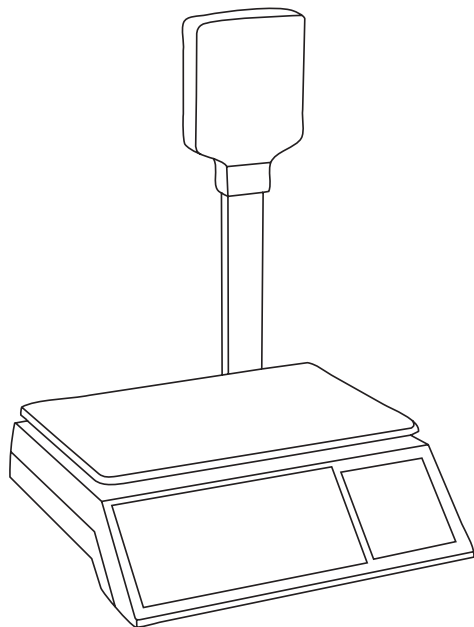


GRAM ▼

SERIES / SERIE / SÉRIE / SERIES

ZFOC-RS



EN

ES

FR

DE

CE

**COMMUNICATION PROTOCOL
PROTOCOLO DE COMUNICACIÓN
PROTOCOLE DE COMMUNICATION
KOMMUNIKATIONS PROTOKOLL**

ENGLISH

COMMUNICATION PROTOCOL (RS Model)

1. **Communication interface:** RS232c, baud rate 9600, 8 digits, no parity, 1 start bit, 1 stop bit.
2. Unit price setting, PLU downloading, PLU uploading, unit price & total price reading, communication process is as follows:
 - PC sends 0X44.
 - PC sends start package after receiving scale response 0x02
 - PC sends command package after receiving scale response 0x02.
 - After PC receiving scale response 0x02 & command response package, if needs to send command, switch to c.
 - PC sends end package.
 - PC terminates after receiving scale response 0x02.

Start package, fixed 6 bytes: 11 00 00 00 00 EF5

End package, fixed 6 bytes: 33 00 00 00 00 CD

Command package command response package, format is as follows:

Cmd type adr1 adr0 datlen [dat] checksum

cmd: Only 55 read command or 77 write command.

Type: Operating data type is as follows:

f9: Unit price or PLU

f4: Total price

adr1,adr0: Form 2 bytes integer $adr=adr1*100+adr0$

When adr is 0, stands for operating current unit price or total price

When adr is not 0, stands for operating memory area PLU, at this time adr is PLU's position.

adr1 = $(DC+pluN*4)/100$

adr0 = $(DC+pluN*4)\%100$

datlen: Data length of reading or writing. Data length of PLU is 4

[dat]: The data of writing command, length is decided by datlen

checksum: checksum, checksum = $100 -$

$(cmd+type+dar1+dar0+datalen+[dat])\%100$

If set current unit price 111.00:

```
DO 44
DI 02
DO 11 00 00 00 00 ef
DI 02
DO 77 f9 00 00 04 00 00 2b 5c 01
DI 02
DO 33 00 00 00 00 cd
DI 02
```

If set unit price of PLU1 111.00:

```
DO 44
DI 02
DO 11 00 00 00 00 ef
DI 02
DO 77 f9 00 e0 04 00 00 2b 5c 21
DI 02
DO 33 00 00 00 00 cd
DI 02
```

If read PLU1: DO 44

```
DI 02
DO 11 00 00 00 00 ef
DI 02
DO 55 f9 00 e0 04 ce
DI 02
DI 55 fd 00 e0 04 00 00 2b 5c 43 ;;
DO 33 00 00 00 00 cd
DI 02
```

If read current unit price & total price: DO 44

```
DI 02
DO 11 00 00 00 00 ef
DI 02
DO 55 f4 00 00 09 ae
DI 02
DI 55 f4 00 00 04
DI 00 00 00 00 de ;; total price 2.22
DI 00 00 2b 5c ;; unit price 111.00
DI 4e
DO 33 00 00 00 00 cd
DI 02
```

3. Read current weight

PC sends 0X05, after receiving scale response 0x06, senfs 0x11, scale will response weight package.

