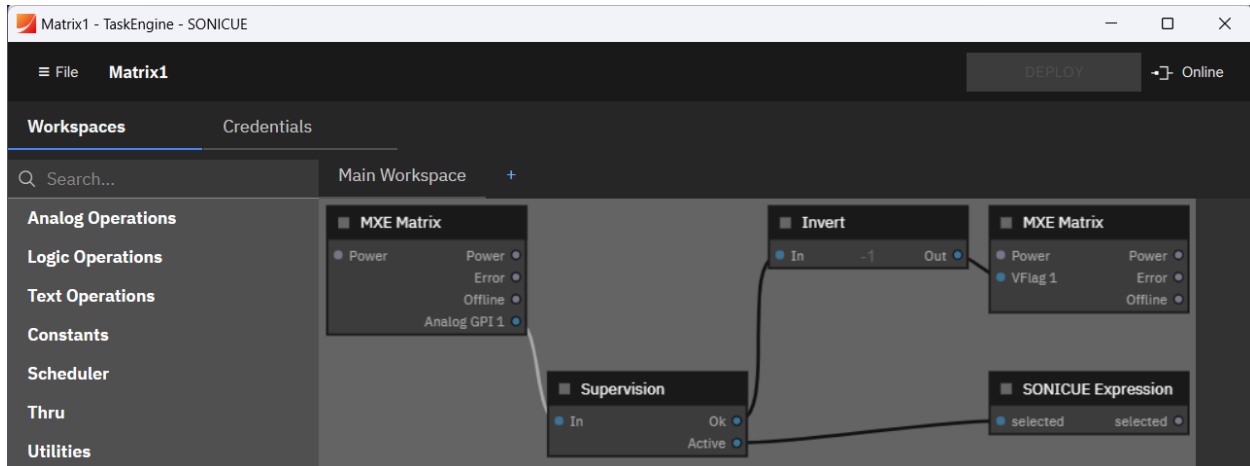
Stellen Sie sicher, dass in SONICUE unter **Setup>GPIO** der GPIO, der verwendet werden soll, korrekt ausgewählt ist. In diesem Beispiel muss **GPIO 1** als **DIGITAL IN** konfiguriert sein.



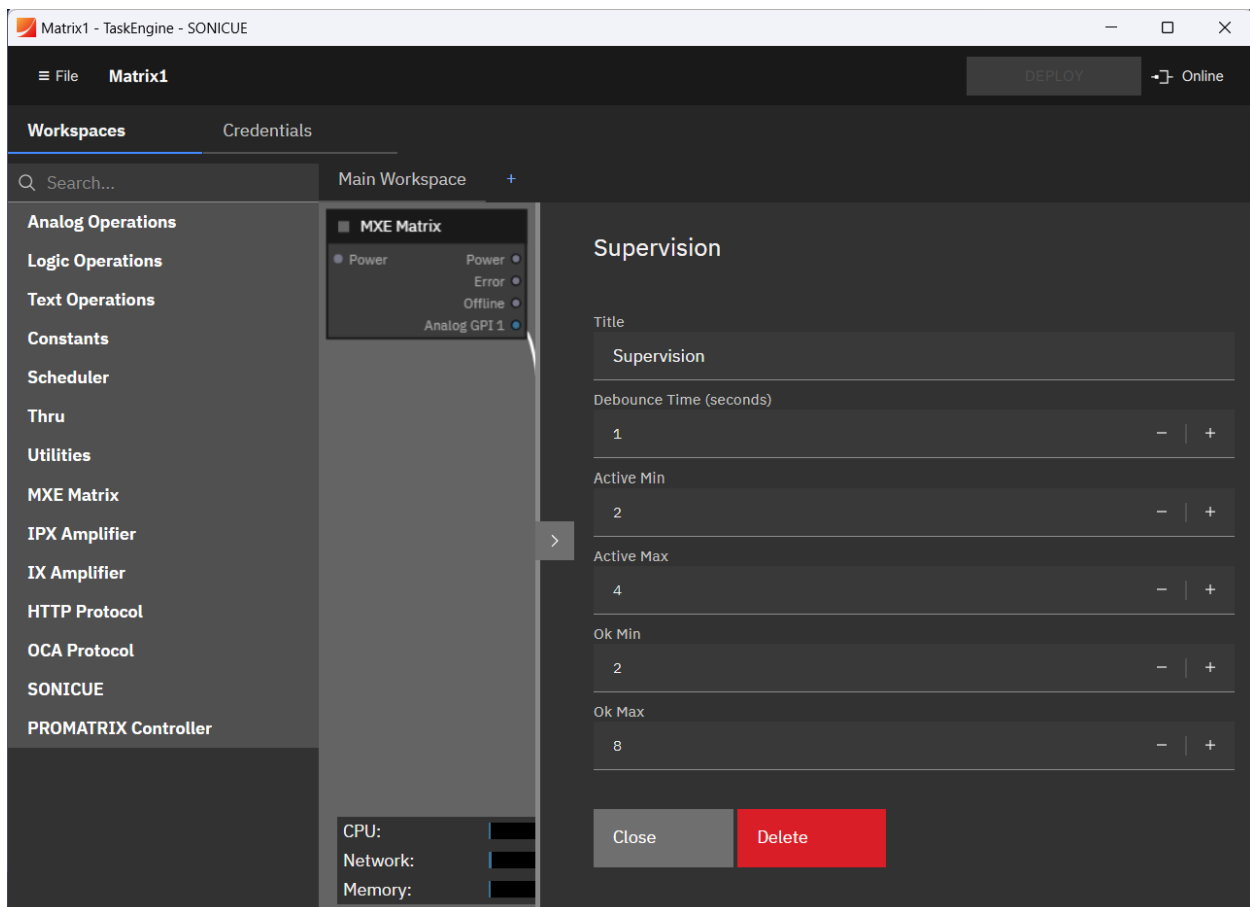
**Bild 23 und 24:** Setzen des *Matrix1* (links) oder *Amp1* (rechts) GPIO 1 als **DIGITAL IN**.

## TaskEngine-Programmierung

In der folgenden TaskEngine-Struktur wird **Analog GPI 1** einer MXE verwendet, um einen DSP-**Mute** (*SONICUE Expression*) in mehreren Ausgängen zu aktivieren. Der **Spannungsbereich** des Analog GPI 1 wird mit einem **Supervision**-Block überwacht. Ein Fehler im **Spannungsbereich** der **überwachten Spannung** wird über ein MXE Virtual Flag **VFlag 1** gemeldet. Bitte beachten Sie den **Invert**-Block, um VFlag 1 (User Flag 1) zu triggern, wenn etwas **nicht OK** ist.



