

Desenvolvimento de Agente SNMP¹ Embarcado para Microprocessador *Rabbit*

Elvis Lopes Monteiro^a, Westter José da Silva Santos^b, Dr. Cláudio Afonso Fleury^c

CEFET-GO – Centro Federal de Educação Tecnológica de Goiás

{elvislm^a, westter^b}@gmail.com, kaw^c@cefetgo.br

Resumo. Este artigo descreve a implementação de um agente SNMPv1 (Simple Network Management Protocol Version 1) para microprocessador Rabbit Core Module 3200, com aplicação no monitoramento e controle de dispositivos de acionamento elétrico em redes de Automação Industrial.

Abstract. This paper describes the implementation of an SNMPv1 agent (Simple Management Protocol Version 1) on a Rabbit Core Module 3200 Microprocessor to be applied on management and control of electrical activation devices in Industrial Automation Networks.

1. Introdução

O protocolo SNMP surgiu em 1988 e tornou-se de fato um padrão para gerenciamento de redes. É bastante difundido em dispositivos de redes de computadores (*hubs*, *switches*, *routers*). Através dele pode-se identificar e localizar diversos tipos de problemas, tanto na rede em si quanto nos elementos que a compõe, facilitando assim o trabalho dos gerenciadores de redes.

Os dispositivos de acionamento, como relés e contadores de potência, são largamente utilizados em automação industrial no acionamento de cargas elétricas como: motores, conjuntos de iluminação, sistemas de ar condicionado, e diversos outros equipamentos elétricos.

A motivação principal deste estudo é a utilização do protocolo SNMP em redes industriais de automação que possui dispositivos de acionamento elétrico, usando eletrônica embarcada, mais especificamente neste caso, o microprocessador *Rabbit* (agente SNMP embarcado).

2. Gerenciamento de Rede com SNMP

Numa rede gerenciada pelo SNMP pode-se distinguir os seguintes componentes:

2.1 *Gerente SNMP*: entidade que realiza consultas SNMP (*pollings*) nos objetos gerenciados contidos nos agentes, estes objetos possuem atributos definidos em uma base de informações gerenciais, a MIB. O gerenciamento é realizado através de operações SNMP de leitura (*GetRequest*, *GetNext*) ou de escrita (*SetRequest*) pela porta UDP 161, com o objetivo de monitorar os agentes SNMP, ou ainda receber informações (*traps*, relatórios gerados por eventos assíncronos) pela porta UDP 162.

2.2 *Agente SNMP*: entidade (programa executável) responsável por coletar os dados dos dispositivos gerenciados. Comunica-se com o gerente SNMP da rede informando os atributos dos objetos definidos na MIB, quando consultados, ou ainda gerando *traps* na

¹ SNMP – Simple Network Management Protocol – Protocolo Simples de Gerenciamento de Rede

hipótese de ocorrência de eventos pré-definidos, tais como anomalias, erros, falhas, etc, que exigem atenção imediata do gerente SNMP.

2.3. *MIB (Management Information Base)*: base de dados conceitual onde são representados os elementos a serem gerenciados. Possui uma estrutura em árvore: o nó posicionado no início é denominado raiz (*root*). Todo nó que possuir filhos será denominado sub-árvore. E tudo que não tiver filhos será considerado nó de folha ou objeto. Um equívoco comum é a confusão entre MIB e arquivo de MIB. MIB, conforme orientação anterior é a definição conceitual dos recursos gerenciados. Arquivo de MIB refere-se às definições realizadas em um arquivo, formato texto, utilizando a linguagem de definição de sintaxe SMI (*Structure of Management Information*). A MIB deve existir tanto no gerente (arquivo de MIB compilado) como nos agentes (geralmente embutido na estrutura do código fonte).

