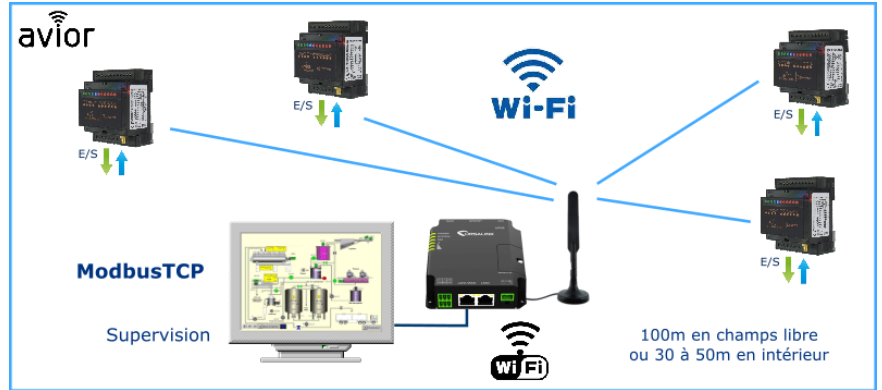


SCHEMA DE PRINCIPE

Nous allons présenter ici une application où l'on va utiliser l'AVIOR comme modules d'E/S déporté sans fil. Un exemple d'application serait de surveiller un parc de machines dans un atelier pour faire du suivi de production, incidents et temps de marche.

Nous allons ici utiliser un Ursalink UR-32 comme point d'accès WiFi relié à notre PC de supervision. Ensuite nous configurerons nos AVIOR pour pouvoir accéder à l'état des E/S en ModbusTCP.



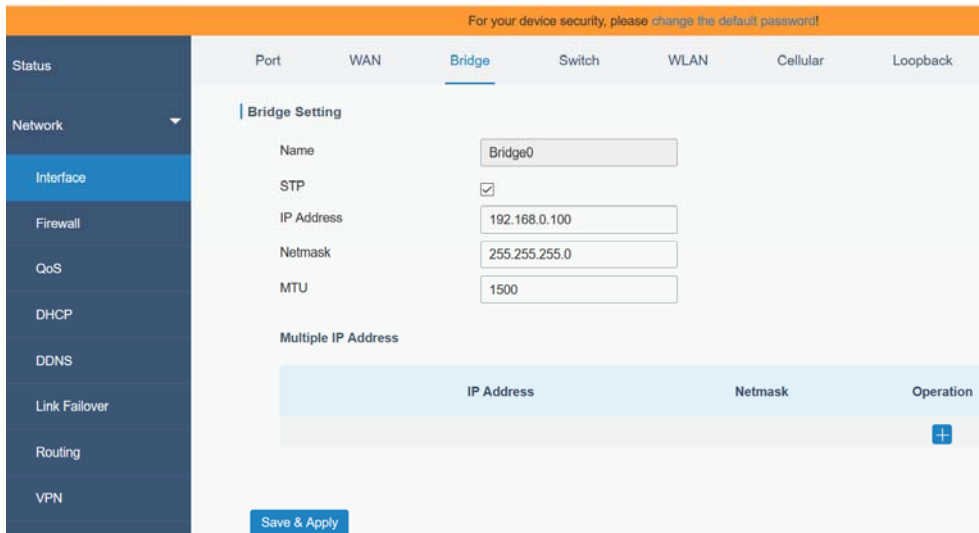
Configuration du point d'Accès WiFi

Le routeur URSALINK UR32-W peut être utilisé comme point d'accès WiFi très simplement. C'est une solution industrielle, fiable et économique pour ce type d'application.

L'adresse IP par défaut du UR32 est 192.168.1.1, login : admin et mot de passe : password. Tout d'abord on configure l'accès WAN ou LAN en fonction du besoin.



Nous configurons les 2 ports Ethernet en LAN.



