

Medição de desempenho de rede em hardware

Racyus Delano (UFV)
racyus.pacifico@ufv.br

Pablo Goulart (UFMG); Ítalo Cunha (UFMG); Marcos Vieira (UFMG);

Dorgival Guedes (UFMG); Alex Vieira (UFJF); José Nacif (UFV).



Introdução

- Situação da área de redes.



O problema

- Existe grande dificuldade em se medir um grande volume de tráfego de maneira exata e também desenvolver estruturas de medição.
- Principais fatores:
 - As estruturas de medição consomem muito recurso (largura de banda e CPU);
 - Grande investimento no desenvolvimento do hardware;
 - Configurar os programas de medição.

Monitoramento e medição

Por que tarefas de monitoramento e medição são importantes?

- Detectar e resolver falhas na rede (gerenciamento da rede);
- Verificar ocorrência de ataques nos nós folhas;
- Verificar se a qualidade do serviço fornecido pelas operadoras está de acordo com o serviço contratado (QoS).

Estruturas de medição

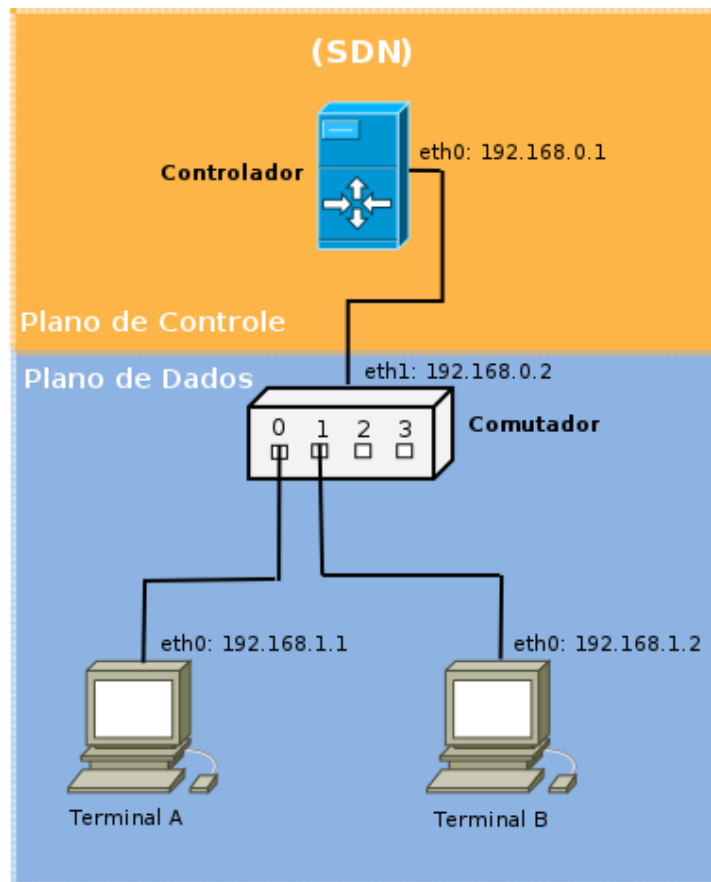
- Tarefas de medição podem ser implementadas utilizando:
 - Contadores de fluxo;
 - Estruturas de dados hash;
 - Fragmentos de código adicionado na CPU do computador.

Objetivo

O objetivo deste trabalho é estender o comutador OpenFlow para realizar medições em redes utilizando um hardware reconfigurável.

Redes definidas por software

- O que são redes definidas por software (SDN)?

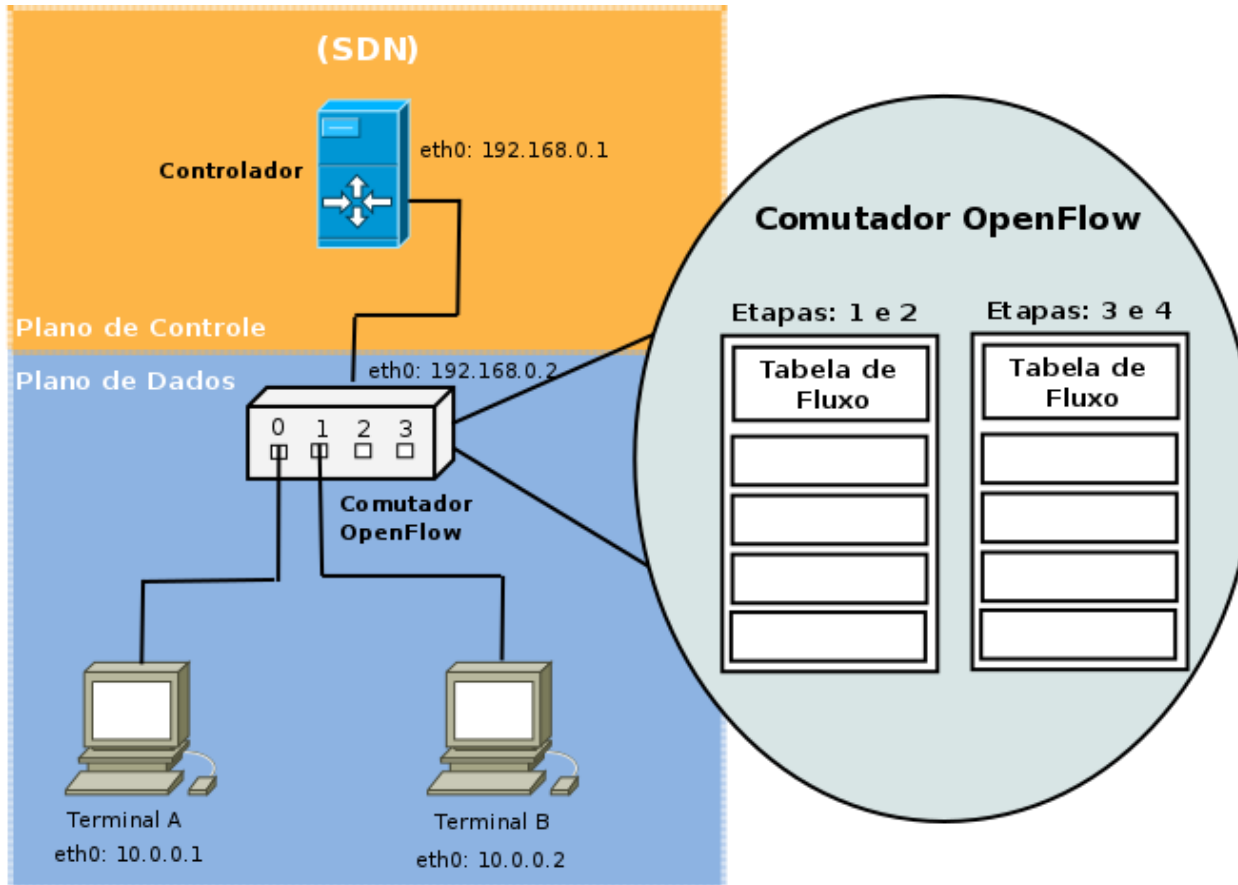


OpenFlow

- **O que é OpenFlow?**
 - É uma API projetada especialmente para SDN. Fornece alto desempenho e controle do tráfego da rede ONF (2012);
 - É composta por uma tabela interna de fluxo e uma interface padronizada para adicionar e remover fluxo de entrada McKeown et al. (2008);
 - Existe uma implementação para ser executada na plataforma NetFPGA;
 - Em relação as tarefas de medição, o OpenFlow apenas mede a contagem de pacotes e *bytes* dos fluxos trafegados.

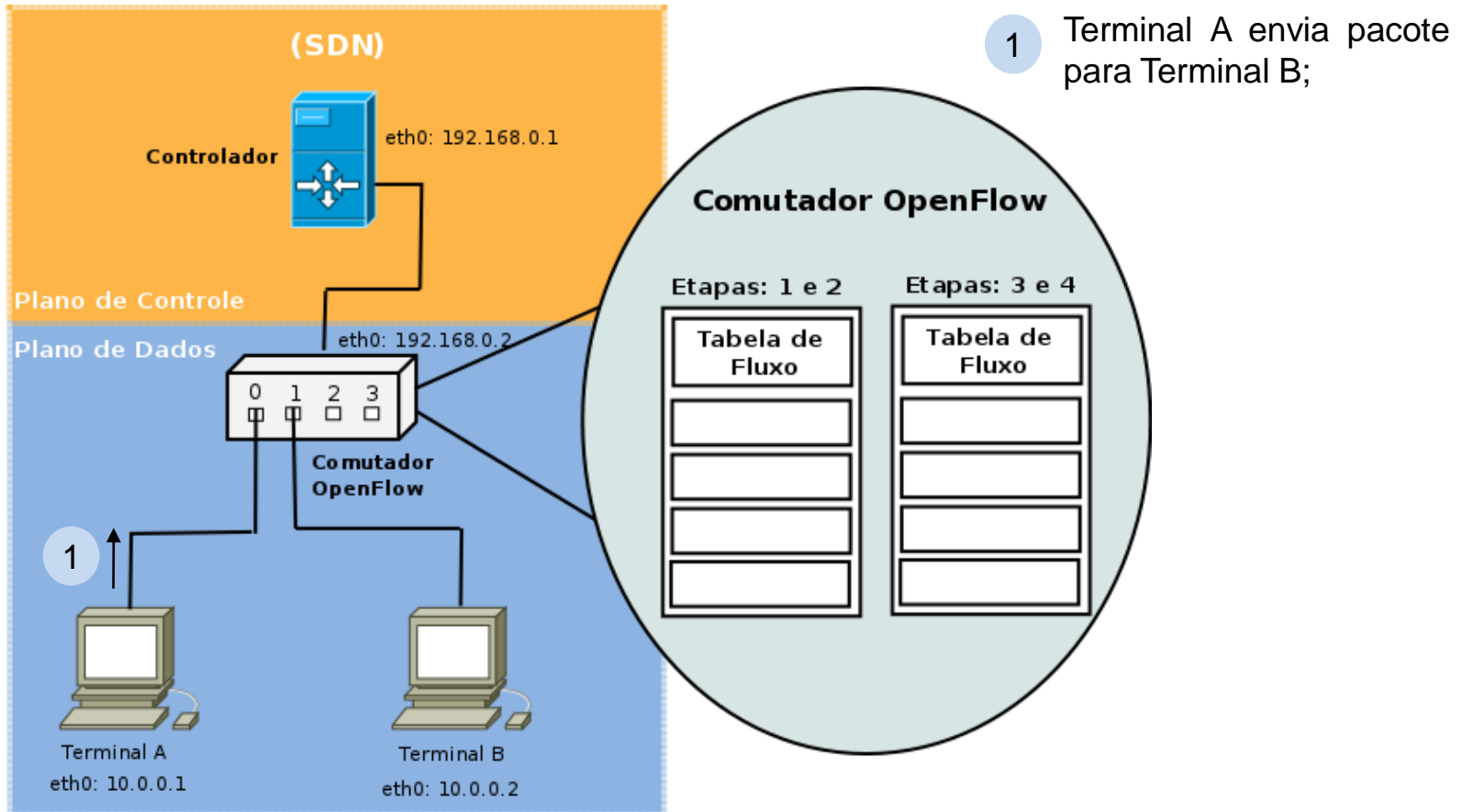
OpenFlow

- Funcionamento



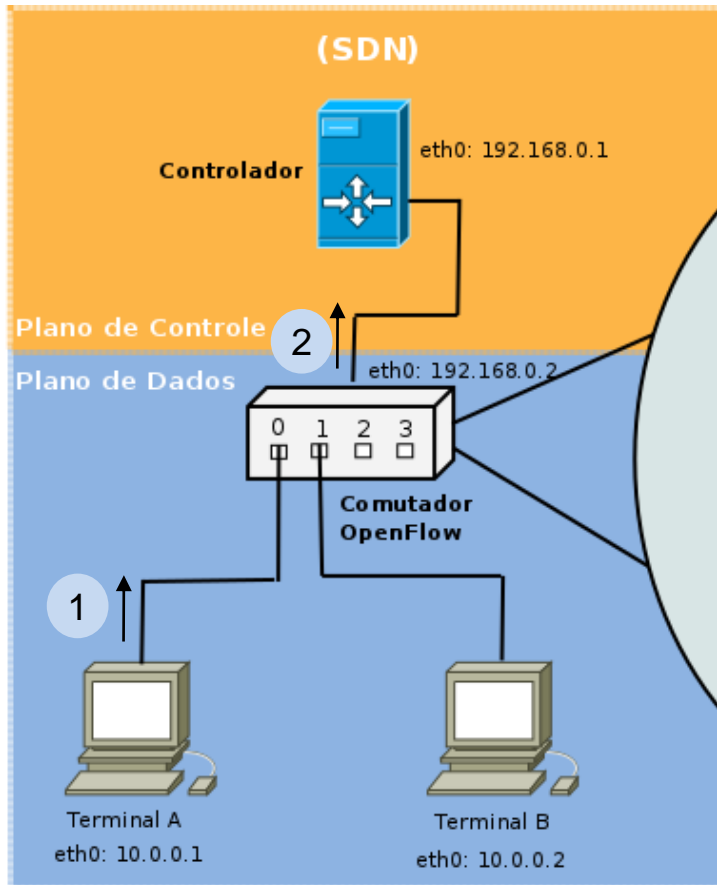
OpenFlow

- Funcionamento



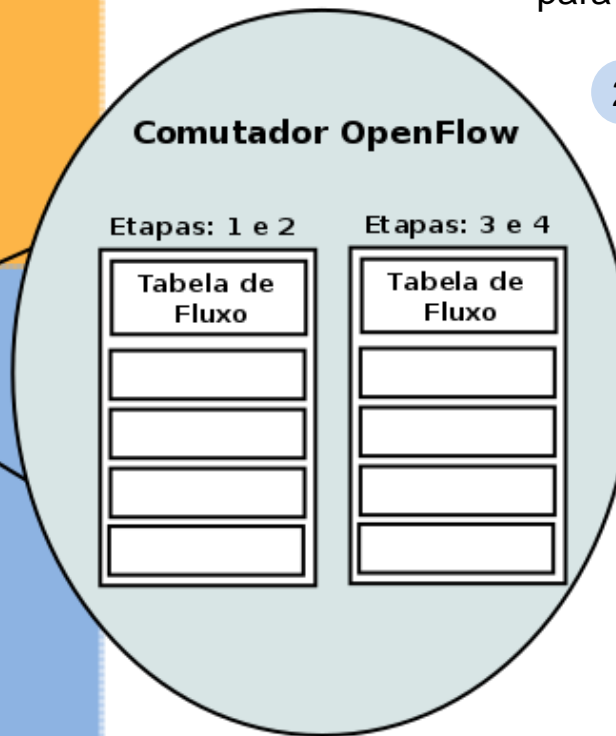
OpenFlow

• Funcionamento



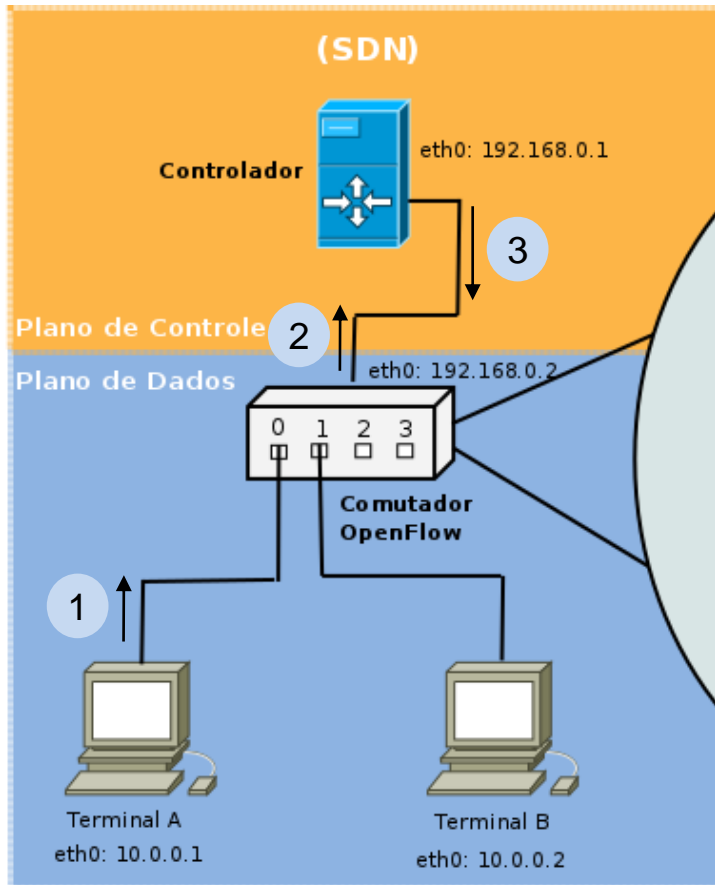
1 Terminal A envia pacote para Terminal B;

2 O pacote não é encontrado na tabela de fluxo. Imediatamente o pacote é enviado ao controlador;



OpenFlow

• Funcionamento



1 Terminal A envia pacote para Terminal B;

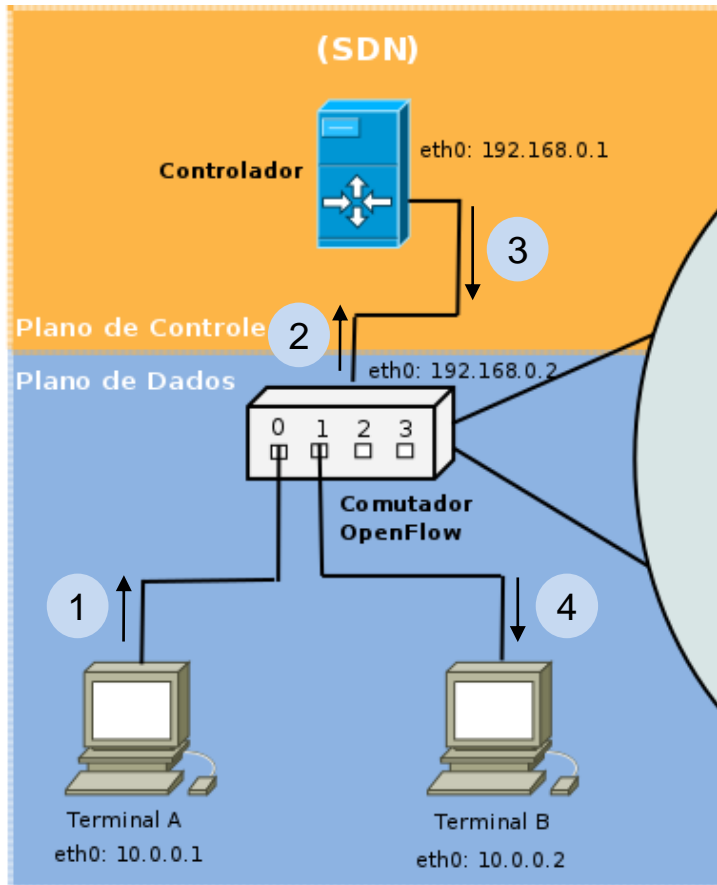
2 O pacote não é encontrado na tabela de fluxo. Imediatamente o pacote é enviado ao controlador;

