

## AR-727-CM - IO

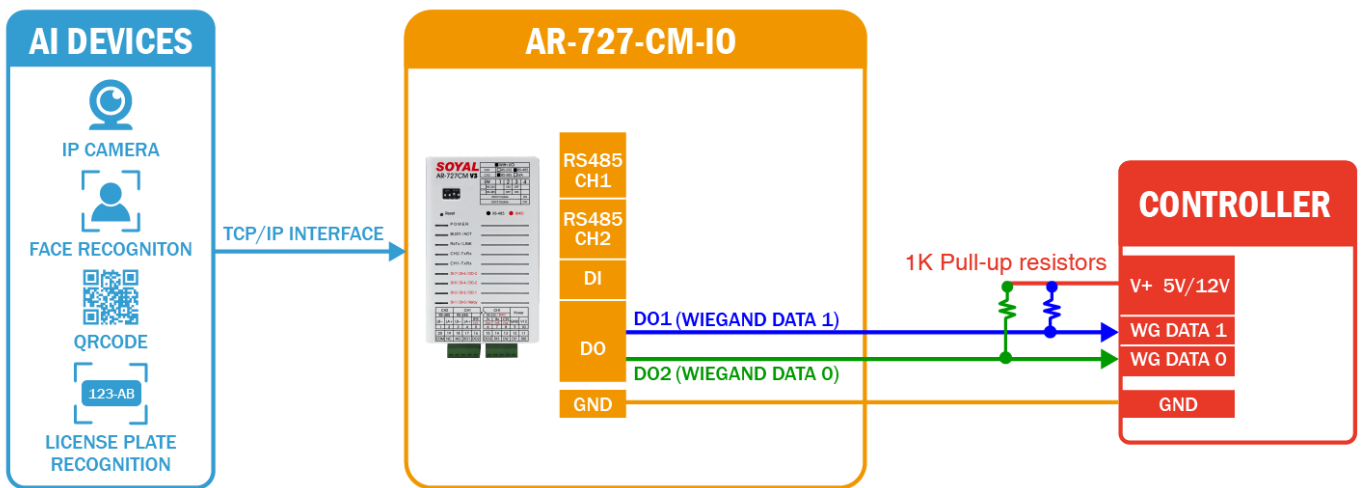
### Modbus Gateway (TCP-RTU) / TCP to Wiegand Converter / Built-in 8DI 1RO 3DO

SOYAL AR-727CM-IO provides multiple credential and authorization with RFID possible. AI devices such as License Plate Recognition, IP Camera, QRCODE, or even Face Recognition device that built-in TCP/IP interface could easily converted to Wiegand interface. The conversion will allows AI devices integrate with SOYAL RFID access controller under one system.

#### APPLICABLE TO:

AR-727CM-IO-0804M with SOYAL Enterprise Series, SOYAL Home Series, SOYAL Control Panel Series

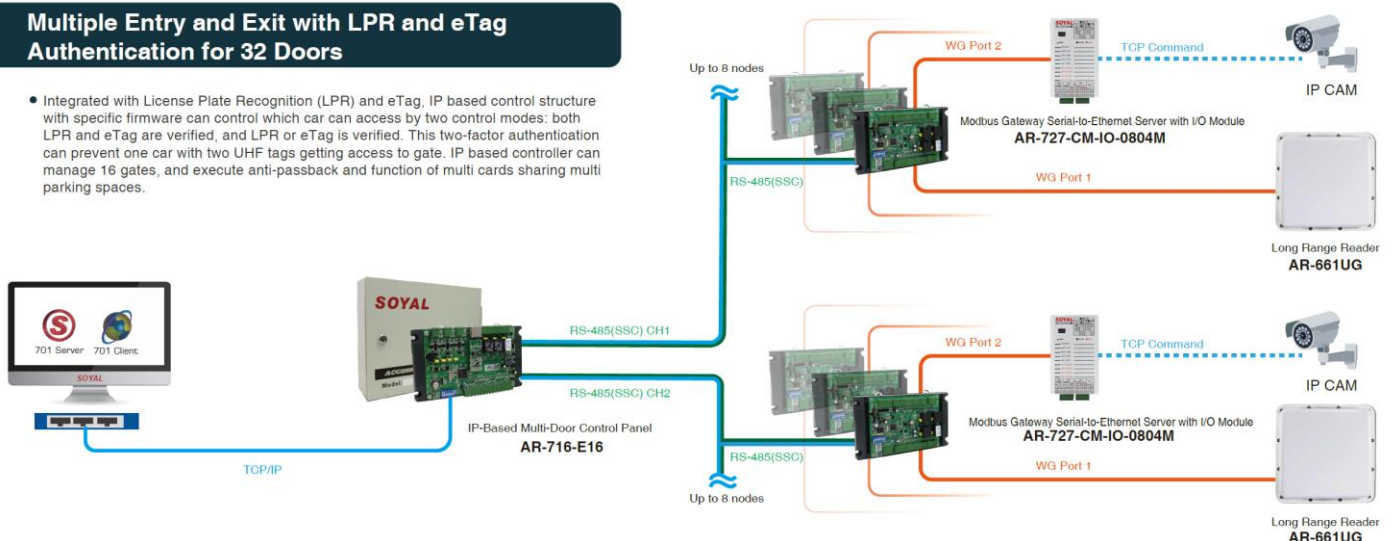
#### SCHEMATIC DIAGRAM:



#### APPLICATION DIAGRAM:

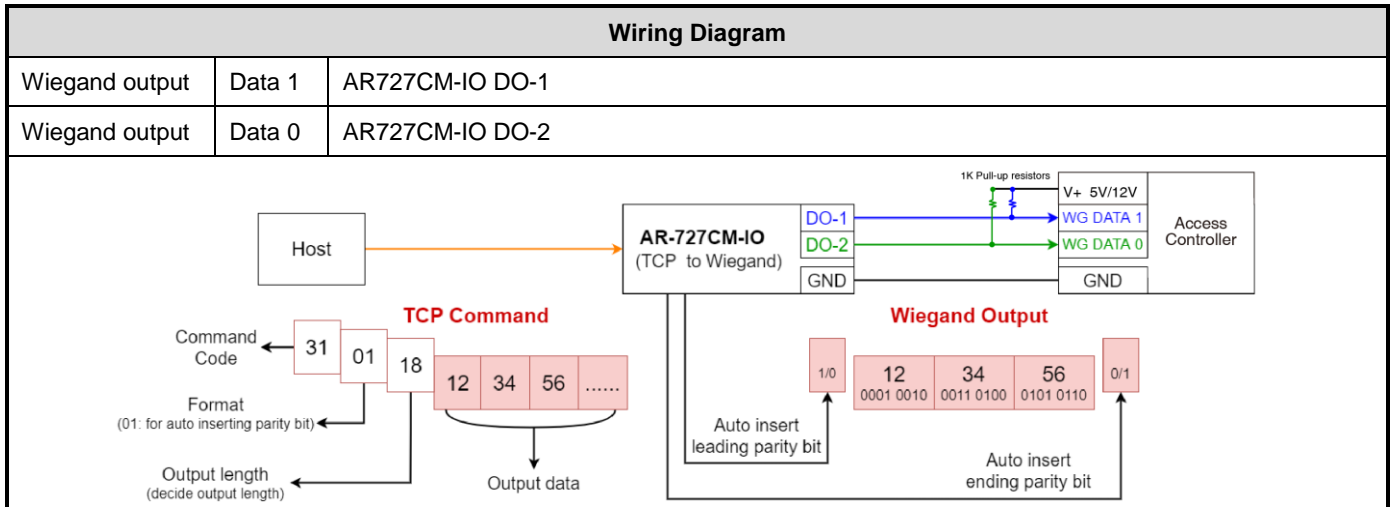
##### Multiple Entry and Exit with LPR and eTag Authentication for 32 Doors

- Integrated with License Plate Recognition (LPR) and eTag, IP based control structure with specific firmware can control which car can access by two control modes: both LPR and eTag are verified, and LPR or eTag is verified. This two-factor authentication can prevent one car with two UHF tags getting access to gate. IP based controller can manage 16 gates, and execute anti-passback and function of multi cards sharing multi parking spaces.



## IMPLEMENTATION STEPS:

### Wiegand output with changeable data length (Simple format): 0x31



Format Explanation		
TCP Port	1601	Output data bit length is changeable depending on demand, supporting 16~96 bits
Command code	0x31	Wiegand output command Output Format: 0x00: Just bypass the data bits through WG1/0 pin 0x01: Auto insert leading/ending parity bit into the bypass data bits on even/odd format 0x02: Auto insert leading/ending parity bit into the bypass data bits on odd/even format

Example					
Command Package					
Command	Format	Auto add parity bit	Data bit length (changeable, support 16~96 bits)	Data bit (hex)	Pass output data bits (bin)
0x31	0x00	None	0x20 / 32dec (actually send 0x20 / 32bits)	12 34 56 78	<b>0001 0010 0011 0100 0101 0110 0111 1000</b>
0x31	0x01	Even / Odd	0x20 / 32dec (actually send 0x22 / 34bits)	12 34 56 78	<b><u>1</u> 0001 0010 0011 0100 0101 0110 0111 1000 <u>1</u></b>
0x31	0x01	Even / Odd	0x18 / 24dec (actually send 0x1A / 26bits)	12 34 56	<b><u>0</u> 0001 0010 0011 0100 0101 0110 <u>0</u></b>
0x31	0x02	Odd / Even	0x18 / 24dec (actually send 0x1A / 26bits)	12 34 56	<b><u>1</u> 0001 0010 0011 0100 0101 0110 <u>1</u></b>

Echo code		
	0x40	COMMAND_RET_SUCCESS
	0x41	COMMAND_RET_UNKNOWN_CMD
	0x42	COMMAND_RET_INVALID_CMD
	0x43	COMMAND_RET_INVALID_ADR

More Examples
<ul style="list-style-type: none"> <li>AR-727-CM sends 96 bits data with parity bit format: <b>31 01 60 10 11 22 33 44 55 66 77 88 99 AA BB</b></li> </ul> Pass output data bits: <b><u>1</u> 0001 0000 0001 0001 0010 0010 0011 0011 0100 0100 0101 0101 0110 0110 0111 0111 1000 1000 1001 1001 1010 1010 1011 1011 <u>0</u></b>