

LogicUniPi

Denne QuickApp bruges til at integrerer enheder fra UniPi serien og præsenterer dem som enheder i Home Center 3.

Den åbner for følgende muligheder

- .

Installation

Indlæs QuickApp, Elpriser.

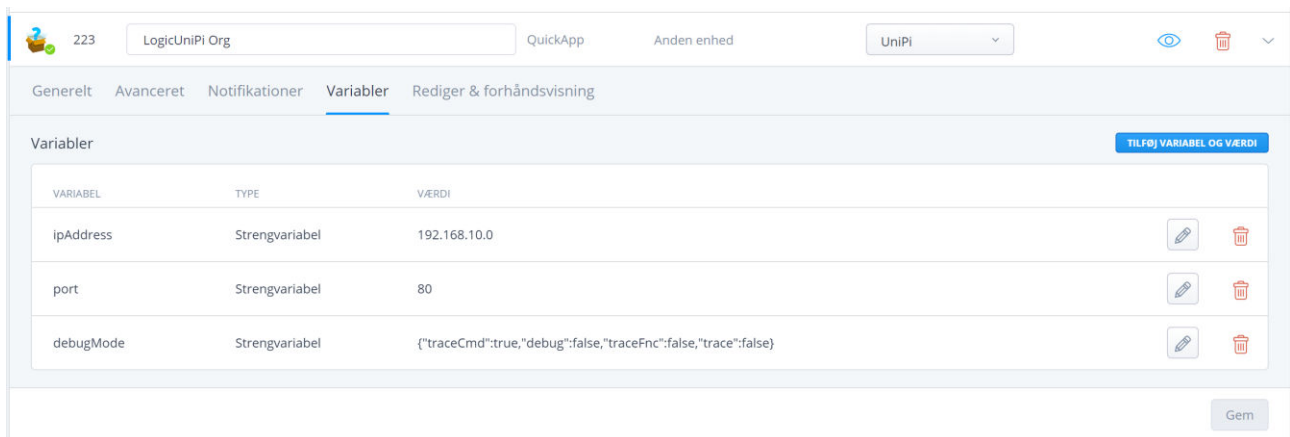
Opret en Lua-scene og tilføj indhold fra denne manual

Variable: Konfiguration

!! Hvis ikke QuickAppen viser de rigtige oplysninger, kan du prøve følgende

Klik på getData.

Klik Rediger/Edit, klik i koden i højre side, klik CTRL+S, klik på Luk/Close



The screenshot shows the configuration interface for the LogicUniPi QuickApp. At the top, there is a header with a user icon, the number '223', the text 'LogicUniPi Org', 'QuickApp', 'Anden enhed', and a dropdown menu set to 'UniPi'. Below the header are navigation tabs: 'Generelt', 'Avanceret', 'Notifikationer', 'Variabler' (which is selected), and 'Rediger & forhåndsvisning'. The main area is titled 'Variabler' and contains a table with three columns: 'VARIABLE', 'TYPE', and 'VÆRDI'. There are three rows of variables, each with edit and delete icons to its right. A blue button 'TILFØJ VARIABLE OG VÆRDI' is located in the top right corner of the table area. At the bottom right, there is a 'Gem' button.

VARIABLE	TYPE	VÆRDI
ipAddress	Strengvariabel	192.168.10.0
port	Strengvariabel	80
debugMode	Strengvariabel	{"traceCmd":true,"debug":false,"traceFnc":false,"trace":false}



1710

LogicUniPi Temp

QuickApp

Anden enhed


Default Room



Generelt Avanceret Notifikationer Variabler **Rediger & forhåndsvisning**

Enhedsforhåndsvisning

Logic Group UniPi devices manager for



QuickApp: **Logic Group**
 sw: 0.9.7
 Manual: **Klik her**
 Netværk: nil
 Status: **Forbundet**

Find enheder Slet alle enhe...

List of devices

Genstart QuickApp

Klik på knappen nedenfor for at åbne redigeringsstilstand. Oprettet eller ændret enhedsvisning vises til venstre.

Rediger

Gem

SONOS Kontor2 7%	SONOS Olivia 20%	?	?	LogicUniPi Temp	input - 11 Aldrig overtrådt	input - 10 Aldrig overtrådt	input - 12 Aldrig overtrådt	input - 1 Aldrig overtrådt	input - 3 Aldrig overtrådt	input - 2 Aldrig overtrådt
input - 5 Aldrig overtrådt	input - 4 Aldrig overtrådt	input - 7 Aldrig overtrådt	input - 6 Aldrig overtrådt	input - 9 Aldrig overtrådt	input - 8 Aldrig overtrådt	relay - 1	relay - 3	relay - 2	relay - 5	
relay - 4	relay - 7	relay - 6	relay - 8	0.00 ai - 1	0.00 ai - 2	ao - 1 Stukket	39.55 ° temp - 2801844B...	39.40 ° temp - 2803D44...	49.45 ° temp - 283C934C...	
57.95 ° temp - 28B9C84C...	55.50 ° temp - 280A855F...	35.25 ° temp - 28A5265F...	53.95 ° temp - 283E7C4C...	51.55 ° temp - 28238D4...	39.25 ° temp - 28631B5F...	46.90 ° temp - 281F2060...	40.75 ° temp - 28A21B5F...	54.40 ° temp - 28E47D4...	53 ° temp - 28733060...	
35.65 ° temp - 2828165F...	39.15 ° temp - 28FB575F...	36.75 ° temp - 28815A4C...	62.80 ° temp - 28498C4C...	44.15 ° temp - 28874360...	LogicUniPi Dier/Vi...	i-2_29:ubruset 210.20N Siden	i-2_28:ubruset 210.20N Siden	i-2_27:ubruset 210.20N Siden	i-2_20:Vindue V 110.17N Siden	

Variabler

TILFØJ VARIABEL OG VÆRDI

VARIABEL	TYPE	VÆRDI		
className	Strengvariabel	uniPi_Input		
uniPiType	Strengvariabel	input		
lastName	Strengvariabel	input - 1_01		
invert	Strengvariabel	1		
uniPICircuit	Strengvariabel	1_01		

Gem

Lua-Scene Elpriser

Du kan lave en scene, som gør det let, at knytte Matrix taster til visning af elpriser.