3 O pacote é examinado. As entradas de fluxo são inseridas na tabela de fluxo do comutador.



OpenFlow

• Funcionamento



1 Terminal A envia pacote para Terminal B;

2 O pacote não é encontrado na tabela de fluxo. Imediatamente o pacote é enviado ao controlador;

3 O pacote é examinado. As entradas de fluxo são inseridas na tabela de fluxo do comutador.

4 O pacote chega no Terminal B;

Comutador OpenFlow

Etapas: 1 e 2

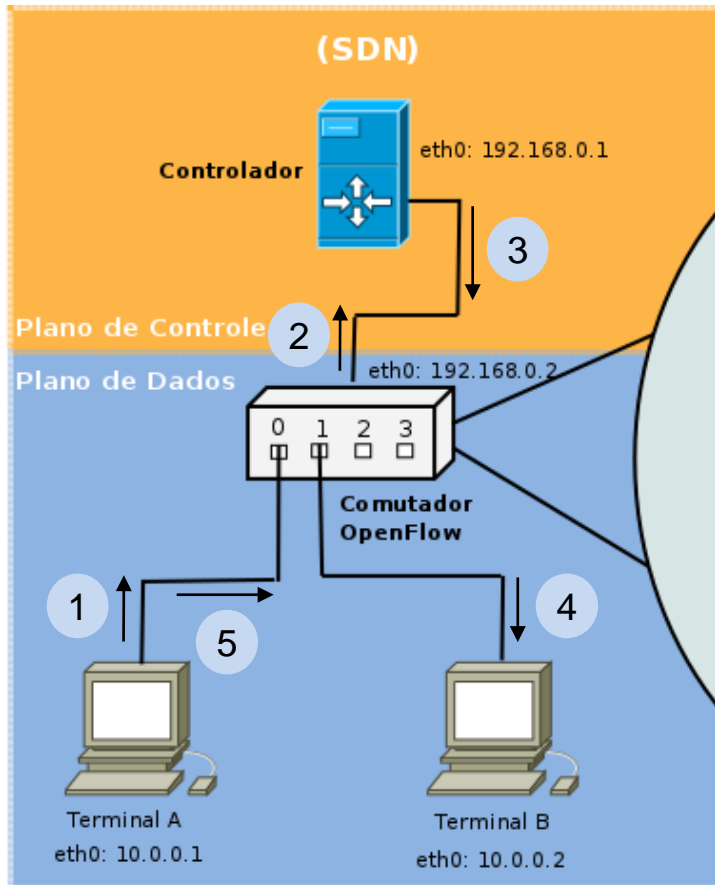
Tabela de Fluxo

Etapas: 3 e 4

Tabela de Fluxo
Terminal A
Terminal B

OpenFlow

• Funcionamento



1 Terminal A envia pacote para Terminal B;

2 O pacote não é encontrado na tabela de fluxo. Imediatamente o pacote é enviado ao controlador;

3 O pacote é examinado. As entradas de fluxo são inseridas na tabela de fluxo do comutador.

4 O pacote chega no Terminal B;

5 Terminal A envia outro pacote para o Terminal B;

Comutador OpenFlow

Etapas: 1 e 2

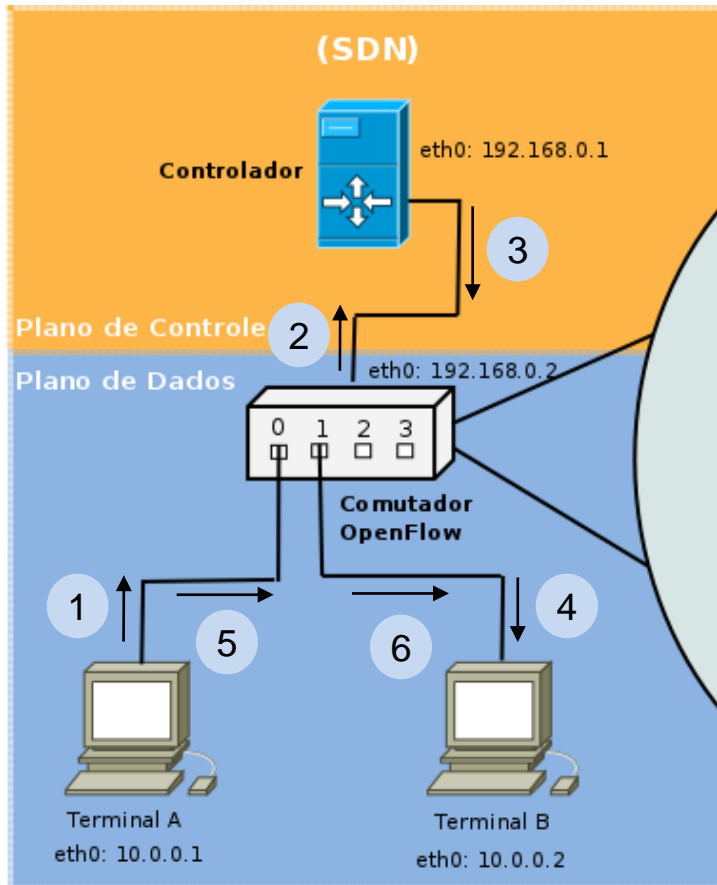
Tabela de Fluxo

Etapas: 3 e 4

Tabela de Fluxo
Terminal A
Terminal B

OpenFlow

• Funcionamento



- 1 Terminal A envia pacote para Terminal B;
- 2 O pacote não é encontrado na tabela de fluxo. Imediatamente o pacote é enviado ao controlador;
- 3 O pacote é examinado. As entradas de fluxo são inseridas na tabela de fluxo do comutador.
- 4 O pacote chega no Terminal B;
- 5 Terminal A envia outro pacote para o Terminal B;
- 6 Pacote chega no Terminal B;



NetFPGA

- **O que é NetFPGA?**

Lockwood et al. (2007) define NetFPGA como sendo uma plataforma que habilita estudantes e pesquisadores a construir sistemas de rede com alto desempenho em hardware.



NetFPGA 1G



NetFPGA 10G



NetFPGA SUME



NetFPGA CML

Métrica de medição

- **Tempo de chegada entre pacotes**

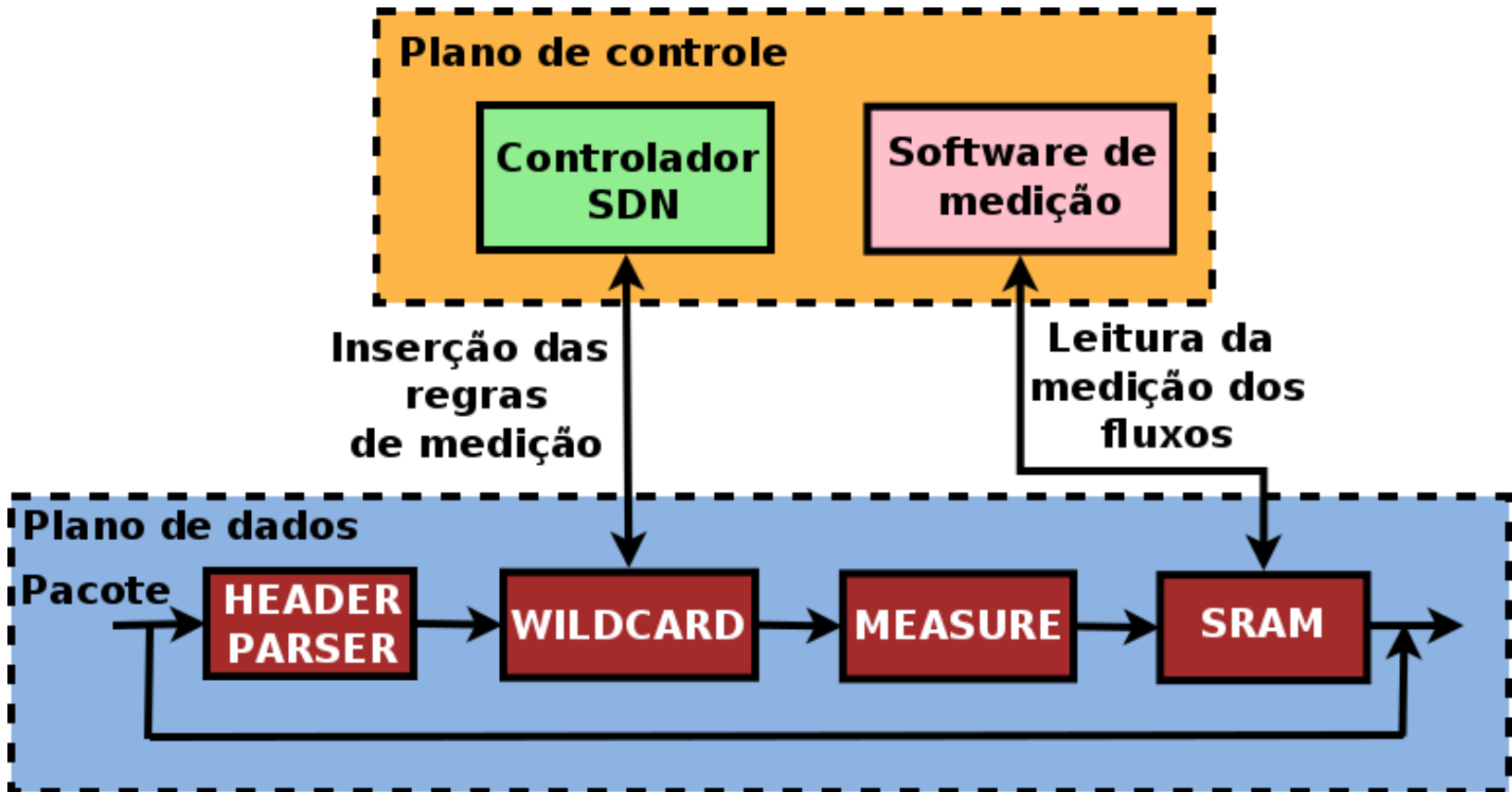
Escolhemos a métrica tempo de chegada entre pacotes por ser uma métrica desafiadora devido os requisitos de tempo real. Implementamos duas variações desta métrica: média **(1)** e variância **(2)**.

$$(1) \text{ media}_j = \frac{\sum_{i=1}^{N_j} dT_i}{(N_j - 1)}, 1 \leq j \leq 28$$

$$(2) \text{ variancia}_j = \frac{\sum_{i=1}^{N_j} (dT_i)^2 - \frac{\left(\sum_{i=1}^{N_j} dT_i\right)^2}{N_j}}{N_j}, 1 \leq j \leq 28$$

Arquitetura proposta

- Plano de controle e dados.



Arquitetura proposta

- **Características**

- Suporta métricas genéricas;
- Escalável;
- Medição centralizada na rede sem depender de medições nos nós folhas da rede;
- Medição de fluxos concorrentemente em intervalos definido pelo usuário na velocidade da interface (line-speed);
- Medição utilizando regra com agregador * (Wildcard).

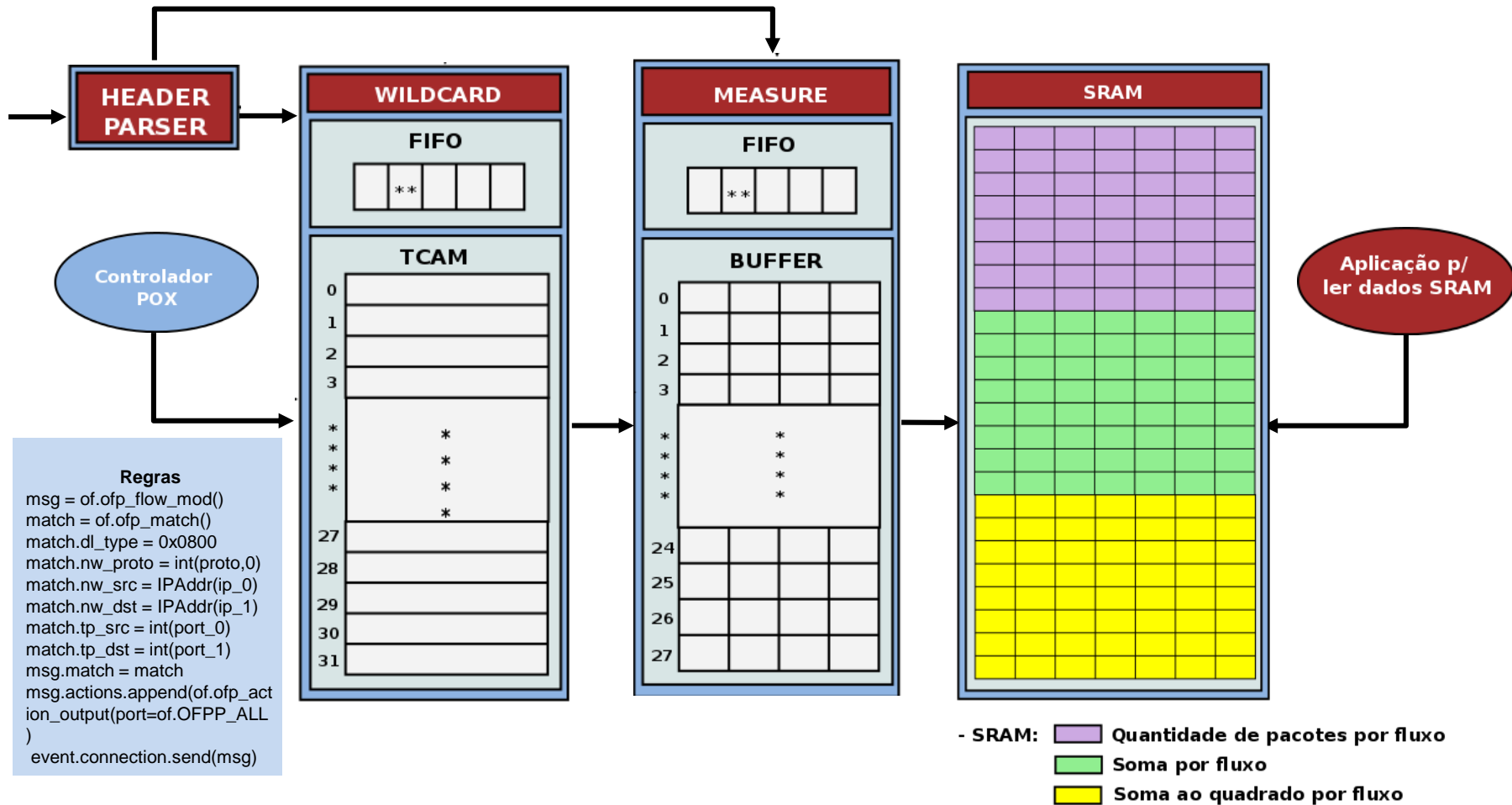
Arquitetura proposta

- **Características**

- Suporta métricas genéricas;
- Escalável;
- Medição centralizada na rede sem depender de medições nos nós folhas da rede;
- Medição de fluxos concorrentemente em intervalos definido pelo usuário na velocidade da interface (line-speed);
- Medição utilizando regra com agregador * (Wildcard).