**Bild 25:** MXE-TaskEngine-Logik mit *Supervision*-Block für die Überwachung der analogen Steuerleitung.

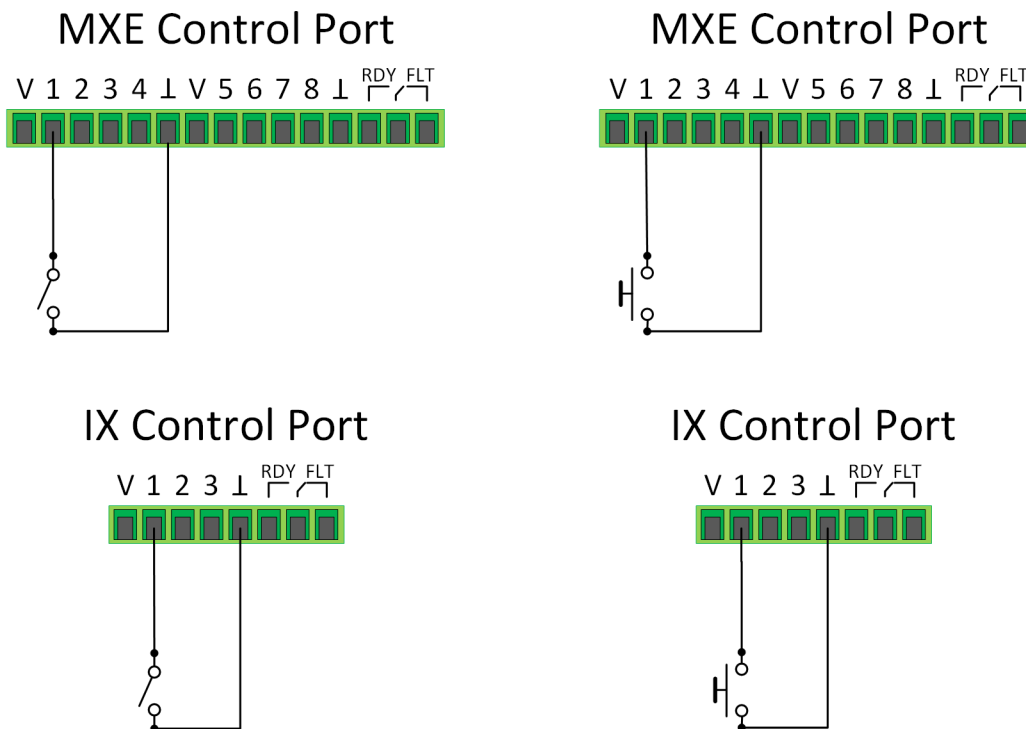


**Bild 26:** Eigenschaftenbereich des *Supervision*-Blocks geöffnet, mit Konfigurations-Details (alles Default-Werte).

### 3. Beispiel Power/Standby-Schalter

Dieses Beispiel zeigt, wie ein **Schalter** oder **Taster**, verbunden mit dem **IX-** oder **MXE-Control-Port (GPIO)**, verwendet werden kann, um im Gerät **Power/Standby** zu schalten.

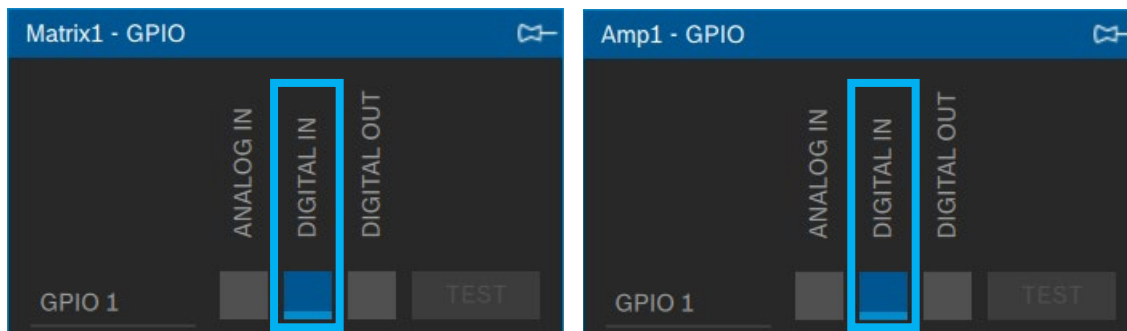
#### Verdrahtungsdiagramm



**Bild 27 und 28:** Verbindung eines Schalters (links) oder Tasters (rechts) mit dem MXE- oder IX-Control-Port.

#### GPIO-Konfiguration

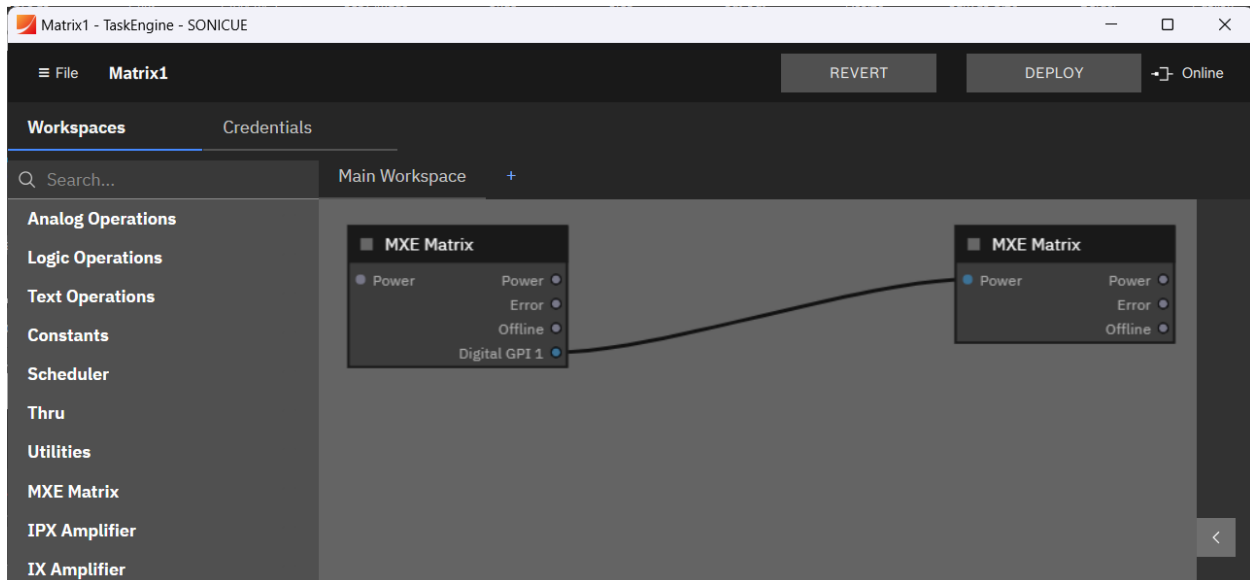
Stellen Sie sicher, dass in SONICUE unter **Setup>GPIO** der GPIO, der verwendet werden soll, korrekt ausgewählt ist. In diesem Beispiel muss **GPIO 1** als **DIGITAL IN** konfiguriert sein.



