



Produto : Rádio Modem

Marca : DATAREGIS WSL

Modelos : WT35 Evolution

WT35 Evolution 3

WT35 Evolution NET

**MANUAL DE COMANDOS AT
PADRÃO DATAREGIS**

Versão: 1.00

ÍNDICE

AT+WTET	Configurar a Rede Ethernet	5
AT+WTCS	Configurar as Interfaces de Conexão	5
AT+WTSS	Configurar os Sockets TCP/UDP	6
AT+WTSO	Abrir os Sockets TCP/UDP	7
AT+WTSC	Fechar os sockets TCP/UDP	8
AT+WTSR	Ler dados recebidos pelo socket TCP/UDP	8
AT+WTSW	Escrever dados em socket TCP/UDP	9
AT+WTVR	Retorna a versão de software	10
AT+WTID	Retorna a identificação do equipamento	10
AT+WTAC	Configura o timeout de ACK para pacotes TCP	10
AT+WTRT	Configura o número máximo de retransmissões de pacotes TCP	11
AT+WTTK	Programa tempo de inatividade para derrubamento de conexão externa ..	11
	Procedimento para recebimento de chamadas:	12
	Auto-bauding (AT+IPR):	12
	Exemplos Comandos Rádio Modems Dataregis WSL	13
	Comandos de IP:	13
	Comandos de leitura de id e de versão:	17
	Comandos de configuração do protocolo TCP:	18
	Exemplo de uso de mensagens SMS:	18
	MODEM SERVIDOR:	18
	MODEM CLIENTE:	18
	Exemplo de realocação de seriais Rádio Modem Evolution3:	19

Introdução

Nomes de produtos e companhias aqui mencionados podem ser marcas comerciais ou marcas registradas de seus respectivos proprietários.

As informações contidas neste manual são para os produtos Rádio Modem DATAREGIS WSL modelos WT35 Evolution, WT35 Evolution3, WT35 Evolution NET.

Em nenhum caso, a DATAREGIS será responsável por rendimentos, perda de dados, ou por qualquer outro dano extraordinário, incidental, conseqüente ou indireto, não obstante a causa.

O conteúdo deste documento é oferecido no estado. A não ser em casos em que a lei em vigor o exija, nenhuma garantia, expressa ou implícita incluindo, porém não limitada, às garantias de negociabilidade ou aptidão para um propósito determinado, é oferecida em relação à precisão, confiabilidade ou conteúdo deste documento.

A DATAREGIS, no contexto de sua política de aprimoramento contínuo, reserva-se o direito de revisar este documento, ou retirá-lo do mercado, sem prévio aviso.

AT+WTET Configurar a Rede Ethernet

Define os parâmetros associados à Rede Ethernet (comando válido apenas para o modelo com rede Ethernet)

Formato: AT+WTET="Parametro", "<Valor>"	Descrição dos campos:	
	"Parâmetro" – Item a ser configurado = address = gateway = timersv = dns = mask	"<Valor>" Define o endereço IP do item definido pelo parâmetro
	Retorno do comando: OK ou ERROR	
	Comando de leitura: AT+WTET? Lista as configurações de todos os parâmetros	
	Retorno do comando: +WTET: "Parametro", "<Valor>" OK	
NOTA	Para que as modificações deste comando sejam gravadas na memória não-volátil e tenham validade deve-se desligar o modem pelo comando AT^SMSO	

AT+WTCS Configurar as Interfaces de Conexão

Define as interfaces de conexão a serem utilizadas com os comandos do TCP-IP relacionando a um Idcon.

Formato: AT+WTCS=Idcon,"Parametro", "<Valor>"	Descrição dos campos:	
	Idcon–Identificador da Interface (ID) = 0 a (NPARAMETROS-1)	
	"Parametro" – Item a ser configurado Os itens restantes são válidos apenas para contype diferente de ETH: = user = passwd = apn - apenas para contype = GPRS0 ou GPRS1 = callednum - apenas para contype = CSD = authmode = alphabet	
	"<Valor>" – Conforme o parâmetro selecionado acima = CSD, ETH (apenas para modelo com interface Ethernet), GPRS0 ou GPRS1 (contype) = 25 caracteres em ASCII (user) = 25 caracteres em ASCII (passwd) = 25 caracteres em ASCII (apn) = 20 numeros em ASCII (callednum) = NONE, PAP, CHAP ou MSCHAPV1 (authmode) = 0 ou 1 (alphabet)	
	Retorno do comando: OK ou ERROR	

	<p>Comando de leitura: AT+WTCS? Lista as configurações das conexões definidas para cada identificador Idcon</p>
	<p>Retorno do comando: +WTCS: Idcon,"Parametro", "<Valor>" OK</p>
NOTA	<p>1) Na versão atual de software, não há diferença em definir o parâmetro contype GPRS0 ou GPRS1 2) Na versão atual de software, o parâmetro alphabet não tem utilidade 3) Para a versão de 3 canais seriais, os parâmetros são independentes para cada canal</p>

AT+WTSS Configurar os Sockets TCP/UDP

Configura todos os parâmetros de uma conexão TCP/UDP-IP a ser utilizada com a interface definida no comando +WTCS.