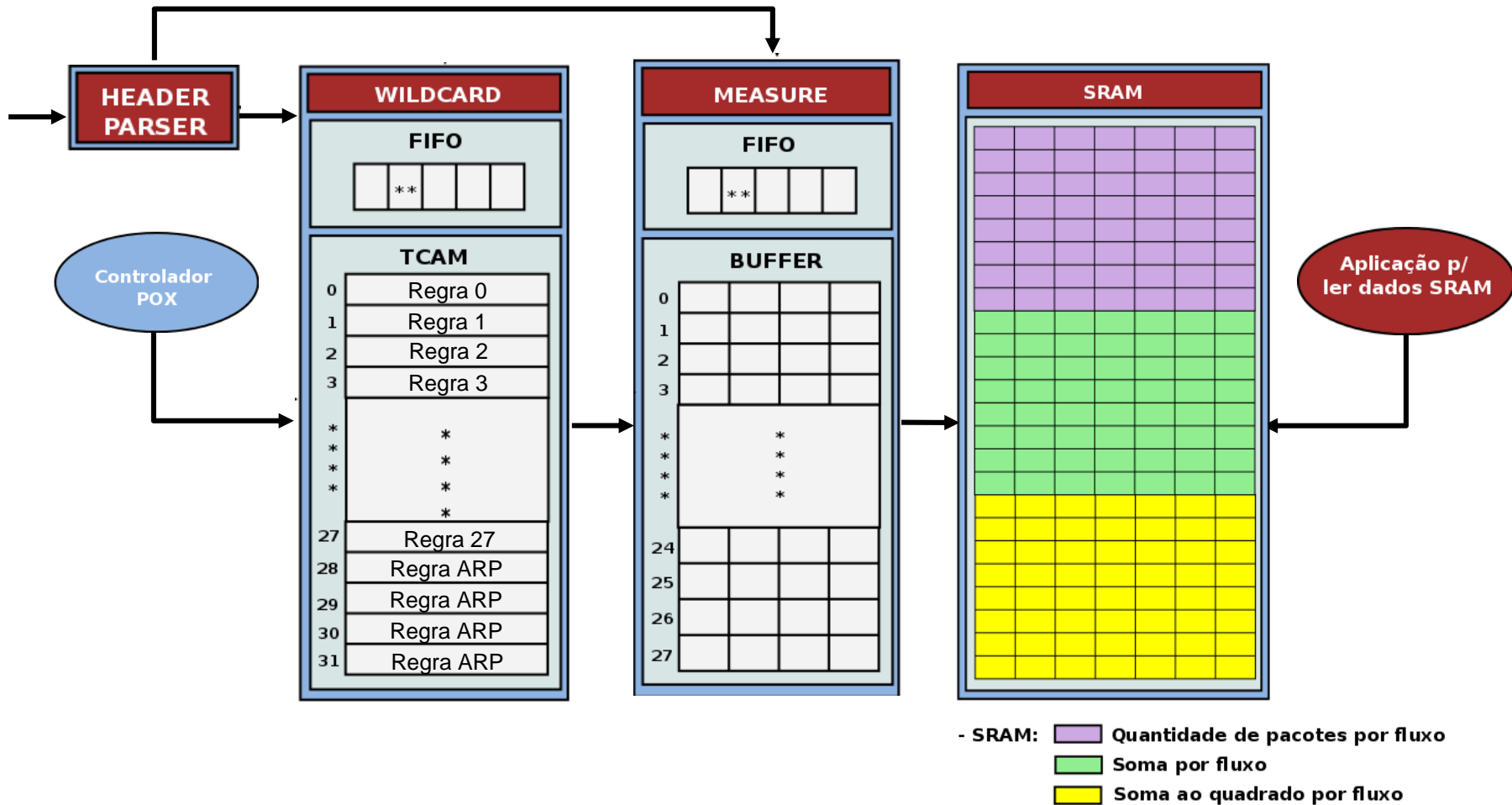
Arquitetura proposta

- Funcionamento



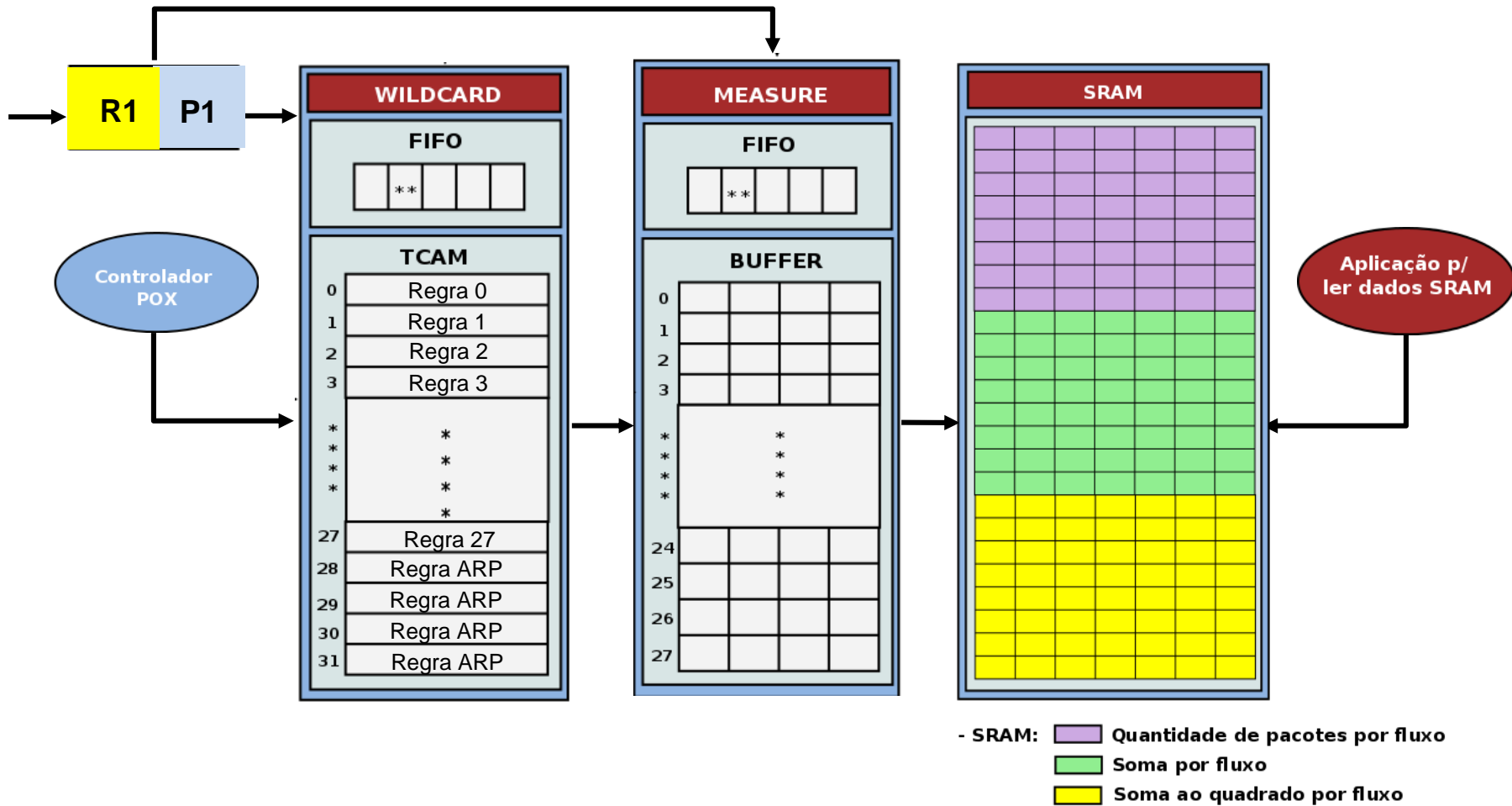
Arquitetura proposta

- Funcionamento



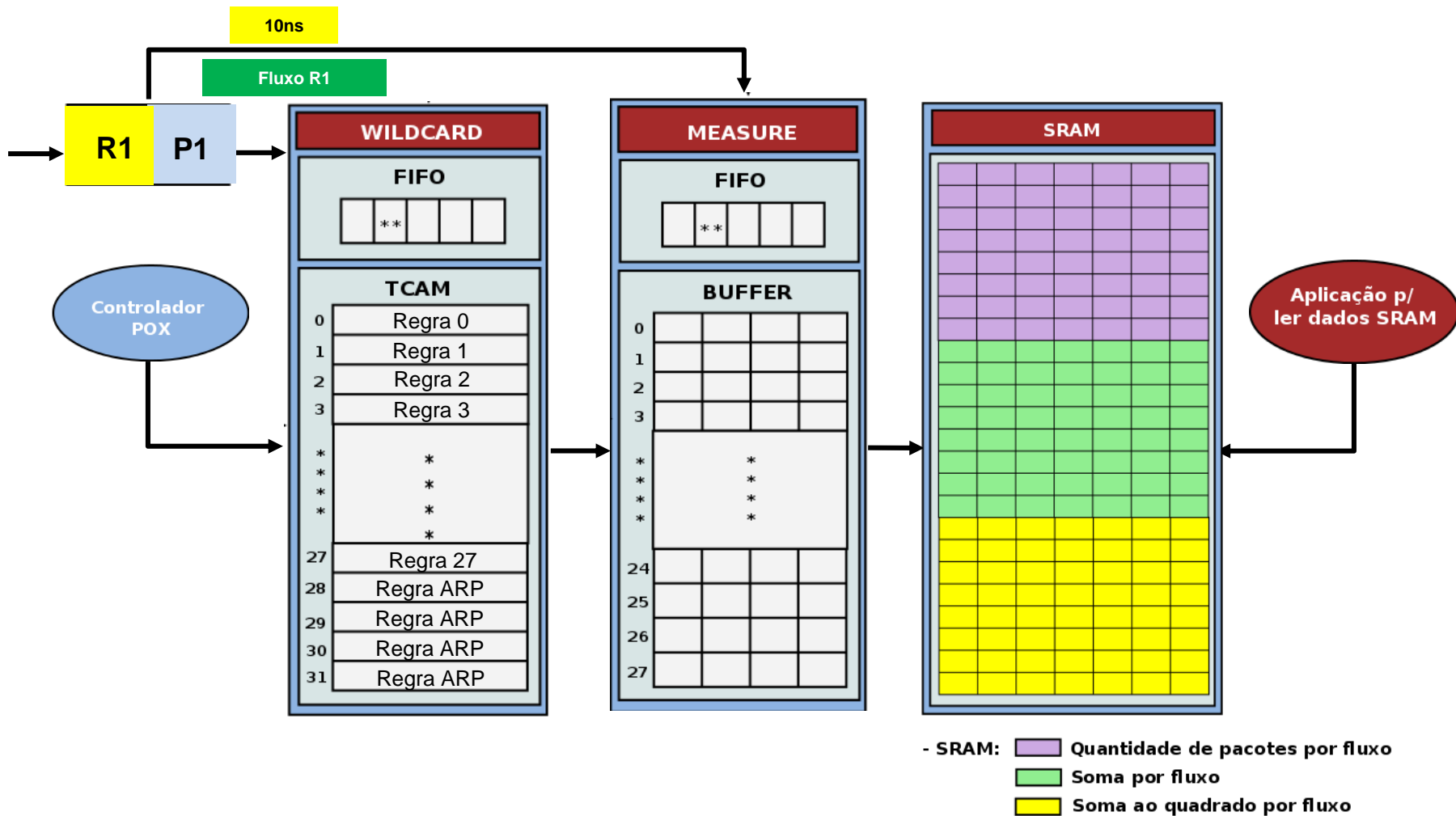
Arquitetura proposta

- Funcionamento



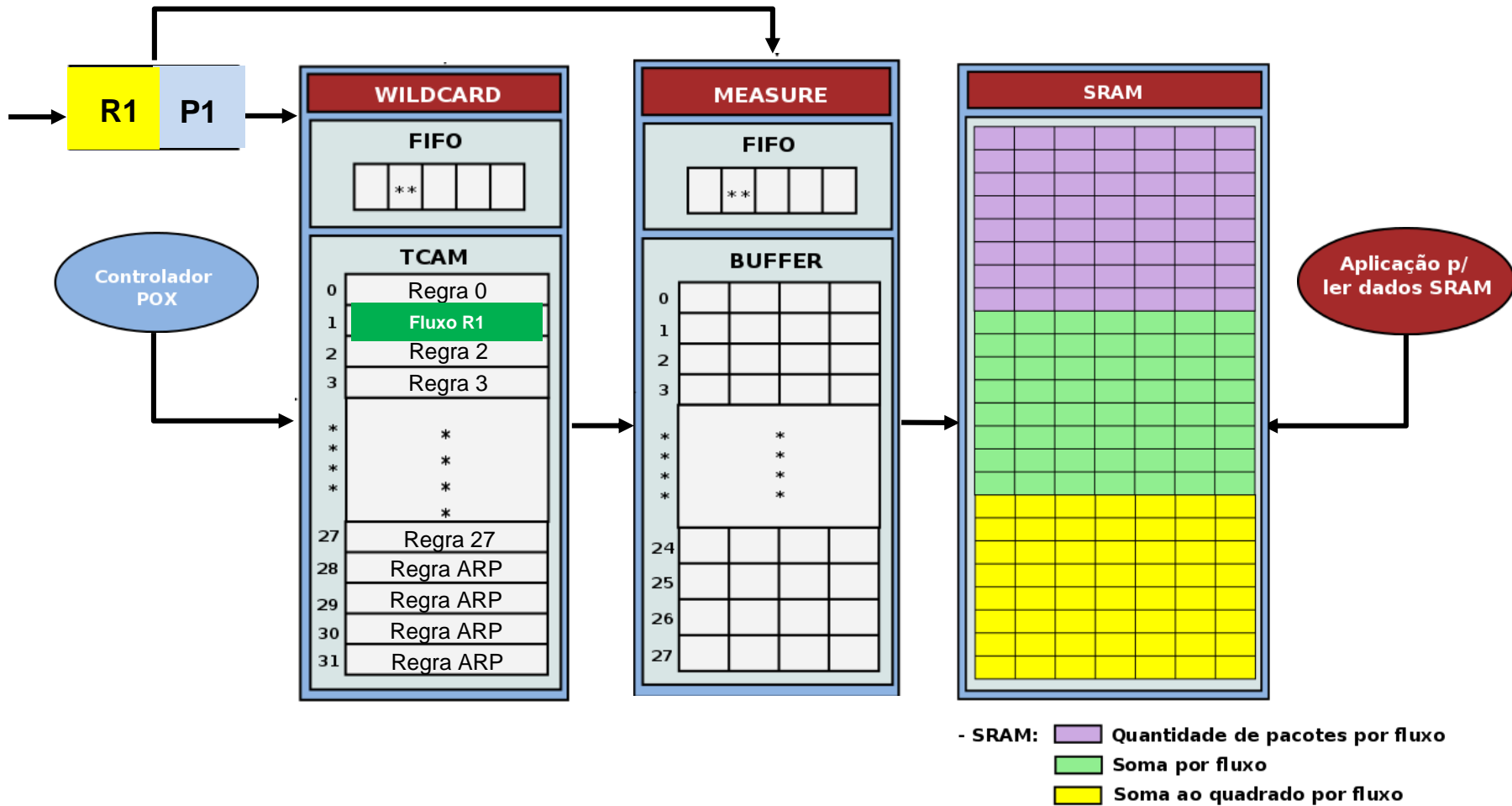
Arquitetura proposta

• Funcionamento



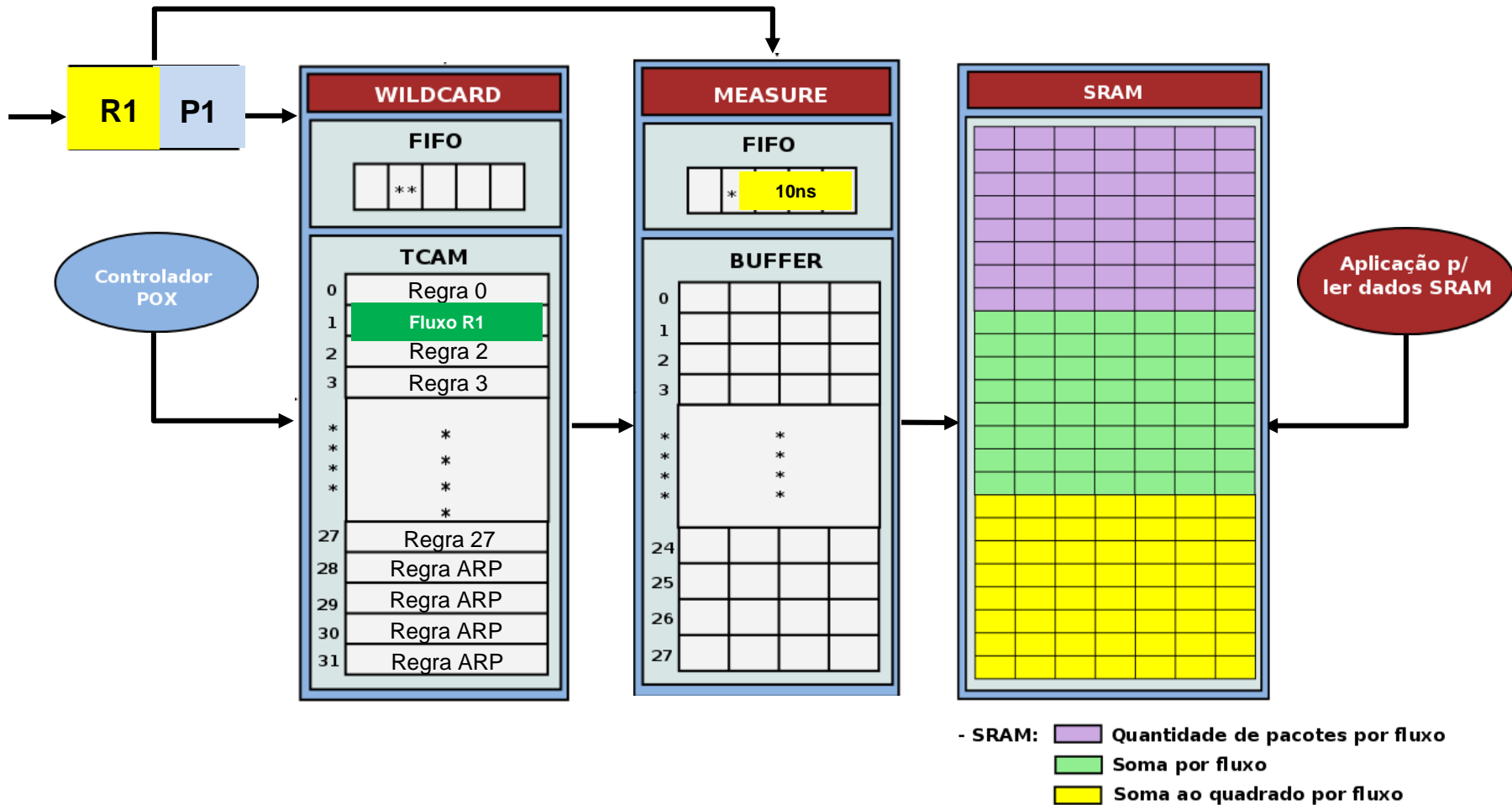
Arquitetura proposta

- Funcionamento



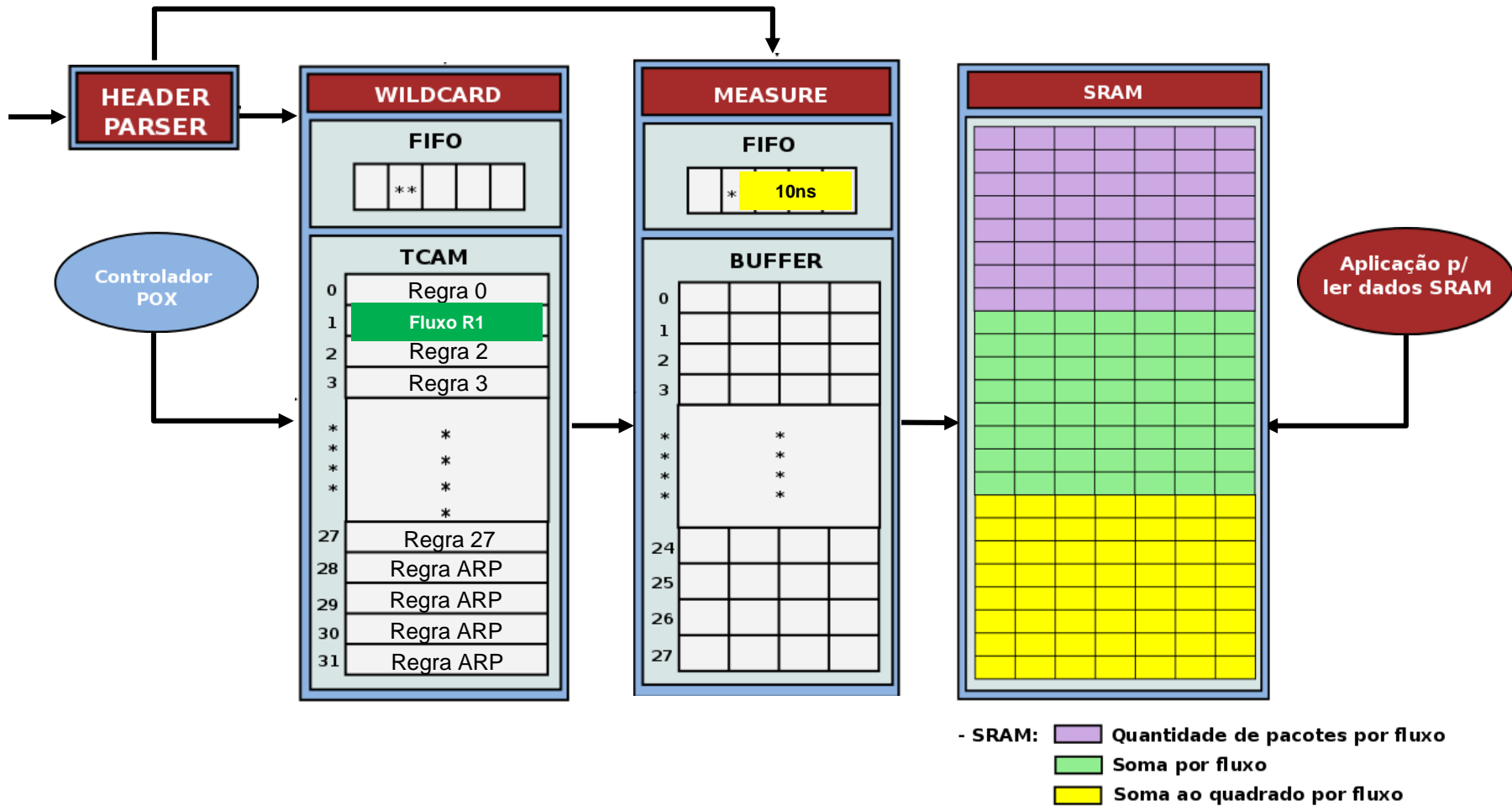
Arquitetura proposta

- Funcionamento



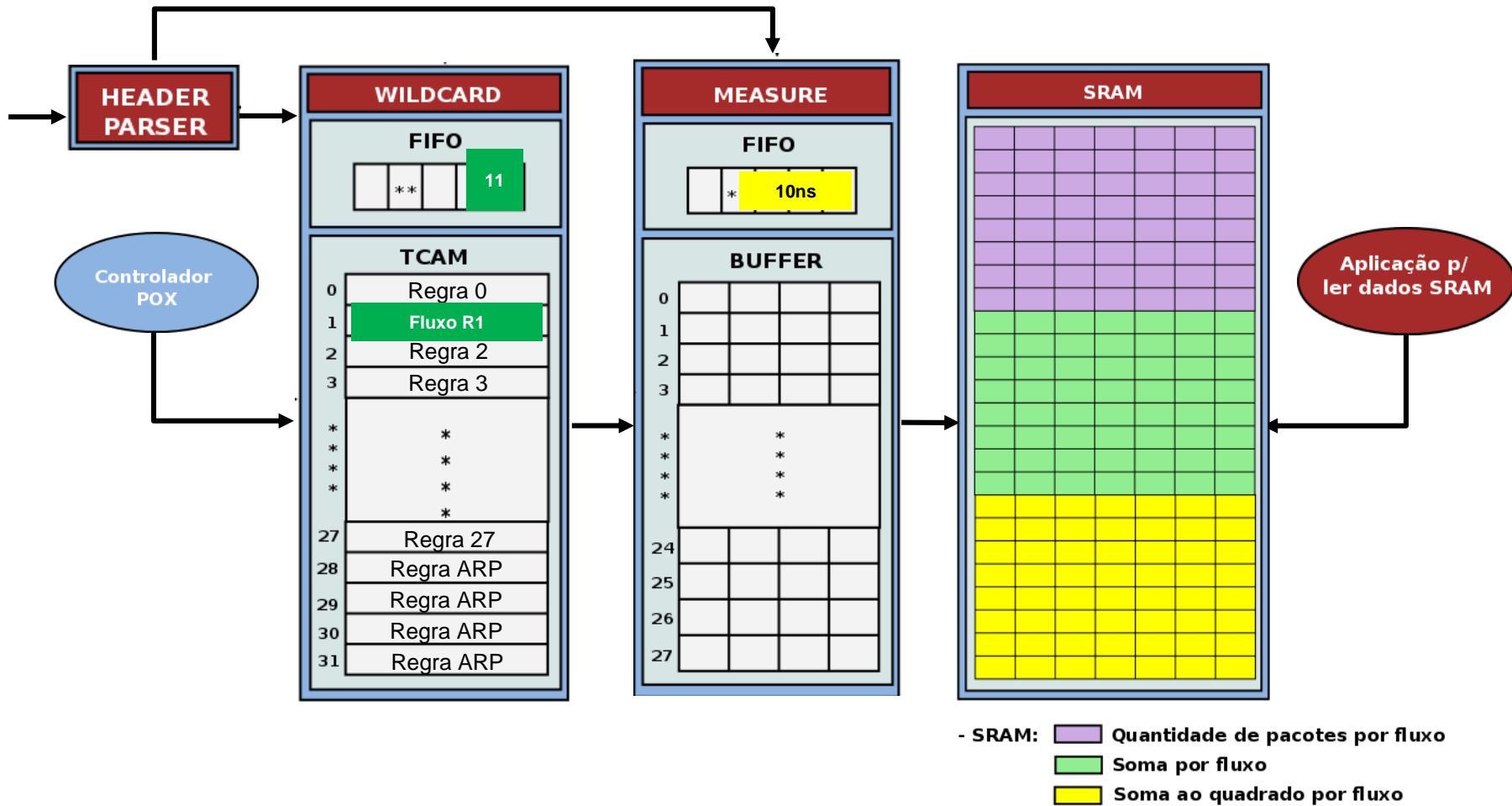
Arquitetura proposta

- Funcionamento



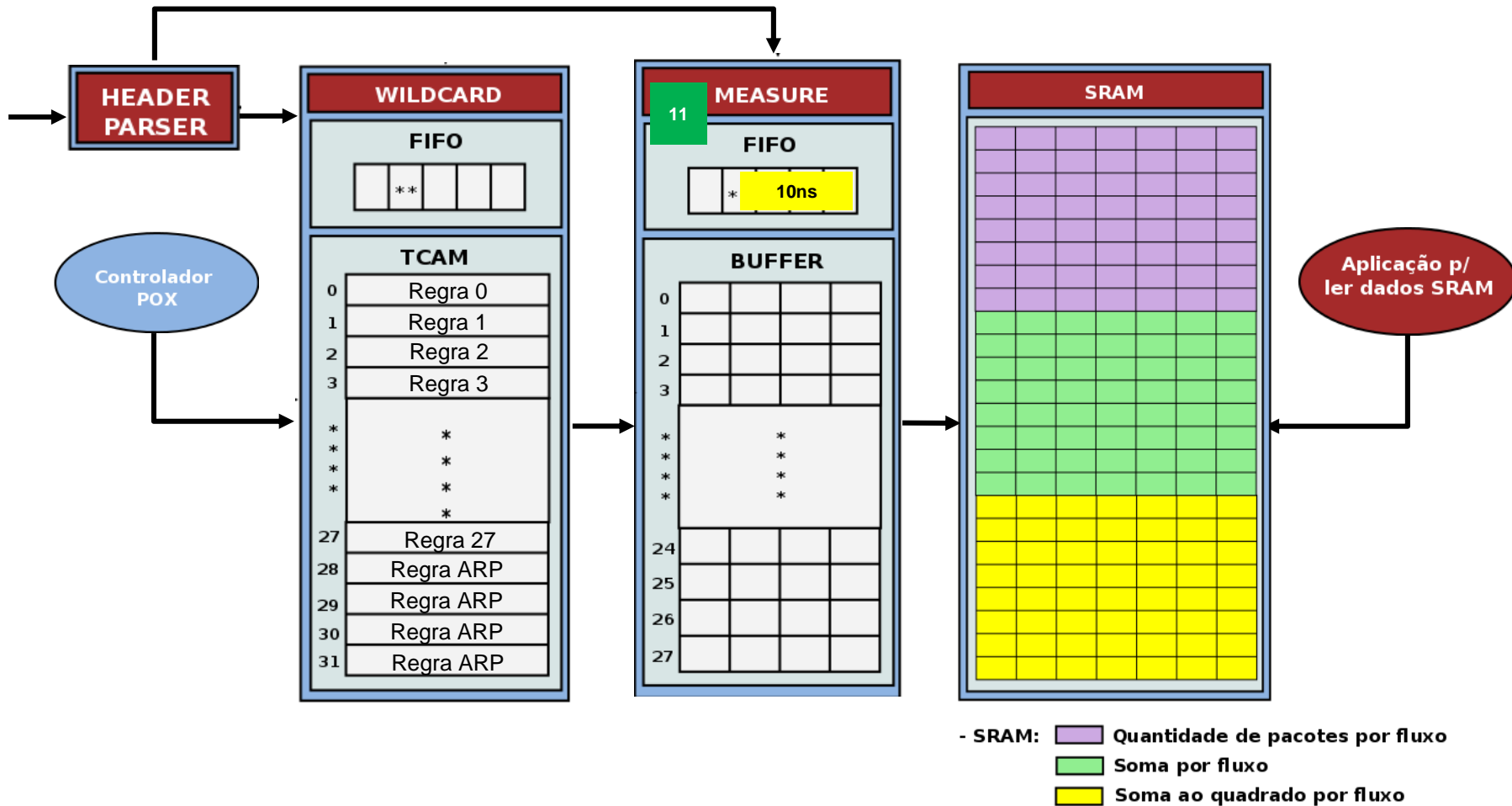
Arquitetura proposta

- Funcionamento



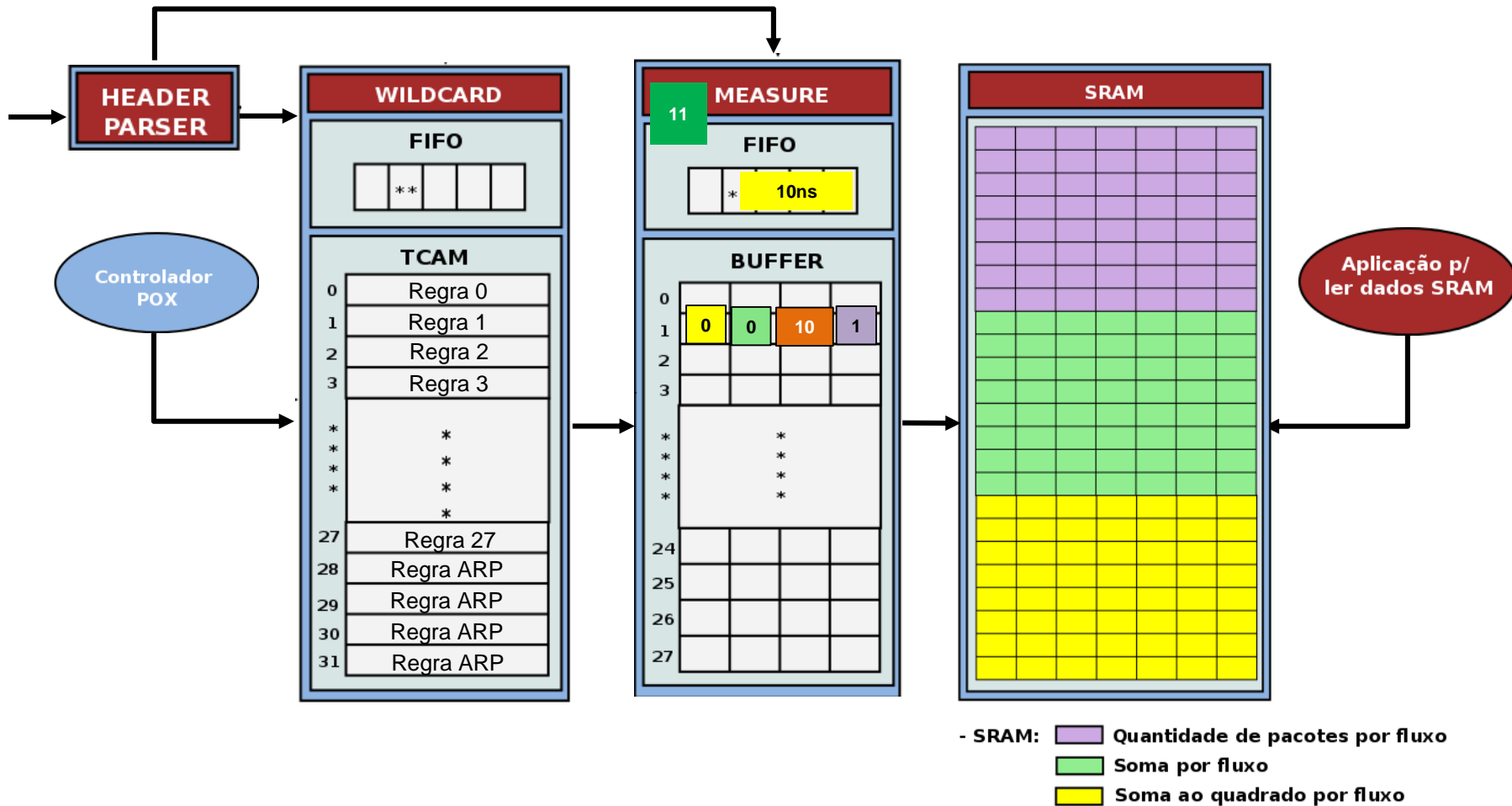
Arquitetura proposta

- Funcionamento



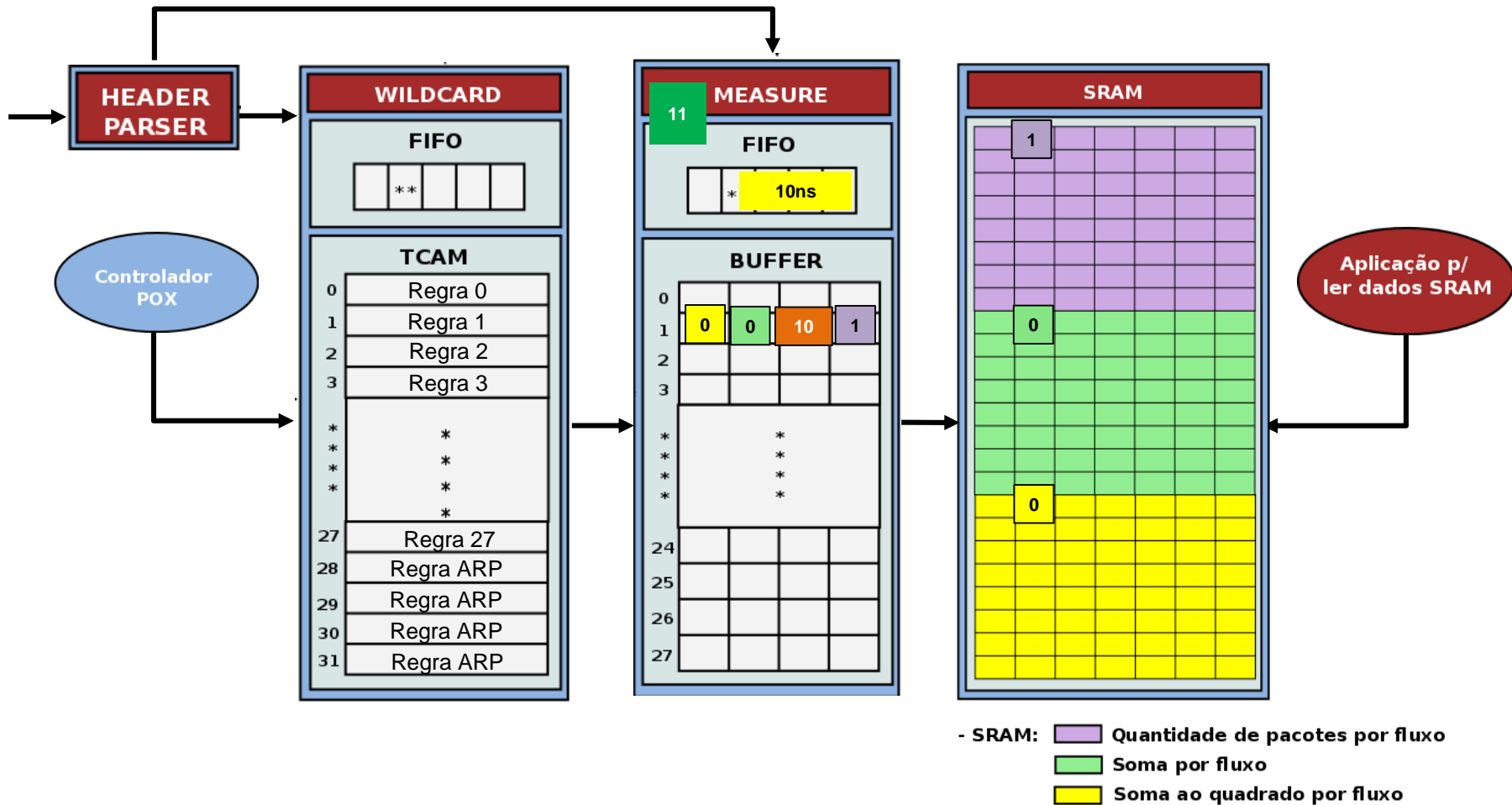
Arquitetura proposta