**Bild 29 und 30:** Setzen des *Matrix1* (links) oder *Amp1* (rechts) GPIO 1 als **DIGITAL IN**.

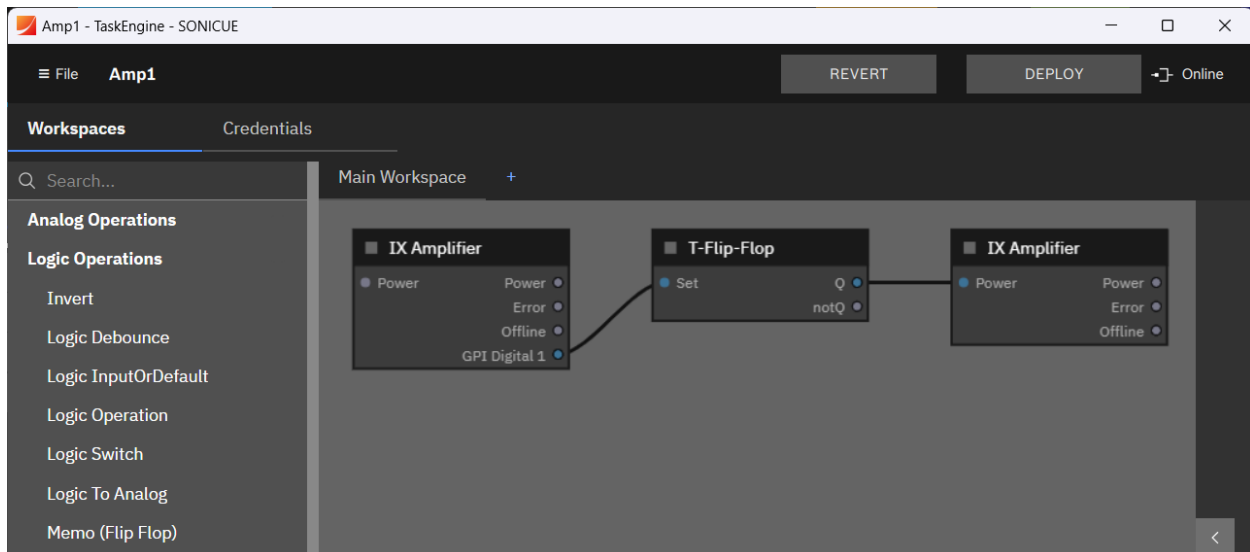
## TaskEngine-Programmierung

In der folgenden TaskEngine-Struktur wird **Digital GPI 1** einer MXE verwendet, um MXE-**Power** über einen Schalter, verbunden mit GPIO 1, zu schalten.



**Bild 31:** MXE-TaskEngine-Konfiguration zum Schalten von MXE-Power über MXE *Digital GPI 1* (mit Schalter).

In der folgenden, modifizierten TaskEngine-Struktur wird **Digital GPI 1** einer IX verwendet, um IX-**Power** über einen Taster, verbunden mit GPIO 1, zu schalten. Ein **T-Flip-Flop** toggelt bei jeder Betätigung des Tasters.

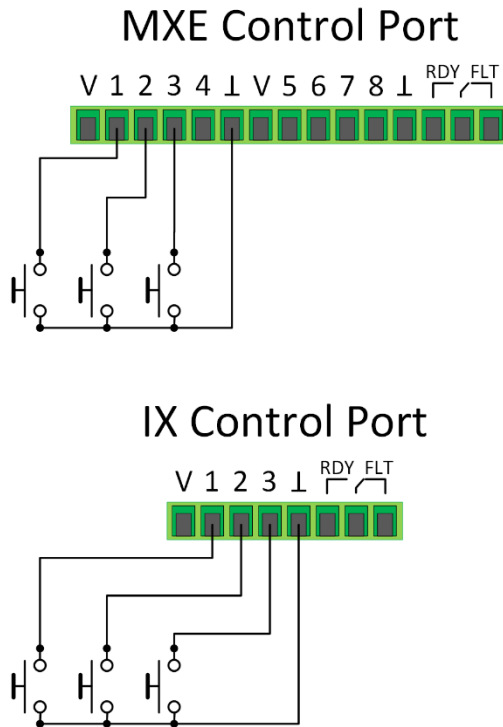


**Bild 32:** IX-TaskEngine-Konfiguration mit *T-Flip-Flop* zum Schalten von IX-Power über IX-*Digital GPI 1* (mit Taster).

## 4. Beispiel Preset-Aufruf oder Szenen-Aufruf

Dieses Beispiel zeigt, wie **Taster**, verbunden mit dem **IX-** oder **MXE-Control-Port (GPIO)**, verwendet werden können, um **Presets** oder **Szenen** aufzurufen.

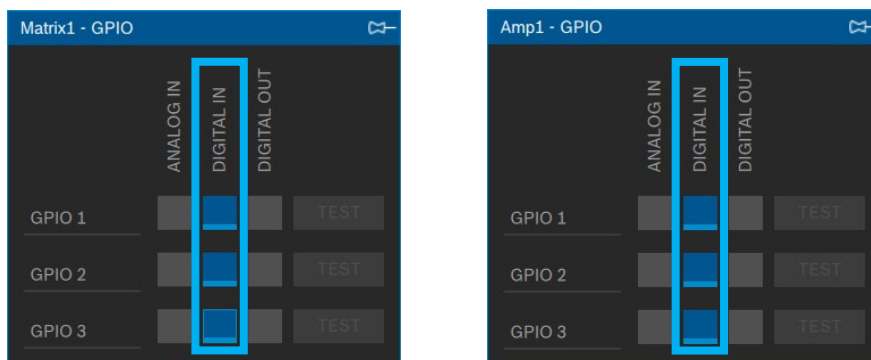
### Verdrahtungsdiagramm



**Bild 33:** Verbindung von drei Tastern mit dem MXE- oder IX-Control-Port.

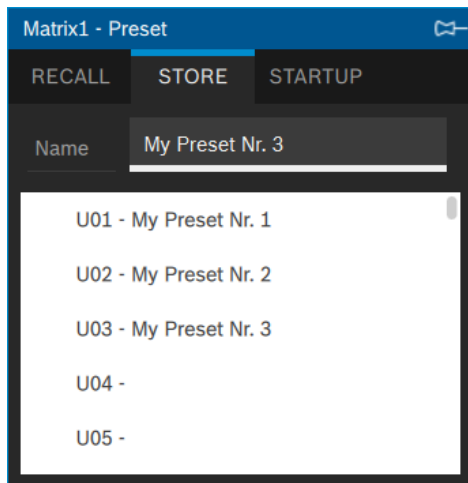
### GPIO-Konfiguration

Stellen Sie sicher, dass in SONICUE unter **Setup>GPIO** die GPIOs, die verwendet werden sollen, korrekt ausgewählt sind. In diesem Beispiel müssen **GPIO 1, 2 und 3** als **DIGITAL IN** konfiguriert sein.



