

# Profibus-PA: byte de status e Fail-Safe

Autor: César Cassiolato - Gerente de Produtos  
Smar Equipamentos Industriais Ltda.

## Introdução

Uma característica muito útil disponibilizada pelo protocolo Profibus PA é o byte de status na estrutura float+status.

A grande vantagem desta característica é permitir que se tenha além do valor de uma medição, uma informação que possa qualificar este valor e que ajuda na condição de falha a levar o equipamento para uma condição de segurança(Fail-Safe). Veremos a seguir alguns detalhes sobre esta funcionalidade.

## Entendo o byte de status(estatdo)

O status da entrada e dos parâmetros de saídas tem a finalidade de informar a outros blocos funcionais o estado atual.

O campo de status é composto de três partes: Qualidade, Sub-estados e Limites.

**Quality** – indica a qualidade do valor do parâmetro.

- *Good Cascade*– A qualidade do valor é boa e pode ser utilizado para controle em cascata.
- *Good Non-cascade*– A qualidade do valor é boa e não pode ser utilizado para controle em cascata.
- *Uncertain*–A qualidade do valor está abaixo do normal, mas o valor ainda pode ser útil.
- *Bad*–O valor não é útil.

**Sub-status** – O sub-status é um complemento do estado da qualidade e leva a informação para inicializar ou parar um controle em cascata, alarmes e outros. Há diferentes configurações do sub-status para cada qualidade.

**Limits** – Fornece informação se o valor associado está limitado ou não, bem como a direção. Os limites são classificados como: Not Limited, High Limited, Low Limited, Constant.

Quando um parâmetro de entrada é “conectado” num parâmetro de saída pela troca de dados cíclicos, a estrutura toda (estado e valor) é recebida via comunicação fieldbus. Se a entrada não é “conectada”, então os estados e os valores podem ser configurados manualmente pelo usuário.

O estado tem a seguinte composição: Quality, sub-status e limits:

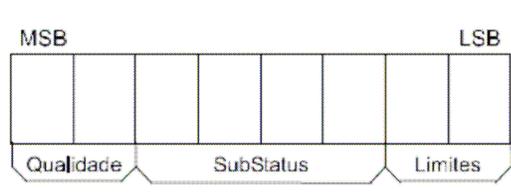


Figura 1 – Estrutura do byte de status

Os componentes do estado são definidos como se segue:

**Quality-** A qualidade usada será determinada pela condição de mais alta prioridade:

0 = Ruim

1 = Incerto

2 = Bom (Não-cascata)

3 = Bom (Cascata)

**Sub-status-** Os valores do sub-status no status attribute são definidos conforme tabela 1.

**Limit** – As seguintes condições de limites sempre estarão disponíveis no estado.

0 = Não limitada

1 = Limite Baixo

2 = Limite Alto

3 = Constante

Exemplos:

- 0xC1 (em hexadecimal) é o estado “Good-Cascade Non Specific and Low Limited”

- 0xCF(em hexadecimal) é o estado “Good-Cascade Not invited and Constant”

- 0x4E(em hexadecimal) é o estado “ Uncertain Initial Value and High Limited”

### **Exemplo: conversão de número para enumerações**

Há muitas formas para converter o número enumerado à string de estado. Abaixo, são mostradas duas formas:

1) Expressando o número em binário.

Estado do valor hexadecimal = 78 = 0x4E = 01001110 (em binário)

Dividindo este número binário em campos Quality, Sub-Status e Limit:

Quality = 01 = 1 = “Uncertain”

Sub-estado = 0011 = 3 = “Valor Inicial”

Limit = 10 = 2 = “Limitado em Alto”

O estado correspondente é “uncertain - initial value - high limited.”

2) Usando o valor do estado em formato decimal.

Decimal Value Status = 78

Dividido o número por 64. O quociente será o Quality e armazenado o resto :

Quality =  $78 / 64 = 1$

Resto = 14

Dividido o resto por 4. O quociente será o Sub-Status e o resto será o limite:

SubStatus =  $14 / 4 = 3$

Limit = 2

Qualidade	Sub-estado	Valor Hexa	Valor Decimal	Não em Cascata	Caminho de avanço da cascata	caminho de recuo da cascata
GoodNC	0 = ok (prioridade mais baixa)	0x80	128	X		
GoodNC	9 = requer manutenção	0xA4	164	X		
GoodNC	1 = Evento de atualização ativo	0x84	132	X		
GoodNC	2 = Alarme de alerta ativo	0x88	136	X		
GoodNC	3 = Alarme crítico ativo	0x8c	140	X		
GoodNC	4 = Atualização de evento não reconhecido	0x90	144	X		
GoodNC	5 = Alarme de alerta não reconhecido	0x94	148	X		
GoodNC	6 = Alarme crítico não reconhecido	0x98	152	X		
GoodNC	8 = início de Fail-Safe (IFS)	0xA0	160	X		
Uncertain	0 = Não especificado	0x40	64	X		
Uncertain	1 = Último valor utilizável	0x44	68	X		
Uncertain	2 = Substituto	0x48	72	X		
Uncertain	3 = Valor inicial	0x4c	76	X		
Uncertain	4 = Conversão de sensor sem precisão	0x50	80	X		
Uncertain	5 = Violada faixa de Unidades de Engenharia	0x54	84	X		
Uncertain	6 = Sub-normal	0x58	88	X		
Uncertain	7 = Erro de configuração	0x5C	92	X		
Uncertain	9 = Calibração de sensor	0x64	100	X		
Uncertain	8 = Valor simulado	0x60	96	X		
GoodC	0 = ok	0xC0	192		X	X
GoodC	1 = Reconhece inicialização (IA)	0xC4	196		X	
GoodC	2 = Requisita inicialização (IR)	0xC8	200			X
GoodC	3 = NÃO Convidado (NI)	0xCC	204			X
GoodC	5 = Não Selecionado (NS)	0xD0	208		X	
GoodC	6 = Override(LO local)	0xD8	216			X
GoodC	8 = Iniciada Fail Safe (IFS)	0xE0	224		X	
Bad	0 = Não-específico	0x00	0	X	X	X
Bad	1 = Erro de configuração	0x04	4	X	X	X
Bad	2 = Não Conectado	0x08	8			
Bad	4 = Falha do sensor	0x10	16	X	X	X
Bad	3 = Falha do equipamento	0x0C	12	X	X	X
Bad	5 = Nenhuma comunicação, com último valor utilizável	0x14	20			
Bad	6 = Nenhuma comunicação, com o valor utilizável	0x18	24			
Bad	7 = Fora de Serviço (a mais alta prioridade)	0x1c	28			

**Tabela 1 – Condições de status**

### **Manipulação em Fail Safe (Falha de Segura)**

O Fail Safe é um estado especial, que permite o bloco de função atuar quando for detectado uma situação anormal.

Uma situação anormal ocorre quando houver uma entrada não usual, por exemplo sensor com defeito ou perda de comunicação entre os blocos de função por um período maior que o especificado (FSAFE\_TIME).

Quando a condição que ativou o estado de falha (Fault State) for normalizada, o Fault State é zerado e o bloco volta à operação normal.

### **Condições que ativam o Fail Safe**

Quando os Blocos de Função de entrada ou saída detectam uma condição anormal, o bloco entra em modo FAIL\_SAFE. Estas situações anormais são detectadas através de diferentes formas.

Os blocos de função de entrada, por exemplo, AI e TOT são conectados com um bloco transdutor via canal. Quando a saída do transdutor tem um estado ruim, (por exemplo, sensor avariado) a condição de FAIL\_SAFE do bloco é ativada.

Os blocos de função de saída, por exemplo: bloco AO recebe os valores de entrada de um dos blocos superiores através da “conexão cíclica.” Estes blocos superiores são blocos de função de controle e, geralmente, vêm de equipamentos de controle. Neste caso, o fail safe é ativado quando uma das seguintes condições é alcançada:

- Perda de comunicação de RCAS\_IN por um tempo que excede FSAFE\_TIME;
- Perda de comunicação de SP durante um tempo que excede FSAFE\_TIME;
- O estado IFS na entrada RCAS\_IN quando o modo designado é RCas;
- O estado IFS no SP quando o modo designado é Auto.

### **Ações de Fail Safe (Falha de Segura)**

As ações que um bloco de entrada ou de saída podem tomar, quando o bloco está em Fail Safe, devem ser selecionadas pelo usuário através do parâmetro FSAFE\_TYPE nos blocos AI e AO ou usando o parâmetro de FAIL\_TOT no bloco TOT.

No parâmetro FSAFE\_TYPE as seguintes opções estão disponíveis:

- Use FSAFE\_VALUE – Neste caso, os blocos AI e AO usam o valor seguro fornecido pelo parâmetro FSAFE\_VALUE para o cálculo quando o Fail Safe está ativo. O estado da saída vai para “Uncertain, substitute value”
- Use Last Usable Value – Neste caso, os blocos AI e AO usam o último valor utilizável para o cálculo do algoritmo. O estado será “Uncertain Last Usable Level”. Se ainda não houver um valor satisfatório, use o Valor Inicial na saída. O estado será “Uncertain Initial Value”.
- “Use the wrong value” (apenas para o bloco AI) – o AI usa os valores e estados errados para o cálculo.
- Use o ACTUATOR\_ACTION (apenas para bloco AO) – O bloco AO entra em posição de segurança baseado em um parâmetro discreto ACTUATOR\_ACTION no bloco transdutor.

No parâmetro FAIL\_TOT (só usado no bloco totalizador) as seguintes opções estão disponíveis:

- Hold - Para a totalização no último valor. O estado da saída vai para “Uncertain non-specific”.

- Memory - Usa o último valor válido para a totalização. O estado será de “Uncertain Last Usable Value”. Se não houver um estado válido na memória use o valor inicial para a totalização. O estado será “Uncertain, Initial Value”
- Run - A totalização é continuada (Reiniciada). Usa o valor e estado incorreto para a saída.

## **Conclusão**

Vimos através deste artigo a importância do argumento status no Profibus-PA e suas particularidades.

## **Referências:**

- Manuais Smar Profibus
- [www.smar.com.br](http://www.smar.com.br)