



# Quido - MODBUS

---

Kompletní popis protokolů MODBUS RTU a TCP  
v I/O modulech Quido

---

# Quido - MODBUS

## Katalogový list

Vytvořen: 9.11.2009

Poslední aktualizace: 16.3 2017 14:01

Počet stran: 12

© 2017 Papouch s.r.o.

---

## Papouch s.r.o.

Adresa:

**Strašnická 3164/1a  
102 00 Praha 10**

Telefon:

**+420 267 314 268**

Internet:

**www.papouch.com**

E-mail:

**papouch@papouch.com**



**OBSAH**

Popis.....	4
MODBUS RTU: Základní komunikační parametry.....	4
MODBUS TCP: Základní komunikační parametry.....	4
Přehled funkčních kódů .....	4
Přepnutí protokolů.....	5
Spinel → MODBUS RTU .....	5
Povolení konfigurace .....	5
Přepnutí .....	5
MODBUS RTU → Spinel .....	5
Registry .....	6
Identifikace zařízení .....	6
Holding Register .....	6
Input register.....	9
Discrete Inputs.....	10
Coils .....	10

## POPIS

Tento dokument popisuje komunikační protokoly MODBUS RTU a MODBUS TCP v I/O modulech Quido. Dokumentace hardwaru a popis jeho funkcí I/O modulů je k dispozici na webu <http://www.papouch.com/> (podrobná dokumentace je ke stažení také ve formátu PDF).

*Tip:* Detailní informace o protokolu MODBUS jsou k dispozici na stránce [modbus.org](http://modbus.org).

*Tip:* Pro vyzkoušení komunikace s Quidem přes Modbus můžete použít například [ModScan32](#).

## MODBUS RTU: Základní komunikační parametry

Následující parametry se týkají Quid s rozhraním RS232 nebo RS485.

Komunikační linka .....	RS485
Komunikační rychlost .....	rozsah od 110 Bd do 230,4 kBd (výchozí: <b>9,6 kBd</b> )
Počet datových bitů .....	<b>8</b>
Parita.....	bez parity, sudá, lichá (výchozí: <b>bez parity</b> )
Počet stopbitů .....	1, 2 (výchozí: <b>1</b> )
Výchozí adresa .....	0x31
Výchozí protokol nastavený z výroby .....	Spinel (Způsob přepnutí do protokolu MODBUS RTU je uveden na následující straně.)

## MODBUS TCP: Základní komunikační parametry

Quida s Ethernetovým rozhraním umí v režimu WEB komunikovat protokolem MODBUS TCP. Port pro komunikaci je nastavitelný na záložce *Ostatní*. Jako výchozí hodnota je nastaven port číslo 502.

## Přehled funkčních kódů

Zařízení umožňuje přistupovat ke své paměti – v závislosti na typu registru – těmito instrukcemi:

- 0x01 .....čtení coils
- 0x02 .....čtení diskretních vstupů
- 0x03 .....čtení holding registrů
- 0x04 .....čtení vstupních registrů
- 0x05 .....nastavení jednoho z diskretních výstupů <sup>1</sup>
- 0x06 .....nastavení jednoho holding registru<sup>1</sup>
- 0x0F .....nastavení několika diskretních výstupů najednou
- 0x10 .....zapsání do několika holding registrů
- 0x11 .....identifikace<sup>1</sup>

<sup>1</sup> Tento funkční kód lze použít jen u MODBUS RTU.

## PŘEPNUTÍ PROTOKOLŮ

Výchozím protokolem u Quido RS je z výroby nastaven Spinel. Pro přepnutí do protokolu MODBUS slouží následující instrukce z protokolu Spinel. Quido RS lze pohodlně přepnout do protokolu Modbus (nebo zpět) pomocí programu **ModbusConfigurator**, který je k dispozici zde: <http://www.papouch.com/cz/website/mainmenu/software/modbus-configurator/>

### Spinel → MODBUS RTU

#### Povolení konfigurace

Povoluje provedení servisní instrukce. Musí předcházet bezprostředně instrukci Přepnutí. Instrukci nelze použít s universální adresou nebo s adresou „broadcast“.

