



**HI** tecnologia

**Indústria e Comércio Ltda**

---

**Notas de Aplicação**

**Uso do Banco de Dados em  
*Flash ROM* nos Controladores  
da HI Tecnologia**

---

---

**HI Tecnologia**

---

Documento de acesso público

---

**ENA.00017**

**Versão 1.02**

**abril-2009**

---

## Apresentação

---

Esta nota de aplicação foi elaborada pela **HI Tecnologia Indústria e Comércio Ltda.** Dúvidas ou esclarecimentos sobre as informações contidas neste documento podem ser obtidas diretamente com o departamento de suporte a clientes, através do telefone (19) 2139-1700 ou do e-mail "[suporte@hitecnologia.com.br](mailto:suporte@hitecnologia.com.br)". Favor mencionar as informações a seguir para que possamos identificar os dados relativos a este documento.

ID da Nota de Aplicação: ENA.00017  
Versão Documento: 1.02

---

## HI Tecnologia Indústria e Comércio Ltda.

Endereço: Av. Dr. Armando de Sales Oliveira, 445

Cidade: Campinas – SP  
CEP: 13076-015

Fone: +55 (19) 2139-1700  
Fax: +55 (19) 2139-1710

*E-mail:* [hi@hitecnologia.com.br](mailto:hi@hitecnologia.com.br)

*Web site:* [www.hitecnologia.com.br](http://www.hitecnologia.com.br)

---

**Este documento abrange os seguintes Controladores nas plataformas especificadas abaixo.**

Equipamentos	Família	Modelo	Plataformas			Abrangência
			GI	GII	GII_DUO	
<b>Controladores</b>	MCI02	MCI02	X			√
		MCI02-QC	X			√
	ZAP500	ZAP500	X			√
		ZAP500-BX	X			√
		ZAP500-BXH	X			√
		ZTK500	X			√
		ZTK501	X			√
		ZAP900		X		√
	ZAP900	ZAP901		X		√
		eZAP900		X		√
		eZAP901		X		√
		ZAP900-BXH		X		√
		ZTK900		X		√
		eZTK900		X		√
	FLEX950	FLEX950-PLC		X		√
	P7C	P7C			X	√
<b>IHMs</b>	MMI600	MMI600		X		
		MMI601		X		
	MM650	MMI650		X		
	MMI800	MMI800		X		
	FLEX950	FLEX950-IHM		X		



# Uso do Banco de Dados em Flash ROM nos Controladores da HI Tecnologia

Referência: ENA.00017  
Arquivo : ENA0001700.doc

Revisão: 2  
Atualizado em: 07/04/2009

## Índice

1.	Introdução .....	5
1.1	Capacidade de Armazenamento.....	5
1.2	Funções de Acesso ao Banco de Dados.....	6
1.3	Limites nos Ciclos de Acesso ao Banco de Dados.....	6
2.	Referências.....	7
3.	Definições .....	7
4.	O Banco de Dados em <i>Flash ROM</i> dos Controladores HI.....	7
5.	Configuração do BD nos Controladores HI .....	8
5.1	Configuração via SPDSW .....	8
5.2	Configuração via SPDS7 .....	11
6.	Descrição das Funções para Acesso ao BD .....	13
6.1	Selecionando o Bloco SCB no Ambiente SPDS.....	14
6.1.1	Utilizando o SPDSW : .....	14
6.1.2	Utilizando o SPDS7 : .....	15
6.2	Função de Leitura do BD .....	16
6.3	Função de Escrita no BD .....	17
6.4	Códigos de Retorno das Funções de Acesso ao BD .....	18
7.	Exemplos de Utilização .....	19
7.1	Exemplo 1 : Registro de Dados .....	19
7.1.1	Leitura do Banco de Dados em <i>Flash ROM</i> .....	19
7.1.2	Escrita no Banco de Dados em <i>Flash ROM</i> .....	20
7.2	Exemplo 2 : Parâmetros de Configuração .....	21
7.2.1	Leitura dos Parâmetros de Configuração do BD.....	22
7.2.2	Escrita dos Parâmetros de Configuração no BD.....	23
7.3	Exemplo 3 : Receitas .....	24
7.3.1	Leitura de uma Receita do BD.....	24
7.3.2	Escrita de uma Receita no BD.....	26
	Controle do Documento.....	27
	Considerações gerais .....	27
	Responsabilidades pelo documento.....	27



## 1. Introdução

---

Este documento descreve como utilizar o Banco de Dados em *Flash ROM* disponível em todos os controladores da HI Tecnologia (ZAP500/501, MIX600, MCI02, MCI02-QC, ZAP900/901, FLEX950 e P7C).

O Banco de Dados em *Flash ROM* pode armazenar de 12 *Kbytes* a 32 *Kbytes* de dados não voláteis (não são perdidos quando se desliga o equipamento) dependendo da família do controlador, e pode ser utilizado através dos ambientes de programação SPDS7 e SPDSW.

O Banco de Dados pode ser utilizado para:

- Armazenar parâmetros de “*setup*” da máquina;
- Armazenar receitas;
- Salvar dados do processo (alarmes, dados históricos, etc.) para uma futura aquisição.

Este documento é dividido nas seguintes seções:

- Descrição do Banco de Dados.
- Descrição das funções de acesso ao Banco de Dados.
- Exemplo de utilização.

### 1.1 Capacidade de Armazenamento

---

Considerando os tipos de variáveis R, M e D dos controladores da HI Tecnologia, se a utilizarmos para armazenar apenas um único tipo de variável, este seria capaz de armazenar:

1. Considerando capacidade de 12 *Kbytes* do Banco de Dados

- 12.288 contatos auxiliares do tipo R (1 *byte*), ou
- 6.144 memórias inteiras do tipo M (2 *bytes*), ou
- 3.072 memórias reais do tipo D (4 *bytes*).

2. Considerando capacidade de 16 *Kbytes* do Banco de Dados

- 16.384 contatos auxiliares do tipo R (1 *byte*), ou
- 8.192 memórias inteiras do tipo M (2 *bytes*), ou
- 4.096 memórias reais do tipo D (4 *bytes*).

3. Considerando capacidade de 32 *Kbytes* do Banco de Dados

- 32.768 contatos auxiliares do tipo R (1 *byte*), ou
- 16.384 memórias inteiras do tipo M (2 *bytes*), ou
- 8.192 memórias reais do tipo D (4 *bytes*).



## Uso do Banco de Dados em Flash ROM nos Controladores da HI Tecnologia

Tipo de Doc.: Notas de Aplicação  
Referência: ENA.00017

Revisão: 2  
Atualizado em: 07/04/2009

Nas aplicações é possível mesclar os tipos de variáveis a serem salvas no Banco de Dados, de modo a permitir salvar registros de dados compostos pelas variáveis R, M e D. De maneira análoga permite-se a leitura de registros de dados compostos por variáveis R, M e D.

**Obs: A capacidade do banco de dados é característica de cada família de controladores da HI Tecnologia.**

### 1.2 Funções de Acesso ao Banco de Dados

---

Para a gerência do acesso de escrita / leitura no Banco de Dados, têm-se as seguintes funções disponíveis:

- Leitura de dados armazenados no Banco de Dados.
- Escrita de dados no Banco de Dados.

Essas funções são detalhadas no item "6 -



Descrição das Funções para Acesso ao BD”.

### 1.3 Limites nos Ciclos de Acesso ao Banco de Dados

Memórias *Flash ROM* são uma evolução das memórias chamadas *EEPROM* (“*Electric Erasable Read Only Memory*”), permitindo serem apagadas e reescritas via *software*. Essas memórias não são voláteis e, portanto não perdem o seu conteúdo quando se desliga o controlador. A diferença entre a memória *Flash* e a *EEPROM* é que a memória *Flash* permite que apenas parte da memória seja escrita durante o processo de escrita, enquanto que na memória *EEPROM* é necessário escrever a memória inteira, mesmo que você precise alterar o valor de apenas um *byte*.

Em linhas gerais a memória *Flash* é uma memória que depende de uma carga elétrica para ser escrito um valor específico e depois de uma descarga para ser gravado o valor inverso. Esse processo de regravação ou reescrita é limitado. Os fabricantes garantem um número máximo de ciclos de escritas e, portanto não são indicadas para escritas repetitivas, enquanto que para leitura essas memórias podem ser acessadas sem restrições.