**Bild 34 und 35:** Setzen der *Matrix1* (links) oder *Amp1* (rechts) GPIOs 1, 2 und 3 als **DIGITAL IN**.

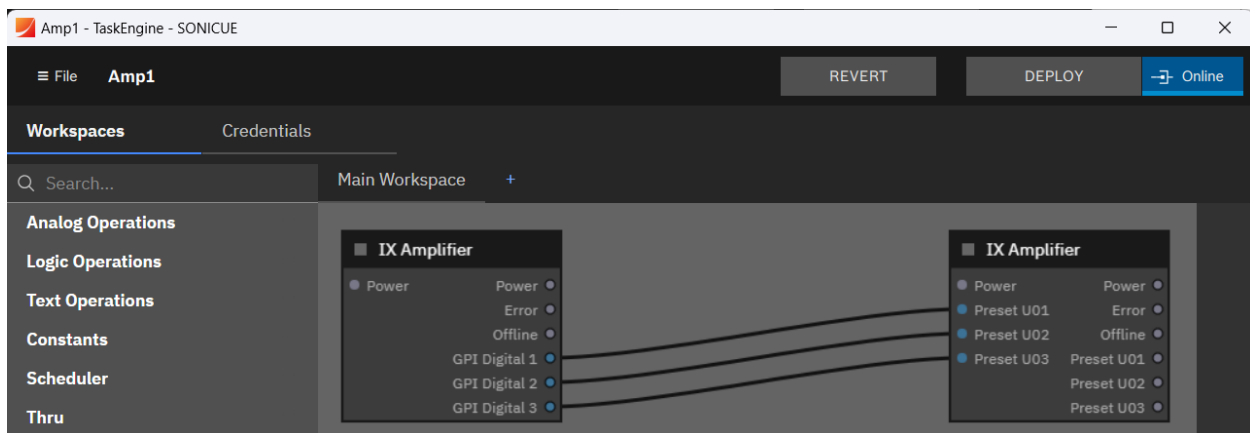
## Presets die aufgerufen werden sollen



**Bild 36:** Matrix1 – Preset-Flyout mit Presets U01 – U03 gespeichert.

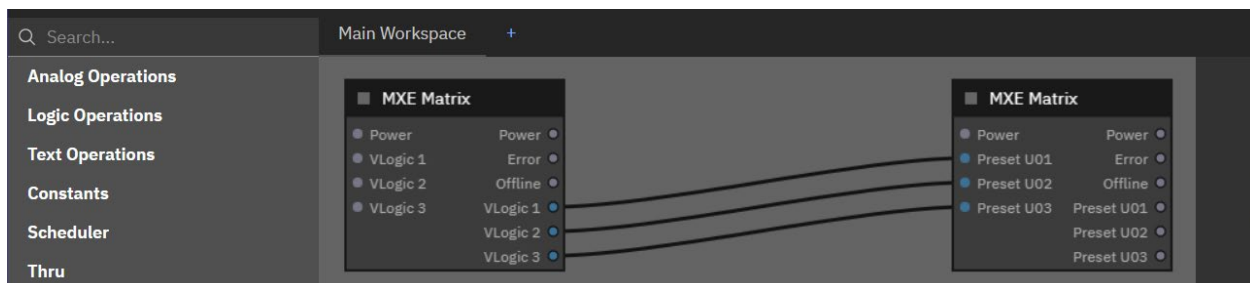
## TaskEngine-Programmierung für Preset-Aufruf

In der folgenden TaskEngine-Struktur werden **Digital GPI 1, 2 und 3** eines IX-Verstärkers verwendet, um User **Preset U01, U02 und U03** aufzurufen.



**Bild 37:** IX-TaskEngine-Konfiguration zum Aufruf von Preset U01, U02 und U03 via GPIOs (GPI Digital 1, 2 und 3).

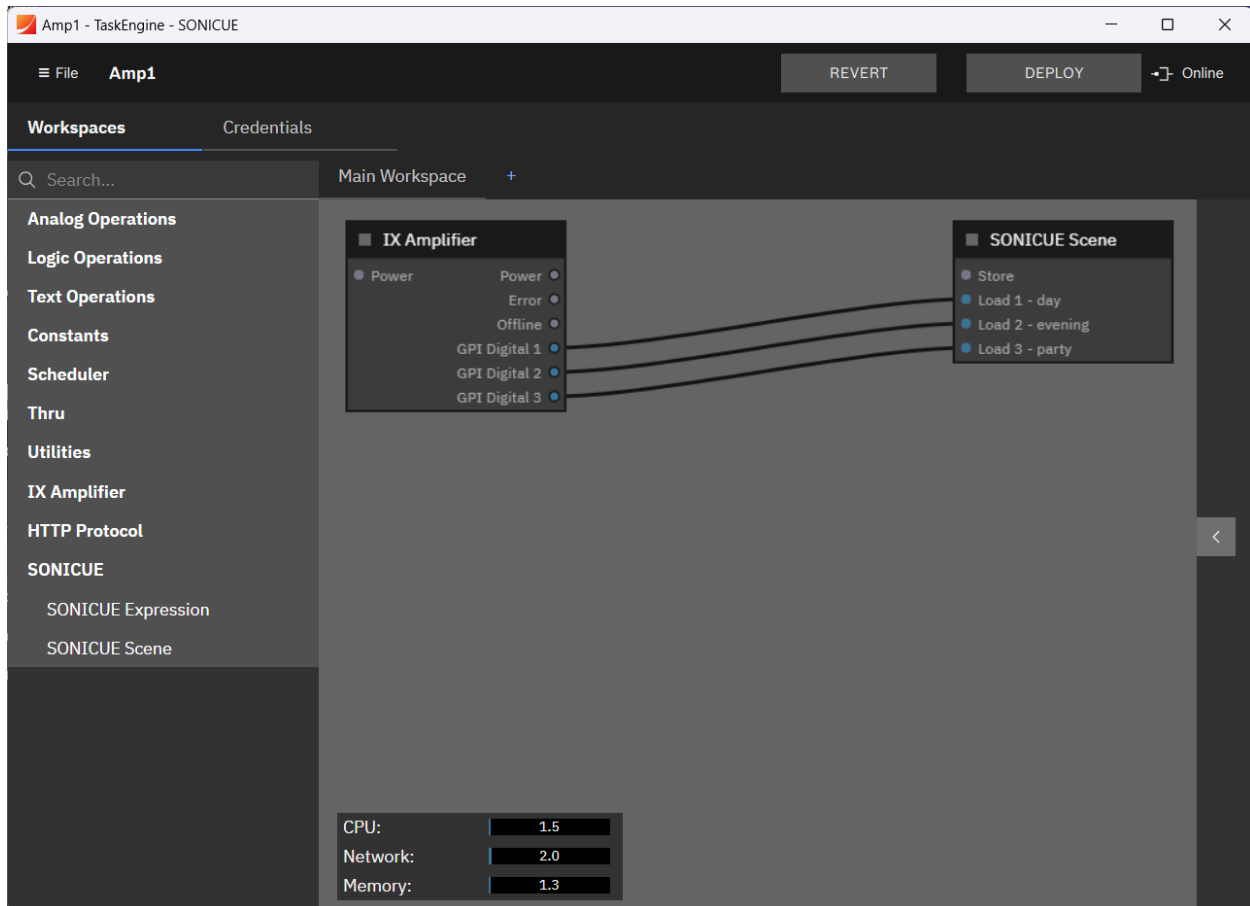
Alternativ können auch **VLogic 1, 2 und 3** verwendet werden für Drittanbieter-Steuerung über die MXE- (oder IX-) **http API**.