#### Dotaz:

*Kód instrukce:* E4H

#### Odpověď:

*Kód potvrzení:* ACK 00H

#### Příklady:

Dotaz:
2AH, 61H, 00H, 05H, 01H, 02H, E4H, 88H, 0DH
Povolení konfigurace.
Odpověď – potvrzení:
2AH, 61H, 00H, 05H, 01H, 02H, 00H, 6CH, 0DH

#### Přepnutí

Přepnutí protokolu se provádí speciální instrukcí protokolu Spinel, formátu 97. Jako adresa musí být použita adresa konkrétního modulu (nelze použít tzv. „broadcast“ ani universální adresu). Instrukci musí bezprostředně předcházet instrukce „Povolení konfigurace“.

#### Dotaz:

*Kód instrukce:* EDH

#### Odpověď:

*Kód potvrzení:* ACK 00H

#### Příklady:

Dotaz:
2AH, 61H, 00H, 06H, 66H, 02H, EDH, 02H, 17H, 0DH
Příkaz k přepnutí protokolu ze Spinel do MODBUS RTU.
Odpověď:
2AH, 61H, 00H, 05H, 66H, 02H, 00H, 07H, 0DH
Přijetí příkazu potvrzeno. Po odeslání této odpovědi již komunikuje Quido protokolem MODBUS RTU.

### MODBUS RTU → Spinel

Přepíná se zápisem do Holding registru – viz str. 7.

## REGISTRY

### Identifikace zařízení

Čtení identifikačního řetězce zařízení (Report slave ID).

#### Funkční kódy:

0x11 – Report slave ID

#### Parametry:

Počet bytů	1 Byte	dle řetězce
ID	1 Byte	ID je totožné s adresou zařízení
RI	1 Byte	Run Indikátor – zde vždy 0xFF (zapnuto)
Data	N Byte	Řetězec stejný jako v protokolu Spinel. Tedy například: <i>Quido RS 4/4; v0209.02.27; f66 97; t1</i>

### Holding Register

Konfigurace zařízení, obsluha počítačů impulzů a analogových výstupů.

Adresa	Přístup	Funkce	Název
0	zápis	0x06	<b>Povolení konfigurace</b> Zápis hodnoty 0x00FF do tohoto paměťového místa musí předcházet všem instrukcím, zapisujícím do holding registru na adresy 0 až 15. Slouží k ochraně před nechtěnou změnou konfigurace. Není povoleno zapisovat Povolení konfigurace pomocí Multiply write zároveň s dalšími parametry!
1	čtení, zápis	0x03, 0x06, 0x10	<b>Adresa (ID)<sup>2, 3</sup></b> Unikátní adresa zařízení v protokolu Modbus. Je očekáváno číslo z rozsahu 1 až 247. Adresa je unikátní pro protokol Modbus. <i>Výchozí adresou je 0x0031.</i>
2	čtení, zápis	0x03, 0x06, 0x10	<b>Komunikační rychlost<sup>2, 3</sup></b> Rychlosti a jim odpovídající kódy: 1 200 Bd..... 0x0003 2 400 Bd..... 0x0004 4 800 Bd..... 0x0005 9 600 Bd..... 0x0006 ( <i>výchozí nastavení</i> ) 19 200 Bd..... 0x0007 38 400 Bd..... 0x0008 57 600 Bd..... 0x0009 115 200 Bd..... 0x000A

<sup>2</sup> Zápisu do tohoto paměťového místa musí předcházet zápis hodnoty 0x00FF na adresu 0 do pozice Povolení konfigurace. Jde o ochranu před nechtěnou změnou konfigurace. Není povoleno zapisovat Povolení konfigurace pomocí Multiply write zároveň s dalšími parametry. **Po zápisu dojde k restartu zařízení a tím i k vynulování čítačů!**

<sup>3</sup> Toto nastavení se ukládá do paměti typu FLASH. To znamená, že (1) nastavení se pamatuje i po vypnutí napájení, a (2) počet zápisů do paměti je omezený (typicky jen několik desítek tisíc zápisů).