O fabricante das memórias *flash* utilizadas nos controladores HI especifica o limite de **10.000** ciclos de escrita efetivos em cada conjunto de 128 *bytes*, denominados de páginas da memória no BD. Novamente frisamos que não existem limites de acesso nos ciclos de leitura no BD.

Para salvar dados repetidamente, tais como dados históricos, pode-se utilizar à memória *NVRAM* (memória não volátil) ou a memória *DataFlash*. Memórias do tipo *NVRAM* são opcionais em algumas linhas de controladores da HI, e memórias do tipo *DataFlash* estão disponíveis somente em algumas de nossas linhas de controladores. Consulte o nosso departamento de vendas ou o nosso *website* ([www.hitecnologia.com.br](http://www.hitecnologia.com.br)) para obter maiores informações.

## 2. Referências

- Controlador Industrial MCI02-QC  
  
PDF.102050.00 – Descritivo funcional do controlador industrial MCI02 QC  
PET.102050.00 – Folha de especificação técnica do controlador industrial MCI02 QC
- Controlador Industrial ZAP500  
  
PDO.09001.01 – Operação da IHM do controlador ZAP500 - Descritivo Operacional  
PDF.09001.00 – Descritivo de hardware do controlador industrial ZAP500
- Controlador Industrial ZAP900  
  
PET.108.001.00 – Controlador Lógico Programável ZAP900/901 - Especificação Técnica



## Uso do Banco de Dados em Flash ROM nos Controladores da HI Tecnologia

Tipo de Doc.: Notas de Aplicação  
Referência: ENA.00017

Revisão: 2  
Atualizado em: 07/04/2009

- Controlador Industrial P7C  
PMU.10700100 – Manual do usuário.
  - Dispositivo de comunicação e armazenamento SCD912  
PET.91291200 – Folha de especificação técnica do dispositivo SCD912
  - Programas de Exemplo:  
HIPJT008 – Utilização do Banco de Dados em Flash ROM nos controladores da HI Tecnologia.
  - Aplicativos:  
SPDSW – Ambiente de programação *Ladder* dos controladores HI Tecnologia  
OPPE – Ambiente de programação das IHM's HI Tecnologia
- Todos os documentos, programas de exemplo e aplicativos referenciados acima estão disponíveis para *download* no *site* da HI Tecnologia.

[www.hitecnologia.com.br](http://www.hitecnologia.com.br)

### 3. Definições

---

BD – **B**anco de **D**ados em *Flash ROM* dos controladores da linha HI Tecnologia.

### 4. O Banco de Dados em *Flash ROM* dos Controladores HI

---

O Banco de Dados corresponde a uma página de 12,16 ou 32 *Kbytes* (dependendo da família) da memória *Flash ROM* existente nos módulos de CPU dos controladores HI. Ou seja, todos os controladores HI possuem esta funcionalidade disponível. Esta página de 12,16 ou 32 *Kbytes* pode ser vista como uma base de dados secundária de variáveis do tipo R, M e D, ou seja, pode-se mapeá-la em até 3 segmentos de variáveis (R, M e D) de acordo com a necessidade de cada aplicação.

Os dados armazenados neste banco de dados em *Flash ROM* **NÃO** estão disponíveis para acesso direto como as variáveis R, M e D da base de dados do CONTROLADOR. Para acessá-los tem-se disponíveis funções para leitura e escrita no banco de dados, e que por sua vez devem ser manipuladas pelo respectivo programa *ladder* de aplicação, como ilustra a figura abaixo.

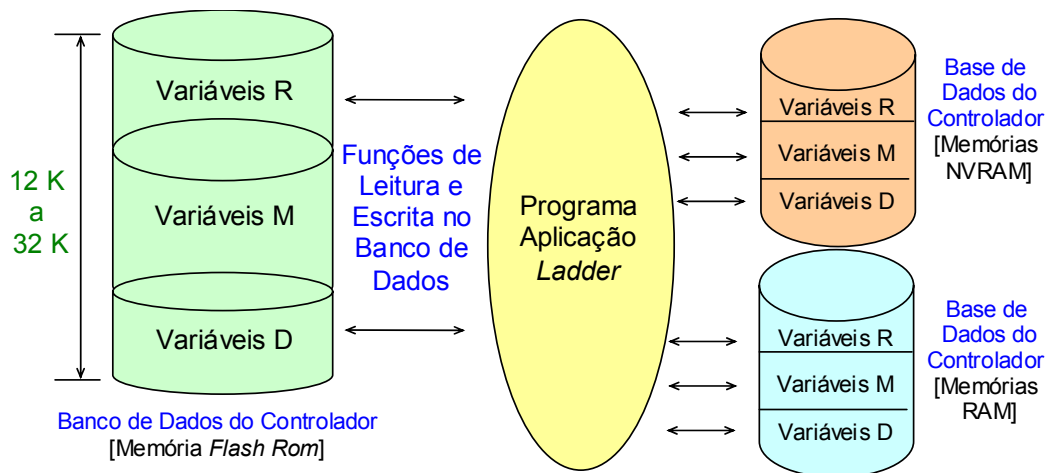


Figura 1 – Banco de Dados e Base de Dados do controlador

Através desta arquitetura o programa de aplicação *ladder* pode salvar conjuntos de dados (do tipo R, M e D) da base de dados do controlador no respectivo banco de dados e vice-versa. Estes conjuntos de dados podem ser, por exemplo, variáveis associadas a parâmetros de configuração do sistema, receitas, dados históricos, etc.

É importante salientar que uma função de leitura / escrita não pode mesclar dados das memórias *NVRAM* e *RAM*, ou seja, todos os valores lidos / escritos da / na base de dados do controlador devem estar alocados na memória *NVRAM* ou na *RAM*.

## 5. Configuração do BD nos Controladores HI

Para utilizar o BD nos controladores HI, é necessário configurar a quantidade de variáveis que serão mapeadas no BD para cada tipo de variável (R, M e D). Para tal pode-se utilizar os aplicativos SPDS7 (G-I) e/ou SPDSW (todos), ambos disponíveis para *download* gratuito em nosso site: [www.hitecnologia.com.br](http://www.hitecnologia.com.br).

### 5.1 Configuração via SPDSW

Para o caso de se utilizar o **SPDSW**, deve-se prosseguir da seguinte forma:

Com o SPDSW em execução selecionar o menu "**Programa / Setup do Programa**". Conforme apresentado na figura a seguir:



Figura 2 – Tela do Menu para a Configuração do Banco de Dados no SPDSW

Pode também ser utilizado o atalho **Ctrl+F4**. Ao abrir sua Aplicação no Ambiente SPDSW pressione as teclas **Ctrl+F4**.

Ao utilizar ambas as opções, será apresentada uma janela, como a figura a seguir:

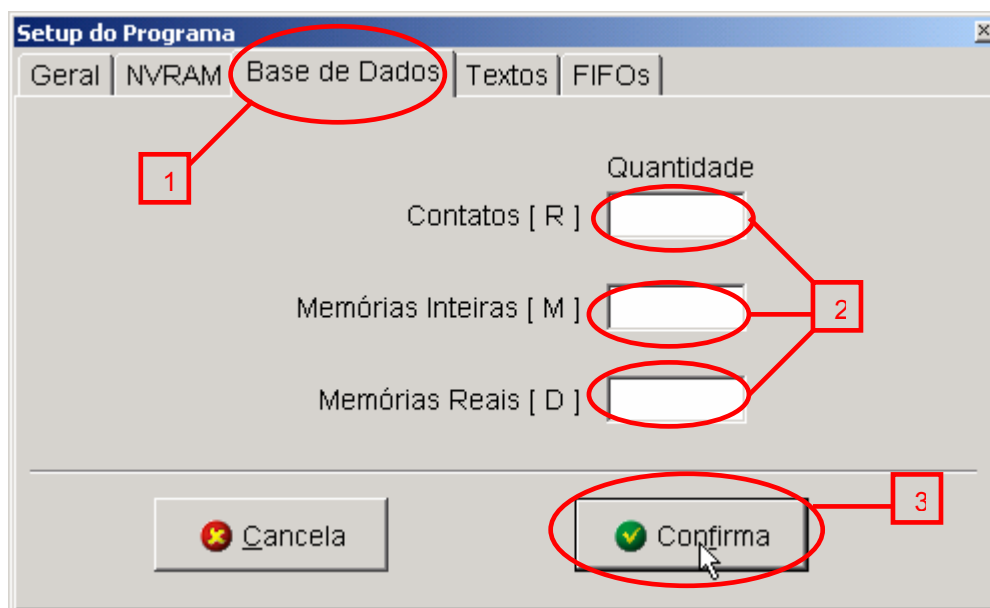


Figura 3 – Tela de Configuração do Banco de Dados em Flash ROM via SPDSW

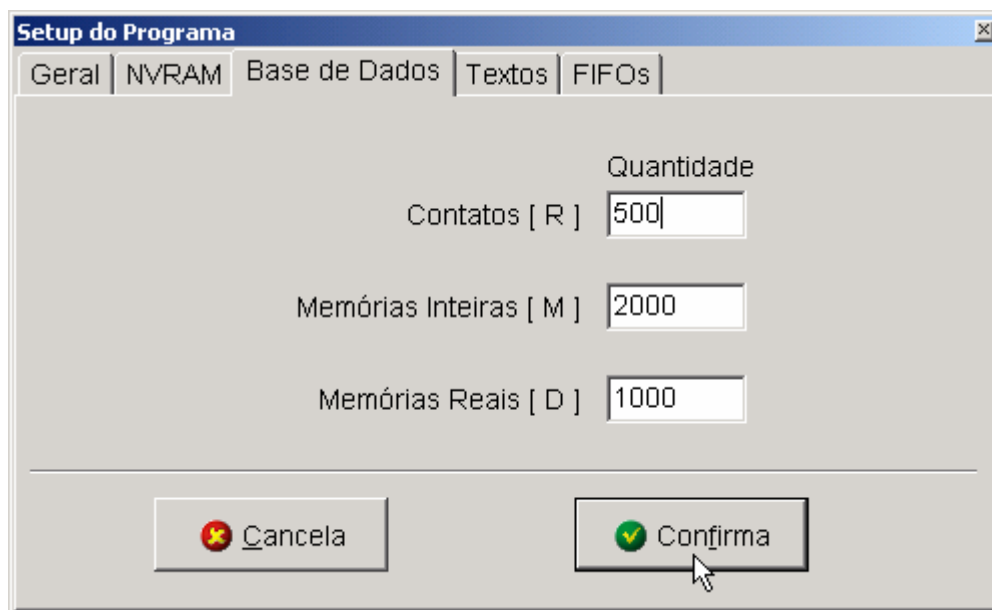
Nesta janela devemos prosseguir da seguinte forma:

1. Selecionar a aba “**Base de Dados**”
2. Especifique a quantidade de variáveis que serão salvas no BD para cada tipo (R, M e D).
3. Pressione o botão confirma e a especificação da quantidade de dados que poderá ser salva no BD estará concluída.

Por exemplo, se especificarmos a seguinte configuração das variáveis no banco de dados, contendo os três tipos de variáveis R, M e D :

- Quantidade de Contatos Auxiliares (R) = 500 variáveis
- Quantidade de Memórias Inteiras (M) = 2000 variáveis
- Quantidade de Memórias Reais (D) = 1000 variáveis

Exemplo de configuração do Banco de Dados na figura a seguir.



**Figura 4 – Tela de exemplo de Configuração do Banco de Dados em Flash ROM via SPDSW**

Teremos o seguinte mapeamento no banco de dados:

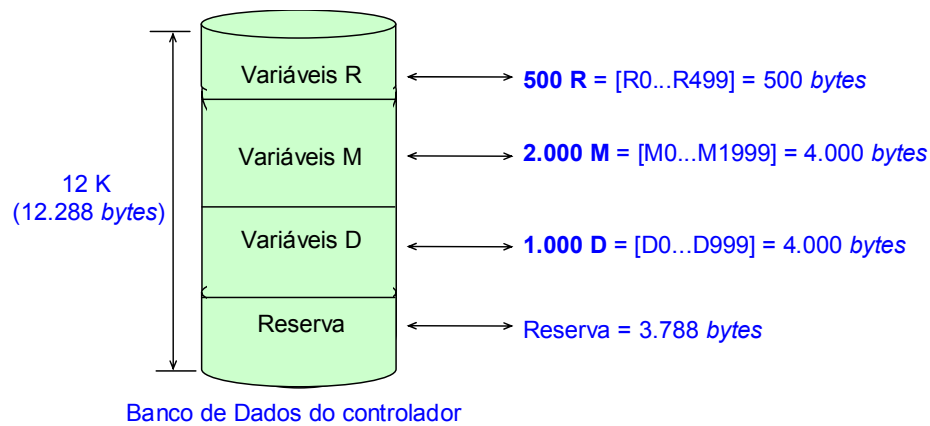


Figura 5 – Exemplo A de Configuração do Banco de Dados em Flash ROM

Em outro exemplo, se especificarmos a seguinte configuração das variáveis no banco de dados, contendo apenas memórias inteiras M :

- Quantidade de Contatos Auxiliares (R) = 0 variáveis
- Quantidade de Memórias Inteiras (M) = 8000 variáveis
- Quantidade de Memórias Reais (D) = 0 variáveis

Teremos o seguinte mapeamento no banco de dados :

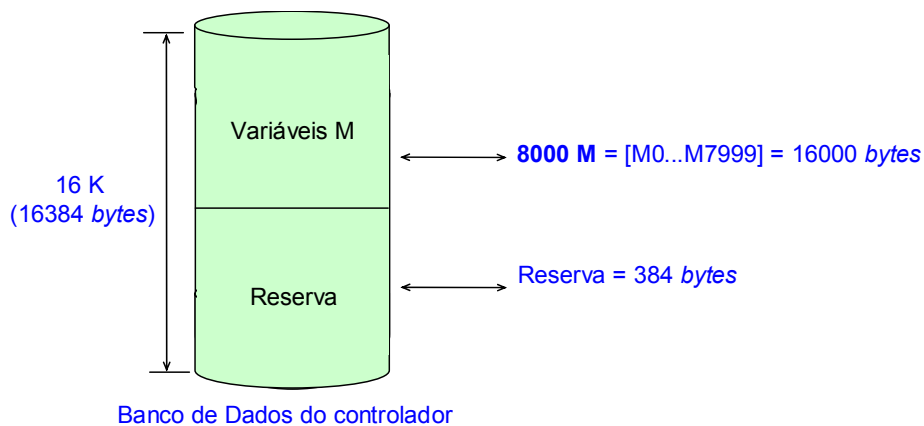


Figura 6 – Exemplo B de Configuração do Banco de Dados em Flash ROM

## 5.2 Configuração via SPDS7

Para o caso de se utilizar o **SPDS7**, deve-se:

Com o SPDS7 em execução seleccionar o menu “**Programa / Conf. Aplicação**”. Conforme apresentado na figura a seguir:



Figura 7 – Tela do Menu de Configuração do Banco de Dados em Flash ROM via SPDS7

Ao seleccionar a opção “Conf. Aplicação” será apresentada uma janela, como a figura a seguir:



Figura 8 – Tela de Configuração do Banco de Dados em Flash ROM via SPDS7

Nesta janela devemos prosseguir da seguinte forma:

1. Especifique a quantidade de variáveis que serão salvas no BD para cada tipo (R, M e D).
2. Pressione o botão confirma e a especificação da quantidade de dados que poderá ser salva no BD estará concluída.