**Bild 38:** MXE-TaskEngine-Konfiguration zum Aufruf von Preset U01, U02 und U03 via VLogic 1, 2 und 3.

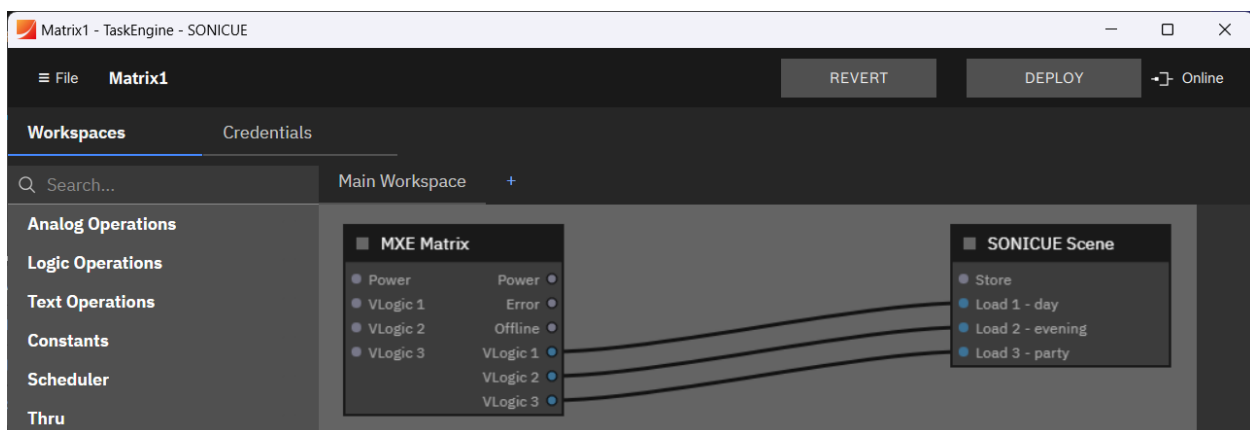
## TaskEngine-Programmierung für Szenen-Aufruf

In der folgenden TaskEngine-Struktur werden **Digital GPI 1, 2 und 3** eines IX-Verstärkers verwendet, um die **Slots Load 1, 2 und 3** in einem **SONICUE Scene**-Block aufzurufen.



**Bild 39:** IX-TaskEngine-Konfiguration zum Aufruf von Szenen-Slots *Load 1, 2 und 3* via *GPI Digital 1, 2 und 3*.

Auch hier können alternativ **VLogic 1, 2 und 3** als **Schnittstelle zu Drittanbieter**-Steuerungen über **http API** oder Buttons in einem **SONICUE Control** Userinterface, erzeugt mit SONICUE PanelDesigner, verwendet werden.

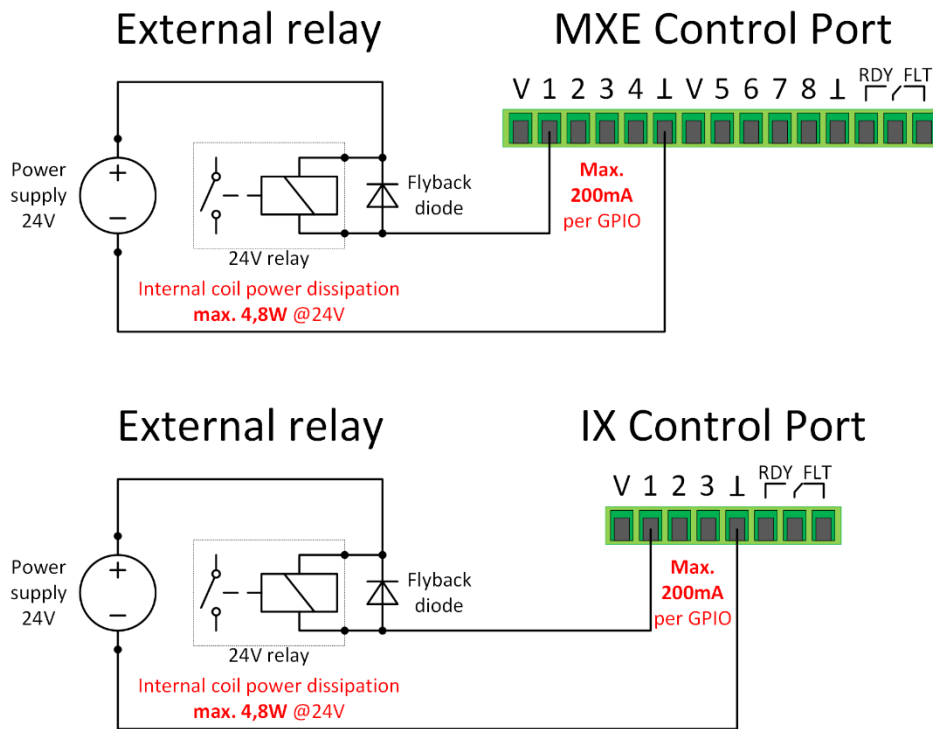


**Bild 40:** MXE-TaskEngine-Konfiguration zum Aufruf von Szenen-Slots *Load 1, 2 und 3* via *VLogic 1, 2 und 3*.

## 5. Beispiel Externes Relais

Dieses Beispiel zeigt, wie ein **externes Relais**, verbunden mit dem **IX- oder MXE-Control-Port (GPIO)**, verwendet werden kann, um **höhere Ströme und Spannungen** zu schalten, als dies mit den GPIOs direkt möglich ist. Ein Relais ist auch zur Potenzialtrennung geeignet.