Adresa	Přístup	Funkce	Název																								
3	čtení, zápis	0x03, 0x06, 0x10	<p><b>Datové slovo</b><sup>2,3</sup> Datové slovo je vždy osmibitové.</p> <table border="1"> <thead> <tr> <th>Hodnota</th> <th>Parita</th> <th>Počet stopbitů</th> </tr> </thead> <tbody> <tr> <td>0x0000 (výchozí)</td> <td>není (N)</td> <td>1</td> </tr> <tr> <td>0x0001</td> <td>sudá (E)</td> <td>1</td> </tr> <tr> <td>0x0002</td> <td>lichá (O)</td> <td>1</td> </tr> <tr> <td>0x0003</td> <td>není (N)</td> <td>2</td> </tr> <tr> <td>0x0004</td> <td>sudá (E)</td> <td>2</td> </tr> <tr> <td>0x0005</td> <td>lichá (O)</td> <td>2</td> </tr> <tr> <td>0x0006 až 0x00FF</td> <td>není (N)</td> <td>1</td> </tr> </tbody> </table>	Hodnota	Parita	Počet stopbitů	0x0000 (výchozí)	není (N)	1	0x0001	sudá (E)	1	0x0002	lichá (O)	1	0x0003	není (N)	2	0x0004	sudá (E)	2	0x0005	lichá (O)	2	0x0006 až 0x00FF	není (N)	1
Hodnota	Parita	Počet stopbitů																									
0x0000 (výchozí)	není (N)	1																									
0x0001	sudá (E)	1																									
0x0002	lichá (O)	1																									
0x0003	není (N)	2																									
0x0004	sudá (E)	2																									
0x0005	lichá (O)	2																									
0x0006 až 0x00FF	není (N)	1																									
4	čtení, zápis	0x03, 0x06, 0x10	<p><b>Rozlišení konce paketu</b><sup>2,3</sup> Konfiguruje, jak velká prodleva mezi byty bude považována za konec paketu. Prodleva se zadává v počtu bytů. Je možné zadat hodnotu 4 až 100. Výchozí hodnota je 10.</p>																								
5	čtení, zápis	0x03, 0x06, 0x10	<p><b>Komunikační protokol</b><sup>2,3</sup> Umožňuje přepnout zařízení do komunikace protokolem Spinel. Po odeslání odpovědi se zařízení přepne do zvoleného protokolu a dále komunikuje pouze jím. (V každém z protokolů existuje instrukce pro přepnutí protokolů.) Kód pro protokol <i>Spinel</i>: 0x0001 (výchozí) Kód pro protokol Modbus RTU: 0x0002</p>																								
100 – 160	<p><i>Modbus TCP:</i> jen čtení</p> <p><i>Modbus RTU:</i> čtení i zápis</p>	<p><i>Modbus TCP:</i> 0x03</p> <p><i>Modbus RTU:</i> 0x03, 0x06, 0x10</p>	<p><b>Stav čítačů</b> <i>Čítač umožňuje počítat jednotlivé změny stavu vstupu. Za změnu je považována změna logického stavu (nebo stavu připojeného kontaktu). Každý vstup má vlastní čítač. K hodnotě čítače je přičtena jednička při vybraných změnách na příslušném vstupu (změna z 1 do 0; změna z 0 do 1; případně obě změny).</i> Zde jsou uloženy aktuální stavy 16ti bitových čítačů na všech vstupech. (Počítání je z výroby vypnuto.) Celkový počet registrů odpovídá počtu vstupů Quida. U Quida s deseti vstupy tedy bude použitých deset 16bit registrů. Maximální počet čítačů je 60. (Na případných dalších vstupech nejsou čítače k dispozici.) Nulování čítačů se provádí zápisem nulové hodnoty. Čítače jsou vynulovány také při vypnutí nebo restartem zařízení. Doporučený postup pro průběžné čtení aktuálního stavu čítačů:  <ol style="list-style-type: none"> <li>1) Přečtete hodnoty registru <i>Stav čítačů</i>.</li> <li>2) Přečtenou hodnotu odečtete pomocí následující sady registrů <i>Odečet od čítače</i>.</li> </ol> Tímto postupem nepřijdete o žádný záznam změny na vstupu.</p>																								
200 – 260	zápis	0x06, 0x10	<p><b>Odečet od čítače</b> Odečte zadanou hodnotu od aktuálního stavu čítače. (Hodnota, která má být odečtena, nesmí být větší než aktuální stav čítače. Při použití funkce 0x10 (Multiply write) nelze zapsat víc než 12 registrů najednou.)</p>																								