- Funcionamento



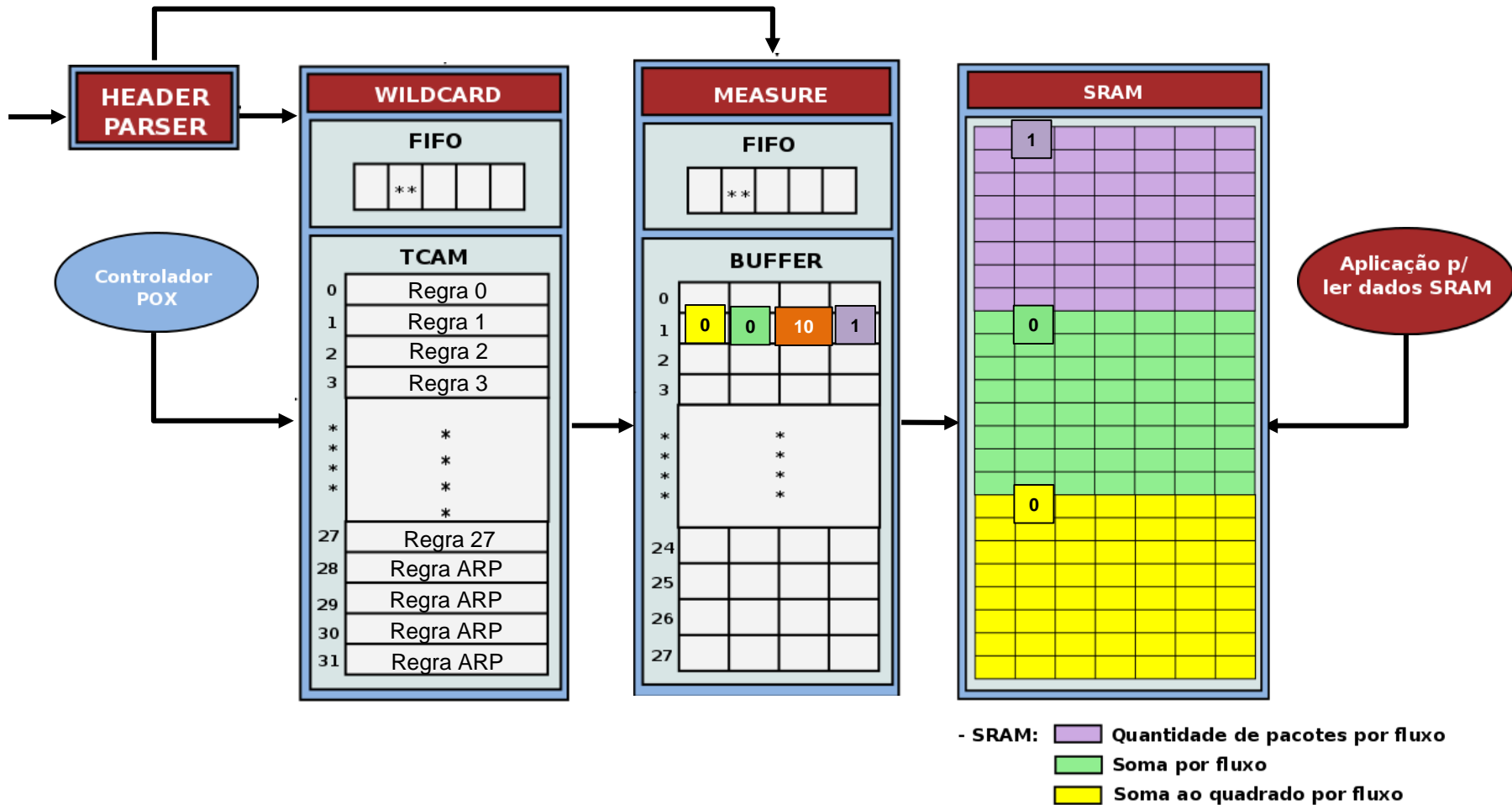
Arquitetura proposta

- Funcionamento



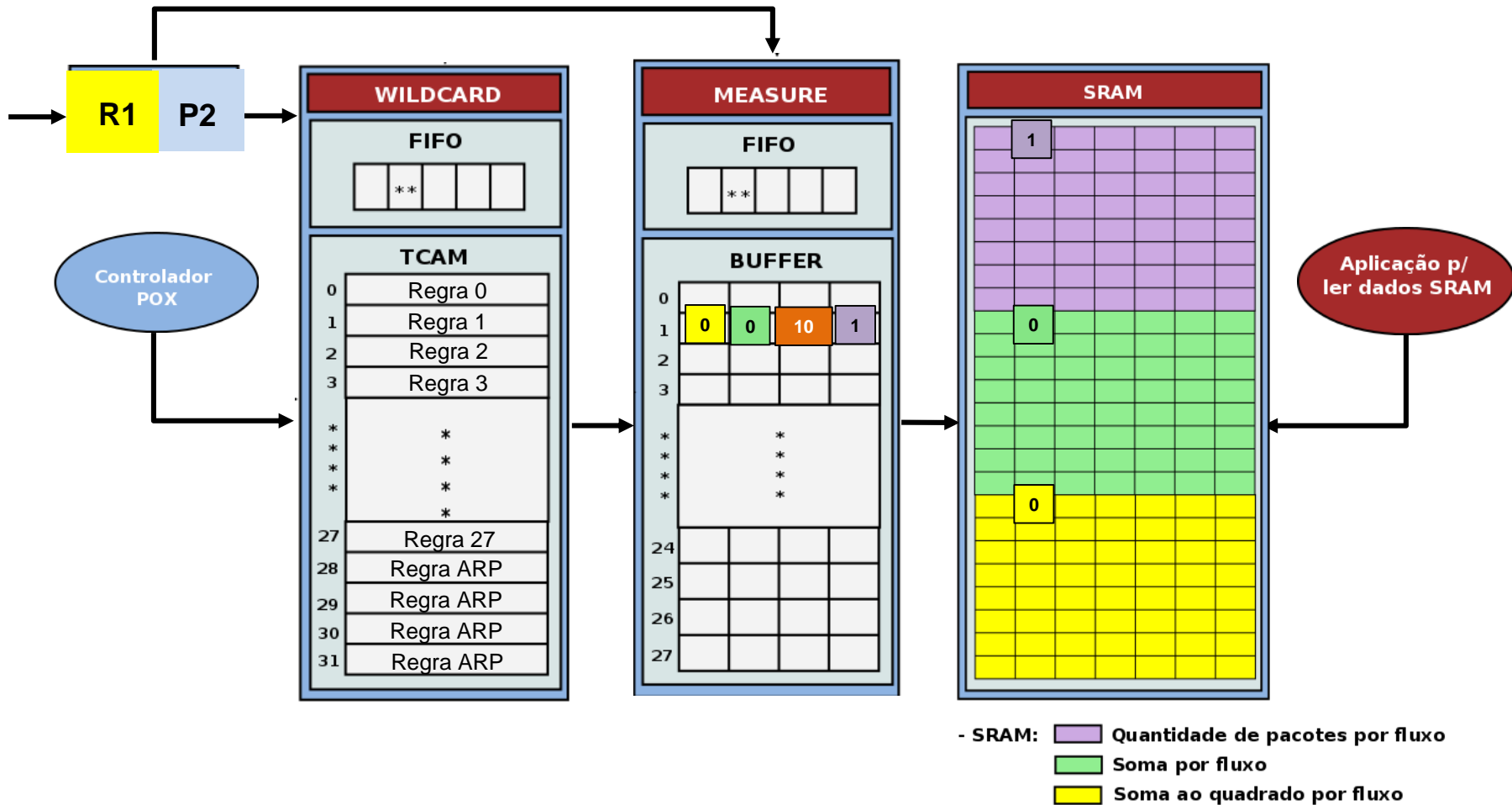
Arquitetura proposta

- Funcionamento



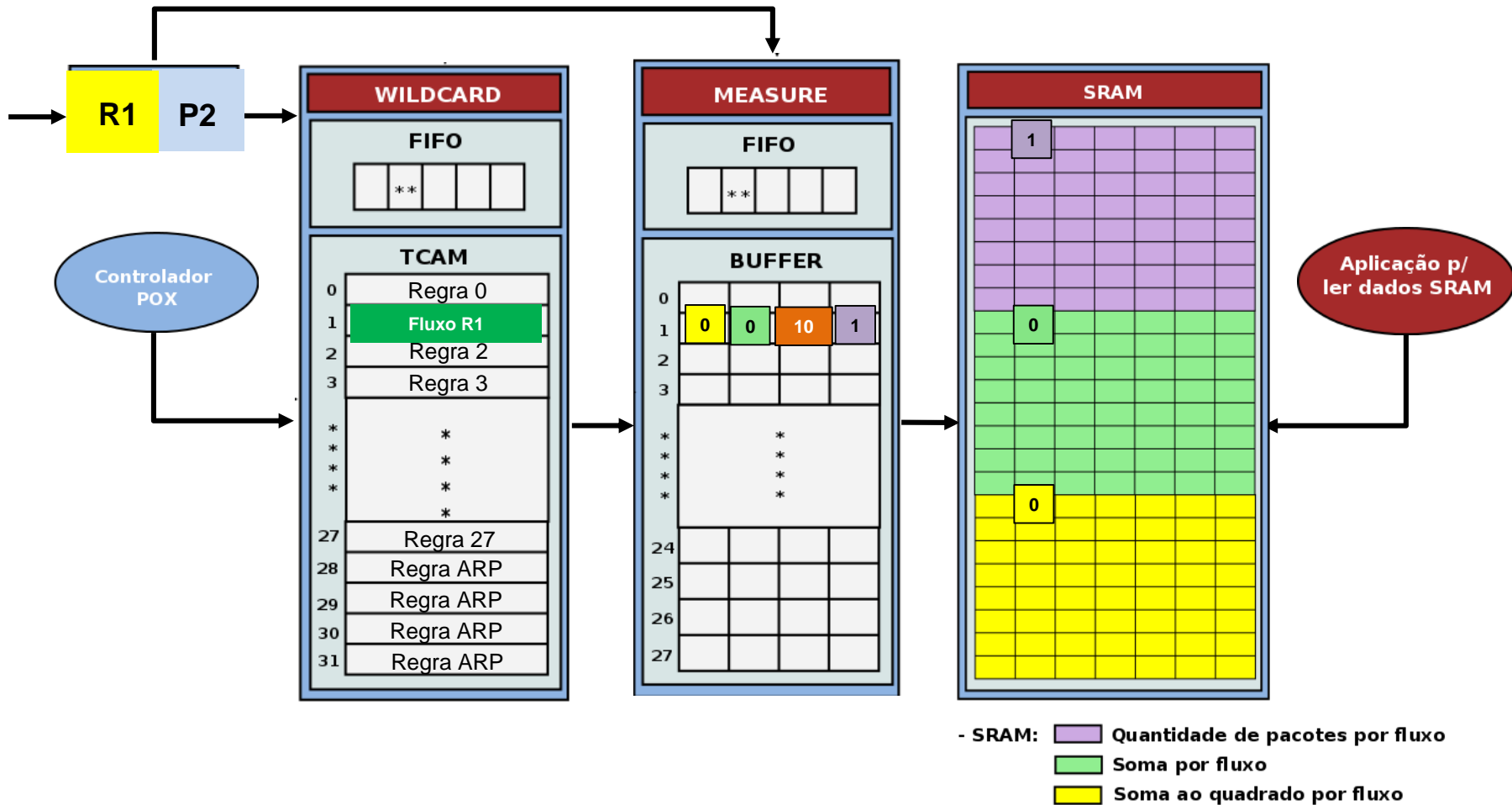
Arquitetura proposta

- Funcionamento



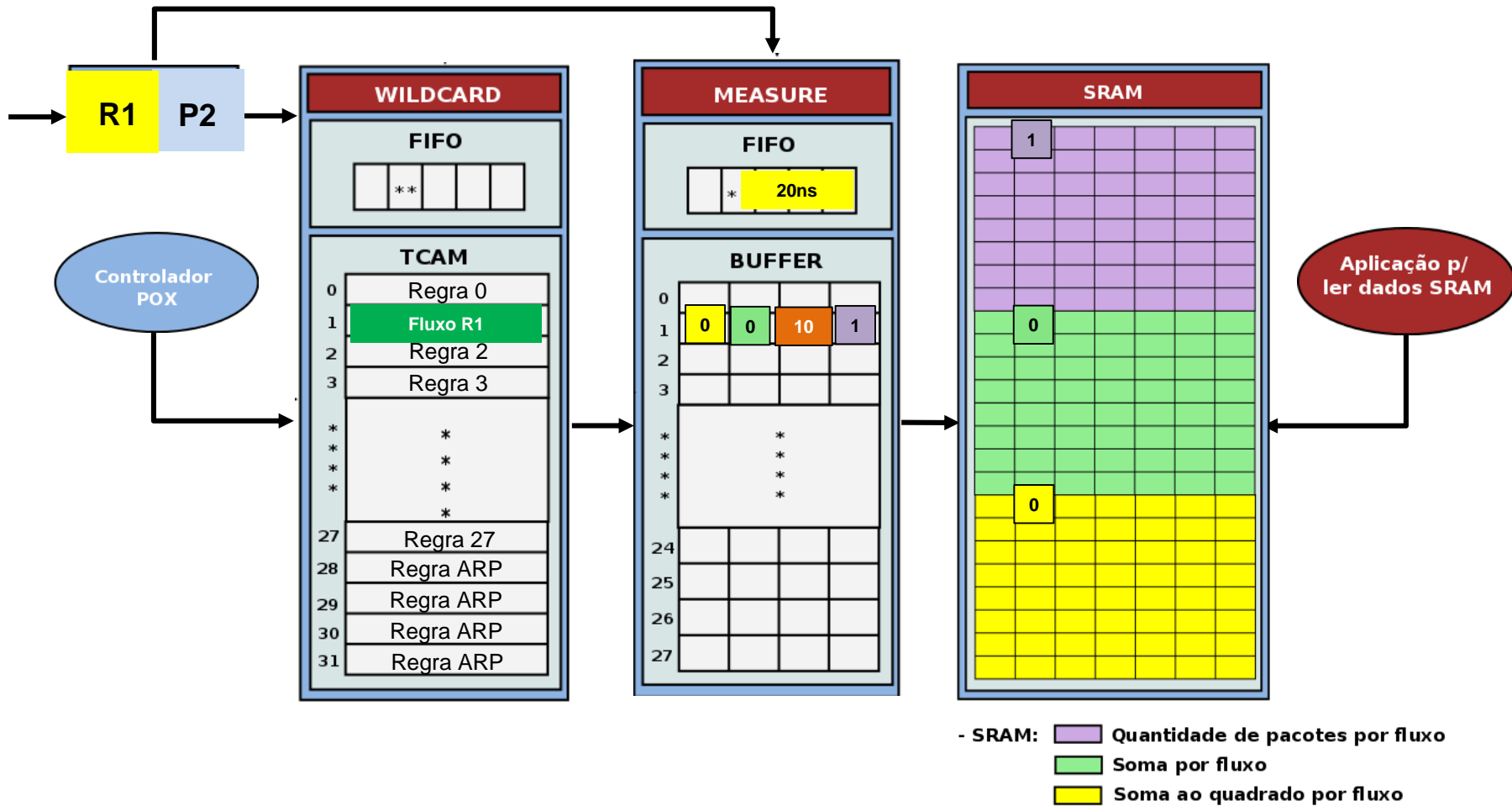
Arquitetura proposta

- Funcionamento



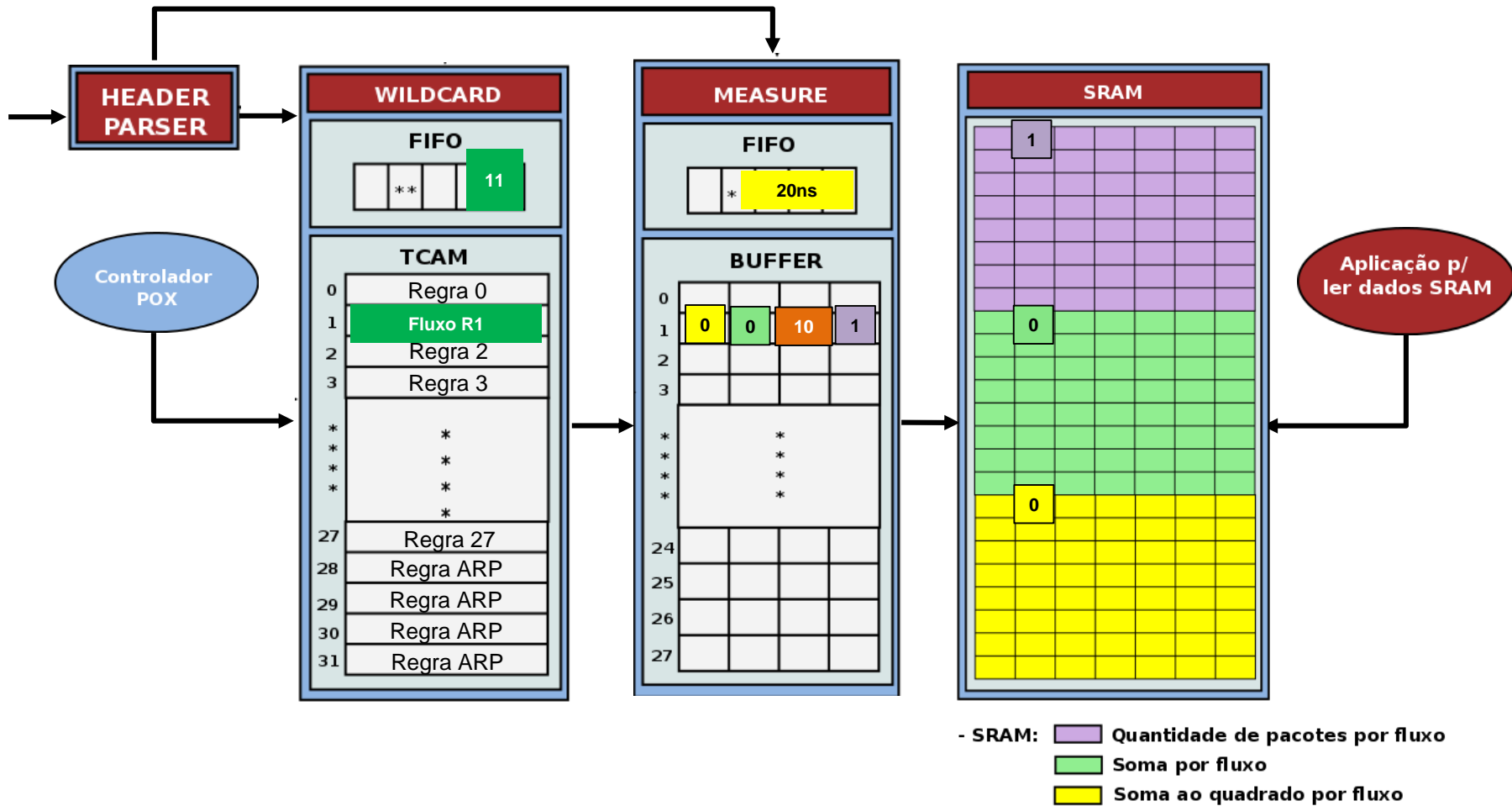
Arquitetura proposta

- Funcionamento



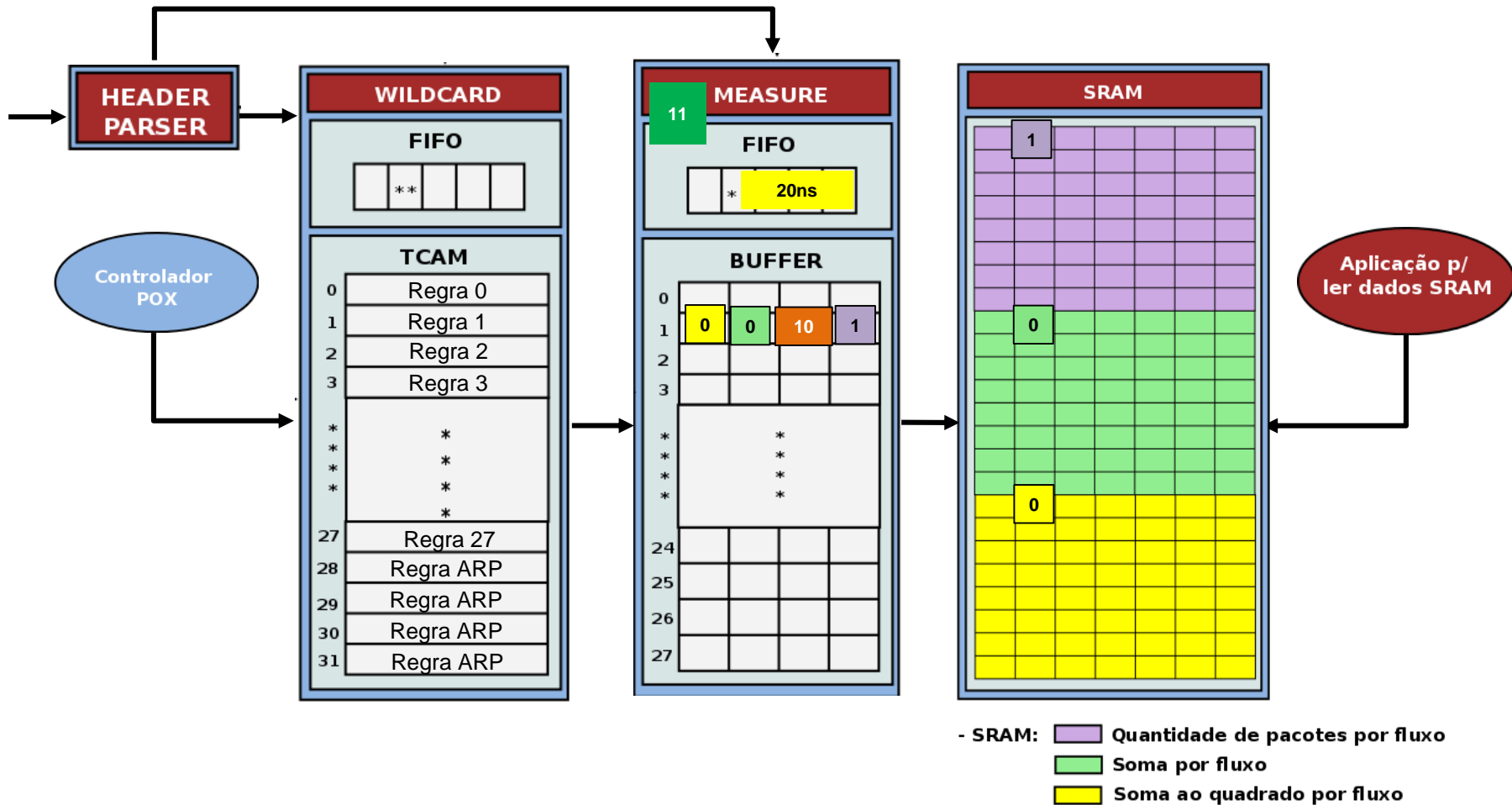
Arquitetura proposta

- Funcionamento



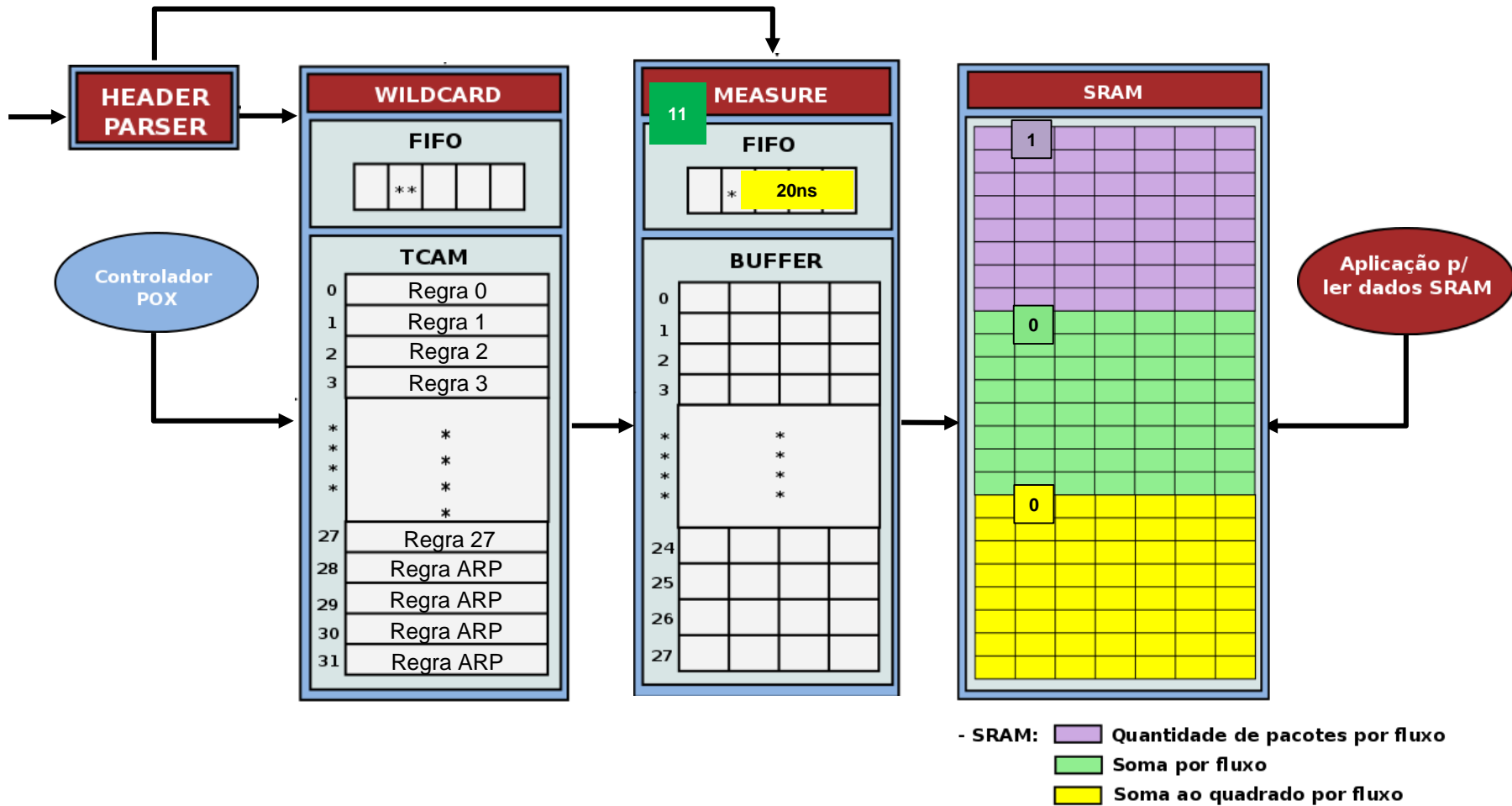
Arquitetura proposta

- Funcionamento



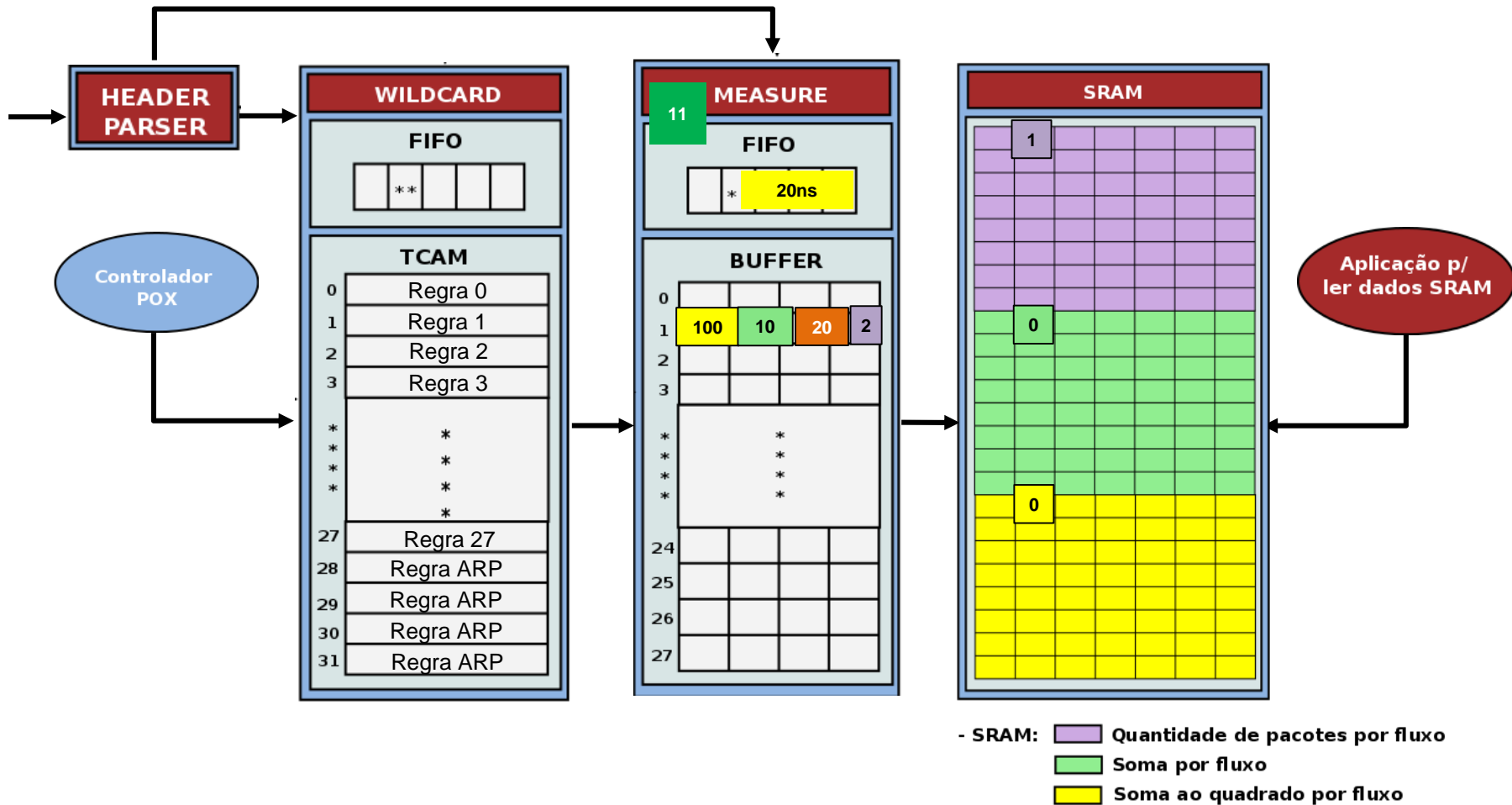
Arquitetura proposta

- Funcionamento



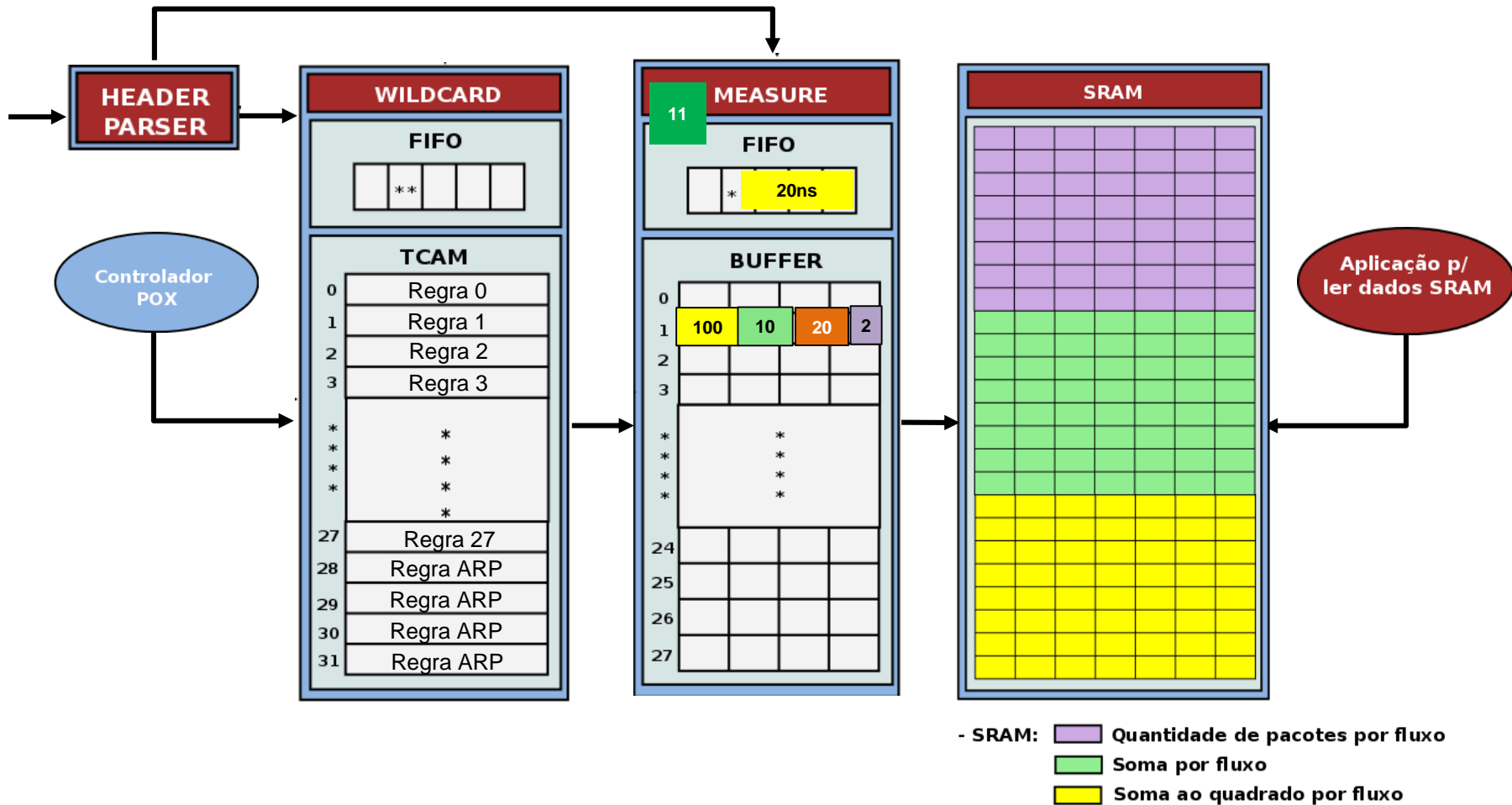
Arquitetura proposta

- Funcionamento



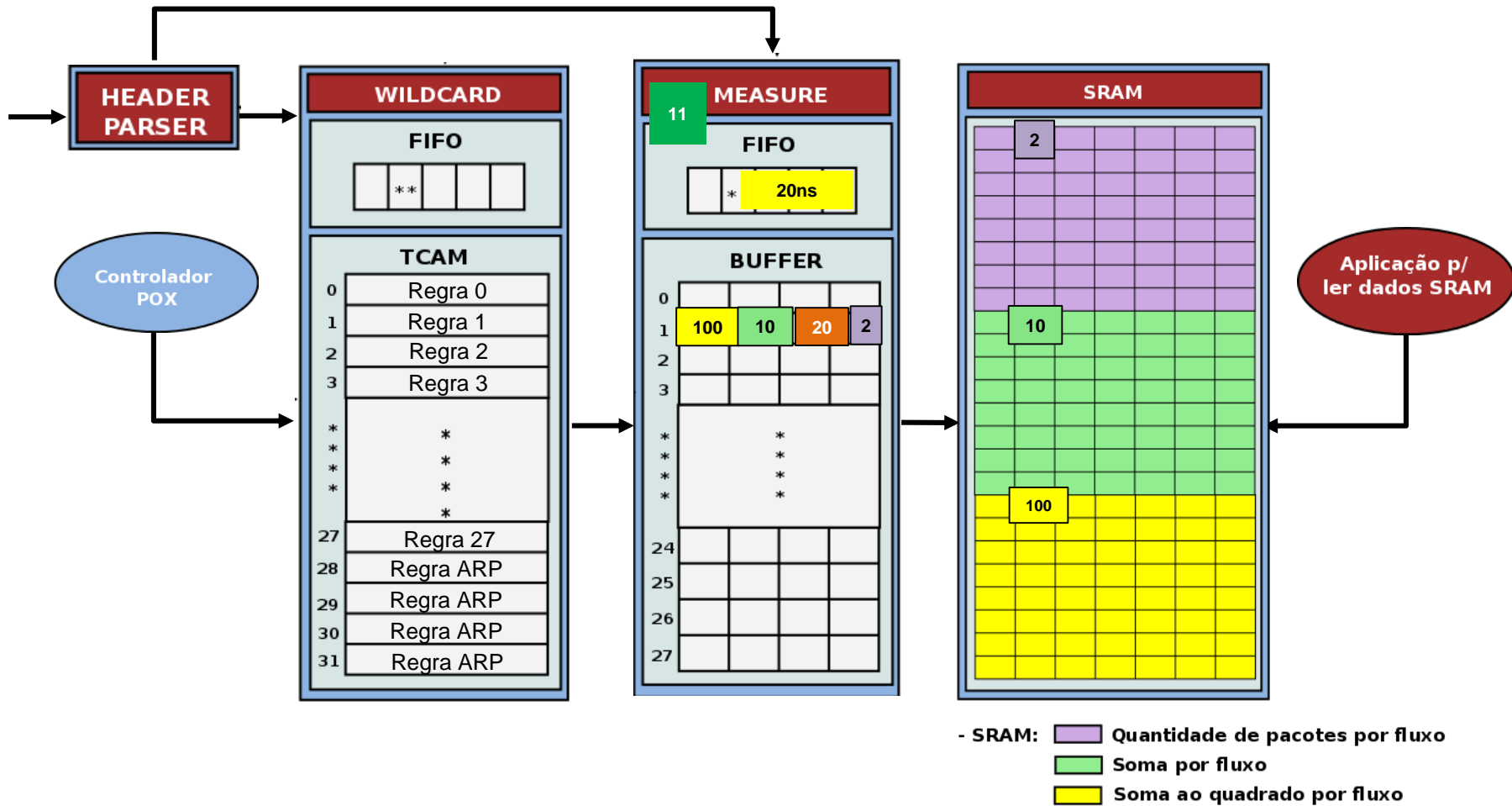