Formato: AT+WTSS=Idsck,"Parametro", "<Valor>"	<p>Descrição dos campos: Idsck – Identificador do Socket = 0 a (NSOCKETS-1)</p>
	<p>"Parâmetro" – Item a ser configurado = srvtype = conid = address</p>
	<p>"<valor>" – Conforme o parâmetro selecionado acima = socket ou none – define um socket ou libera recursos (srvtype) = 0 a (NPARAMETROS-1) (conid), identifica o conjunto de parâmetros definidos por AT+WTSC que serão utilizados na tentativa de conexão =socktcp://xxx.xxx.xxx.xxx:yyyy, (socket cliente TCP) sockudp://xxx.xxx.xxx.xxx:yyyy ou (socket cliente UDP) socktcp://listener:yyyy (socket servidor TCP) sendo xxx.xxx.xxx.xxx o endereço IP remoto e yyyy a porta remota, no caso de socket cliente, ou apenas yyyy (porta local), no caso de socket servidor (address)</p>
	<p>Retorno do comando: OK ou ERROR</p>
	<p>Comando de leitura: AT+WTSS? Lista todos os parâmetros definidos para cada Idsck</p>
	<p>Retorno do comando: +WTSS: Idsck,"Parâmetro", "<Valor>" OK</p>

AT+WTSO Abrir os Sockets TCP/UDP

O comando faz todas as ações necessárias (abre a conexão IP caso ainda não esteja aberta e abre socket TCP ou UDP) para habilitar o tráfego de dados através dos parâmetros identificados por Idsck

Formato: AT+WTSO=Idsck	Descrição dos campos: Idsck – Identificador do Socket = 0 a (NSOCKETS-1)
	Retorno do comando: OK ou ERROR
	Comando de leitura: AT+WTSO? Lista o estado dos sockets
	Retorno do comando: AT+WTSO: Idsck,<id serial>,<estado server>,<estado socket>,<contador rx>,<contador tx>,<end local>,<end remoto> OK Descrição dos valores retornados: Idsck – Identificador do Socket = 0 a (NSOCKETS-1) Tsock - Indica que o socket está definido <estado server> - Indica o estágio de conexão do socket: 2 = parâmetros do socket alocados 3 = conectando 4 = conectado 5 = fechando <estado socket> - Indica o estado do socket: 1 = socket não ativo 2 = cliente 3 = ouvindo (socket definido como listener) 4 = servidor (conexão de um cliente remoto aceita) <contador rx> - Número de caracteres recebidos <contador tx> - Número de caracteres transmitidos <end local> - Endereço local e porta local <end remoto> - Endereço remoto e porta remota
URC: +WTS: Idsck,<causa>,Sockserv Descrição dos valores retornados: Idsck – Identificador do Socket = 0 a (NSOCKETS-1) <causa> - Indica a causa da URC: 0 = queda de conexão 1 = indicação de conexão cliente entrante bem sucedida 2 = indicação de falha na conexão cliente entrante (falta de recursos disponíveis) Sockserv – Identifica o socket no qual a conexão cliente entrante foi alocada (apenas quando causa =1). A conexão pode ser aceita com AT+WTSO=Sockserv ou rejeitada por AT+WTSC=Sockserv	

NOTA	<p>1) Para que um cliente remoto possa ser alocado, é necessário que haja sockets disponíveis (não estejam definidos por AT+WTSS=N,srvtype,socket). Caso o socket já esteja definido, e deseje-se liberá-lo deve-se usar AT+WTSS=N,srvtype,none)</p> <p>2) Pode haver no máximo 2 conexões GPRS simultâneas, ou 1 conexão CSD. Para o modelo de um canal serial, se já houver 2 sockets conectados com parâmetros conid diferentes, e um terceiro tentar conexão com outro parâmetro conid, este não conseguirá efetuar a conexão (será retornado erro).</p> <p>3) No caso do modelo de três canais seriais, cada canal serial só pode usar um parâmetro de conexão ao mesmo tempo.</p>
-------------	---

AT+WTSC Fechar os sockets TCP/UDP

O comando fecha o socket TCP ou UDP identificado por Idsck

Formato: AT+WTSC=Idsck NOTA	Descrição dos campos: Idsck – Identificador do Socket = 0 a (NSOCKETS-1)
	Retorno do comando: OK ou <i>ERROR</i>
	Caso não haja mais sockets conectados à mesma interface de conexão do socket que está sendo fechado, esta interface será derrubada logo após o fechamento do socket.