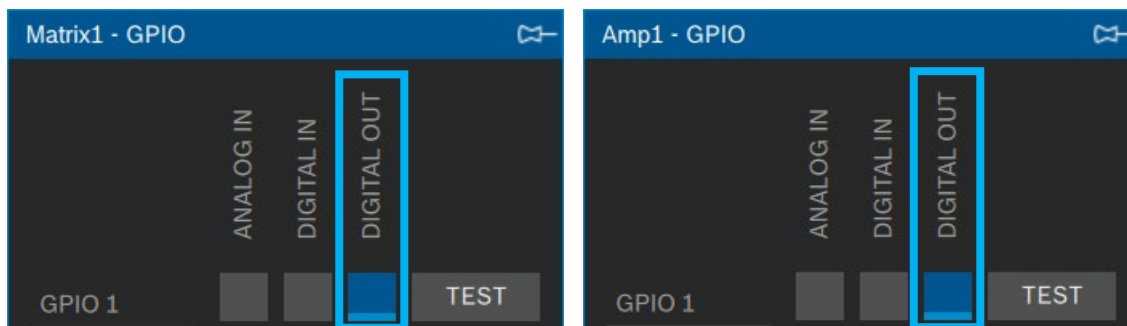
### Verdrahtungsdiagramm



**Bild 41:** Verbindung eines externen Relais mit dem MXE- oder IX-Control-Port.

### GPIO-Konfiguration

Stellen Sie sicher, dass in SONICUE unter **Setup>GPIO** der GPIO, der verwendet werden soll, korrekt ausgewählt ist. In diesem Beispiel muss **GPIO 1** als **DIGITAL OUT** konfiguriert sein.



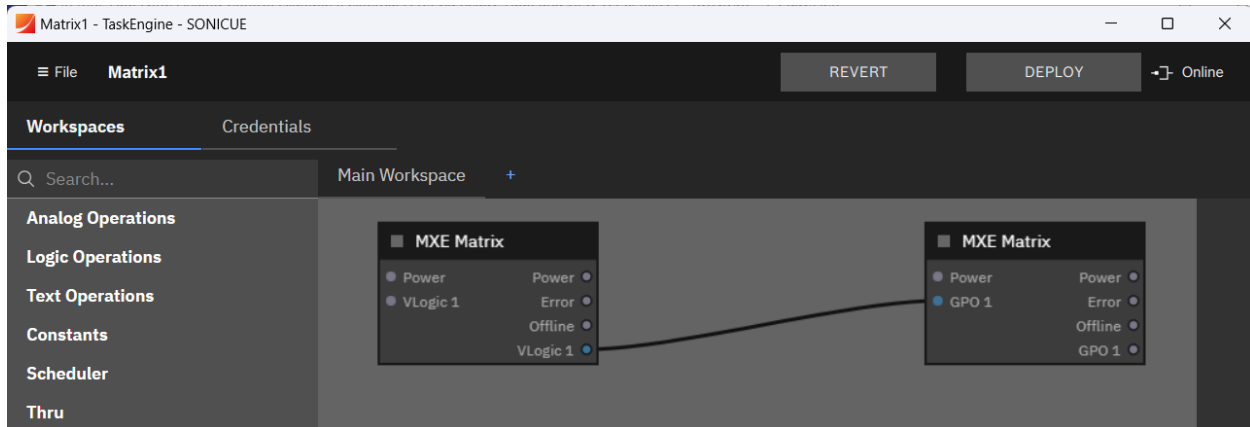
**Bild 42 und 43:** Setzen des *Matrix1* (links) oder *Amp1* (rechts) GPIO 1 als **DIGITAL OUT**.

## TaskEngine-Programmierung

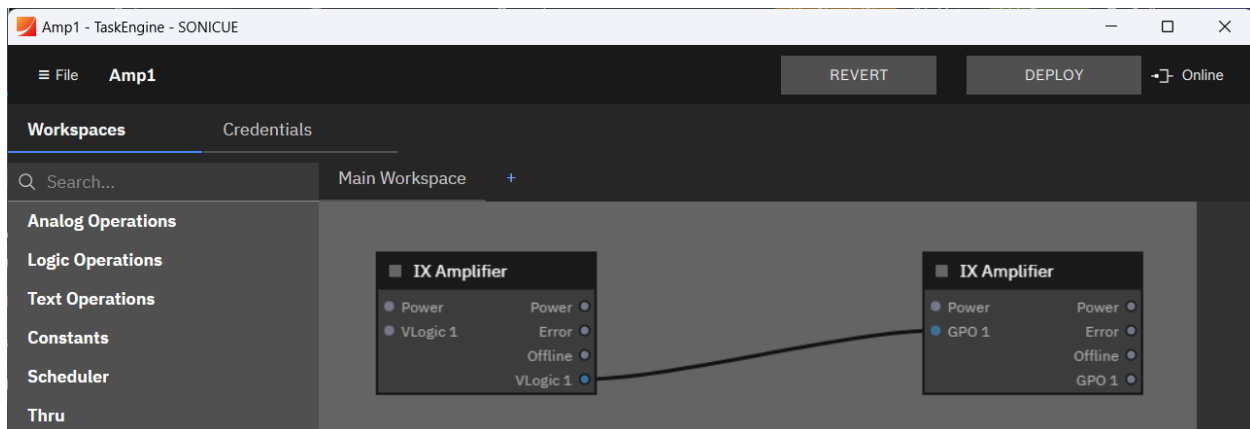
In der folgenden TaskEngine-Struktur wird **VLogic 1** eines IX-Verstärkers oder MXE-Matrix verwendet, um **GPO 1** zum Schalten eines **externen Relais** zu aktivieren.

Die IX- und MXE **virtuellen analogen** und **logischen** Werte sind eine perfekte Möglichkeit ein **SONICUE Control**-Userinterface mit der IX- oder MXE-**TaskEngine** zu verbinden.

Außerdem können die **virtuellen analogen** und **logischen** Werte über die **IX-/MXE-http API** adressiert werden.



**Bild 44:** MXE-TaskEngine-Konfiguration zum Schalten eines externen Relais, verbunden mit *GPO 1*, über *VLogic 1*.

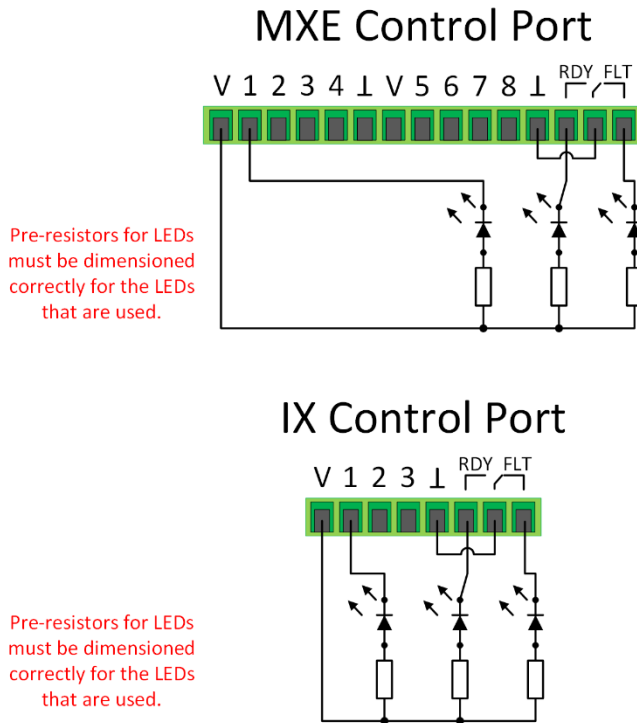


**Bild 45:** IX-TaskEngine-Konfiguration zum Schalten eines externen Relais, verbunden mit *GPO 1*, über *VLogic 1*.

## 6. Beispiel Externe LEDs

Dieses Beispiel zeigt, wie **externe LEDs**, verbunden mit dem **IX-** oder **MXE-Control-Port** (**GPIOs** oder **Ready/Fault-Kontakt**), verwendet werden können, um **Geräte-** oder **System-Status** zu signalisieren