Arquitetura proposta

- Funcionamento



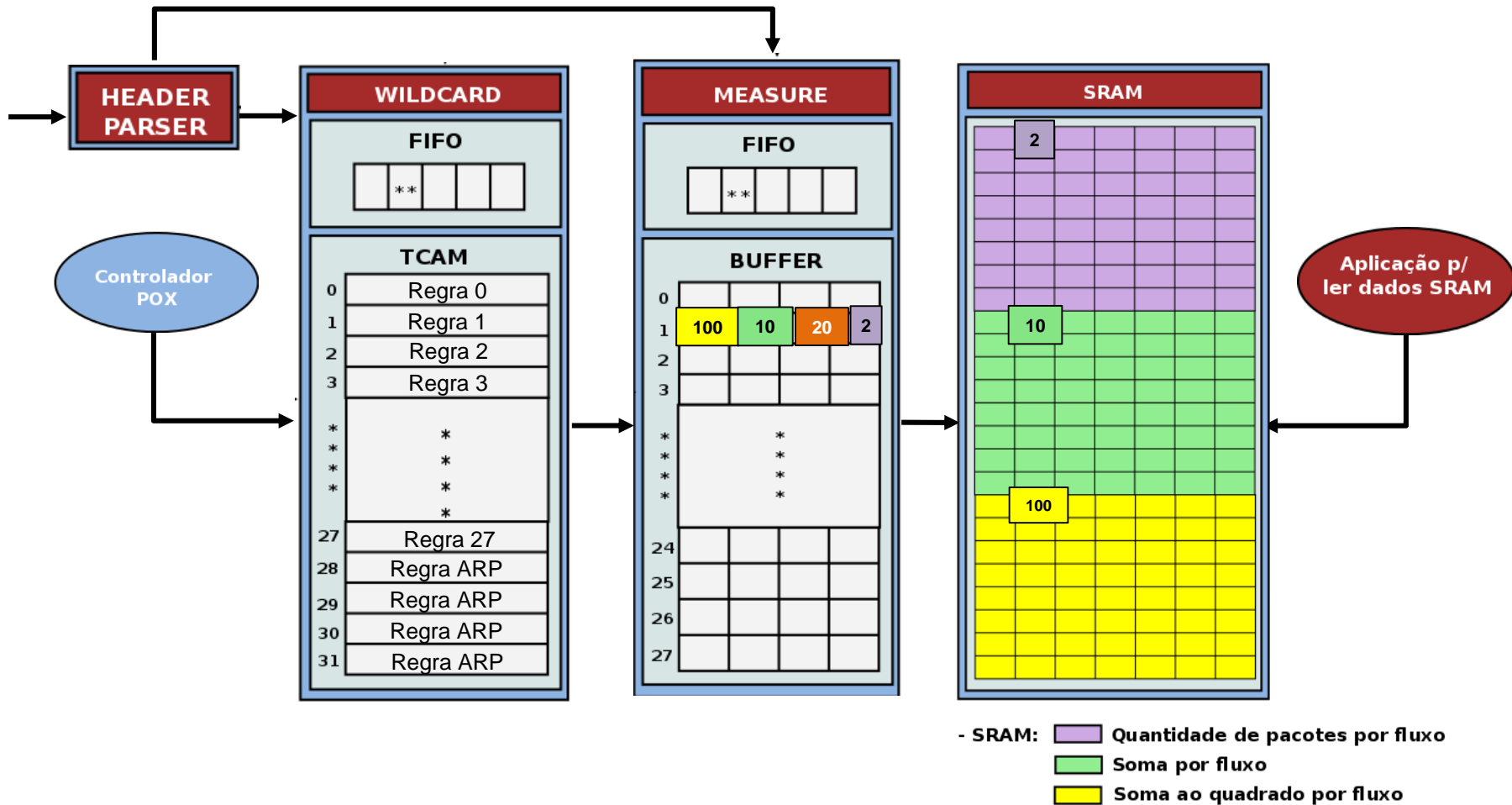
Arquitetura proposta

- Funcionamento



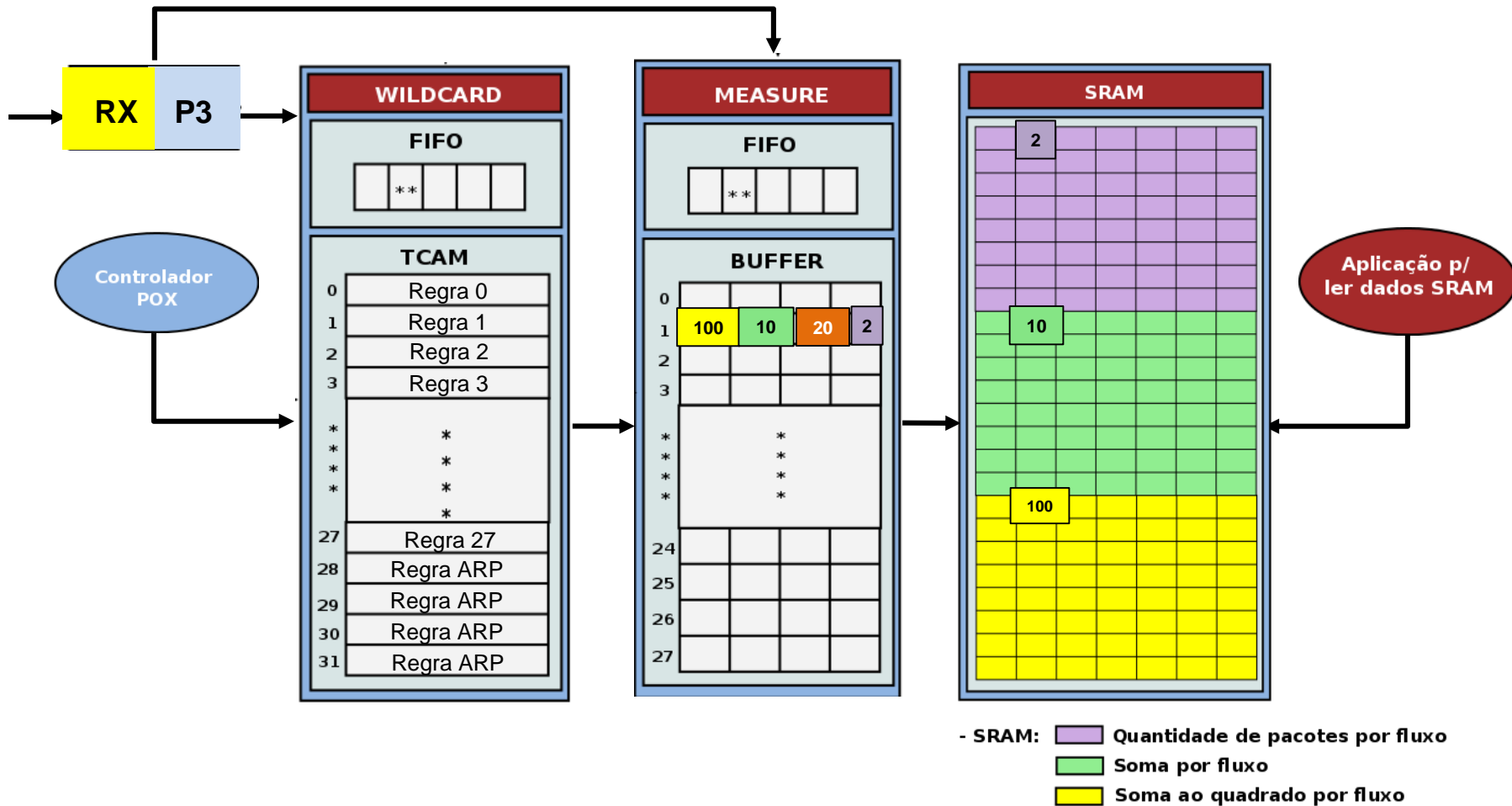
Arquitetura proposta

- Funcionamento



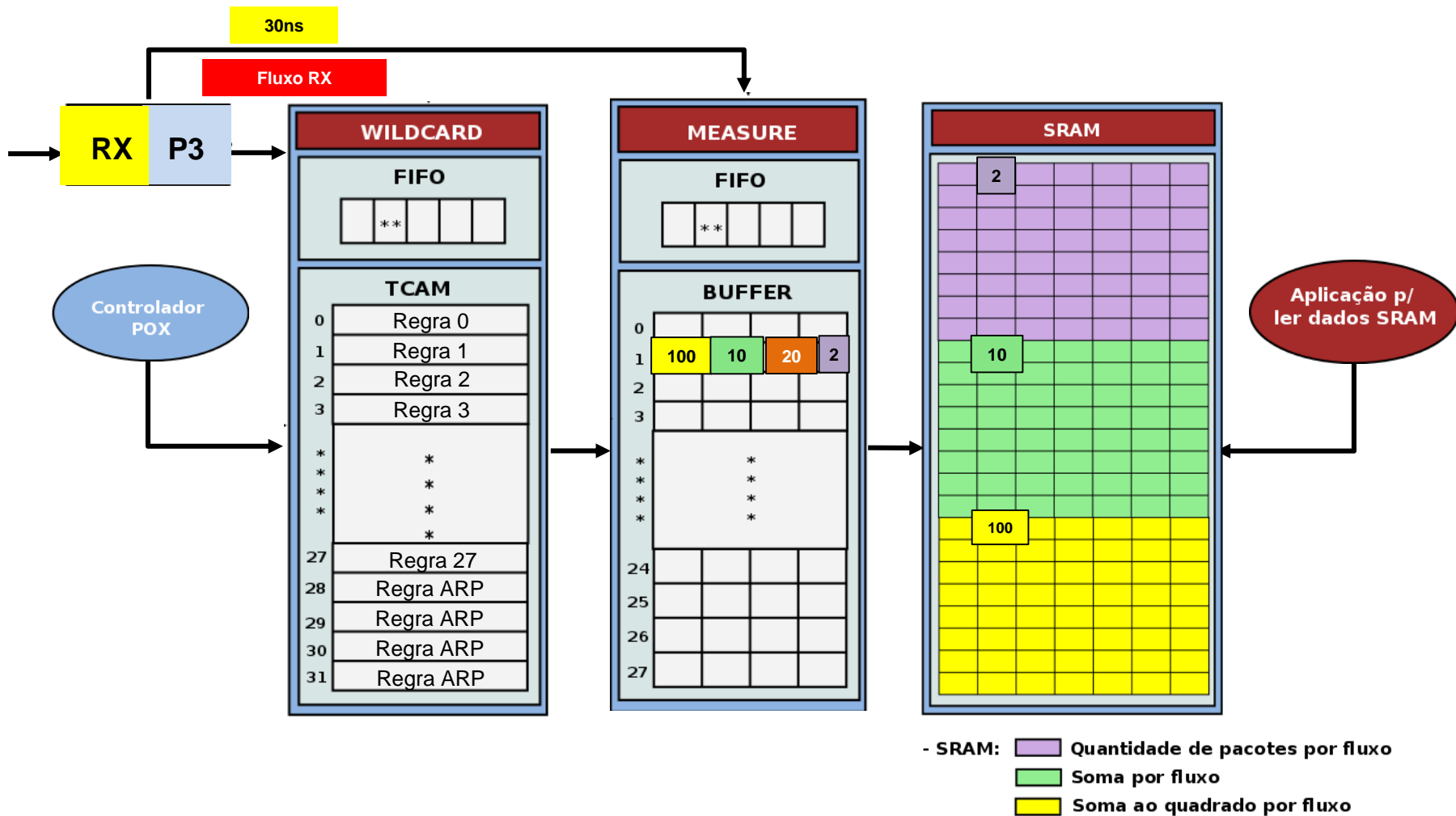
Arquitetura proposta

- Funcionamento



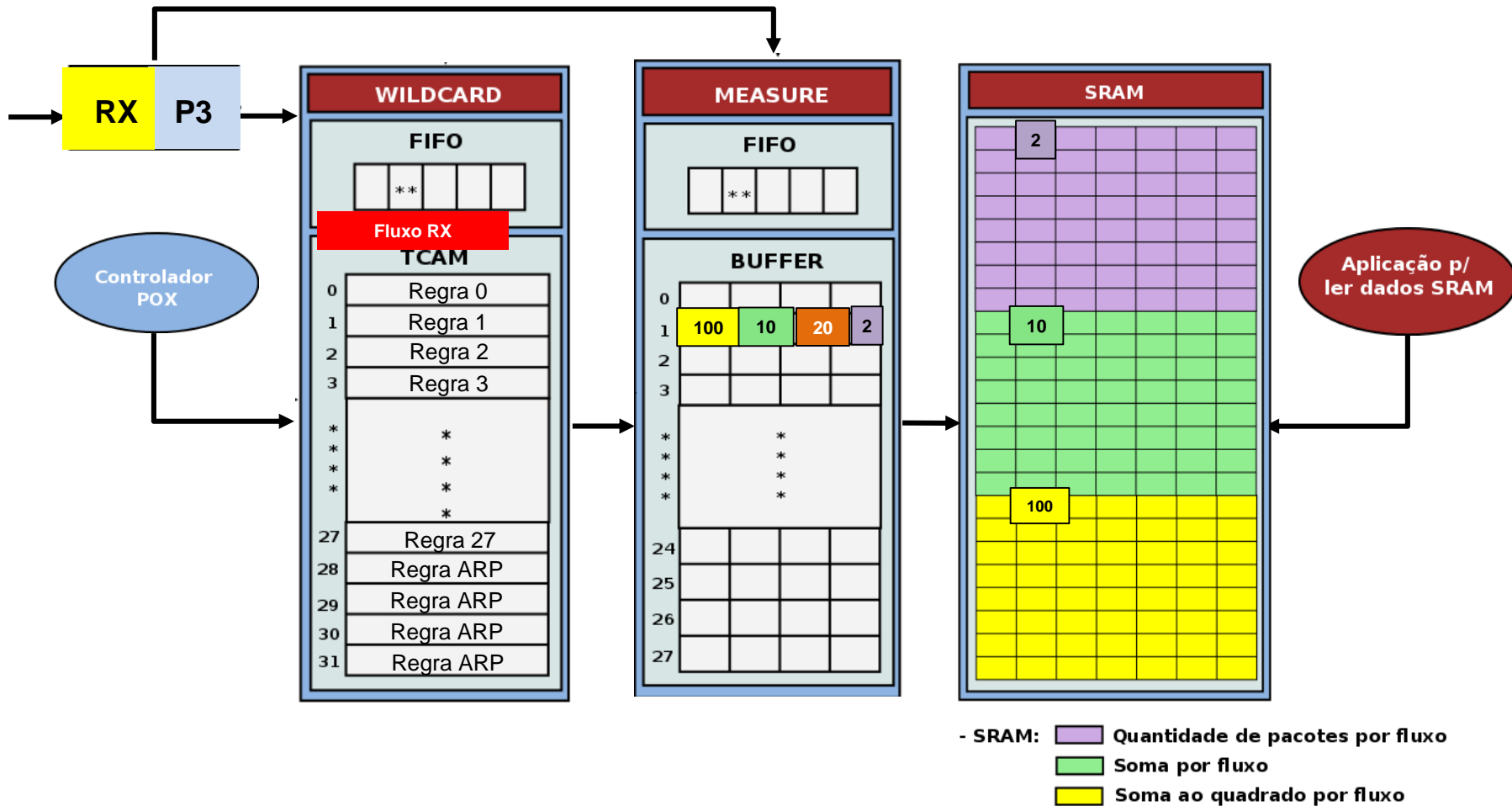
Arquitetura proposta

• Funcionamento



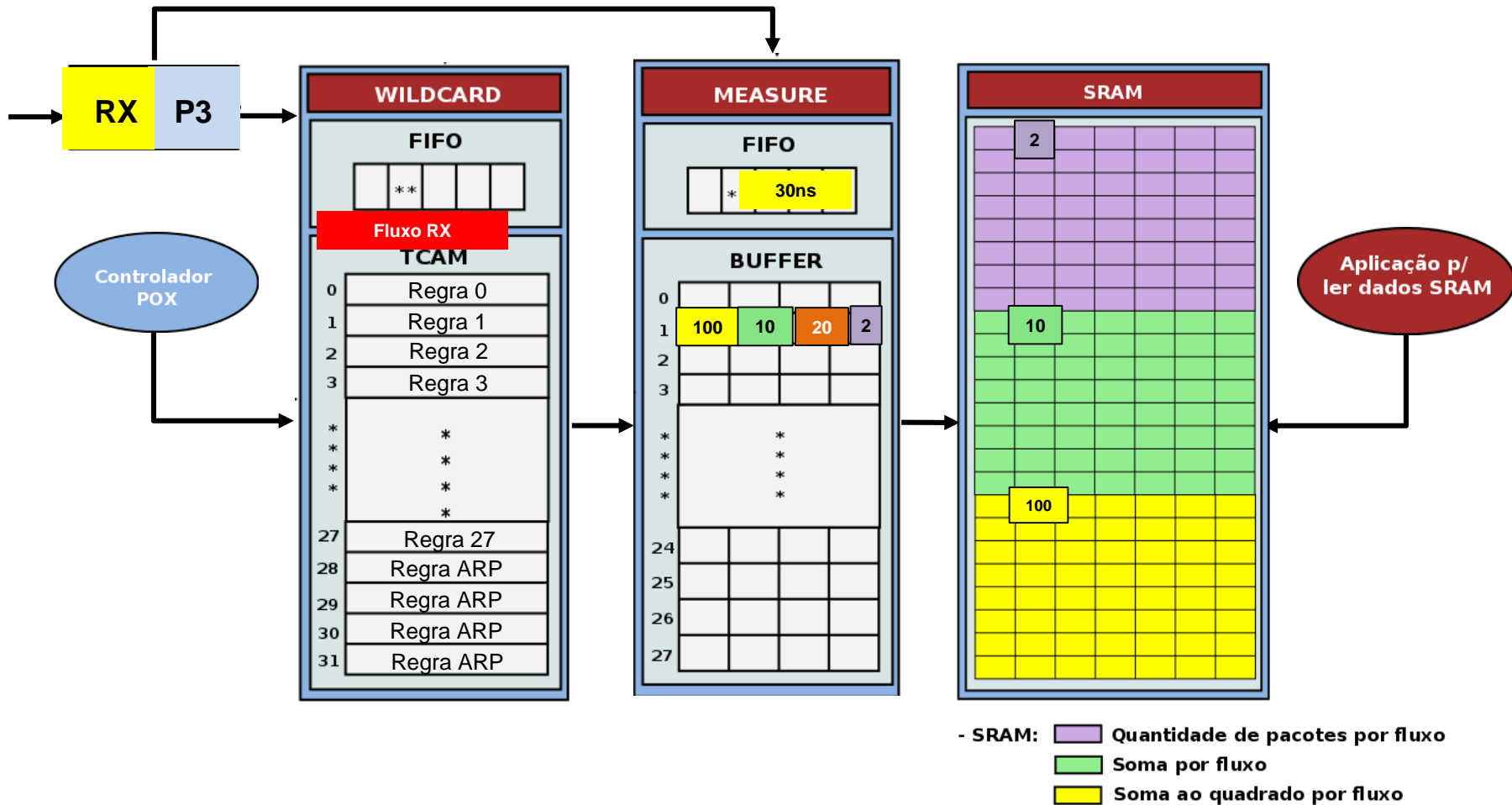
Arquitetura proposta

- Funcionamento



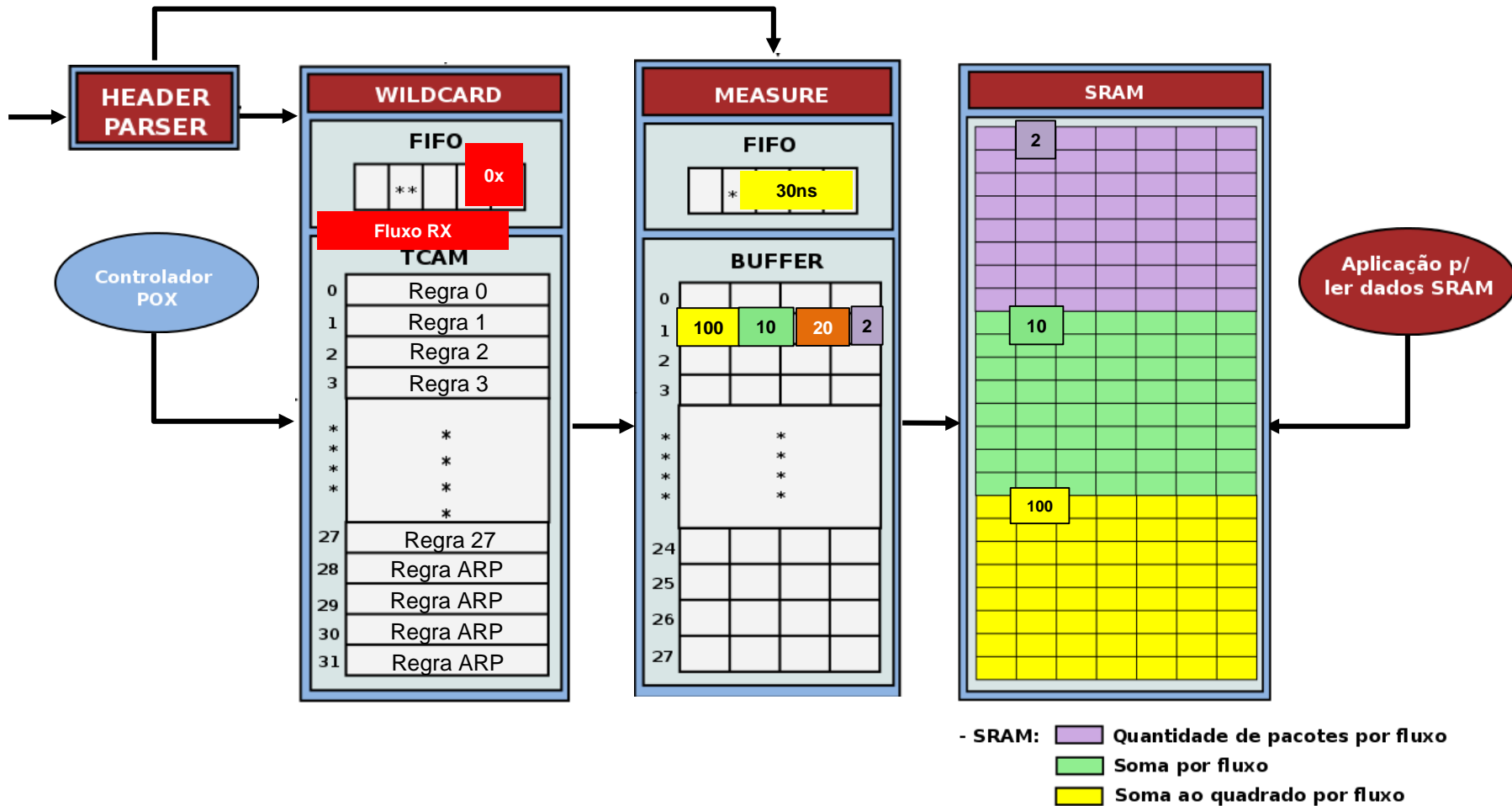
Arquitetura proposta

- Funcionamento



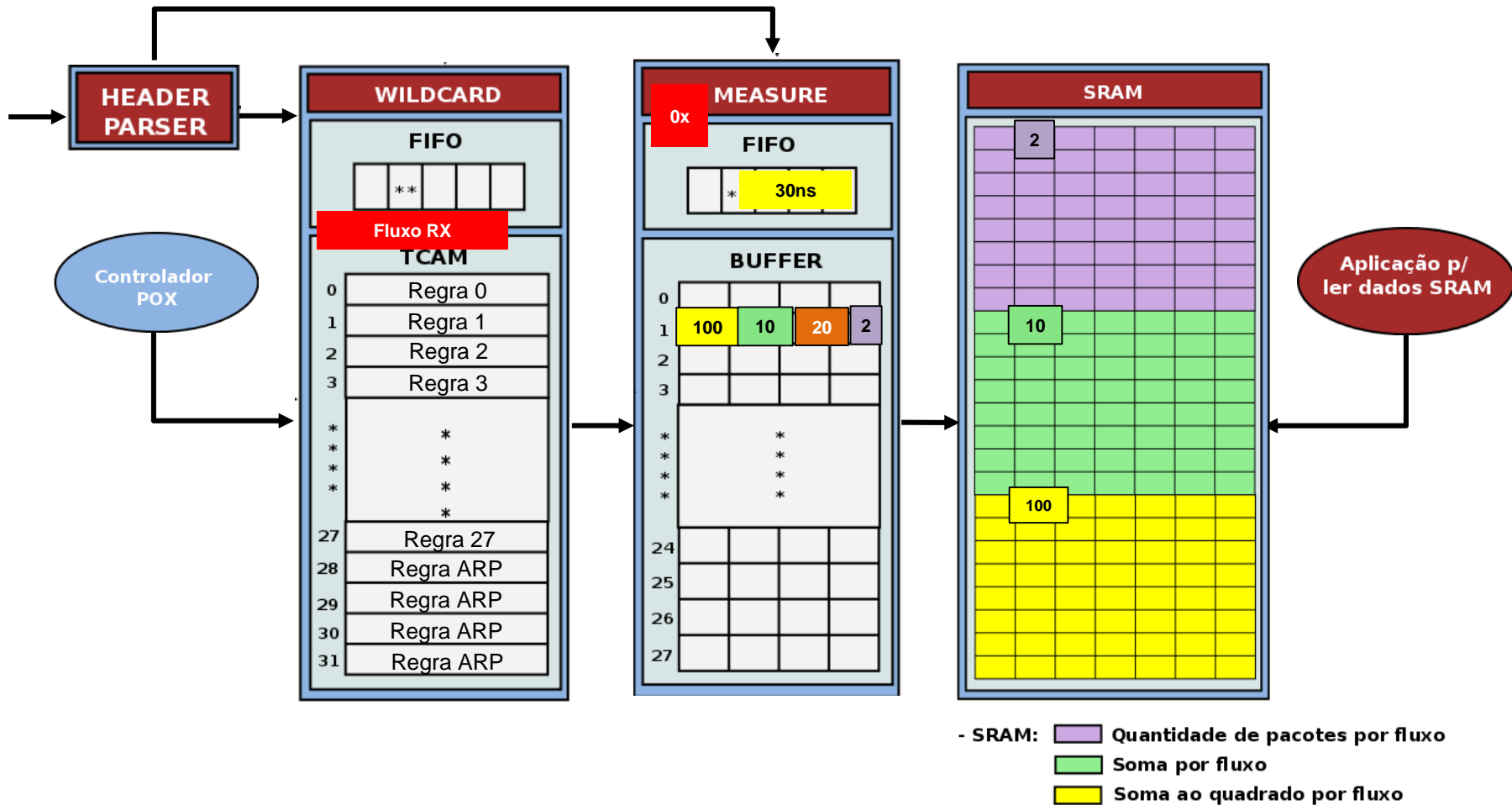
Arquitetura proposta

- Funcionamento



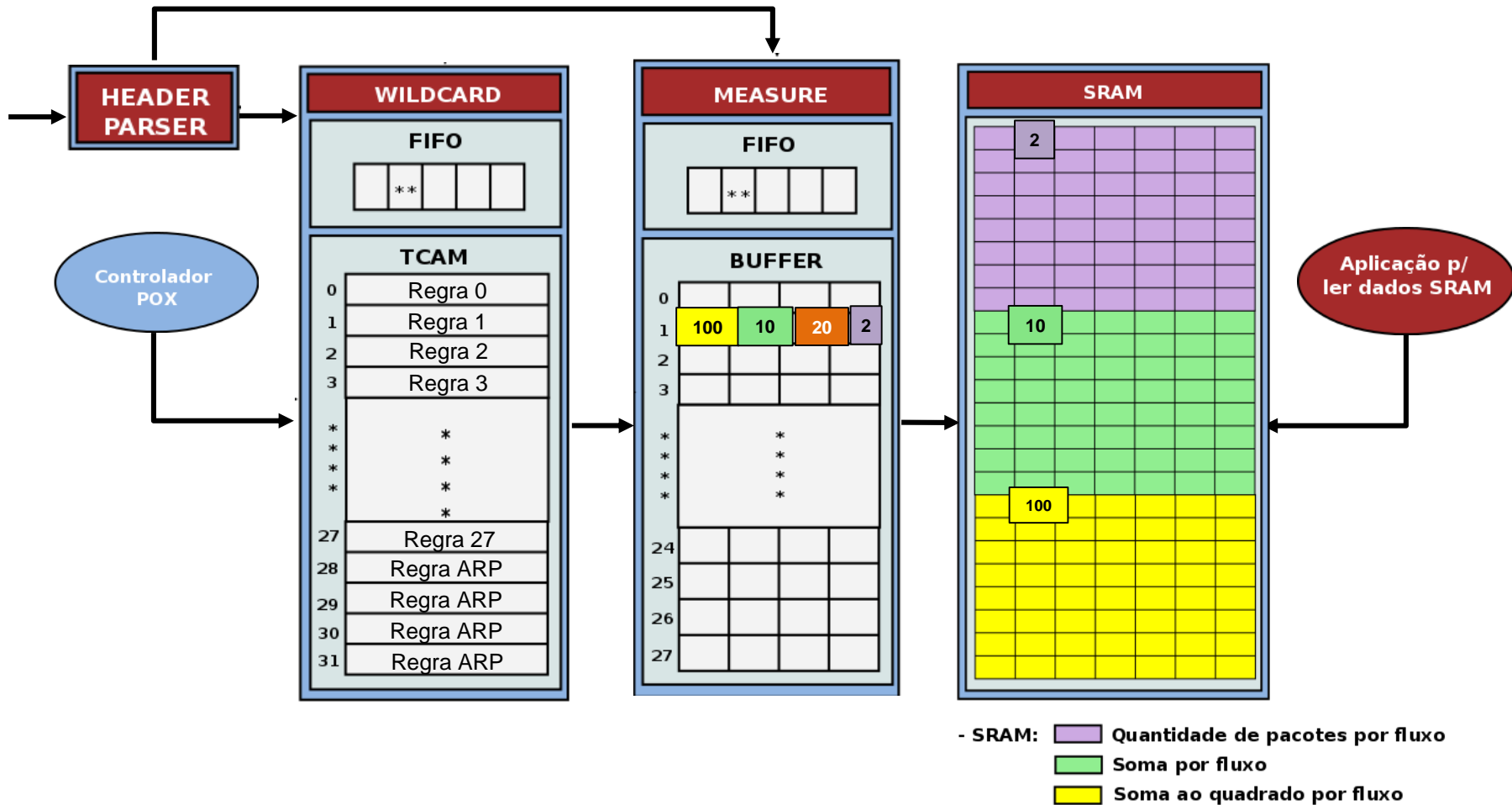
Arquitetura proposta

- Funcionamento



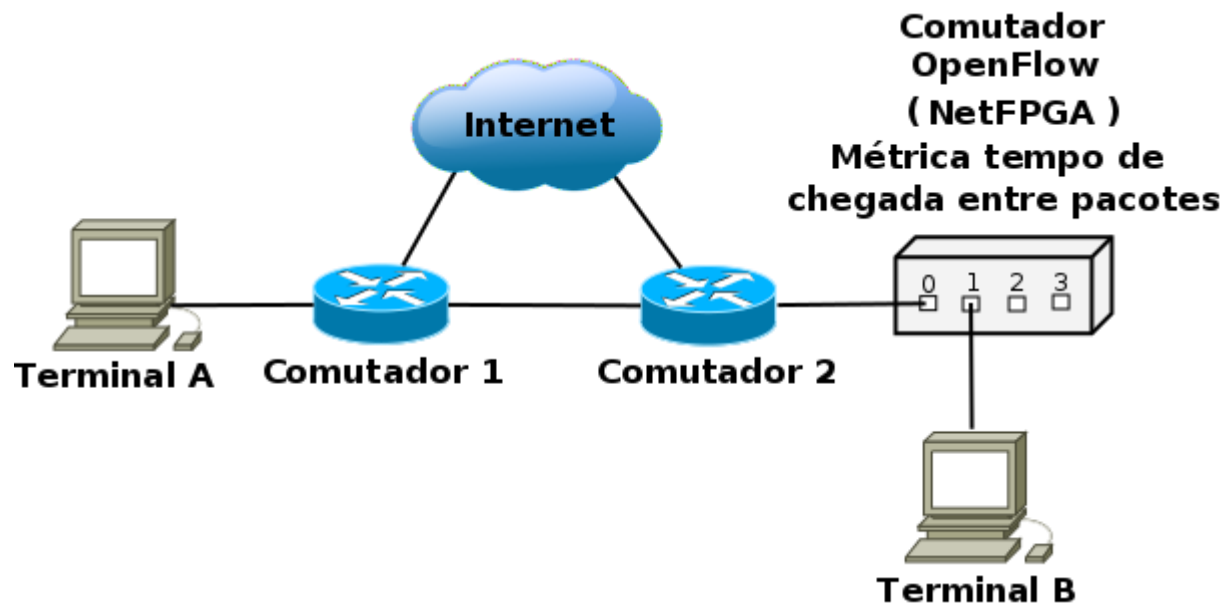
Arquitetura proposta

- Funcionamento



Ambiente de testes