## 6. Descrição das Funções para Acesso ao BD

Em aplicações desenvolvidas no ambiente SPDS deve-se utilizar o bloco SCB ("Bloco de Controle Padrão") para acessar o BD em operações de leitura e escrita de dados. O bloco SCB é composto por uma entrada E1, 4 parâmetros (P1, P2, P3 e P4) e uma saída S1, como ilustra a figura abaixo:

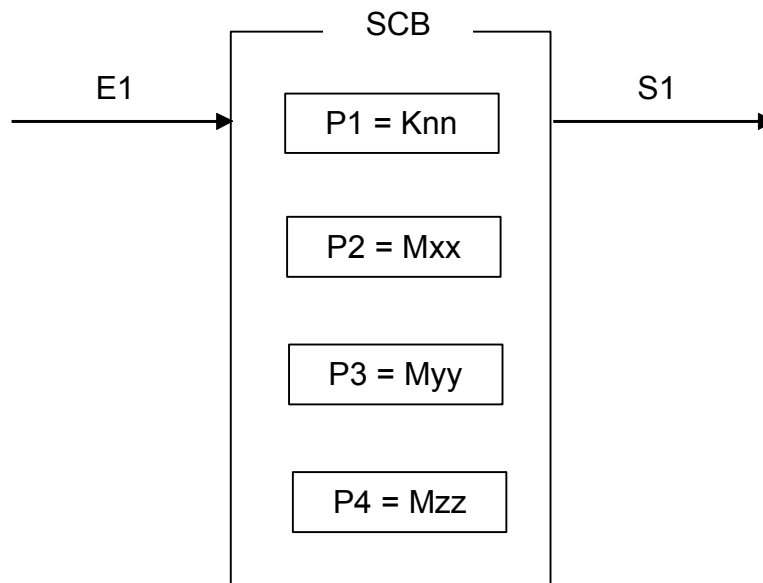


Figura 9 - Bloco "SCB"

Os elementos que compõem o bloco SCB são descritos abaixo:

- Parâmetro P1: Deve ser uma constante inteira do tipo **Knn**, onde o valor especificado para essa constante indica a função do bloco, podendo ser:
  - Igual a 1 para Leitura de variáveis do BD;
  - Igual a 2 para Escrita de variáveis no BD.
- Parâmetro P2: Deve ser uma memória inteira do tipo **Mxx**, a partir dessa deve existir uma seqüência de oito memórias consecutivas com parâmetros para a função. Exemplo: Sendo Mxx igual a M10, a seqüência será M10, M11, M12, ....., M18.
- Parâmetro P3: Deve ser uma memória inteira do tipo **Myy**, Este parâmetro não é utilizado nas funções de acesso ao BD, podendo ser especificada qualquer memória disponível para o mesmo.
- Parâmetro P4: Deve ser uma memória inteira do tipo **Mzz**. Nesta memória Mzz sempre será retornado o código de retorno da execução da função de acesso ao BD. Neste caso, se retornar um valor 0 (ZERO) indica função executada com sucesso, caso contrário indica o código de erro associado à execução da função selecionada.

- Entrada E1: sinal de habilitação do bloco SCB, onde:

Energizado - Bloco habilitado, executa função de acesso ao BD.

Desenergizado - Bloco desabilitado, não executa nenhuma função.

**Observação:** Cabe ressaltar que na funcionalidade deste bloco SCB, o tratamento da função de acesso ao BD somente é iniciado após a transição de subida desta entrada E1, devendo permanecer ativa até o término da execução da função, no caso indicado quando a saída S1 torna-se energizada.

- Saída S1: sinal de saída do bloco SCB, onde:

Energizado - Término da execução da função de acesso ao BD.

Desenergizado – Bloco não habilitado, ou está habilitado, mas ainda não concluiu a execução da função de acesso ao BD.

## 6.1 Selecionando o Bloco SCB no Ambiente SPDS

Para inserir um bloco SCB em seu programa de aplicação *ladder*, deve-se proceder como descrito a seguir, dependendo do ambiente em que se estiver trabalhando:

### 6.1.1 Utilizando o SPDSW :

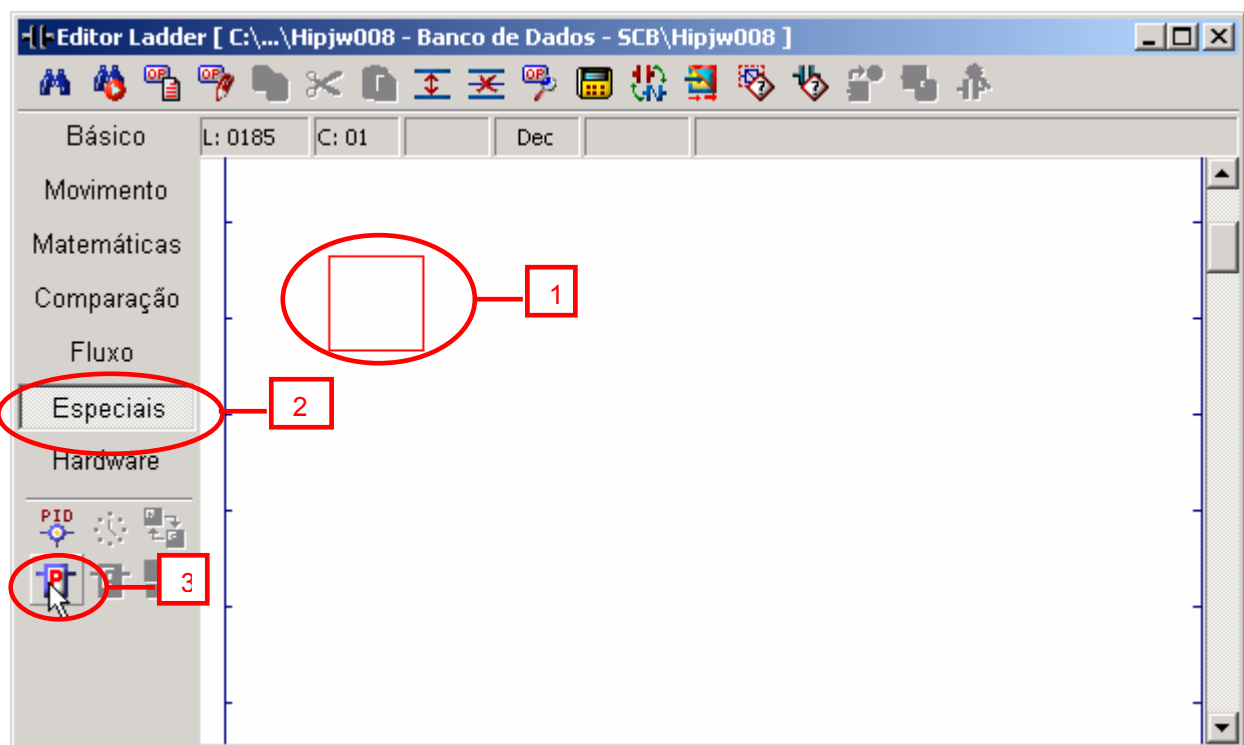



Figura 10 – Menu do Comando “Bloco SCB” no SPDSW

Prossiga da seguinte forma:

1. Posicione o cursor no local desejado do programa *ladder*.
2. Selecione a opção “Especiais” na palheta de comandos à esquerda da janela do editor *ladder*.
3. Em seguida selecione o botão .