Adresa	Přístup	Funkce	Název										
300 – 360	čtení, zápis	0x03, 0x06, 0x10	<p><b>Konfigurace čítače</b><sup>3</sup></p> <p>0.....čítač na tomto vstupu je <u>vypnutý</u></p> <p>1.....čítač přičte ke své hodnotě jednotku při každé zaznamenané <u>náběžné hraně</u> signálu na příslušném vstupu</p> <p>2.....čítač přičte ke své hodnotě jednotku při každé zaznamenané <u>sestupné hraně</u> signálu na příslušném vstupu</p> <p>3.....čítač přičte ke své hodnotě jednotku <u>při každé</u> zaznamenané hraně (náběžné <i>i</i> sestupné) signálu na příslušném vstupu</p>										
500 – 532	čtení, zápis	0x03, 0x06, 0x10  (i pomocí 0x10 pouze jeden výstup!)	<p><b>Nastavení jednoho výstupu na určitou dobu</b></p> <p>Aktivuje vybraný výstup na určitou dobu – spustí na vybraném výstupu pulz zadané polaroty na zadanou dobu. Pulz se spustí okamžitě po přijetí této instrukce. Opětovné spouštění pulzu, když ještě neskončil předchozí, je možné.</p> <p><b>Horní byte</b></p> <p>0xFF → Sepnutí</p> <p>0x00 → Rozeptnutí</p> <p><b>Dolní byte</b></p> <p>0x00 až 0xFF → Doba, po kterou má být výstup sepnutý nebo rozeptnutý (podle horního bytu). Jednotkou je 0,5 sec. Je tedy možné nastavit sepnutí na 0,5 až 127,5 sec.</p>										
600 – 728  (Pouze v Modbus TCP!)	čtení, zápis	0x03, 0x10  Jak při čtení, tak při zápisu je dovoleno zapisovat/číst pouze celou čtveřici registrů najednou!	<p><b>Termostat</b><sup>3</sup></p> <p>Po sobě následující čtveřice registrů se vztahují k jednotlivým výstupům a nastavení termostatu pro ně. Pro první výstup Quida se jedná o registry 600 (první) až 603 (čtvrtý).</p> <table border="1"> <thead> <tr> <th>Registr</th> <th>Význam</th> </tr> </thead> <tbody> <tr> <td>První</td> <td> <p>Formát <u>dolního</u> bytu: <b>FSSKTTTT</b></p> <p>„<b>F</b>“ – Zapnuta/vypnuta funkce hlídání teploty pro výstup (OUTx); (1 = zapnuto; 0 = vypnuto)</p> <p>„<b>SS</b>“ – Akce, která se má při nastavené teplotě provést</p> <p>00 = sepnout výstup</p> <p>01 = rozeptnout výstup</p> <p>10 = sepnout výstup na nastavenou dobu („kladný pulz“)</p> <p>11 = rozeptnout výstup na nastavenou dobu („záporný pulz“)</p> <p>„<b>K</b>“ – Kritická teplotní tendence – uplatní se pouze u sepnutí na nastavenou dobu:</p> <p>0 – vzestup teploty</p> <p>1 – pokles teploty</p> <p>„<b>TTTT</b>“ – Binární číslo teploměru, ke kterému se vztahují následující teplotní meze.</p> </td> </tr> <tr> <td>Druhý</td> <td>Teplota ve formátu signed int – vyšší mez.</td> </tr> <tr> <td>Třetí</td> <td>Teplota ve formátu signed int – nižší mez.</td> </tr> <tr> <td>Čtvrtý</td> <td> <p><u>Horní</u> byte:</p> <p>Čas sepnutí relé ve vteřinách, pokud je nastaveno sepnutí na určitou dobu.</p> </td> </tr> </tbody> </table>	Registr	Význam	První	<p>Formát <u>dolního</u> bytu: <b>FSSKTTTT</b></p> <p>„<b>F</b>“ – Zapnuta/vypnuta funkce hlídání teploty pro výstup (OUTx); (1 = zapnuto; 0 = vypnuto)</p> <p>„<b>SS</b>“ – Akce, která se má při nastavené teplotě provést</p> <p>00 = sepnout výstup</p> <p>01 = rozeptnout výstup</p> <p>10 = sepnout výstup na nastavenou dobu („kladný pulz“)</p> <p>11 = rozeptnout výstup na nastavenou dobu („záporný pulz“)</p> <p>„<b>K</b>“ – Kritická teplotní tendence – uplatní se pouze u sepnutí na nastavenou dobu:</p> <p>0 – vzestup teploty</p> <p>1 – pokles teploty</p> <p>„<b>TTTT</b>“ – Binární číslo teploměru, ke kterému se vztahují následující teplotní meze.</p>	Druhý	Teplota ve formátu signed int – vyšší mez.	Třetí	Teplota ve formátu signed int – nižší mez.	Čtvrtý	<p><u>Horní</u> byte:</p> <p>Čas sepnutí relé ve vteřinách, pokud je nastaveno sepnutí na určitou dobu.</p>
Registr	Význam												
První	<p>Formát <u>dolního</u> bytu: <b>FSSKTTTT</b></p> <p>„<b>F</b>“ – Zapnuta/vypnuta funkce hlídání teploty pro výstup (OUTx); (1 = zapnuto; 0 = vypnuto)</p> <p>„<b>SS</b>“ – Akce, která se má při nastavené teplotě provést</p> <p>00 = sepnout výstup</p> <p>01 = rozeptnout výstup</p> <p>10 = sepnout výstup na nastavenou dobu („kladný pulz“)</p> <p>11 = rozeptnout výstup na nastavenou dobu („záporný pulz“)</p> <p>„<b>K</b>“ – Kritická teplotní tendence – uplatní se pouze u sepnutí na nastavenou dobu:</p> <p>0 – vzestup teploty</p> <p>1 – pokles teploty</p> <p>„<b>TTTT</b>“ – Binární číslo teploměru, ke kterému se vztahují následující teplotní meze.</p>												
Druhý	Teplota ve formátu signed int – vyšší mez.												
Třetí	Teplota ve formátu signed int – nižší mez.												
Čtvrtý	<p><u>Horní</u> byte:</p> <p>Čas sepnutí relé ve vteřinách, pokud je nastaveno sepnutí na určitou dobu.</p>												