AT+WTSR Ler dados recebidos pelo socket TCP/UDP

O comando lê os caracteres disponíveis no buffer de recepção do socket

Formato: AT+WTSR=Idsck,<tamanho>	Descrição dos campos: Idsck – Identificador do Socket = 0 a (NSOCKETS-1) <tamanho> - Número de caracteres solicitados
	Retorno do comando: + <i>WTSR</i> : Ident, <tamanho confirmado> O modem fornece o número de dados indicado por <tamanho confirmado> OK
	Descrição dos valores retornados: Idsck – Identificador do Socket = 0 a 7 <tamanho confirmado> - se houver menos dados no buffer de recepção do que o número solicitado retorna o número de dados disponíveis, caso contrário retorna <tamanho>

	<p>URC:</p> <p><i>+WTSR: Idsck, <causa></i></p> <p>Descrição dos valores retornados:</p> <p>Idsck – Identificador do Socket = 0 a (NSOCKETS-1)</p> <p><causa> - Indica a causa da URC: 1 = novos dados estão disponíveis através de AT+WTSR se <tamanho confirmado> é menor que <tamanho></p>
NOTA	O número máximo de caracteres que podem ser lidos a cada comando AT+WTSR é 1500

AT+WTSW Escrever dados em socket TCP/UDP

O comando coloca os dados fornecidos no buffer de transmissão do socket

<p>Formato:</p> <p>AT+WTSW=Idsck, <tamanho></p>	<p>Descrição dos campos:</p> <p>Idsck – Identificador do Socket = 0 a (NSOCKETS-1)</p> <p><tamanho> - Número de caracteres que serão fornecidos</p>
	<p>Retorno do comando:</p> <p><i>+WTSW: Idsck, <tamanho confirmado></i></p> <p>Fornece-se ao modem o número de dados indicado por <tamanho confirmado></p> <p>OK</p> <p><tamanho confirmado> é igual ou menor a <tamanho></p> <p>Descrição dos valores retornados:</p> <p>Idsck – Identificador do Socket = 0 a (NSOCKETS-1)</p> <p><tamanho confirmado> - se houver menos espaço disponível no buffer de transmissão do que <tamanho>, retorna o espaço disponível, caso contrário retorna <tamanho></p>
	<p>URC:</p> <p><i>+WTSW: Idsck, <causa></i></p> <p>Descrição dos valores retornados:</p> <p>Idsck – Identificador do Socket = 0 a (NSOCKETS-1)</p> <p><causa> - Indica a causa da URC: 1 = os dados retidos já foram enviados</p>
NOTA	O número máximo de caracteres que podem ser transmitidos a cada comando AT+WTSW é 1500

AT+WTVR Retorna a versão de software

O comando retorna a versão de software do microcontrolador e do modem

Formato: AT+WTVR	Retorno do comando: +WTVR:"<versaomcu>","<versaomod>" OK Descrição dos valores retornados: <versaomcu> - String ASCII (versão de software do microcontrolador) <versaomod> - String ASCII (versão de software do modem)
----------------------------	--

AT+WTID Retorna a identificação do equipamento

O comando retorna a identidade da serial que está fornecendo o comando, o número de série do microcontrolador, o número de série do modem e o número de identificação do cartão GSM

Formato: AT+WTID	Retorno do comando: +WTID:<idser>,"<idmcu>","<idmod>","<idcard>" OK Descrição dos valores retornados: <idser> - número de identificação da serial que está fornecendo o comando (1,2 ou 3) <idmcu> - 16 caracteres em ASCII (número de série do microcontrolador) <idmod> - 15 caracteres em ASCII (número de série do modem) <idcard> - 20 caracteres em ASCII (número de identificação do cartão GSM)
----------------------------	--

AT+WTAC Configura o timeout de ACK para pacotes TCP

O comando configura o valor em segundos de tempo de espera de ACK de um determinado pacote TCP para que o mesmo pacote seja reenviado. O valor ideal para este parâmetro depende da velocidade da rede na qual o pacote TCP está trafegando. Um valor baixo pode acarretar em reenvios desnecessários, um valor alto pode tornar a comunicação mais lenta. Este parâmetro é válido apenas para sockets TCP internos e é único para todas as seriais. O valor de fábrica é 5 segundos.