2.4. *Objetos SNMP*: são os elementos que compõem a MIB. Estes objetos são na verdade variáveis, às quais estão associados: um tipo e um valor. Objetos se relacionam com as características das entidades físicas gerenciadas (*hardware*) ou com propriedades de determinado serviço ou aplicação (*software*).

As principais vantagens resultantes do uso do protocolo SNMP para a gerência de redes são:

- Monitoramento dos objetos gerenciados via *polling* (evento síncrono);
- Utilização de um protocolo padrão e aberto, regulamentado pelo IETF (RFCs);
- Protocolo bastante difundido e largamente utilizado;
- Interoperabilidade entre diversos dispositivos de vários fabricantes;
- Possibilidade de geração de eventos do tipo alarme (*traps*) que indicam quando algo de errado está acontecendo ou está para acontecer, e pode ser configurado para eventos específicos.

3. Metodologia

Adotou-se o estudo de caso como método de análise da viabilidade técnica da proposta, implementando-se um controle simples, que consiste no acionamento de um contator usado na ativação de uma carga elétrica de potência considerável. Após os testes, realizados em laboratório, dos circuitos e programas desenvolvidos, mostrou-se possível a implementação de um agente SNMP embarcado para microprocessador *Rabbit*, visando o gerenciamento de dispositivos interligados em uma rede de automação industrial.

4. Implementação do Estudo de Caso

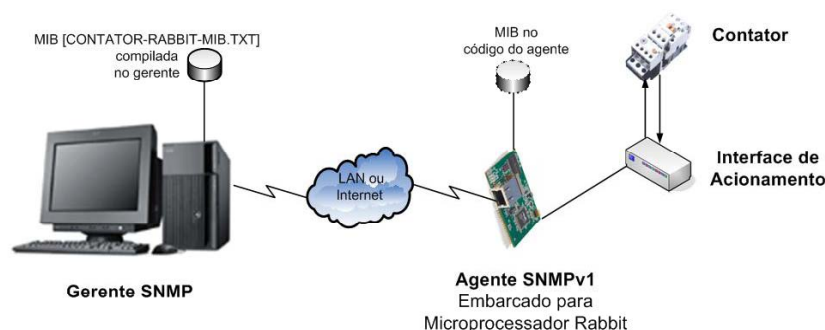


Figura 1 – Arquitetura do estudo de Caso - Interação entre Agente e Gerente SNMP

4.1 Características da implementação:

A seguir as características e funcionalidades dos recursos de *Hardware* e *Software*:

4.1.1 Recursos de *Software*:

a) *Aplicativo MG-Soft MIB Browser*: utilizado para visualização dos valores dos objetos da MIB, assim como para execução de operações de gerenciamento SNMP (*GetRequest*, *SetRequest*, etc).

b) *Dynamic C 8.1* [7]: compilador C para o desenvolvimento da aplicação Agente.

4.1.2 Recursos de *Hardware*:

a) *Kit de Desenvolvimento do Microprocessador Rabbit Core Module RCM3200* [7]: composto pelo microprocessador e sua respectiva placa de desenvolvimento. O microprocessador opera em frequência de 44.2 MHz, e possui memória Flash de 512 KB, memória SRAM de 256 KB, memória de código de 512 KB, 52 linhas de I/O², possibilidade de implementação de serviços TCP/IP (HTTP, POP3, SNMP, etc).

b) *Contator*: foi utilizado um dispositivo SIEMENS, modelo 3TF40 22-0A, para acionamento de carga elétrica.

c) *Circuito para acionamento e monitoramento do contator*: foi desenvolvido um circuito eletrônico para ligar o contator ao *Rabbit*. Com as seguintes funções: proteger o microprocessador através de optoacopladores³ e condicionar os sinais provenientes do microprocessador de forma que estes possam realizar o acionamento do contator (Figura 2), e ainda fazer o monitoramento de um dos contatos auxiliares do contator (Figura 3).