Weight Package formats: SOH STX STA SIGN WEIGHT_ASCII WEIGHT_UNIT
BCC ETX EOT

SOH. Start to sending symbol, one byte, fixed at 01H.

STX. Start symbol of data item, one byte, fixed at 02H.

STA. Words to describe weight status, one byte, only possible to be 53H or 55H46H. 53H –weight is stable, 55H – weight is unstable, 46H –weight is abnormal.

SIGN. Sign of weight, one byte, only possible to be 2DH or 20H. 2DH-weight is negative, 20H-weight is positive.

WEIGHT_ASCII. Character string of weight. 5~6 byte, only possible to be 30H~39H, decimal (2EH), space (20H)

WEIGHT_UNIT. Character string of weight unit. 1~2 byte, only possible to be: 'TJ': catty of Taiwan, 'TL': tael of Taiwan, 'SJ': 'SJ': jin, 'LB': pound, 'KG': kilogram, 'G': gram.

BCC. Character of data verify. One byte. Set STA to D1, SIGN to D2, byte before BCC to DN, so $BCC = D1 \wedge D2 \wedge \dots \wedge DN$. '^' is a logic XOR operator.

ETX. Symbol of ending data item, one byte, fixed at 03H.

EOT. Symbol of ending sending, one byte, fixed at 04H.

PROTOCOLO DE COMUNICACIÓN (Modelo RS)

1. **Interfaz de comunicación:** RS232c, velocidad en baudios 9600, 8 dígitos, sin paridad, 1 bit de arranque, 1 bit de parada.
2. Configuración de precio unitario, descarga de PLU, carga de PLU, lectura de precio unitario y de precio total. El proceso de comunicación es el siguiente:
 - El PC transmite 0X44.
 - El PC transmite el paquete de inicio una vez recibida la respuesta de la balanza 0x02.
 - El PC transmite el paquete de comando una vez recibida la respuesta de la balanza 0x02.
 - Después de que el PC recibe la respuesta de la balanza 0x02 y el paquete de respuesta de comando, si necesita enviar un comando, cambie a c.
 - El PC transmite el paquete final.
 - El PC finaliza después de recibir la respuesta de la balanza 0x02.

Paquete de inicio, fijo 6 bytes: 11 00 00 00 00 EF5.

Paquete final, fijo 6 bytes: 33 00 00 00 00 CD.

Paquete de comando, paquete de respuesta de comando, el formato es el siguiente:

Cmd type adr1 adr0 datlen [dat] checksum.

cmd: Solo 55 comandos de lectura o 77 comandos de escritura.

Type: El tipo de datos operativos es el siguiente:

f9: Precio unitario o PLU.

f4: Precio total.

adr1,adr0: Forma 2 bytes enteros $adr=adr1*100+adr0$.

Si adr es 0, significa precio unitario actual operativo o precio total.

Si adr es distinto de 0, significa área de memoria operativa PLU, en ese momento adr es la posición de PLU.

adr1 = $(DC+pluN*4)/100$

adr0 = $(DC+pluN*4)\%100$

datlen: Longitud de los datos de lectura o escritura. La longitud de los datos de PLU es 4.

[dat]: Los datos del comando de escritura, la longitud la decide datlen

checksum: checksum, checksum = $100 - (cmd+type+dar1+dar0+datalen+[dat])\%100$.

Si se fija el precio unitario actual 111,00:

DO	44
DI	02
DO	11 00 00 00 00 ef
DI	02
DO	77 f9 00 00 04 00 00 2b 5c 01
DI	02
DO	33 00 00 00 00 cd
DI	02

Si se fija un precio unitario de PLU1 111,00:

DO 44
DI 02
DO 11 00 00 00 00 ef
DI 02
DO 77 f9 00 e0 04 00 00 2b 5c 21
DI 02
DO 33 00 00 00 00 cd
DI 02

Si se lee PLU1: DO 44

DI 02
DO 11 00 00 00 00 ef
DI 02
DO 55 f9 00 e0 04 ce
DI 02
DI 55 fd 00 e0 04 00 00 2b 5c 43 ;;
DO 33 00 00 00 00 cd
DI 02

Si se lee el precio unitario actual y el precio total: DO 44

DI 02
DO 11 00 00 00 00 ef
DI 02
DO 55 f4 00 00 09 ae
DI 02
DI 55 f4 00 00 04
DI 00 00 00 00 de ;; precio total 2,22
DI 00 00 2b 5c ;; precio unitario 111,00
DI 4e
DO 33 00 00 00 00 cd
DI 02

3. Leer peso actual

El PC transmite 0X05 tras recibir la respuesta de la balanza 0x06, transmite 0x11, la balanza responderá al paquete de peso.