Adresa	Přístup	Funkce	Název										
			<p><u>Dolní byte:</u> Určuje co se má stát, pokud bude odpojen nebo přerušen kabel k teplotnímu senzoru.</p> <p>0 – ponechat kontakt relé beze změny 1 – rozepnout kontakt relé 2 – sepnout kontakt relé</p>										
800 – 832  (Pouze v Modbus TCP!)	čtení, zápis	0x03, 0x10  Jak při čtení, tak při zápisu je dovoleno zapisovat/číst pouze celou čtveřici registrů najednou!	<p><b>Hlídnání teploty</b><sup>3</sup> Zařízení hlídá dvě teplotní meze. Při jejich opuštění pošle automatickou zprávu (např. HTTP GET). Po sobě následující čtveřice registrů se vztahují k jednotlivým teploměřům. Pro první teploměr se jedná o registry 800 (první) až 803 (čtvrtý).</p> <table border="1"> <thead> <tr> <th>Registr</th> <th>Význam</th> </tr> </thead> <tbody> <tr> <td>První</td> <td>Zapnutí (0001H) nebo vypnutí (0000H) hlídání teploty</td> </tr> <tr> <td>Druhý</td> <td>Pokud je teplota mimo meze a má se upozornění na tento stav odesílat opakovaně, zadejte sem periodu v sekundách, jak často se má zpráva odesílat.</td> </tr> <tr> <td>Třetí</td> <td>Horní mez teploty zadaná jako celé číslo (signed int). Jde o teplotu vynásobenou deseti. Příklad: Hodnotu 24.6 sem vložte jako číslo 246.</td> </tr> <tr> <td>Čtvrtý</td> <td>Dolní mez teploty zadaná jako celé číslo.</td> </tr> </tbody> </table>	Registr	Význam	První	Zapnutí (0001H) nebo vypnutí (0000H) hlídání teploty	Druhý	Pokud je teplota mimo meze a má se upozornění na tento stav odesílat opakovaně, zadejte sem periodu v sekundách, jak často se má zpráva odesílat.	Třetí	Horní mez teploty zadaná jako celé číslo (signed int). Jde o teplotu vynásobenou deseti. Příklad: Hodnotu 24.6 sem vložte jako číslo 246.	Čtvrtý	Dolní mez teploty zadaná jako celé číslo.
Registr	Význam												
První	Zapnutí (0001H) nebo vypnutí (0000H) hlídání teploty												
Druhý	Pokud je teplota mimo meze a má se upozornění na tento stav odesílat opakovaně, zadejte sem periodu v sekundách, jak často se má zpráva odesílat.												
Třetí	Horní mez teploty zadaná jako celé číslo (signed int). Jde o teplotu vynásobenou deseti. Příklad: Hodnotu 24.6 sem vložte jako číslo 246.												
Čtvrtý	Dolní mez teploty zadaná jako celé číslo.												