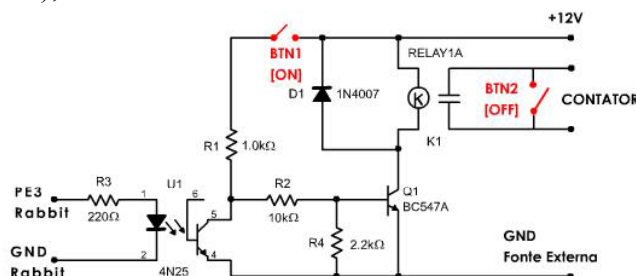


Figura 2 – Parte do circuito que efetua o acionamento do contator

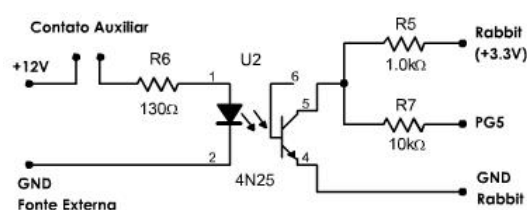


Figura 3 – Parte do circuito que efetua o monitoramento do contato auxiliar

4.2. Procedimentos para implementação:

a) *Criação da MIB de acordo com SMIV1*: Nesta fase foi necessário criar um modelo conceitual do gerenciamento proposto. Como a proposta era ser um modelo simples e didático, foram definidas duas ramificações, a saber: **controle** - responsável pelas características de controle; e **monitoramento** - responsável pelas características de monitoramento do contator. Dentro de cada uma destas entradas foram criados objetos, que gerenciam e controlam determinadas características. A Tabela 1 apresenta as funções associadas a cada objeto da MIB definida para o agente embarcado, assim

² I/O – Input/Output (Entrada/Saída)

³ Optoacopladores – Dispositivos que realizam acoplamento ótico isolando o circuito eletricamente.

como o OID (*Object Identification*) para cada um destes. O arquivo contendo a MIB pode ser visualizado em detalhes no site do projeto [8].

Tabela 1 – Objetos da MIB – Contator

Ramo	Objeto	OID	Descrição
Controle	contator0	1.3.6.1.4.1.12807.3.1.1.1.0	Liga ou desliga o contator0
Monitoramento	contFalha0	1.3.6.1.4.1.12807.3.1.2.1.0	Contagem de falhas do contator0
	contAcciona0	1.3.6.1.4.1.12807.3.1.2.2.0	Contagem de acionamentos / desligamentos
	TempoFunc	1.3.6.1.4.1.12807.3.1.2.3.0	Tempo de funcionamento do agente SNMP

Na Figura 4 visualiza-se a estrutura da árvore de MIB contendo os objetos a serem gerenciados pelo agente SNMP embarcado.

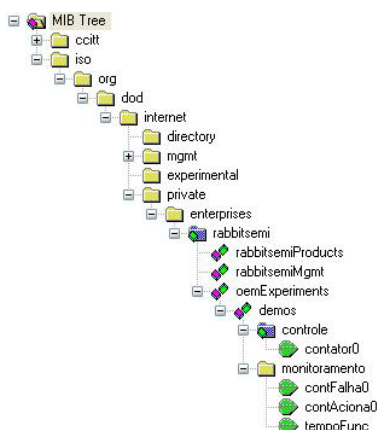


Figura 4 – Representação da Árvore de MIB do agente SNMP

b) *Configuração do Ambiente de Desenvolvimento (Dynamic C 8.1)*: Mudanças nos arquivos de biblioteca (rcm3200.lib e tcp_config.lib) de maneira a configurar características específicas (configuração dos bits das portas paralelas, mascaramento e direcionamento de entrada ou saída, endereço IP, portas utilizadas, etc.).

c) *Desenvolvimento do programa para o microprocessador Rabbit (código do agente SNMP embarcado)*: Escrita do código fonte em Linguagem C (para Dynamic C), onde foram especificadas as condições para o funcionamento do agente, com a utilização de funções do compilador para a implementação das operações SNMP e da MIB. O código fonte pode ser visualizado no site do desenvolvimento do projeto [8].