Formatos de paquetes de peso: SOH STX STA SIGN WEIGHT_ASCII WEIGHT_UNIT BCC ETX EOT.

SOH. Empieza a transmitir símbolo, un byte, fijado en 01H.

STX. Símbolo de inicio del elemento de datos, un byte, fijado en 02H.

STA. Palabras para describir el estado del peso, un byte, solo puede ser 53H o 55H46H. 53H: el peso es estable; 55H: el peso es inestable; 46H: el peso no es normal.

SIGN. Signo de peso, un byte, solo puede ser 2DH o 20H. El peso 2DH es negativo, el peso 20H es positivo.

WEIGHT_ASCII. Cadena de caracteres de peso. 5~6 bytes, solo puede ser 30H~39H, decimal (2EH), espacio (20H).

WEIGHT_UNIT. Cadena de caracteres de unidad de peso. 1~2 bytes, solo puede ser: 'TJ': medida «catty» de Taiwán, 'TL': tael de Taiwán, 'SJ': 'SJ': jin, 'LB': libra, 'KG': kilogramo, 'G': gramo.

BCC. Verificar carácter de datos. Un byte. Definir STA en D1, SIGN en D2, byte antes de BCC en DN, así como $BCC=D1^D2^...^DN$. '^' es un operador XOR lógico.

ETX. Símbolo de fin del elemento de datos, un byte, fijado en 03H.

EOT. Símbolo de fin de envío, un byte, fijado en 04H.

FRANÇAIS

PROTOCOLE DE COMMUNICATION (Modèle RS)

1. **Interface de communication** : RS232c, débit en bauds 9600, 8 chiffres, sans parité, 1 bit de démarrage, 1 bit d'arrêt.
2. Fixation du prix unitaire, téléchargement PLU, affichage du prix unitaire et du prix total, le processus de communication est le suivant :
 - PC envoie 0X44.
 - PC envoie le package de début après réception de la réponse à l'échelle 0x02.
 - PC envoie le package de commande après réception de la réponse à l'échelle 0x02.
 - Après réception par le PC de la réponse à l'échelle 0x02 et de la réponse du package de commande, il doit envoyer la commande, passer à c.
 - PC envoie le package de fin.
 - PC termine après réception de la réponse à l'échelle 0x02.

Package de début, 6 octets fixes : 11 00 00 00 00 EF5.

Package de fin, 6 octets fixes : 33 00 00 00 00 CD.

Package de commande, package de réponse commande, le format est le suivant :

Cmd type adr1 adr0 datlen [dat] checksum.

cmd : Uniquement commande READ 55 ou commande WRITE 77.

Type : Le type de données de fonctionnement est le suivant :

f9 : Prix unitaire ou PLU.

f4 : Prix total.

adr1,adr0 : Forme 2 octets entiers $adr=adr1*100+adr0$.

Lorsque adr est 0, cela signifie prix unitaire actuel ou prix total de fonctionnement.

Lorsque adr n'est pas 0, cela signifie zone de mémoire de fonctionnement PLU, au moment où adr est en position PLU.

adr1 = $(DC+pluN*4)/100$.

adr0 = $(DC+pluN*4)\%100$.

datlen : Longueur des données d'affichage ou d'écriture. La longueur des données de PLU est 4.