### Input register

Adresa	Přístup	Funkce	Název
0	čtení	0x04	<p><b>Status naměřené teploty</b> 0..... hodnota je platná 1..... chyba senzoru nebo odpojený senzor</p>
1	čtení	0x04	<p><b>Naměřená hodnota – celé číslo</b> Naměřená teplota jako celé číslo se znaménkem (signed integer). Číslo znamená teplotu vynásobenou deseti. <i>Příklad:</i> Teplota 23,4 °C je v tomto registru reprezentována hodnotou 234.</p>
2, 3	čtení	0x04	<p><b>Naměřená hodnota – float</b> Naměřená teplota jako 32 bit float podle IEEE 754.<sup>4</sup></p>

<sup>4</sup> Popis normy IEEE 754 je k dispozici například zde: [http://en.wikipedia.org/wiki/IEEE\\_754](http://en.wikipedia.org/wiki/IEEE_754)

## Discrete Inputs

Funkční kód **0x02** je určen pro **čtení stavu vstupů**. Čte 1 až X vstupů (maximum dle počtu vstupů na Quidu). V dotazu je specifikováno číslo prvního čteného vstupů i počet vstupů, které se mají přečíst. Vstupy jsou číslovány od nuly. Tedy například vstupy 1 až 10 mají pořadová čísla 0 až 9.

V odpovědi jsou stavy vstupů představovány jednotlivými bity. Hodnota 1 znamená aktivní vstup (připojeno napětí nebo sepnutý kontakt), hodnota 0 neaktivní vstup. Nejnižší bit v prvním bytu odpovědi představuje stav prvního vstupu, který byl adresován v dotazu.

Pokud není počet vstupů násobkem osmi, jsou přebytečné bity vyplněny nulami. V proměnné *Počet hodnot* je specifikován počet bytů se stavem vstupů.

Adresa	Přístup	Funkce	Obsah
0	čtení	0x02	<b>Stav prvního</b> požadovaného vstupu
1	čtení	0x02	<b>Stav druhého</b> požadovaného vstupu
...			

### Příklad:

Příklad čtení vstupů 1 až 8.

Dotaz:		Odpověď:	
Funkční kód	0x02	Funkční kód	0x02
Adresa MSB	0x00	Počet bytů	0x01
Adresa LSB	0x00	Stav vstupů	0xA7
Počet vstupů MSB	0x00		
Počet vstupů LSB	0x08		

Výsledkem dotazu je byte 0xA7, což je v binárním vyjádření 1010 0111. Jednotlivé bity odpovídají stavům vstupů. Nejnižší bit představuje vstup číslo 1.