### 6.1.2 Utilizando o SPDS7 :

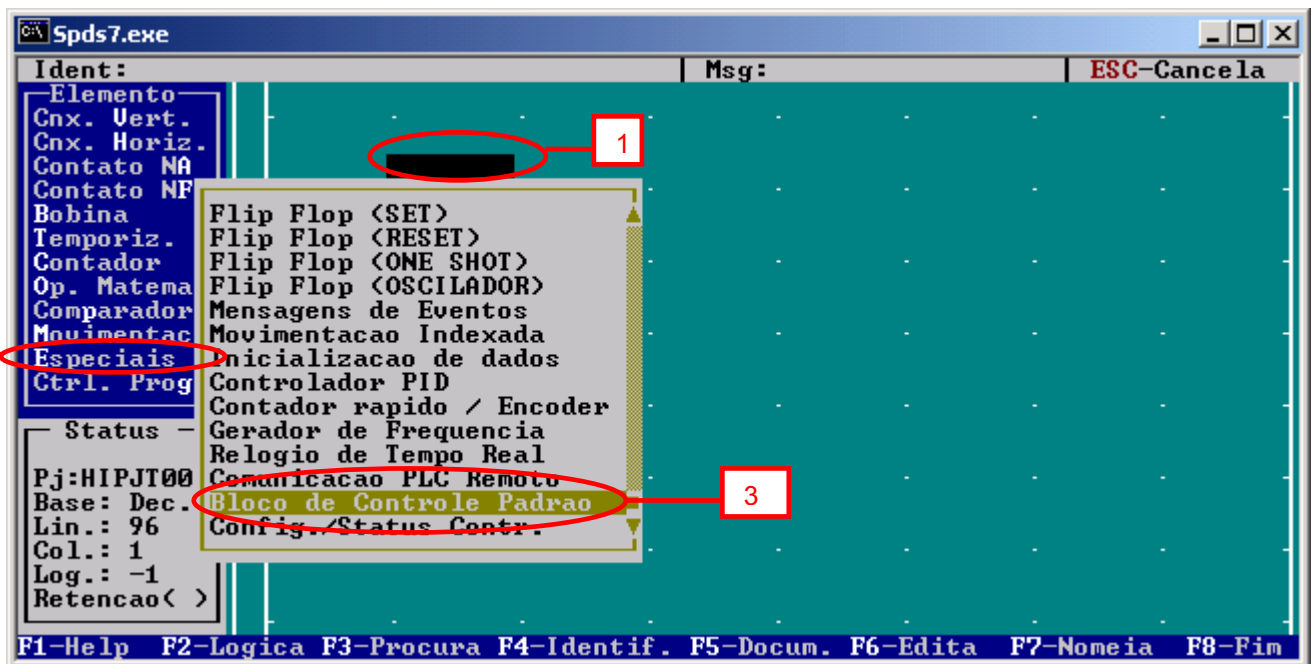


Figura 11 – Menu do Comando “Bloco de Controle Padrão” no SPDS7

Prossiga da seguinte forma:

1. Posicione o cursor no local desejado do programa *ladder* com quatro linhas disponíveis para inserir o bloco.
2. Pressione a tecla “E” para abrir a opção de comandos “Especiais” na palheta de comandos à esquerda da janela do editor *ladder*.
3. Em seguida selecione a opção “Bloco de Controle Padrão”.

O bloco SCB será inserido no programa de aplicação assim como mostra a figura 11.



## 6.2 Função de Leitura do BD

### Função 01

### Leitura

#### Descrição da Função

Ler um registro de dados do BD e transferi-lo para a base de dados do controlador. Considera-se como registro de dados uma combinação qualquer de variáveis dos tipos R, M ou D

#### P1: Parâmetros de Entrada

**Constante Knn** Função do bloco SCB de acesso de leitura no BD  
Deve ser uma constante inteira (Knn) cujo valor deve ser 1.

#### P2: Parâmetros de Entrada

- Memória xx + 0** Número inicial da variável tipo contato (R) ser lido no BD  
(Endereço origem R)
- Memória xx + 1** Quantidade de variáveis do tipo contato (R) a serem lidos no BD. Caso não existam variáveis do tipo R a serem lidas deve-se especificar o valor 0 (zero).  
(Quantidade de variáveis R)
- Memória xx + 2** Número inicial da variável tipo contato (R) na base de dados do CONTROLADOR, a partir do qual serão armazenados os dados lidos do BD (Endereço destino R)
- Memória xx + 3** Número inicial da variável tipo memória inteira (M) ser lida no BD. Caso não existam variáveis do tipo M a serem lidas deve-se especificar o valor 0 (zero).  
(Endereço origem M)
- Memória xx + 4** Quantidade de variáveis do tipo memória inteira (M) a serem lidas no BD  
(Quantidade de variáveis M)
- Memória xx + 5** Número inicial da variável tipo memória inteira (M) na base de dados do controlador, a partir do qual serão armazenados os dados lidos do BD (Endereço destino M)
- Memória xx + 6** Número inicial da variável tipo memória real (D) ser lida no BD  
(Endereço origem D)
- Memória xx + 7** Quantidade de variáveis do tipo memória real (D) a serem lidos no BD. Caso não existam variáveis do tipo D a serem lidas deve-se especificar o valor 0 (zero).  
(Quantidade de variáveis D)
- Memória xx + 8** Número inicial da variável tipo memória real (D) na base de dados do controlador, a partir do qual serão armazenados os dados lidos do BD (Endereço destino D)

#### P3: Parâmetros de Saída

**Memória yy + 0** Parâmetro não utilizado para esta função (parâmetro *dummy*)

#### P4: Parâmetros de Saída

**Memória zz + 0** Código de retorno da execução da função de acesso ao BD, onde :  
Igual a 0 Indica função executada com sucesso.  
Diferente de 0 Indica condição de falha. Vide a lista de códigos de retorno



## 6.3 Função de Escrita no BD

Função 02

**Escrita**

### Descrição da Função

Salvar um registro de dados definido na base de dados do controlador no BD. Considera-se como registro de dados uma combinação qualquer de variáveis dos tipos R, M ou D.

#### P1: Parâmetros de Entrada

**Constante Knn** Função do bloco SCB de acesso de escrita no BD  
Deve ser uma constante inteira (Knn) cujo valor deve ser 2.

#### P2: Parâmetros de Entrada

- Memória xx + 0** Número inicial da variável tipo contato (R) na base de dados do controlador a ser escrita no BD. (Endereço origem R)
- Memória xx + 1** Quantidade de variáveis do tipo contato (R) a serem escritas no BD. Caso não existam variáveis do tipo R a serem escritas deve-se especificar o valor 0 (zero). (Quantidade de variáveis R)
- Memória xx + 2** Número inicial da variável tipo contato (R) no BD a partir do qual serão escritas os dados da base de dados do CONTROLADOR (Endereço destino R)
- Memória xx + 3** Número inicial da variável tipo memória inteira (M) na base de dados do CONTROLADOR a ser escrita no BD. (Endereço origem M)
- Memória xx + 4** Quantidade de variáveis do tipo memória inteira (M) a serem escritas no BD. Caso não existam variáveis do tipo M a serem escritas deve-se especificar o valor 0 (zero). (Quantidade de variáveis M)
- Memória xx + 5** Número inicial da variável tipo memória inteira (M) no BD a partir do qual serão escritas os dados da base de dados do CONTROLADOR (Endereço destino M)
- Memória xx + 6** Número inicial da variável tipo memória real (D) na base de dados do CONTROLADOR a ser escrita no BD. (Endereço origem D)
- Memória xx + 7** Quantidade de variáveis do tipo memória real (D) a serem escritas no BD. Caso não existam variáveis do tipo D a serem escritas deve-se especificar o valor 0 (zero). (Quantidade de variáveis D)
- Memória xx + 8** Número inicial da variável tipo memória real (D) no BD a partir do qual serão escritas os dados da base de dados do CONTROLADOR (Endereço destino D)

#### P3: Parâmetros de Saída

**Memória yy + 0** Parâmetro não utilizado para esta função (parâmetro *dummy*)

#### P4: Parâmetros de Saída

**Memória zz + 0** Código de retorno da execução da função de acesso ao BD, onde :  
Igual a 0 Indica função executada com sucesso.  
Diferente de 0 Indica condição de falha. Vide a lista de códigos de retorno



## 6.4 Códigos de Retorno das Funções de Acesso ao BD

Código de Retorno	Descrição
211	Programa de aplicação está no modo de DESENVOLVIMENTO. Esta função opera apenas no modo de APLICAÇÃO.
212	Variável R, M ou D a ser lida / escrita no BD não foi alocada no BD.  <u>Exemplo</u> : Solicitar a leitura de dados do BD e nem todas as variáveis R, M ou D indicadas como origem dos dados estão alocadas no BD. Por exemplo, solicitar a leitura de 100 memórias M do BD, mas na configuração do BD tem-se, alocadas, apenas 80 variáveis do tipo M.  <u>Exemplo</u> : Solicitar a escrita de dados no BD e nem todas as variáveis R, M ou D indicadas como destino dos dados estão alocadas no BD. Por exemplo, solicitar a escrita de 50 memórias D no BD, mas na configuração do BD não existe nenhuma variável do tipo D alocada.
206	Variável R, M ou D acessada na base de dados do CONTROLADOR pelo programa de aplicação não está alocada.  <u>Exemplo</u> : Solicitar a leitura de dados do BD e nem todas as variáveis R, M ou D indicadas como destino dos dados estão alocadas na base de dados da aplicação. Por exemplo, solicitar a leitura de 100 memórias M do BD e que serão salvas nas memórias M200 a M299 da base de dados do CONTROLADOR, mas na base de dados da aplicação tem-se alocadas até a memória M250.  <u>Exemplo</u> : Solicitar a escrita de dados no BD e nem todas as variáveis R, M ou D indicadas como origem dos dados estão alocadas na base de dados da aplicação. Por exemplo, solicitar a escrita de 50 memórias D no BD, cujos valores estão nas memórias D100 a D149 da base de dados da aplicação, mas na base de dados da aplicação tem-se alocadas até a memória D120.
213	Variáveis R, M ou D acessadas na base de dados do CONTROLADOR estão distribuídas na memória RAM e NVRAM. Neste caso deve-se executar a função desejada ( <i>read / write</i> ) em duas etapas, uma contemplando todas as variáveis desejadas que estão na RAM e outra contemplando todas as variáveis desejadas que estão na NVRAM.  <u>Exemplo</u> : Solicitar a leitura de dados do BD e uma parte das variáveis R, M ou D indicadas como destino dos dados estão alocadas em memória RAM e outra parte das variáveis estão alocadas em memória NVRAM da base de dados do CONTROLADOR.  <u>Exemplo</u> : Solicitar a escrita de dados no BD e uma parte das variáveis R, M ou D indicadas como origem dos dados estão alocadas em memória VORAM e outra parte das variáveis está alocada em memória NVRAM da base de dados do CONTROLADOR.