Dans notre cas nous allons configurer le routeur pour un réseau en 192.168.0.100

Port WAN Bridge Switch **WLAN** Cellular Loopback

WLAN

Enable

Work Mode

BSSID

Radio Type

Channel

Bandwidth

SSID

Encryption Mode

Cipher

Key

SSID Broadcast

AP Isolation

Guest Mode

Max Client Number

Save & Apply

Ensuite on configure le routeur en point d'accès WiFi. Dans notre cas, le réseau WiFi sera nommé ursalink.

DHCP Server DHCP Relay

DHCP Server_1

Enable

Interface

Start Address

End Address

Netmask

Lease Time(Min)

Primary DNS Server

Secondary DNS Server

Windows Name Server

Static IP

MAC Address	IP Address	Operation
<input type="text" value="24:0a:c4:bf:85:5c"/>	<input type="text" value="192.168.0.200"/>	<input type="button" value="✕"/>
<input type="text" value="24:0a:c4:bf:85:1a"/>	<input type="text" value="192.168.0.201"/>	<input type="button" value="✕"/>
<input type="text" value="24:0a:c4:bf:82:4e"/>	<input type="text" value="192.168.0.202"/>	<input type="button" value="✕"/>
<input type="text" value="24:0a:c4:bf:81:2f"/>	<input type="text" value="192.168.0.204"/>	<input type="button" value="✕"/>
		<input type="button" value="⊕"/>

Save

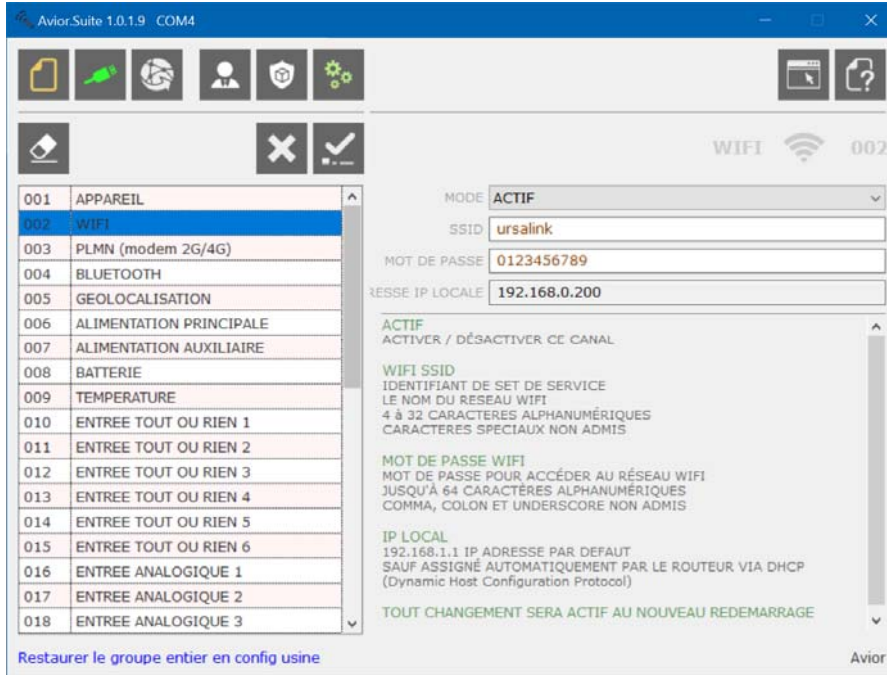
Voilà pour la partie point d'accès.

Maintenant nous allons définir l'affectation des adresses IP fixe en fonction des mac Adresses des AVIOR, pour pouvoir les reconnaître et y accéder depuis notre superviseur.

Sur la figure ci-contre nous associons l'adresse Mac de chaque AVIOR à une adresse IP de notre réseau.