## Coils

Přístup k aktuálním stavům a ovládání výstupních relé.

### Funkční kód 0x01

Tento funkční kód je určen pro **čtení stavu výstupů**. Čte 1 až X výstupů (maximum dle počtu výstupů na Quidu). V dotazu je specifikováno číslo prvního čteného výstupů i počet výstupů, které se mají přečíst. Výstupy jsou číslovány od nuly. Tedy například výstupy 1 až 10 mají pořadová čísla 0 až 9.

V odpovědi jsou stavy výstupů představovány jednotlivými bity. Hodnota 1 znamená sepnutý výstup, hodnota 0 rozepnutý výstup. Nejnižší bit v prvním bytu odpovědi představuje stav prvního výstupu, který byl adresován v dotazu.

Pokud není počet výstupů násobkem osmi, jsou přebytečné bity vyplněny nulami.

**Funkční kódy 0x05 a 0x0F**

Tyto funkční kódy jsou určeny pro **ovládání výstupů**. V dotazu je specifikováno které výstupy mají být nastaveny. Výstupy jsou číslovány od nuly. Tedy například výstup 5 má pořadové číslo 4.

Logická 1 znamená sepnutí výstupu, logická 0 rozepnutí výstupu.

V odpovědi je uveden funkční kód, adresa a počet výstupů, které byly změněny.

Adresa	Přístup	Funkce	Výstup
0	čtení, zápis	0x01, 0x05, 0x0F	<b>První</b> adresovaný výstup
1	čtení, zápis	0x01, 0x05, 0x0F	<b>Druhý</b> adresovaný výstup
...			

**Příklad čtení:**

Příklad čtení výstupů 1 a 2.

Dotaz:		Odpověď:	
Funkční kód	0x01	Funkční kód	0x01
Adresa MSB	0x00	Počet bytů	0x01
Adresa LSB	0x00	Stav výstupů	0x02
Počet výstupů MSB	0x00		
Počet výstupů LSB	0x02		

Výsledkem dotazu je byte 0x02, což je v binárním vyjádření 0000 0010. Je nastaven druhý nejnižší bit. To znamená, že výstup 1 je rozepnutý a 2 je sepnutý. (Zbylé bity jsou vyplněny nulami.)

**Příklad zápisu:**

Příklad zápisu stavu výstupů 20 až 29 (celkem deset výstupů):

Data pro výstupy, jsou uložena ve dvou Bytech: 0xCD a 0x01 (1100 1101 0000 0001 binárně)

Bit: 1 1 0 0 1 1 0 1 0 0 0 0 0 0 0 1  
 Číslo výstupu: 27 26 25 24 23 22 21 20 - - - - - 29 28

Jako první je odeslán byte 0xCD se stavem výstupů 27 až 20. Nejnižší bit představuje nejnižší výstup 20. Další byte (0x01) obsahuje zbylé bity 28 a 29. Ostatní bity jsou vyplněny nulami.

Dotaz:		Odpověď:	
Funkční kód	0x0F	Funkční kód	0x0F
Adresa MSB	0x00	Adresa MSB	0x00
Adresa LSB	0x13	Adresa LSB	0x13
Počet výstupů MSB	0x00	Počet výstupů MSB	0x00
Počet výstupů LSB	0x0A	Počet výstupů LSB	0x0A
Počet bytů	0x02		
Hodnoty MSB	0xCD		
Hodnoty LSB	0x01		

Výsledkem příkazu je změna stavu některých výstupů na Quidu.

# Papouch s.r.o.

Přenosy dat v průmyslu, převodníky linek a protokolů, RS232/485/422/USB/Ethernet/GPRS/WiFi, měřicí moduly, inteligentní teplotní čidla, I/O moduly, elektronické aplikace dle požadavků.

Adresa:

**Strašnická 3164/1a  
102 00 Praha 10**

Telefon:

**+420 267 314 268**

Internet:

**[www.papouch.com](http://www.papouch.com)**

E-mail:

**[papouch@papouch.com](mailto:papouch@papouch.com)**