## 7. Exemplos de Utilização

Apresentamos a seguir exemplos de utilização das funções de acesso ao banco de dados em *Flash ROM*. No *website* da HI Tecnologia ([www.hitecnologia.com.br](http://www.hitecnologia.com.br)) pode-se obter os seguintes programas com exemplos adicionais de utilização do BD.

- HIPJT008 – programa de exemplo para **SPDS7** - Utilização do BANCO DE DADOS em *FLASH ROM* para linha de controladores HI.
- HIPJW008 - programa de exemplo para **SPDSW** - Utilização do BANCO DE DADOS em *FLASH ROM* para linha de controladores HI.

Para uma melhor explanação sobre o acesso ao BD nos controladores HI, mostra-se a seguir exemplos de utilização das funções de leitura e escrita no BD.

### 7.1 Exemplo 1 : Registro de Dados

#### 7.1.1 Leitura do Banco de Dados em *Flash ROM*

Este exemplo mostra a configuração dos parâmetros para efetuar a leitura no BD de:

- 10 contatos auxiliares (R): por exemplo, de [R0..R9]
- 20 memórias inteiras (M): por exemplo, de [M0..M19]
- 30 memórias reais (D): por exemplo, de [D0..D29]

A figura a seguir ilustra este registro de dados:

10 Variáveis R	20 Variáveis M	30 Variáveis D
----------------	----------------	----------------

**Figura 12: Exemplo de registro de dados a ser lido do BD**

Para realizar a leitura deste registro de dados deve-se:

Habilitar a entrada (**E1**) do bloco SCB, para efetuar a leitura no BD.

A saída deste bloco (**S1**) será ligada quando a entrada (**E1**) estiver ligado e o Bloco SCB concluir a função.



PARAMETROS PARA LEITURA DO BD :

Param.	Parâmetro	Valor	Significado do Valor do Parâmetro
P1:	Knn	01	Função do bloco SCB para leitura do BD
P2:	Mxx + 0	0	Número inicial da variável "R" no BD, no caso a partir de "R0"
	Mxx + 1	10	Quantidade de variáveis "R" do BD a partir da variável inicial "R0"
	Mxx + 2	0	Destino inicial dos contatos lidos do BD que será escrito na base de dados do CONTROLADOR, no caso a partir de "R0"
	Mxx + 3	0	Número inicial da variável "M" no BD, no caso a partir de "M0"
	Mxx + 4	20	Quantidade de variáveis "M" do BD a partir da variável inicial "M0"
	Mxx + 5	0	Destino inicial das memórias lidas do BD que será escrita na base de dados do CONTROLADOR, no caso a partir de "M0"
	Mxx + 6	0	Número inicial da variável "D" no BD, no caso a partir de "D0"
	Mxx + 7	30	Quantidade de variáveis "D" do BD a partir da variável inicial "D0"
	Mxx + 8	0	Destino inicial das memórias lidas do BD que será escrita na base de dados do CONTROLADOR, no caso a partir de "D0"
P3:	Myy		Parâmetro não utilizado para esta função (parâmetro <i>dummy</i> )
P4:	Mzz		Código de retorno da execução da função de acesso ao BD

### 7.1.2 Escrita no Banco de Dados em Flash ROM

Este exemplo mostra a configuração dos parâmetros para efetuar a escrita no BD de:

- 10 contatos auxiliares (R): por exemplo, de [R0..R9]
- 20 memórias inteiras (M): por exemplo, de [M0..M19]
- 30 memórias reais (D): por exemplo, de [D0..D29]

A figura a seguir ilustra este registro de dados:

10 Variáveis R	20 Variáveis M	30 Variáveis D
----------------	----------------	----------------

**Figura 13: Exemplo de registro de dados a ser escrito no BD**

Para realizar a leitura deste registro de dados deve-se:

Habilitar a entrada (**E1**) do bloco SCB, para efetuar a leitura no BD.

A saída deste bloco (**S1**) será ligada quando a entrada (**E1**) estiver ligado e o Bloco SCB concluir a função.



PARAMETROS PARA ESCRITA NO BD:

Param.	Parâmetro	Valor	Significado do Valor do Parâmetro
P1:	Knn	02	Função do bloco SCB para escrita no BD
P2:	Mxx + 0	0	Número inicial da variável "R" na base de dados do CONTROLADOR, no caso a partir de "R0"
	Mxx + 1	10	Quantidade de variáveis "R" na base de dados do CONTROLADOR a partir da variável inicial "R0"
	Mxx + 2	0	Destino inicial dos contatos lidos da base de dados do CONTROLADOR que será escrito no BD, no caso a partir de "R0"
	Mxx + 3	0	Número inicial da variável "M" na base de dados do CONTROLADOR, no caso a partir "M0"
	Mxx + 4	20	Quantidade de variáveis "M" na base de dados do CONTROLADOR a partir da variável inicial "M0"
	Mxx + 5	0	Destino inicial das memórias lidas da base de dados do CONTROLADOR que será escrita no BD, no caso a partir de "M0"
	Mxx + 6	0	Número inicial da variável "D" na base de dados do CONTROLADOR, no caso a partir de "D0"
	Mxx + 7	30	Quantidade de variáveis "D" na base de dados do CONTROLADOR a partir da variável inicial "D0"
	Mxx + 8	0	Destino inicial das memórias lidas da base de dados do CONTROLADOR que será escrita no BD, no caso a partir de "D0"
P3:	Myy		Parâmetro não utilizado para esta função (parâmetro <i>dummy</i> )
P4:	Mzz		Código de retorno da execução da função de acesso ao BD

## 7.2 Exemplo 2 : Parâmetros de Configuração

Suponhamos uma aplicação onde se tem as seguintes variáveis associadas a parâmetros de configurações do sistema:

- 20 variáveis do tipo memórias inteiras (M), mapeadas de M10...M29 na base de dados do CONTROLADOR, e
- 5 variáveis do tipo memórias reais (D), mapeadas de D0...D4, também na base de dados do CONTROLADOR.

O objetivo consiste em manter uma cópia destes parâmetros de configuração do sistema no banco de dados em Flash ROM, de modo a garantir que tais parâmetros de configuração não sejam perdidos quando o controlador for desligado.

No programa de aplicação *ladder* pode-se construir a seguinte funcionalidade:

Ao detectar alteração em qualquer um dos parâmetros de configuração pode-se ativar lógicas para salvar estas variáveis (M e D) no BD;

Na inicialização do sistema pode-se automaticamente chamar funções para realizar a leitura destes parâmetros de configuração do BD para a base de dados do CONTROLADOR. Assim, torna-se transparente para o usuário do sistema o local de armazenamento destes dados de configuração do sistema.



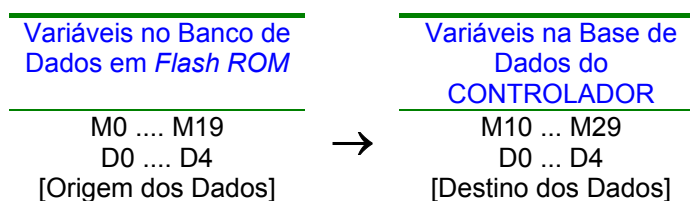
O primeiro passo consiste em configurar o banco de dados da aplicação. Neste caso o BD terá somente 20 variáveis do tipo M e 5 variáveis do tipo D, de tal modo que deve ser configurado com os seguintes valores:

- Quantidade de Contatos Auxiliares (R) = 0 variáveis
- Quantidade de Memórias Inteiras (M) = 20 variáveis
- Quantidade de Memórias Reais (D) = 5 variáveis

A seguir apresentam-se os parâmetros necessários quando da ativação de um bloco SCB tanto para leitura como para escrita destes parâmetros de configuração no BD.