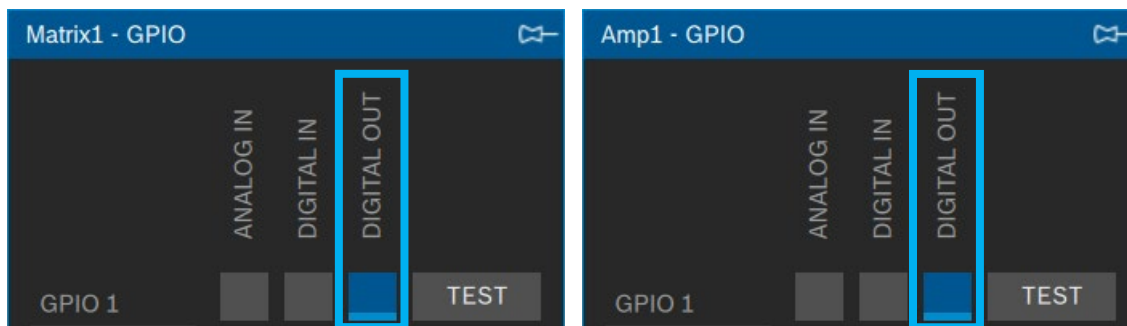
### Verdrahtungsdiagramm



**Bild 46:** Verbindung von externen LEDs mit dem MXE- oder IX-Control-Port.

### GPIO-Konfiguration

Stellen Sie sicher, dass in SONICUE unter **Setup>GPIO** der GPIO, der verwendet werden soll, korrekt ausgewählt ist. In diesem Beispiel muss **GPIO 1** als **DIGITAL OUT** konfiguriert sein.

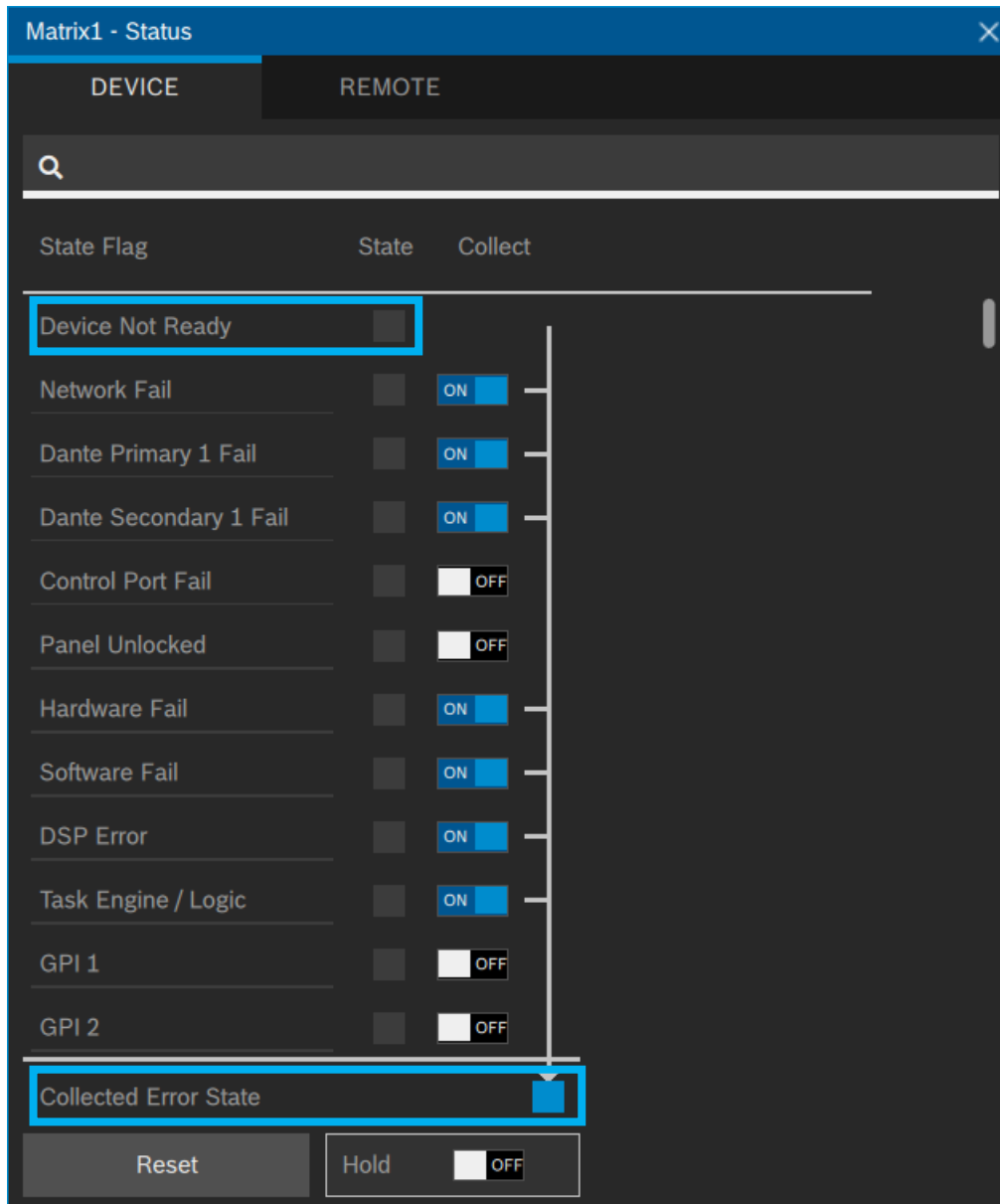


**Bild 47 und 48:** Setzen des *Matrix1* (links) oder *Amp1* (rechts) GPIO 1 als DIGITAL OUT.

## Ready/Fault-Konfiguration

Der Status des potenzialfreien **Ready/Fault**-Relaiskontakts hängt von der Konfiguration der IX- oder MXE-**State Flags** unter **Setup>Status** ab und davon, ob das Gerät **komplett gebootet** hat (= **ready**).