d) *Desenvolvimento dos circuitos (interface) de acionamento/monitoramento*: O circuito de acionamento (Figura 2) permite o acionamento/desligamento da carga elétrica (ligada ao contator) tanto via operação SNMP (*SetRequest*), quanto via chave [sw1] na placa de desenvolvimento. Já o circuito de monitoramento (Figura 3) realiza o monitoramento do contato auxiliar do contator, verificando seu estado (aberto ou fechado). Quando aberto, deduz-se que, ou o contator está desligado, ou o contator está com problemas. Este estado é monitorado pelo código do agente, gerando-se uma *trap* sempre que este evento ocorrer.

e) *Chaves de manutenção e de acionamento manual*: A chave de manutenção permite a manutenção do Rabbit e do circuito de acionamento/monitoramento do contator. Já a chave de acionamento manual (botoeira) efetua o acionamento ou desligamento da carga elétrica de forma manual, quando a chave de manutenção estiver ligada. A Figura 5 apresenta um diagrama de blocos da interligação do hardware que compõe este estudo.

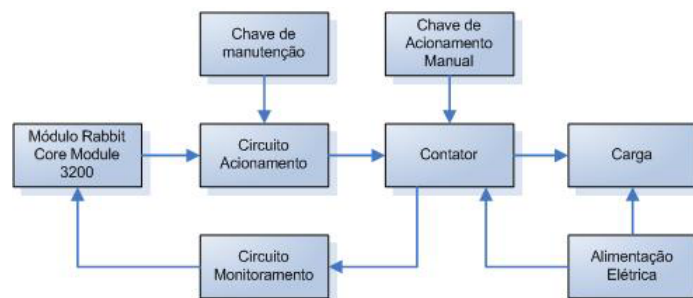


Figura 5 - Diagrama de Blocos da Interligação do Hardware

f) *Testes com agente SNMP*: visualização dos valores dos objetos com software MG-SOFT MIB Browser, e coleta de pacotes com o *Sniffer* (Analisador de Pacotes) Ethereal para visualização das mensagens SNMP e suas PDU's (*Protocol Data Unit*).

4.3. Funcionamento do Agente

Os objetos da MIB são definidos de maneira bastante intuitiva, de modo que ao se observar a árvore da MIB, tem-se uma clara percepção das funções dos objetos do agente SNMPv1 a serem monitorados. Este agente possui as seguintes funcionalidades:

- **Acionamento do contador** – acionamento através de comando SNMP [*set*] enviado pelo gerente SNMP da rede ou através de uma chave tipo *push-button* [*sw1*] localizado na placa do kit de desenvolvimento;
- **Monitoramento de algumas características do contador**
 - **Contador de Acionamento** – conta os acionamentos e desligamentos;
 - **Contador de Falhas** – conta as falhas detectadas pelo agente;
 - **Tempo de Funcionamento** – armazena o tempo de funcionamento do agente SNMPv1, desde o início das operações, ou desde o último *reset*;
 - **Checagem de erros** – a cada falha ocorrida é gerado imediatamente um aviso assíncrono (*trap* tipo 20 – erro), e novos avisos continuarão a ser gerados a cada 30 segundos. No caso de restabelecimento do serviço, o envio destas *traps* será automaticamente interrompido, e partir disto será gerado um outro tipo de *trap* (*trap* tipo 30 – restabelecimento), informando o retorno do contador às suas operações.

5. Conclusões

A partir dos resultados obtidos nos experimentos descritos nesta investigação, observou-se uma solução para o monitoramento de dispositivos de acionamento elétrico via protocolo SNMP, que pode ser utilizada em automação industrial para acionamento de cargas como motores ou similares.