Formato: AT+WTAC=Timeout	Descrição dos campos: Timeout – Valor do tempo de espera em segundos = 1 a 15
	Retorno do comando: OK
	Descrição dos valores retornados: <idser> - número de identificação da serial que está fornecendo o comando (1,2 ou 3) <idmcu> - 16 caracteres em ASCII (número de série do microcontrolador) <idmod> - 15 caracteres em ASCII (número de série do modem) <idcard> - 20 caracteres em ASCII (número de identificação do cartão GSM)
	Comando de leitura: AT+WTAC?

	<p>Retorno do comando: +WTAC: Timeout OK</p> <p>Descrição dos valores retornados: Timeout - Valor do tempo de espera em segundos</p>
--	--

AT+WTRT Configura o número máximo de retransmissões de pacotes TCP

O comando configura o número máximo de retransmissões de um mesmo pacote TCP quando não é recebido o ACK. Este parâmetro é válido apenas para sockets TCP internos e é único para todas as seriais. O valor de fábrica é 5 retransmissões.

<p>Formato: AT+WTRT=Retransmissões</p>	<p>Descrição dos campos: Retransmissões – Número de retransmissões = 1 a 9</p>
	<p>Retorno do comando: OK</p>
	<p>Comando de leitura: AT+WTRT?</p> <p>Retorno do comando: +WTRT: Retransmissões OK</p>
	<p>Descrição dos valores retornados: Retransmissões - Número de retransmissões</p>

AT+WTTK Programa tempo de inatividade para derrubamento de conexão externa

Programa o tempo de inatividade de uma conexão GPRS para permitir que ela seja derrubada no caso em que haja duas seriais conectadas GPRS com PPP externo e a terceira solicite outra conexão GPRS com PPP externo.

Se o tempo de inatividade for menor que o tempo programado pelo comando nas duas seriais já conectadas, a serial que está tentando se conectar esperará pelo tempo programado para esta pela inatividade das outras. Caso dentro deste período de tempo não haja um período de inatividade maior que o especificado para as outras duas seriais já conectadas, estas não serão derrubadas e a terceira não conseguirá se conectar. Caso o tempo de inatividade for maior que o tempo programado para as duas seriais já conectadas, será derrubada a que tiver o maior tempo de inatividade e a serial que estiver solicitando a conexão conseguirá se conectar.

Caso o valor de +WTTK seja 0, a serial nunca será derrubada.

Formato: AT+WTTK=<timconexao>	Descrição dos campos: <timconexao> - valor em segundos de tempo de inatividade (0 – 255)
NOTA	Retorno do comando: OK ou <i>ERROR</i>
<p>1)O modem permite que no máximo duas seriais estejam conectadas ao serviço GPRS ao mesmo tempo, e que no máximo uma esteja conectada ao serviço CSD</p> <p>2)O comando apenas é válido para conexões GPRS. No caso de uma serial estar conectada ao serviço CSD e uma outra tentar outra conexão CSD, a conexão da primeira será preservada e a segunda não conseguirá conexão, recebendo "NO DIALTONE"</p> <p>3)Enquanto houver uma conexão CSD ativa, é possível haver conexão GPRS nas outras duas seriais, mas não será possível transferência de dados nas seriais conectadas ao GPRS</p> <p>4)É considerada "atividade da conexão" qualquer transferência de dados (entrada ou saída) pela conexão especificada, não importando o protocolo de comunicação usado, nem o número de bytes trafegados.</p> <p>O valor <timconexao> valerá para a serial que estiver fornecendo o comando (o valor é individual para cada serial).</p> <p><u>Comando de leitura:</u> AT+WTTK?</p> <p><u>Retorno do comando:</u> +WTTK: <timconexao> OK</p> <p><u>Descrição dos valores retornados:</u> <timconexao> - valor programado de tempo de inatividade da serial que está fornecendo o comando</p>	

Procedimento para recebimento de chamadas:

Caso o modem receba uma chamada de dado, ocorrerá a URC "RING" nas 3 seriais a cada 5 segundos, até que uma das seriais atenda a chamada (atenderá a chamada a serial que primeiro atender a chamada através do comando "ATA")

NOTA: Caso o cartão gsm do modem não esteja com o código PIN automaticamente habilitado, é necessário fornecer o comando AT+CSNS=4 em qualquer uma das seriais para que o modem possa receber chamada de dados

Auto-bauding (AT+IPR):

O auto-bauding é programável individualmente para cada canal serial (ativado com AT+IPR=0), e detecta velocidades de 1200bps a 115200bps.

A serial deve estar configurada para 8 bits, 1 stop bit e sem paridade.