[dat] : Les données de commande de l'écriture, la longueur est décidée par datlen.

checksum : checksum, checksum = 100-

$(cmd+type+dar1+dar0+datalen+[dat])\%100$

S'il est établi le prix unitaire actuel de 111.00 :

```
DO    44
DI    02
DO    11 00 00 00 00 ef
DI    02
DO    77 f9 00 00 04 00 00 2b 5c 01
DI    02
DO    33 00 00 00 00 cd
DI    02
```

S'il est établi un prix unitaire de PLU1 111.00 :

```
DO    44
DI    02
```

DO 11 00 00 00 00 ef
DI 02
DO 77 f9 00 e0 04 00 00 2b 5c 21
DI 02
DO 33 00 00 00 00 cd
DI 02

S'il est affiché PLU1 : DO 44

DI 02
DO 11 00 00 00 00 ef
DI 02
DO 55 f9 00 e0 04 ce
DI 02
DI 55 fd 00 e0 04 00 00 2b 5c 43 ;;
DO 33 00 00 00 00 cd
DI 02

S'il est affiché le prix unitaire actuel et le prix total : DO 44

DI 02
DO 11 00 00 00 00 ef
DI 02
DO 55 f4 00 00 09 ae
DI 02
DI 55 f4 00 00 04
DI 00 00 00 00 de ;; prix total 2.22
DI 00 00 2b 5c ;; prix unitaire 111.00
DI 4e
DO 33 00 00 00 00 cd
DI 02

3. Afficher le poids actuel.

PC envoie 0X05, après réception de la réponse à l'échelle 0x06, envoie 0x11, l'échelle répondra le poids du package.

Formats de poids d'emballage : SOH STX STA SIGN WEIGHT_ASCII WEIGHT_UNIT BCC ETX EOT.

SOH. Lancer le symbole d'envoi, un octet, fixé à 01H.

STX. Symbole de commencement de la donnée, un octet, fixée à 02H.

STA. Termes pour décrire le statut pondéral, un octet, uniquement possible avec 53H ou 55H/46H. 53H –le poids est stable, 55H – le poids est instable, 46H – le poids est anormal.

SIGN. Signe du poids, un octet, uniquement possible avec 2DH ou 20H. 2DH-le poids est négatif, 20H-le poids est positif.

WEIGHT_ASCII. Chaîne de caractères du poids. 5~6 octet, uniquement possible avec 30H~39H, décimale (2EH), espace (20H).

WEIGHT_UNIT. Chaîne de caractères de l'unité de poids. 1~2 octets, uniquement possible avec : 'TJ': catty de Taiwan, 'TL': taël de Taiwan, 'SJ': 'SJ': Jin, 'LB': livre, 'KG': kilogramme, 'G': gramme.

BCC. Vérifier les caractères des données. Un octet. Établir STA à D1, SIGN à D2, octet avant BCC à DN, soit $BCC = D1 \wedge D2 \wedge \dots \wedge DN$. '^' est un opérateur XOR logique.

ETX. Symbole d'une donnée de fin, un octet, fixé à 03H.

EOT. Symbole d'envoi de fin, un octet, fixé à 04H.

DEUTSCH

KOMMUNIKATIONSPROTOKOLL (Modell RS)

1. **Kommunikationsschnittstelle:** RS232c, Baudrate 9600, 8-stellig, ohne Parität, 1 Startbit, 1 Stoppbit.
2. Festlegung Stückpreis, PLU herunterladen, PLU hochladen, Anzeige Stückpreis und Gesamtpreis. Der Kommunikationsprozess ist wie folgt:
 - Der PC sendet 0X44.
 - Der PC sendet das Startpaket nach Erhalt der Waagenantwort 0x02.
 - Der PC sendet das Befehlspaket nach Erhalt der Waagenantwort 0x02.
 - Nachdem der PC die Waagenantwort 0x02 und das Befehls-/Antwortpaket erhalten hat, muss auf „c“ gewechselt werden, falls ein Befehl gesendet werden muss.
 - Der PC sendet das Endpaket.
 - Der PC beendet den Vorgang nach Erhalt der Waagenantwort 0x02.

Startpaket, auf 6 Byte festgelegt: 11 00 00 00 00 EF5.

Endpaket, auf 6 Byte festgelegt: 33 00 00 00 00 CD.

Das Befehlspaket und das Befehlsantwortpaket haben das folgende Format:

Cmd type adr1 adr0 datlen [dat] checksum.

cmd: Nur 55 Lesebefehl oder 77 Schreibbefehl.

Type: Der Betriebsdatentyp ist wie folgt:

f9: Stückpreis oder PLU.

f4: Gesamtpreis.

adr1,adr0: Form 2 bytes integer $\text{adr}=\text{adr1}\cdot 100+\text{adr0}$.

Wenn adr 0 ist, steht dies für den aktuellen Betriebsstückpreis oder Gesamtpreis.