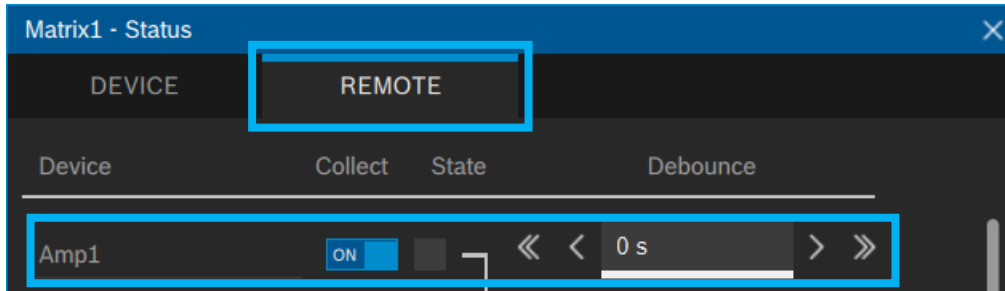
Wenn entweder das **Device Not Ready**-Flag, oder **ein anderes Flag**, das in der **Collect-Spalte** als Teil des **Collected Error State** ausgewählt wurde, aktiv ist, zeigt die IX oder MXE auf dem Frontpanel Fehlerzustand an. Die Fault-LED ist aktiv und der Fault-Relaiskontakt geschlossen.



**Bild 49:** MXE-Status-Flyout – *DEVICE*-Tab.

Im selben **Status**-Flyout, unter dem **REMOTE**-Tab, kann der **Status anderer Geräte** (MXE, IPX, IX) auf demselben Netzwerk in den **Collected Error State** ingegriert werden. Indem man

dies tut, kann das **MXE-Fehlerrelais** dazu verwendet werden, nicht nur einen **Fehler im Gerät**, sondern auch einen **Systemfehler** anzuzeigen.

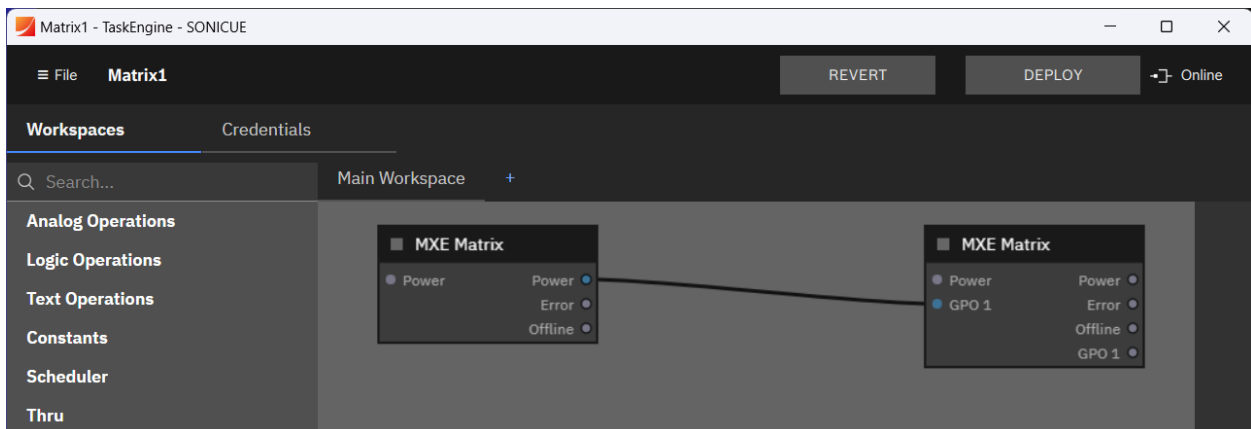


**Bild 50:** MXE-Status-Flyout – REMOTE-Tab.

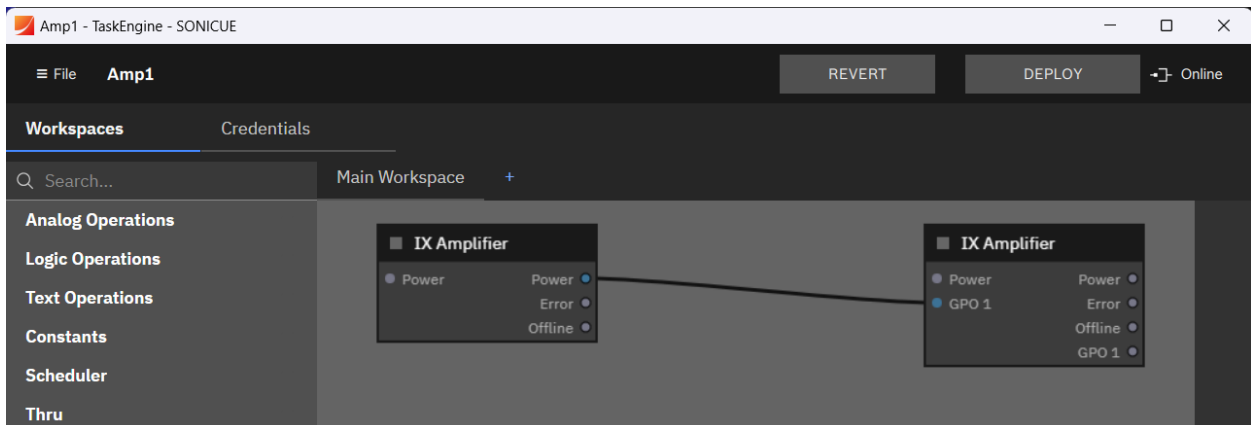
## TaskEngine-Programmierung

Wenn **andere Informationen** als Geräte- oder Systemstatus (Ready/Fault) signalisiert werden sollen, kann ein **GPIO** als **DIGITAL OUT** konfiguriert werden. Die TaskEngine-Logik kann dann entsprechend konfiguriert werden, um den GPIO mit daran angeschlossener LED zu triggern.

Im folgenden Beispiel wird der **Power**-Status des Geräts über **GPO 1** angezeigt.



**Bild 51:** MXE-TaskEngine-Konfiguration zum Anzeigen des *Power*-Status über eine ext. LED, verbunden auf *GPO 1*.



**Bild 52:** IX-TaskEngine-Konfiguration zum Anzeigen des *Power*-Status über eine ext. LED, verbunden auf *GPO 1*.

**Drittanbieter-Produkthaftungsausschluss:**

Dynacord übernimmt keine Verantwortung für die Garantie, Qualität oder Verfügbarkeit von Standard-Elektronikkomponenten (Potentiometer, Widerstände, Relais, LEDs, etc.). Die in diesem Dokument enthaltenen Standard-Elektronikkomponenten wurden zum Zeitpunkt der Veröffentlichung erfolgreich getestet. Jedoch kann Dynacord die Kompatibilität oder Verfügbarkeit solcher Standard-Elektronikkomponenten in der Zukunft nicht garantieren.