Dependendo da velocidade da serial, pode ser necessário entrar 2 ou 3 vezes AT<enter> ou at<enter> para que a velocidade seja detectada (quando isto ocorre o modem devolve "OK").

Para desativar o auto-bauding deve-se fornecer o comando AT+IPR=x, onde x é a velocidade da serial. Esta configuração é armazenada automaticamente na memória não-volátil e será validada quando o modem for reiniciado e valerá apenas para a serial que forneceu o comando.

Exemplo: AT+IPR=9600 fixa a velocidade da serial em 9600bps

Exemplos Comandos Rádio Modens Dataregis WSL

Comandos de IP:

1) Programa os parâmetros de conexão que serão usados pelos sockets definidos em seguida:

Defina um parâmetro de conexão GPRS	at+wtcs=0,contype,GPRS0 OK	//Define o tipo de conexão: GPRS
	at+wtcs=0,user,dtgsm OK	//Define o usuário PPP: dtgsm
	at+wtcs=0,passwd,dtgsm OK	//Define a senha PPP: dtgsm
	at+wtcs=0,apn,dtgsm.br OK	//Define a APN: dtgsm.br
	at+wtcs=0,authmode,PAP OK	//Define o protocolo de autenticação: PAP

Outro parâmetro de conexão GPRS, agora com outra APN	at+wtcs=1,contype,GPRS0 OK	//Define o tipo de conexão: GPRS
	at+wtcs=1,user,tim2 OK	//Define o usuário PPP: tim
	at+wtcs=1,passwd,tim2 OK	//Define a senha PPP: tim
	at+wtcs=1,apn,wap.tim.br OK	//Define a APN: wap.tim.br
	at+wtcs=1,authmode,PAP OK	//Define o protocolo de //autenticação: PAP

Defina um parâmetro de conexão CSD	at+wtcs=2,contype,CSD OK	//Define o tipo de conexão: CSD
	at+wtcs=2,user,testecsd OK	//Define o usuário PPP: testecsd
	at+wtcs=2,passwd,csd OK	//Define a senha PPP: csd
	at+wtcs=2,callednum,12345678 OK	//Define o número a ser discado: 12345678
	at+wtcs=2,authmode,PAP OK	//Define o protocolo de //autenticação: PAP

Define um parâmetro de conexão Ethernet (apenas para o modelo Evolution NET)

at+wtcs=3,contype,ETH OK	//Define o tipo de conexão: //Ethernet
-----------------------------	--

2) Programa os parâmetros de sockets TCP ou UDP que em seguida serão usados:

Parâmetros de sockets UDP	at+wtss=0,svrtype,socket OK at+wtss=0,conid,0 OK at+wtss=0,address,sockudp://10.205.10.10:5490 OK	//Define o socket 0 //O socket 0 usará as configurações de //parâmetro de conexão definida por //at+wtcs=0 //Define o socket 0 como cliente UDP //e define o IP do servidor que o socket //tentará se conectar
Parâmetros de sockets TCP	at+wtss=2,svrtype,socket OK at+wtss=2,conid,0 OK at+wtss=2,address,socktcp://10.205.10.15:3490 OK	//Define o socket 2 //O socket 2 usará as configurações de //parâmetro de conexão definida por //at+wtcs=0 //Define o socket 2 como cliente TCP //e define o IP do servidor que o socket //tentará se conectar
Parâmetros de sockets TCP	at+wtss=3,svrtype,socket OK at+wtss=3,conid,1 OK at+wtss=3,address,socktcp://10.201.1.30:4490 OK	//Define o socket 3 //O socket 3 usará as configurações de //parâmetro de conexão definida por //at+wtcs=1 //Define o socket 3 como cliente TCP //e define o IP do servidor que o socket //tentará se conectar
Parâmetros de sockets TCP	at+wtss=4,svrtype,socket OK at+wtss=4,conid,2 OK at+wtss=4,address,socktcp://200.98.2.10:23902 OK	//Define o socket 4 //O socket 4 usará as configurações de //parâmetro de conexão definida por //at+wtcs=2 //Define o socket 4 como cliente TCP //e define o IP do servidor que o socket //tentará se conectar
Parâmetros de sockets TCP (servidor TCP)	at+wtss=5,svrtype,socket OK at+wtss=5,conid,0 OK at+wtss=5,address,socktcp://listener:50001 OK	//Define o socket 5 //O socket 5 usará as configurações de //parâmetro de conexão definida por //at+wtcs=0 //Define o socket 5 como servidor TCP
Parâmetros de sockets TCP (cliente TCP)	at+wtss=6,svrtype,socket OK at+wtss=6,conid,3 OK at+wtss=6,address,socktcp://192.168.0.20:50006 OK	//Define o socket 6 //O socket 6 usará as configurações de //parâmetro de conexão definida por //at+wtcs=3 (Ethernet) //Define o socket 6 como cliente TCP

