



Sistemas Embarcados 2

Semana 11

Professor: Éder Moura

Aluno: Ítalo Marangoni De Souza

11811EAU014

1. DISTRIBUIÇÕES LINUX CUSTOMIZADAS – PROJETO YOCTO E BUILDROOT.

1.1 Anatomia de um software embarcado a partir de uma distribuição Linux.

Uma distribuição Linux embarcada é diferente de uma de desktop tradicional, em relação ao tamanho reduzido, quantidade menor de módulos, processamento menor, etc., dessa forma, sendo adequada para aplicações dedicadas. O software desse tipo, de distribuição, tem a arquitetura básica dividida nas seguintes partes, do nível mais baixo até o topo:

- 1.1.1 Hardware: apesar de não fazer parte da arquitetura do software, esta parte é essencial para qual o sistema deve ser compilado, para ser executado na arquitetura específica da CPU;
- 1.1.2 Bootloader: ao inicializar o hardware, o primeiro programa a ser executado é o bootloader, responsável por toda a inicialização básica do sistema, o que inclui o carregamento e a execução do kernel Linux;
- 1.1.3 Kernel Linux: é o núcleo do sistema operacional, sendo responsável por gerenciar o hardware, como: CPU, memória, dispositivos de entrada e saída; e exporta os serviços para a aplicação do usuário, conhecido por realizar a interface entre o hardware e o sistema operacional;
- 1.1.4 Rootfs: o root file system é o sistema de arquivos principal, possuindo as bibliotecas do sistema, destinadas ao uso dos serviços exportados pelo kernel e bibliotecas de aplicações do usuário;
- 1.1.5 Toolchain: não constitui a arquitetura de software do sistema em execução, porém é essencial ser delineada, por que é um conjunto de ferramentas utilizadas em uma máquina hospedeira (host) para gerar os artefatos de software do sistema para uma máquina alvo (target), atendendo as especificadas de cada arquitetura alvo e as necessidades do usuário;

1.2 Projeto Yocto

É um projeto de código aberto composto de várias ferramentas, as quais buscam a criação de uma distribuição Linux embarcado customizada. A filosofia do projeto é que “O projeto Yocto não é uma distribuição Linux embarcada, ele cria uma customizada para você”.

Sua completude que o diferencia dos demais sistemas de compilação, tem uma comunidade ativa e receptiva, com suporte aos principais fabricantes de semicondutores e é liderado pela The Linux Foundation.

O projeto Yocto tem seus pontos positivos e negativos:

- Positivos:
 - Muito esforço colocado pela comunidade;
 - A compilação pode ser feita por interface, por linha de comando ou interface gráfica;
 - Possui vários pacotes;
 - Fácil uso, incluindo na construção de rootfs, mais fácil que o buildroot;
- Negativos:
 - Grande esforço para encontrar erros;
 - Muito tempo e esforço para ser configurado corretamente;
 - Terminologia confusa a princípio;

1.2.1 Como gerar uma distribuição customizada com o Projeto Yocto.

O passo-a-passo de uma geração de distribuição em projeto Yocto é bem mais complicado que o buildroot, dessa forma, possui uma curva de aprendizado mais íngreme e que assusta principiantes. Podendo ser simplificado como:

1. Baixar o código fonte;
2. Aplicar *patches*;
3. Configuração e compilação;
4. Analisar o resultado e trabalhar na divisão das dependências a partir do conceito de camada;
5. Gerar pacotes;
6. Realização de testes para garantir a qualidade dos pacotes gerados;
7. Gerar *feed* de pacotes;

8. Gerar a imagem final do *rootfs*.

1.3 Buildroot

É uma alternativa ao projeto Yocto, para auxiliar e automatizar a criação de distribuições Linux embarcado. Este possui muitas referências na comunidade, com vários exemplos.

Os pontos positivos e negativos do buildroot:

- Positivos:
 - Ótima documentação e página da wiki;
 - Grande seletude de pacotes;
 - Escalabilidade ótima, o que permite compilação de grandes ou pequenos projetos embarcados;
 - O rootfs podem ser criados como gerenciadores de pacote, mas leva tempo;
- Negativos:
 - Muitos problemas de compilação e depuração;
 - Apesar de poder ser instalado em vários dispositivos, só funciona bem com algumas distribuições de ponta;

1.3.1 Como gerar uma distribuição customizada com o Buildroot.

Primeiramente, o buildroot deve ser instalado na máquina hospedeira que esteja rodando alguma distribuição Linux. Os pacotes essenciais a serem instalados são:

- build-essencial;
- ncurses5;
- bazaar;
- cvs;
- git;
- mercurial;
- rsync;
- scp;
- subversion.

Após a instalação dos pacotes essenciais, o buildroot pode ser baixado do seu site oficial.

Agora devemos realizar as configurações necessárias, prezando por:

- Especificar uma plataforma alvo para a compilação (ou compilação cruzada);
- Escolha da *toolchain*;
- Escolha do *bootloader*;
- Escolha da versão do *kernel* Linux;
- Customizações para o *rootfs*;

Logo após essas configurações, a próxima etapa é a geração das imagens. Se define o local de downloads dos pacotes, inicialmente, permitindo reaproveitamento dos arquivos baixados para outro processo de compilação. Deve ser definido também o tipo do sistema de arquivos. A porta serial de comunicação também deve ser definida. Dessa forma a geração da imagem pode ser feita.

Após as imagens serem geradas, estas devem ser gravadas em um dispositivo de armazenamento a ser conectado ao sistema de destino para executar imagens.