### 7.2.1 Leitura dos Parâmetros de Configuração do BD

Quando o programa de aplicação *ladder* desejar recuperar os parâmetros de configuração que estão armazenados no BD e transferi-los para variáveis da Base de Dados do CONTROLADOR, no caso realizando a seguinte transferência de valores do Banco de Dados em *Flash ROM* para a Base de Dados do CONTROLADOR:



Deve-se passar os seguintes parâmetros para o bloco SCB:

Param.	Parâmetro	Valor	Significado do Valor do Parâmetro
P1:	Knn	01	Função do bloco SCB para leitura do BD
P2:	Mxx + 0	0	Número inicial da variável "R" no BD. Como não existem variáveis do tipo "R", este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)
	Mxx + 1	0	Quantidade de variáveis "R" a serem lidas no BD. Como não existem variáveis do tipo "R", deve-se obrigatoriamente especificar o valor 0 (Zero)
	Mxx + 2	0	Destino inicial dos contatos lidos na base do CONTROLADOR. Como não existem variáveis do tipo "R", este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)
	Mxx + 3	0	Origem inicial da variável "M" a ser lida no BD. No caso deseja-se ler a partir da variável "M0" do BD
	Mxx + 4	20	Quantidade de variáveis "M" a serem lidas no BD. No caso, deseja-se ler 20 variáveis do tipo "M" do BD, a partir da memória "M0"
	Mxx + 5	10	Destino inicial das memórias "M" na base de dados do CONTROLADOR. No caso deseja-se armazená-las a partir da memória "M10" da base de dados do CONTROLADOR
	Mxx + 6	0	Origem inicial da variável "D" no BD. No caso deseja-se ler a partir da memória "D0" do BD
	Mxx + 7	05	Quantidade de variáveis "D" a serem lidas no BD. No caso deseja-se ler 5 variáveis do tipo "D" do BD, a partir da memória "D0"
	Mxx + 8	0	Destino inicial das memórias "D" na base de dados do CONTROLADOR. No caso deseja-se armazená-las a partir da



memória “D0” da base de dados do CONTROLADOR

P3: Myy

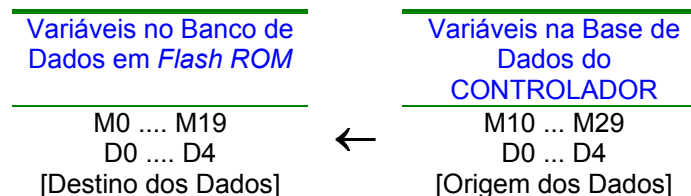
Parâmetro não utilizado para esta função (parâmetro *dummy*)

P4: Mzz

Código de retorno da execução da função de acesso ao BD

## 7.2.2 Escrita dos Parâmetros de Configuração no BD

Quando o programa de aplicação *ladder* desejar salvar os parâmetros de configuração que estão armazenados na Base de Dados de trabalho do CONTROLADOR e transferi-los para a Banco de dados em *Flash ROM*, no caso realizando a seguinte transferência de valores da Base de Dados do CONTROLADOR para o Banco de Dados em *Flash ROM*:



Deve-se passar os seguintes parâmetros para o bloco SCB:

Param.	Parâmetro	Valor	Significado do Valor do Parâmetro
<u>P1:</u>	Knn	02	Função do bloco SCB para escrita no BD
<u>P2:</u>	Mxx + 0	0	Número inicial da variável “R” na base de dados do CONTROLADOR. Como não existem variáveis do tipo “R”, este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)
	Mxx + 1	0	Quantidade de variáveis “R” a serem escritas no BD. Como não existem variáveis do tipo “R”, deve-se obrigatoriamente especificar o valor 0 (Zero)
	Mxx + 2	0	Destino inicial dos contatos no BD. Como não existem variáveis do tipo “R”, este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)
	Mxx + 3	10	Origem inicial da variável “M” a ser salva na base de dados do CONTROLADOR. No caso deseja-se escrever a partir da variável “M10” do CONTROLADOR
	Mxx + 4	20	Quantidade de variáveis “M” a serem escritas no BD. No caso, deseja-se escrever 20 variáveis do tipo M no BD, a partir da memória “M10” do CONTROLADOR
	Mxx + 5	0	Destino inicial das memórias “M” no BD. No caso deseja-se armazená-las a partir da memória “M0” do BD
	Mxx + 6	0	Origem inicial da variável “D” na base de dados do CONTROLADOR. No caso deseja-se escrever a partir da variável “D0” do CONTROLADOR
	Mxx + 7	05	Quantidade de variáveis “D” a serem escritas no BD. No caso deseja-se escrever 5 variáveis do tipo D no BD, a partir da memória “D0” do CONTROLADOR
	Mxx + 8	0	Destino inicial das memórias “D” no BD. No caso deseja-se armazená-las a partir da memória “D0” do BD
<u>P3:</u>	Myy		Parâmetro não utilizado para esta função (parâmetro <i>dummy</i> )
<u>P4:</u>	Mzz		Código de retorno da execução da função de acesso ao BD

### 7.3 Exemplo 3 : Receitas

Suponhamos uma aplicação onde tem a necessidade de manipulação de receitas. No caso, suponhamos que cada receita pode ser mapeada em 10 memórias do tipo “M”, e que a aplicação permita armazenar até 100 receitas distintas.

Neste caso, pode-se armazenar as 100 receitas no banco de dados em *Flash ROM*, e criar uma área de trabalho na base de dados do CONTROLADOR com a receita corrente a ser manipulada. Suponhamos que na base de dados do CONTROLADOR estejam alocadas as memórias [M20...M29] como área de trabalho visando ser uma área de transferência de dados de uma receita da aplicação e o banco de dados em *Flash ROM*.

O objetivo consiste em ler e salvar qualquer uma das 100 receitas para esta área de trabalho do CONTROLADOR. Neste caso faz-se necessário indexar a receita a ser acessada no BD. Suponhamos que indexemos as receitas de [0..99], ou seja, em função do número da receita calcula-se o endereço das variáveis associadas a receita desejada no BD, de tal forma a poder ler e escrevê-la novamente. A tabela abaixo ilustra o mapeamento das variáveis utilizadas para o gerenciamento das receitas, tanto no BD como na base de dados do CONTROLADOR.

Índice da Receita	Variáveis no Banco de Dados em <i>Flash ROM</i>	Variáveis na Base de Dados do CONTROLADOR
0	M00 .... M09	M20 ... M29
1	M10 .... M19	
2	M20 .... M29	
3	M30 .... M39	
4	M40 .... M49	
...	...	
96	M960 .... M969	
97	M970 .... M979	
98	M980 .... M989	
99	M990 .... M999	

Assim, faz-se necessário alocar 1000 variáveis do tipo “M” (M0...M1999) no BD para permitir o armazenamento das 100 receitas do usuário, de tal modo que o banco de dados em *Flash ROM* deve possuir a seguinte configuração:

- Quantidade de Contatos Auxiliares (R) = 0 variáveis
- Quantidade de Memórias Inteiras (M) = 1000 variáveis
- Quantidade de Memórias Reais (D) = 0 variáveis

A seguir apresentam-se os parâmetros necessários quando da ativação de um bloco SCB tanto para leitura como para escrita de uma receita no BD.