3) Abre os sockets definidos por AT+WTSS

Abre o socket 2	at+wtso=2 OK +WTSW: 2, 1	//Abre o socket 2 (e inicia a primeira //conexão GPRS) //URC informando que o socket está //pronto para transmitir dados
Abre o socket 0	at+wtso=0 OK +WTSW: 0, 1	//Abre o socket 0 //URC informando que o socket está //pronto para transmitir dados
Abre o socket 6	at+wtso=6 OK +WTSW: 6, 1	//Abre o socket 6 //URC informando que o socket está //pronto para transmitir dados
Abre o socket 3	at+wtso=3 OK +WTSW: 3, 1	//Abre o socket 3 (e inicia a segunda //conexão GPRS)
Abre o socket 5	at+wtso=5 OK +WTSW: 5, 1	//Abre o socket 5 //Obs: <i>pode-se informar aos clientes</i> // remotos o número de IP do modem //através de SMS (ver exemplo no final //do documento)

Comando AT+WTSO?
retorna (modelo com 10 sockets):

```
+WTSO: 0, "Socket", "4", "2", "0", "0", "0.0.0.0", "10.205.10.10:5490"
+WTSO: 1, ""
+WTSO: 2, "Socket", "4", "2", "0", "0", "10.205.181.135:49233", "10.205.10.15:3490"
+WTSO: 3, "Socket", "4", "2", "0", "0", "10.201.72.231:49234", "10.205.1.30:4490"
+WTSO: 4, "Socket", "2", "1", "0", "0", "0.0.0.0", "0.0.0.0"
+WTSO: 5, "Socket", "3", "3", "0", "0", "10.205.181.135:50001", "0.0.0.0"
+WTSO: 6, "Socket", "4", "2", "0", "0", "192.168.0.50:49237", "192.168.0.20:50006"
+WTSO: 7, ""
+WTSO: 8, ""
+WTSO: 9, ""
OK
```

4) Transfere dados pelos sockets abertos e os fecha:

<pre>at+wtsw=0,200 +WTSW: 0, 200 <O host deve fornecer os dados a serem transmitidos> OK</pre>	//Transfere 200 caracteres pelo socket 0
<pre>+WTSW: 0, 1 +WTSR: 0, 1</pre>	<pre>//URC: os dados fornecidos foram //enviados //URC: novos dados foram recebidos //pelo socket 0 //Obs: esta URC só é apresentada ao //chegarem os primeiros dados após o //socket ser aberto ou quando na última //leitura por at+wtsr havia menos dados //disponíveis que a quantidade solicitada //e novos dados foram recebidos</pre>
<pre>at+wtsr=0,200 +WTSR: 0, 200 ABCDEFGHI@ABCDEFGHI@ABCDEFGHI@ABCDE FGHI@ABCDEFGHI@ABCDEFGHI@ABCDEFGHI@ ABCDEFGHI@ABCDEFGHI@ABCDEFGHI@ABCDE FGHI@ABCDEFGHI@ABCDEFGHI@ABCDEFGHI@ ABCDEFGHI@ABCDEFGHI@ABCDEFGHI@ABCDE FGHI@ABCDEFGHI@ABCDEFGHI@ OK</pre>	<pre>//Solicita 200 caracteres do socket 0 //O socket confirma que há 200 //caracteres disponíveis</pre>
<pre>+WTSR: 2, 1</pre>	//URC: novos dados foram recebidos //pelo socket 2
<pre>at+wtsr=2,20 +WTSR: 2, 10 ABABABABAB OK</pre>	<pre>//Solicita 20 caracteres do socket 2 //O socket 2 confirma o recebimento de //10 //caracteres</pre>
<pre>+WTSR: 2, 1</pre>	//URC: novos dados foram recebidos pelo //socket 2
<pre>at+wtsr=2,30 +WTSR: 4, 16 ABCDABCDABCDABCD OK</pre>	<pre>//Tenta ler mais 30 caracteres, mas só há 16 //disponíveis no buffer interno do socket 4</pre>
<pre>at+wtsc=2 OK</pre>	//Fecha o socket 2
<pre>+WTSO: 5, 1, 1</pre>	//Indica que um cliente está tentando se //conectar com a porta definida pelo socket 5 e este foi alocado no socket 1
<pre>at+wtso=1 OK +WTSW: 1, 1</pre>	//Aceita o cliente
<pre>+WTSR: 1, 1 at+wtsr=1,50 +WTSR: 1, 50</pre>	//Lê 50 caracteres do socket 1
<pre>þÿüúúø+ðóóòññííëëèèçæääååååååßÞÝÛÛÛÛÛÛ×ÖÕ ÓÓÑÑİİ OK</pre>	