- O Terminal A envia pacotes TCP à uma taxa controlada para o Terminal B;
- Entre o Terminal A e B são inseridos três comutadores. Dois comutadores tradicionais para aumentar o tempo de chegada entre pacotes e o comutador OpenFlow com suporte a métrica tempo de chegada entre pacotes.



Resultados

- **Warm up**

- Todos os experimentos foram compostos de cinquenta iterações, tendo cada iteração 1s;
- São gastos 5s para inicializar o controlador, os processos de monitoramento e transmissão de pacotes e 1s para finalizar todos os processos;
- O tempo total do experimento é 56s ($5s + 50 \times 1s + 1s$);
- Para cada experimento foram descartados 10% das amostras devido ao período de warm up. Foram retirados os 6s no começo e no final do experimento. Cada segundo equivale à 28 amostras.

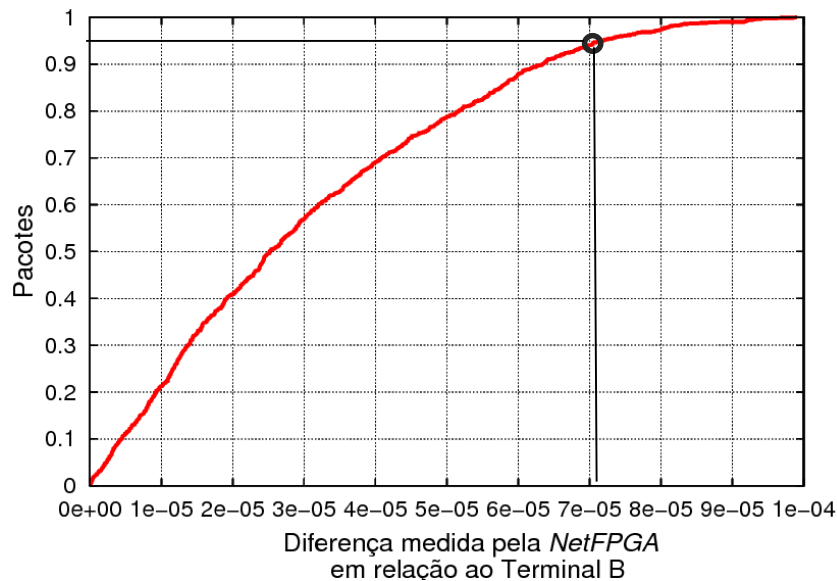
Resultados

- **Foram realizados três tipos de experimentos:**
 - **Experimento com atraso fixo:** Foram enviados pacotes com atrasos pré-definidos de 20/40ms para cada um dos 28 fluxos;
 - **Experimento com atraso aleatório:** Foram enviados pacotes com atrasos aleatórios para cada um dos 28 fluxos. Para gerar os atrasos aleatórios utilizamos três sementes;
 - **Experimento com regra wildcard:** Foram enviados pacotes com atrasos pré-definidos 20/40ms para quatro fluxos. O objetivo deste experimento foi medir todos os quatro fluxos utilizando uma regra agregada inserida no comutador.

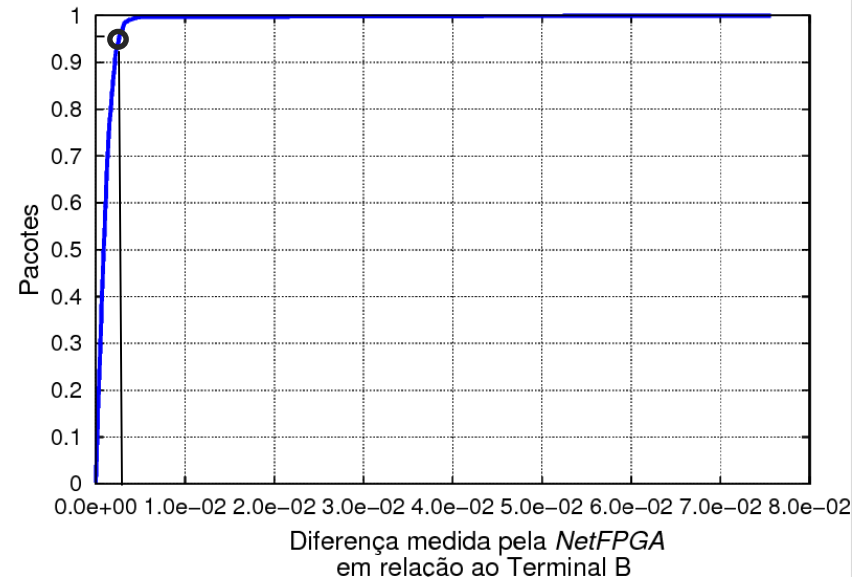
Resultados

- Experimento com atraso fixo

Média	
Porc. de amostras	95%
Diferença relativa	0,0072%
Intervalo de confiança	[2,897:3,166] x 10 ⁻⁵