Os trabalhos relacionados com desenvolvimento de agentes SNMP são escassos, principalmente no que tange ao SNMP aplicado ao *Rabbit*. Mas dentre a literatura pesquisada destacam-se: as referências [4] e [6], que divulgam soluções de extensibilidade de agentes SNMP utilizando o NET-SNMP, a referência [5] aborda o desenvolvimento de agente SNMP embarcado, similar ao desenvolvido neste trabalho, mas com diferenças: a) No artigo em referência foi utilizada a plataforma TINI (*Tiny Internet Interface*); b) Não suporta SNMP; c) Não suporta o envio de notificações; d) execução de uma máquina virtual Java.

Dentre as vantagens da implementação de monitoramento através do SNMP, destacam-se: o acompanhamento on-line de algumas características pré-determinadas do sistema gerenciado, que reduz o tempo médio entre falhas, o caráter pró-ativo à manutenção do sistema, através de verificações síncronas (*polling*) e assíncronas (*traps*), que reduz prejuízos oriundos da demora na detecção de problemas.

O microprocessador *Rabbit* pode ser uma alternativa interessante quando se pretende realizar uma implementação de qualidade com custos reduzidos. O custo final é cerca de 10 vezes menor que o de uma estação de trabalho convencional (tipo PC), com a vantagem de redução da manutenção, visto que trata-se de um sistema muito mais simplificado, não tendo a necessidade de instalação e configuração de sistema operacional, que poderia ainda estar sujeito a ataques às fragilidades do sistema, como por exemplo vírus de computador ou falha de algum componente de software.

Embora este projeto tenha implementado o SNMPv1, ressalta-se que esta versão foi adotada devido à indisponibilidade de versões mais atualizadas para o módulo SNMP da biblioteca existente no *Dynamic C* v8.1. Recomenda-se que em desenvolvimentos de projetos para chão de fábricas, o protocolo SNMPv1 deva ser utilizado somente para o monitoramento, mas nunca para o controle de dispositivos. Isto deve-se à fragilidade do SNMPv1 em relação ao quesito segurança. Para situações de controle real indica-se o uso do SNMPv3, que incorpora conceitos de segurança, com protocolos de criptografia (DES, MD5 e SHA-1) e protocolos mais sofisticados de autenticação de mensagem como o HMAC.

Bibliografia

- [1] MAURO, Douglas; Devin J. Schmidt; **SNMP Essencial** – Tradução de Tereza Cristina Feliz de Souza – Rio de Janeiro: Campus, 2003.
- [2] STALLINGS, William. **SNMP, SNMPV2, SNMPV3, and 1 RMON and 1 and 2. 3.** ed. Massachusetts: Addison Wesley, 1999.
- [3] SUBRAMANIAN, Mani, - **NETWORK MANAGEMENT**, - Principles and Practice, Addison Wesley, 2000.
- [4] ZAMBENEDETTI, Lisandro. **Construção de Agente SNMP em Ambiente Linux**, www.inf.ufgrs.br/granville/AgentesSNMP/FevMar-2002 (acessado em 04/2005).
- [5] NACAMURA, Luiz. GARCIA, Osvaldo. FERRASA, Adriano. AUGUSTO, Rômulo. **Desenvolvimento de um agente SNMP embarcado para o gerenciamento de carga em redes de distribuição elétrica**, XXX Seminário integrado de Hardware e Software.
- [6] KADIONIK, Patrice - **USING NET-SNMP UNDER LINUX AND µCLINUX - INTRODUCTION TO NETWORK MANAGEMENT**, www.enseirb.fr/~kadionik/embedded/snmp/english/NET-SNMP_english.html (acessado em 04/2005).
- [7] Documentação referente ao *Rabbit Core Module RCM3200* e *Dynamic C* RCM3200, www.rabbitsemiconductor.com/products/rcm3200/index.shtml (acessado em 04/2005).
- [8] http://www.redes.cefetgo.br/gl_artigos/001/index.htm - site desenvolvido pelos integrantes do projeto.