Comando AT+WTSO?

retorna (modelo com 10 sockets):

at+wtso?

```
+WTSO: 0, "Socket", "4", "2", "0", "0", "10.205.181.135:49232", "10.205.10.10:5490"  
+WTSO: 1, "Socket", "4", "4", "0", "0", "10.205.181.135:49235", "10.205.129.14:21340"  
+WTSO: 2, "Socket", "2", "1", "0", "0", "0.0.0.0:0", "0.0.0.0:0"  
+WTSO: 3, "Socket", "4", "2", "0", "0", "10.201.72.231:49234", "10.205.1.30:4490"  
+WTSO: 4, "Socket", "2", "1", "0", "0", "0.0.0.0:0", "0.0.0.0:0"  
+WTSO: 5, "Socket", "3", "3", "0", "0", "10.205.181.135:50001", "0.0.0.0:0"  
+WTSO: 6, "Socket", "4", "2", "0", "0", "192.168.0.50:49237", "192.168.0.20:50006"  
+WTSO: 7, ""  
+WTSO: 8, ""  
+WTSO: 9, ""  
OK
```

```
at+wtsc=0 //Fecha socket 0  
OK  
+WTS: 1, 0 //Indica que o cliente conectado ao socket 1  
derrubou //a conexão  
+WTS: 3, 0 //Indica que o servidor conectado ao socket  
//3 derrubou //a conexão -neste instante a  
//segunda conexãoGPRS aberta é desfeita  
at+wtsc=5 //Fecha socket 5 – neste instante a primeira  
conexão  
OK //GPRS aberta é desfeita
```

Nota

Neste instante, como não há nenhuma conexão GPRS, é possível abrir o socket que usa os parâmetros de conexão CSD:

```
at+wtso=4 //Abre o socket 4 – e inicia a conexão CSD  
OK  
+WTSW: 4, 1  
+WTSR: 4, 1  
at+wtso=4,20  
+WTSR: 4, 5  
ABCDE  
OK  
at+wtsc=4 //Fecha o socket 4 – neste instante a  
conexão CSD é desfeita  
OK
```

Comandos de leitura de id e de versão:

at+wtid

+WTID:1,"11050000000056","351248000504043","89550311000026306789"

OK

at+wtvr

+WTVR:"DATAREGIS VERSAO 1.1","SIEMENS Gipsy Soft Protocolstack V2.550"

OK

Comandos de configuração do protocolo TCP:

<i>at+wtac=3</i> OK	//Configura o timeout de reenvio de pacotes //TCP em 3 segundos
<i>at+wrt=4</i> OK	//Configura o número máximo de //retransmissões de um pacote TCP = 4

Exemplo de uso de mensagens SMS:

Pode-se usar mensagens de SMS para informar aos clientes qual o número de IP obtido pelo modem quando há um socket ouvinte (listener), já que o IP obtido pelo PPP é dinâmico.

MODEM SERVIDOR:

```
at+wtso=4 //Abre um socket ouvinte
OK

at+wtso? //Verifica o número de IP obtido
...
+WTSO: 4, "Socket", "3", "3", "0", "0", "10.205.181.135:50001", "0.0.0.0:0"
...

at+cmgf=1 //Seleciona envio no modo texto
OK

at+cmgs=87654321 //Envia uma mensagem ao modem cliente
// (87654321 é o número de telefone do
// modem cliente)
>> Escrever a mensagem (no caso, o IP obtido), finalizá-la com (CTRL+Z) ou 0x1A
Deve-se aguardar a confirmação do envio antes de pressionar alguma tecla:
+CMGS: XX
OK
```

MODEM CLIENTE:

```
at+cmgf=1 //Seleciona recebimento no modo texto
OK

at+cmgl=all //Verifica se há mensagens SMS disponíveis
+CMGL: 1, "REC UNREAD", "01187872222", "05/11/18, 13:54:25+04"
10.201.181.135 //Mensagem com o IP do servidor
OK

at+cmgd=1 //Apaga a mensagem recebida
OK

at+wtss=2,address,socktcp://10.201.181.135:50001 //Cria um socket cliente
OK //apontando para o IP do modem servidor

at+wtso=2 //Abre o socket, conectando-se ao servidor
OK
```

Exemplo de realocação de seriais Rádio Modem Evolution3:

Exemplo de uso do comando AT+WTTK para realocação de conexões GPRS (válido apenas para o Modem com 3 seriais):