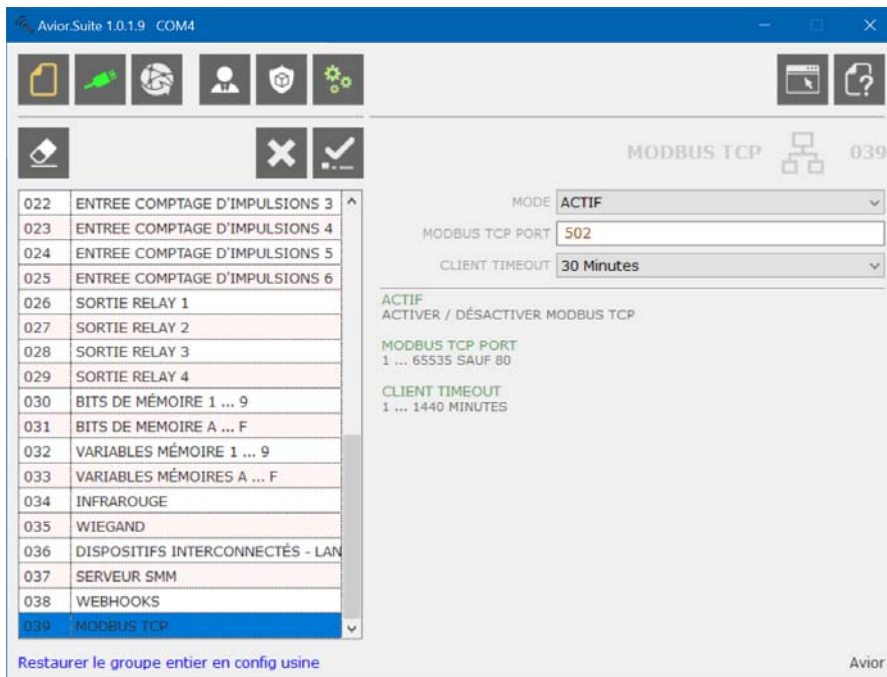
Configuration de l'AVIOR

A l'aide du logiciel AVIOR.Suite nous allons vérifier que l'accès via WiFi et et ModbusTCP soit bien activé



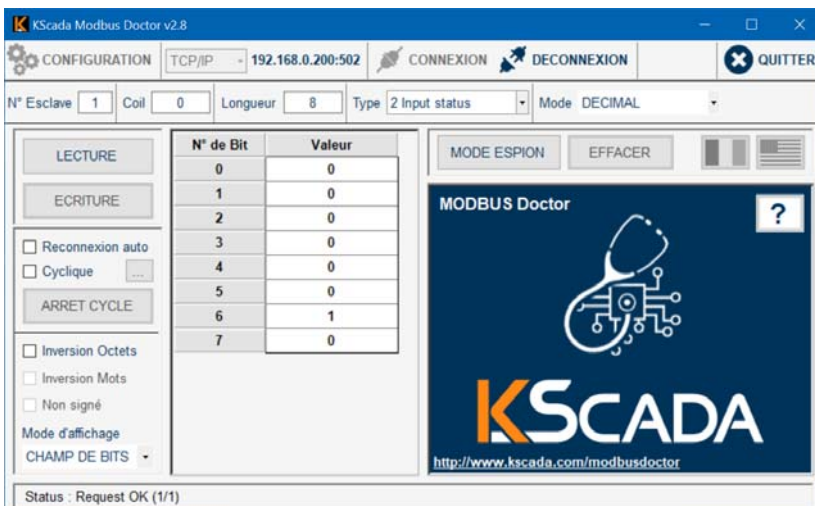
On renseigne le SSID du réseau WiFi ainsi que le mot de passe (Key WPA2)

On pourra ensuite, après validation et redémarrage de l'AVIOR de vérifier que l'adresse IP voulue à bien été associé à notre appareil.



Il faut activer le mode ModbusTCP

Nous allons ensuite utiliser un utilitaire ModbusTCP Master, comme Modbus Doctor ici, pour vérifier que nous arrivons bien à nous connecter à l'AVIOR.



Dans l'exemple ci-contre nous allons lire l'état des 8 bits d'entrées, qui regroupent les 6 entrées digitales ainsi que l'état de l'alimentation principale et l'auxiliaire.

Vous pouvez vous reporter au tableau suivant pour connaître l'intégralité des données qui sont accessibles dans l'AVIOR.