Wenn adr nicht 0 ist, steht dies für den Betriebssystembereich PLU. Zu diesem Zeitpunkt ist adr die PLU-Position.

adr1 = $(\text{DC}+\text{pluN}\cdot 4)/100$.

adr0 = $(\text{DC}+\text{pluN}\cdot 4)\%100$.

datlen: Datenlese- oder Datenschreiblänge. Die Datenlänge der PLU ist 4.

[dat]: Die Daten des Schreibbefehls. Die Länge wird durch datlen checksum bestimmt: $\text{checksum,checksum} = 100 - (\text{cmd}+\text{type}+\text{dar1}+\text{dar0}+\text{datalen}+[\text{dat}])\%100$.

Falls der festgelegte aktuelle Stückpreis 111,00 ist:

```
DO    44
DI    02
DO    11 00 00 00 00 ef
DI    02
DO    77 f9 00 00 04 00 00 2b 5c 01
DI    02
DO    33 00 00 00 00 cd
DI    02
```

Falls der festgelegte aktuelle Stückpreis der PLU1 111,00 ist:

```
DO    44
DI    02
DO    11 00 00 00 00 ef
DI    02
DO    77 f9 00 e0 04 00 00 2b 5c 21
```

DI 02
DO 33 00 00 00 00 cd
DI 02

Bei Ablesung PLU1: DO 44

DI 02
DO 11 00 00 00 00 ef
DI 02
DO 55 f9 00 e0 04 ce
DI 02
DI 55 fd 00 e0 04 00 00 2b 5c 43 ;;
DO 33 00 00 00 00 cd
DI 02

Falls der abgelesene aktuelle Stückpreis und Gesamtpreis wie folgt ist: DO 44

DI 02
DO 11 00 00 00 00 ef
DI 02
DO 55 f4 00 00 09 ae
DI 02
DI 55 f4 00 00 04
DI 00 00 00 00 de ;; Gesamtpreis 2.22
DI 00 00 2b 5c ;; Stückpreis 111,00
DI 4e
DO 33 00 00 00 00 cd
DI 02

3. Ablesung aktuelles Gewicht.

Der PC sendet 0x05, nach Erhalt der Waagenantwort 0x06, Sendung von 0x11, Waage antwortet mit dem Gewichtspaket.

Gewichtspaketformate: SOH STX STA SIGN WEIGHT_ASCII WEIGHT_UNIT
BCC ETX EOT.

SOH. Symbol Start des Sendevorgangs, ein Byte, festgelegt als 01H.

STX. Startsymbol des Datenelements, ein Byte, festgelegt als 02H.

STA. Beschreibung des Gewichtsstatus, ein Byte, kann nur 53H oder 55H46H sein. 53H – Gewicht ist stabil, 55H – Gewicht ist nicht stabil, 46H – Gewicht ist anormal.

SIGN. Gewichtszeichen, ein Byte, kann nur 2DH oder 20H sein. 2DH – Gewicht ist negativ, 20H – Gewicht ist positiv.

WEIGHT_ASCII. Gewichtszeichenfolge. 5~6 Byte, kann nur 30H~39H sein, Dezimal (2EH), Leerstelle (20H).

WEIGHT_UNIT. Zeichenfolge der Gewichtseinheit. 1~2 Byte, kann nur Folgendes sein: 'TJ': catty of Taiwan, 'TL': tael of Taiwan, 'SJ': 'SJ': jin, 'LB': pfund, 'KG': Kilogramm, 'G': gramm.

BCC. Zeichen der Datenverifizierung. Ein Byte. STA als D1, SIGN als D2, Byte vor BCC als DN festlegen, sodass $BCC=D1^D2^{\dots}DN$. '^' ist ein logischer XOR-Operator.

ETX. Endsymbol des Datenelements, ein Byte, festgelegt als 03H.

EOT. Symbol Ende des Sendevorgangs, ein Byte, festgelegt als 04H.

GRAM ▼

Gram Precision S.L.

Travesía Industrial, 11 · 08907 Hospitalet de Llobregat · Barcelona (Spain)

Tel. +34 902 208 000 · +34 93 300 33 32

Fax +34 93 300 66 98

info@gram-group.com

www.gram-group.com