Exemplo1:

Neste exemplo o valor de WTTK é igual a 10 segundos para todas as seriais

SERIAL1:	<i>ATD*99***1#</i>	//Serial1 se conecta	-	9:00:00
SERIAL1:	<i>CONNECT</i>			
SERIAL2:	<i>ATD*99***1#</i>	//Serial2 se conecta	-	9:01:00
SERIAL2:	<i>CONNECT</i>			
SERIAL1:		//Serial1 recebe pacote	-	9:01:40
SERIAL2:		//Serial2 transmite pacote	-	9:02:00
SERIAL1:		//Serial1 transmite pacote	-	9:02:03
SERIAL3:	<i>ATD*99***1#</i>	//Serial3 tenta conexão enquanto as seriais 1 e 2 //já estão conectadas	-	9:02:05
SERIAL2:		//Serial2 recebe pacote	-	9:02:08
SERIAL3:	<i>CONNECT</i>	//Serial 3 consegue conexão e conexão da serial1		
SERIAL1:	<i>NO CARRIER</i>	//é derrubada, após 10 segundos de inatividade //na serial1 (recebe NO CARRIER) e serial3 //recebe CONNECT	-	9:02:13

Exemplo2:

Neste exemplo o valor de WTTK é igual a 10 segundos para todas as seriais

SERIAL1:	<i>ATD*99***1#</i>	//Serial1 se conecta	-	9:00:00
SERIAL1:	<i>CONNECT</i>			
SERIAL2:	<i>ATD*99***1#</i>	//Serial2 se conecta	-	9:01:00
SERIAL2:	<i>CONNECT</i>			
SERIAL1:		//Serial1 recebe pacote	-	9:01:40
SERIAL2:		//Serial2 transmite pacote	-	9:02:00
SERIAL1:		//Serial1 transmite pacote	-	9:02:03
SERIAL3:	<i>ATD*99***1#</i>	//Serial3 tenta conexão enquanto as seriais 1 e 2 //já estão conectadas	-	9:02:05
SERIAL2:		//Serial2 recebe pacote	-	9:02:08
SERIAL1:		//Serial1 transmite pacote	-	9:02:09

```

SERIAL3:  NO DIALTONE           //Serial 3 não consegue conexão, pois dentro do tempo de 10
                                           //segundos de espera não houve um período de inatividade maior que
                                           //10 segundos para nenhuma das outras duas seriais já conectadas e
                                           //permanecem conectadas) -           9:02:15

```

Exemplo3:

Neste exemplo o valor de WTTK é igual a 15 segundos para a serial1, 0 para a serial2 (esta serial nunca será derrubada) e 20 para a serial3

```

SERIAL1:  ATD*99***1#           //Serial1 se conecta -           9:00:00
SERIAL1:  CONNECT
SERIAL2:  ATD*99***1#           //Serial2 se conecta -           9:01:00
SERIAL2:  CONNECT
SERIAL1:                               //Serial1 recebe pacote -           9:01:40
SERIAL2:                               //Serial2 transmite pacote -           9:02:00
SERIAL1:                               //Serial1 transmite pacote -           9:05:23
SERIAL3:  ATD*99***1#           //Serial3 tenta conexão enquanto as seriais 1 e 2
                                           //já estão conectadas -           9:05:27
SERIAL1:                               //Serial1 transmite pacote -           9:05:36
SERIAL3:  NO DIALTONE           //Serial 3 não consegue conexão, pois dentro do tempo de 20
                                           //segundos de espera não houve um período de inatividade maior que
                                           //15 segundos para a serial1 e apesar de a serial2 não transferir
                                           //pacotes a mais de 3 minutos, como seu valor de timeout é igual a 0,
                                           //ela nunca será derrubada -           9:05:47

```

Serviço de Atendimento ao Cliente

O Serviço de Atendimento ao Cliente da DATAREGIS compromete-se a ajudá-lo a obter um ótimo rendimento do seu equipamento.

Caso existam dúvidas ou observações sobre este manual, favor entrar em contato com nosso Serviço de Atendimento ao Cliente.

Tenha em mãos o número de série de seu equipamento.

DATAREGIS

SAC (Serviço de Atendimento ao Cliente)

Tel.: (0xx11) 5682-6644 / Fax: (0xx11) 5682-6633

Assunto: Rádio Modem WSL

e-mail: sac@dataregis.com.br

Site: www.dataregis.com.br



Rua Dom Aguirre, 190 – Jd. Marajoara – CEP 04671-245
São Paulo – SP – Fone: (11) 5682 6644 – Fax (11) 5682 6633
Site:www.dataregis.com.br