Table Annexe des Données ModbusTCP

PHYSICAL DISCRETE INPUTS: Read

Function Code	# Coils (dec)	Modbus ADR hex	Digital Inputs	AVIOR Tag	Long
02	0000	0x 00	1	I12	1 Bit
02	0001	0x 01	2	I22	
02	0002	0x 02	3	I32	
02	0003	0x 03	4	I42	
02	0004	0x 04	5	I52	
02	0005	0x 05	6	I62	
02	0006	0x 06	Main	I72	
02	0007	0x 07	Aux	I82	

COILS: Read/Write Discrete output

Function Code	# Coils (dec)	Modbus ADR hex	Outputs	AVIOR Tag	Long
01 05	0100	0x 64	1	O12	1 Bit
01 05	0101	0x 65	2	O22	
01 05	0102	0x 66	3	O32	
01 05	0103	0x 67	4	O42	

COILS: Read/Discrete output or coils

Function Code	# Coils (dec)	Modbus ADR hex	Memory bits	AVIOR Tag	Long
01 05	0200	0x C8	1	M11	1 Bit
01 05	0201	0x C9	2	M21	
01 05	0202	0x CA	3	M31	
01 05	0203	0x CB	4	M41	
01 05	0204	0x CC	5	M51	
01 05	0205	0x CD	6	M61	
01 05	0206	0x CE	7	M71	
01 05	0207	0x CF	8	M81	
01 05	0208	0x D0	9	M91	
01 05	0209	0x D1	A	MA1	
01 05	0210	0x D2	B	MB1	
01 05	0211	0x D3	C	MC1	
01 05	0212	0x D4	D	MD1	
01 05	0213	0x D5	E	ME1	
01 05	0214	0x D6	F	MF1	

REGISTER: Read Input Registers

Function Code	# Register (dec)	Modbus ADR hex	Analog Inputs	AVIOR Tag	Long
04	0000	0x 00	1	A19	2 Words 32 Bits
04	0002	0x 02	2	A29	
04	0004	0x 04	3	A39	
04	0006	0x 06	4	A49	
04	0008	0x 08	Batt	A52	
04	0010	0x 0A	Temp	A62	

REGISTER: Read/Write Output or Holding Registers

Function Code	# Register (dec)	Modbus ADR hex	Word Device (CW)	AVIOR Tag	Long
03 06	0100	0x 64	P1	P12	2 Words 32 Bits
03 06	0102	0x 66	P2	P22	
03 06	0104	0x 68	P3	P32	
03 06	0106	0x 6A	P4	P42	
03 06	0108	0x 6C	P5	P52	
03 06	0110	0x 6E	P6	P62	

REGISTER: Read/Write Output or Holding Registers

03 06	0200	0x C8	T1	I13	2 Words 32 Bits
03 06	0202	0x CA	T2	I23	
03 06	0204	0x CC	T3	I33	
03 06	0206	0x CE	T4	I43	
03 06	0208	0x D0	T5	I53	
03 06	0210	0x D2	T6	I63	

REGISTER: Read/Write Output or Holding Registers

03 06	0300	0x 012C	var1	V11	2 Words 32 Bits
03 06	0302	0x 012E	var2	V21	
03 06	0304	0x 0130	var3	V31	
03 06	0306	0x 0132	var4	V41	
03 06	0308	0x 0134	var5	V51	
03 06	0310	0x 0136	var6	V61	
03 06	0312	0x 0138	var7	V71	
03 06	0314	0x 013A	var8	V81	
03 06	0316	0x 013C	var9	V91	
03 06	0318	0x 013E	varA	VA1	
03 06	0320	0x 0140	varB	VB1	
03 06	0322	0x 0142	varC	VC1	
03 06	0324	0x 0144	varD	VD1	
03 06	0326	0x 0146	varE	VE1	
03 06	0328	0x 0148	varF	VF1	



Rémy GUÉDOT

Gsm: +33 (0) 662 80 65 57
guedot@rg2i.fr

Olivier BENAS

Gsm: +33 (0) 666 84 26 26
olivier.benas@rg2i.fr

ATTENTION - NOUVELLE ADRESSE

14 rue Edouard Petit
F42000 Saint Etienne

Tél: +33 (0) 477 92 03 56

Fax: +33 (0) 477 92 03 57

www.rg2i.fr