#### 7.3.1 Leitura de uma Receita do BD

Suponhamos que o usuário deseja ler a receita de número 15 que está armazenada no BD. No caso o programa *ladder* de aplicação deve localizar as memórias “M” no BD que estão associadas a esta receita de



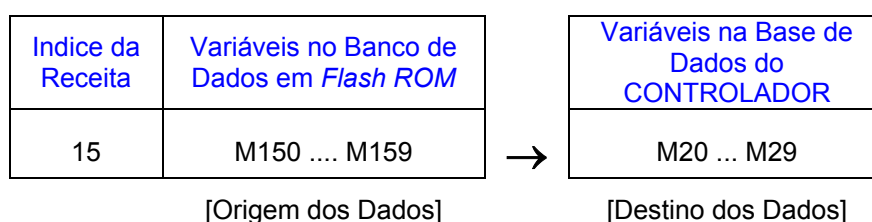
## Uso do Banco de Dados em Flash ROM nos Controladores da HI Tecnologia

Tipo de Doc.: Notas de Aplicação  
Referência: ENA.00017

Revisão: 2  
Atualizado em: 07/04/2009

numero 15, e transferir o conteúdo destas memórias “M” do BD para a área de trabalho do CONTROLADOR, ou seja, deve-se armazená-las nas variáveis [M20...M29] na base de dados do CONTROLADOR. Através do número da receita é possível calcular as respectivas memórias “M” de onde se deve ler os dados desta receita no BD, no caso, como cada receita consome 10 memórias “M” e as receitas são armazenadas seqüencialmente no BD, conclui-se que as memórias [M150...M159] são as memórias do BD que armazenam a receita de número 15.

Assim quando o programa de aplicação *ladder* desejar recuperar os valores da receita de número 15 que está armazenada no BD e transferi-la para as variáveis da base de dados do CONTROLADOR, deve-se realizar a seguinte transferência de valores do banco em dados em *Flash ROM* para a base de dados do CONTROLADOR:



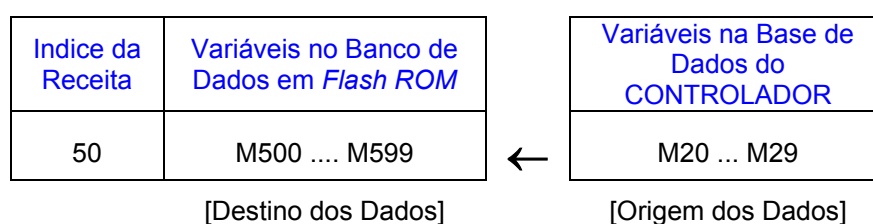
Devendo passar os seguintes parâmetros para o bloco SCB:

Param.	Parâmetro	Valor	Significado do Valor do Parâmetro
P1:	Knn	01	Função do bloco SCB para leitura do BD
P2:	Mxx + 0	0	Número inicial da variável “R” no BD. Como não existem variáveis do tipo “R” na receita, este parâmetro é desprezado, podendo possuir qualquer valor, como, por exemplo, o valor 0 (Zero)
	Mxx + 1	0	Quantidade de variáveis “R” a serem lidas no BD. Como não existem variáveis do tipo “R” na receita, deve-se obrigatoriamente especificar o valor 0 (Zero)
	Mxx + 2	0	Destino inicial dos contatos lidos na base do CONTROLADOR. Como não existem variáveis do tipo “R” na receita, este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)
	Mxx + 3	150	Origem inicial da variável “M” a ser lida no BD. No caso deseja-se ler a partir da variável “ <b>M150</b> ” do BD
	Mxx + 4	10	Quantidade de variáveis “M” a serem lidas no BD. No caso, deseja-se ler 10 variáveis do tipo “M” do BD, a partir da memória “ <b>M150</b> ”
	Mxx + 5	20	Destino inicial das memórias “M” lidas na base de dados do CONTROLADOR. No caso deseja-se armazená-las a partir da memória “ <b>M20</b> ” da base de dados do CONTROLADOR
	Mxx + 6	0	Origem inicial da variável “D” no BD. Como não existem variáveis do tipo D na receita, este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)
	Mxx + 7	0	Quantidade de variáveis “D” a serem lidas no BD. Como não existem variáveis do tipo D na receita, deve-se obrigatoriamente especificar o valor 0 (Zero)
Mxx + 8	0	Destino inicial das memórias “D” lidas na base de dados do CONTROLADOR. Como não existem variáveis do tipo “D” na receita, este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)	
P3:	Myy		Parâmetro não utilizado para esta função (parâmetro <i>dummy</i> )
P4:	Mzz		Código de retorno da execução da função de acesso ao BD

### 7.3.2 Escrita de uma Receita no BD

Suponhamos que o usuário tenha preenchido uma receita em sua aplicação e agora deseja guardá-la como receita número 50. Esta receita deve estar na área de trabalho do CONTROLADOR, ou seja, deve estar armazenada nas variáveis M20...M29 na base de dados do CONTROLADOR. Através do número da receita é possível calcular as respectivas memórias "M" onde deve-se salvar os dados desta receita no BD, no caso, como cada receita consome 10 memórias "M" e as receitas são armazenadas seqüencialmente no BD, conclui-se que as memórias M500...M599 são as memórias do BD que armazenam a receita de número 50.

Quando o programa de aplicação *ladder* desejar salvar a receita corrente da aplicação, receita esta que deve estar armazenada na base de dados de trabalho do CONTROLADOR, e transferi-la para a base de dados em *Flash ROM* como receita de número 50, deve-se realizar a seguinte transferência de valores da base de dados do CONTROLADOR para o banco de dados em *Flash ROM*:



devendo passar os seguintes parâmetros para o bloco SCB:

Param.	Parâmetro	Valor	Significado do Valor do Parâmetro
P1:	Knn	02	Função do bloco SCB para escrita no BD
P2:	Mxx + 0	0	Número inicial da variável "R" na base de dados do CONTROLADOR. Como não existem variáveis do tipo "R" na receita, este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)
	Mxx + 1	0	Quantidade de variáveis "R" a serem escritas no BD. Como não existem variáveis do tipo "R" na receita, deve-se obrigatoriamente especificar o valor 0 (Zero)
	Mxx + 2	0	Destino inicial dos contatos no BD. Como não existem variáveis do tipo "R" na receita, este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)
	Mxx + 3	20	Origem inicial da variável "M" ser salva na base de dados do CONTROLADOR. No caso deseja-se escrever a partir da variável "M20" do CONTROLADOR
	Mxx + 4	10	Quantidade de variáveis "M" a serem escritas no BD. No caso, deseja-se escrever 10 variáveis do tipo "M" no BD, a partir da memória "M20" do CONTROLADOR
	Mxx + 5	500	Destino inicial das memórias "M" no BD. No caso deseja-se armazená-las a partir da memória "M500" do BD
	Mxx + 6	0	Origem inicial da variável "D" na base de dados do CONTROLADOR. Como não existem variáveis do tipo "D" na receita, este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)
	Mxx + 7	0	Quantidade de variáveis "D" a serem escritas no BD. Como não existem variáveis do tipo "D" na receita, deve-se obrigatoriamente especificar o valor 0 (Zero)
	Mxx + 8	0	Destino inicial das memórias "D" no BD. Como não existem variáveis do tipo D na receita, este parâmetro é desprezado, podendo possuir qualquer valor, como por exemplo o valor 0 (Zero)
P3:	Myy		Parâmetro não utilizado para esta função (parâmetro <i>dummy</i> )
P4:	Mzz		Código de retorno da execução da função de acesso ao BD



## Uso do Banco de Dados em Flash ROM nos Controladores da HI Tecnologia

Tipo de Doc.: Notas de Aplicação  
Referência: ENA.00017

Revisão: 2  
Atualizado em: 07/04/2009

## Controle do Documento

---

### Considerações gerais

---

1. Este documento é dinâmico, estando sujeito a revisões, comentários e sugestões. Toda e qualquer sugestão para seu aprimoramento deve ser encaminhada ao departamento de suporte ao cliente da **HI Tecnologia**, especificado na “Apresentação” deste documento.
2. Os direitos autorais deste documento são de propriedade da **HI Tecnologia**.

### Responsabilidades pelo documento

---

	<b>Data</b>	<b>Responsável</b>	
Elaboração	29/08/2003	Paulo C. M. Inazumi	
Revisão	11/03/2009	Paulo C. M. Inazumi	<i>Revisado em mídia</i>
Aprovação	07/04/2009	Isaias M. C. Ribeiro	<i>Aprovado em mídia</i>

#### Histórico de Revisões

07/04/2009	2	Inclusão dos equipamentos GII e GII Duo
25/01/2008	1	Revisão nos cálculos da base de dados em flash rom de 16k e 12K
27/10/2003	0	Documento original
Data	Rev	Descrição