Den venstre del af scenen er kriteriet for, hvornår scenen skal kaldes.

Denne scene tjekker prisen hver 5 minut

```
{
  conditions = { {
    isTrigger = true,
    operator = "matchInterval",
    property = "cron",
    type = "date",
    value = {
      date = { "00", "12", "22", "5", "*", "2022" },
      interval = 300
    }
  } },
  operator = "all"
}
```

Den højre del af scenen er den der vælger hvilke taster, som skal styre og hvilken enhed der skal styres.

I følgende eksempel er de to Matrix'er lagt ind, til styring af 3 Hue enheder

```
-- Angiv en eller flere taster, som skal vise farven for prisen
local dayIds = {} -- taster, som viser prisen sammenlignet med i dag
local dayPIds = {{},{},{}} -- taster, som viser prisen +1, +2 og +3 timer
local weekIds = {} -- taster, som viser prisen sammenlignet med sidste uge
local priceIds = {} -- taster, som viser prisen i forhold til Low og High

-- skal ikke ændres
local qa = (api.get("/devices?name=Elpriser") or {})[1];
local qaId = qa.id or 0;
if (qaId == nil or qaId == 0) then return; end

function fibaro.getQAVariable(id,name)
  local props = (api.get("/devices/"..(id or MID)) or {}).properties or {}
  for _, v in ipairs(props.quickAppVariables or {}) do
    if v.name==name then return v.value end
  end
end
```

```

    end
end

function fibaro.setQAVariable(id,name,value)
    return fibaro.call(id,"setVariable",name,value)
end

local cDkk = fibaro.getQAVariable(qaId,"currentPrice");
local cDkkPct = fibaro.getQAVariable(qaId,"currentPricePct");
local dType = fibaro.getQAVariable(qaId,"currentTypeDay");
local cDkkP1 = fibaro.getQAVariable(qaId,"currentPriceP1");
local dTypeP1 = fibaro.getQAVariable(qaId,"currentTypeP1Day");
local cDkkP2 = fibaro.getQAVariable(qaId,"currentPriceP2");
local dTypeP2 = fibaro.getQAVariable(qaId,"currentTypeP2Day");
local cDkkP3 = fibaro.getQAVariable(qaId,"currentPriceP3");
local dTypeP3 = fibaro.getQAVariable(qaId,"currentTypeP3Day");
local wType = fibaro.getQAVariable(qaId,"currentTypeWeek");

function calcRandG(_type)
    local r = 0;
    local g = 0;
    local colType = 64;
    if (_type >= 70) then
        g = 0;
        r = (192 + (63 * ((_type - 70) / 30)));
    elseif (_type < 0) then
        colType = 138;
        g = 64;
        r = 64;
    elseif (_type <= 30) then
        g = (192 + (63 * ((30 - _type) / 30)));
        r = 0;
    else
        g = (192 - (192 * ((_type - 30) / 40)));
        r = (192 - (192 * ((40 - (_type - 30)) / 40)));
    end
    return math.floor(r),math.floor(g);
end

function showColOnButton(_dkk,_Ids,_Type)
    print("Elpriser: _ids:" .. json.encode(_Ids) .. ", _Type:" .. (_Type or 'nil') ..
        ", _dkk:" .. (_dkk or 'nil'));
    if (type(_Ids) == "number") then _Ids = {_Ids}; end
    if (_Ids ~= nil and #_Ids > 0 and _Type ~= nil) then

```

```

    local r,g = calcRandG(_Type);
    print("_Type r,g:" .. r .. ", " .. g)
    hub.call(_Ids, 'setColor', r, g, 0, 64);
end
end

print("Elpriser: dkk:" .. cDkk .. ", dtype:" .. dtype .. ", wType:" .. wType);
showColOnButton(cDkk,dayIds,dType);
showColOnButton(cDkkP1,dayPIds[1],dtypeP1);
showColOnButton(cDkkP2,dayPIds[2],dtypeP2);
showColOnButton(cDkkP3,dayPIds[3],dtypeP3);
showColOnButton(cDkk,weekIds,wType);
showColOnButton(cDkkPct,priceIds,dType);

```

Den første 3 linje skal rettes, så id'erne for tasterne er angivet

```

local dayIds = {}           -- taster, som viser prisen sammenlignet med i dag
local dayPIds = {{},{},{}} -- taster, som viser prisen +1, +2 og +3 timer
local weekIds = {}        -- taster, som viser prisen sammenlignet med sidste uge

```

```
dayIds = {65}
```

Dette kunne være en opsætning, hvor 65 er id på en tast, der ønskes at lyse med aktuelle pris, sammenlignet med de 48 timer.

```
dayPIds = { 67, 69, 71}
```

Dette ville få tast med id 67 til at lyse med prisen om en time, sammenlignet med de 48 timer.

Tast med id 69 lyser med prisen om to time, sammenlignet med de 48 timer.

Tast med id 71 lyser med prisen om tre time, sammenlignet med de 48 timer.

```
weekIds = {77,81}
```

Dette ville være en opsætning, hvor 77 og 81 er id'er på taster, der ønskes at lyse med aktuelle pris, sammenlignet med sidste uge.