Variância *	
Porc. de amostras	95%
Diferença relativa	0,26%
Intervalo de confiança	[1,0043:1,3574] x 10 ⁻³



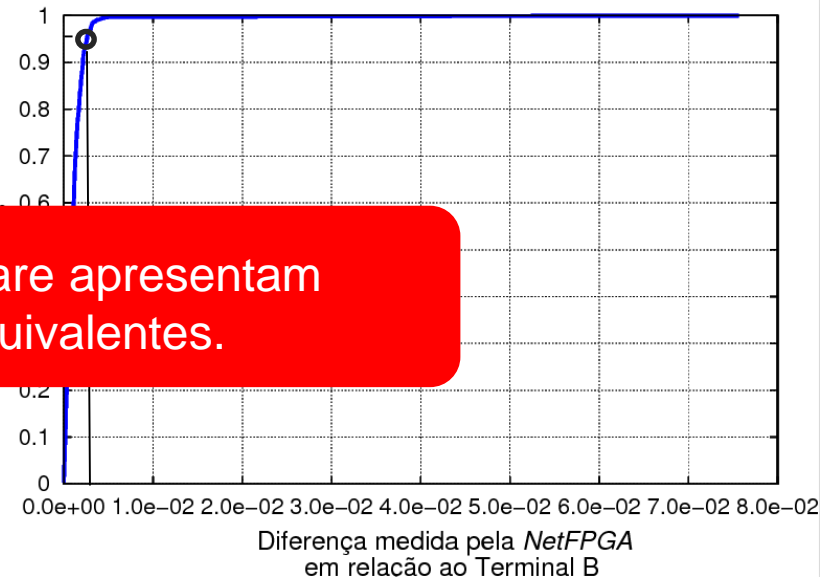
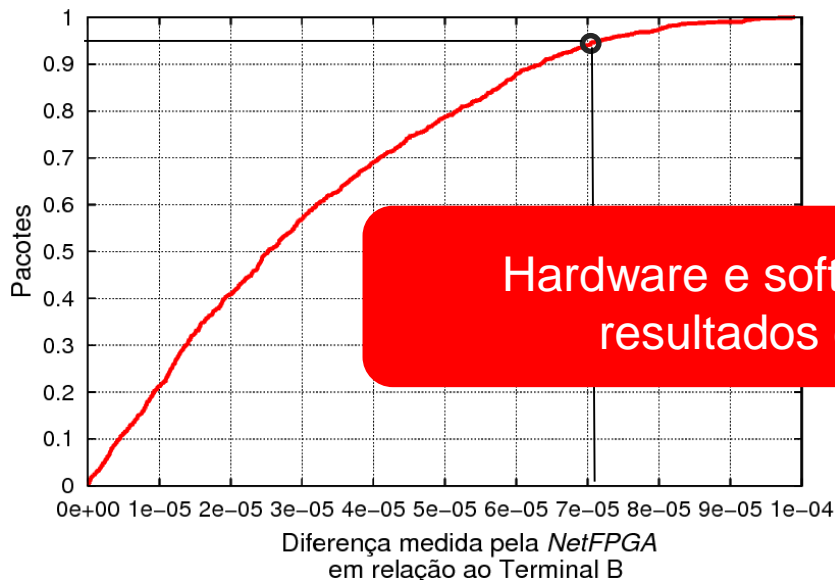
* No gráfico da variância três amostras ficaram entre 3% e 7%, mas as diferenças absolutas foram menores que 10⁻⁴.

Resultados

- Experimento com atraso fixo

Média	
Porc. de amostras	95%
Diferença relativa	0,0072%
Intervalo de confiança	[2,897:3,166] x 10 ⁻⁵

Variância *	
Porc. de amostras	95%
Diferença relativa	0,26%
Intervalo de confiança	[1,0043:1,3574] x 10 ⁻³



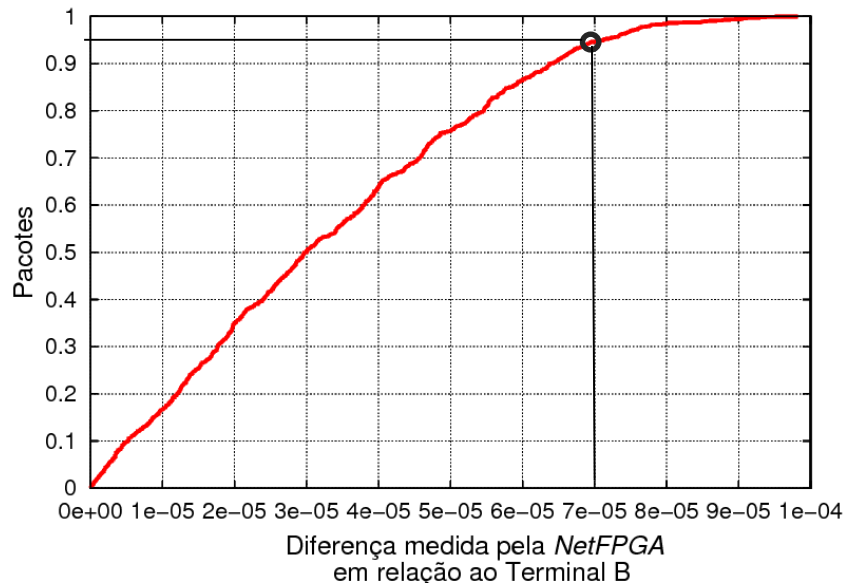
Hardware e software apresentam resultados equivalentes.

* No gráfico da variância três amostras ficaram entre 3% e 7%, mas as diferenças absolutas foram menores que 10⁻⁴.

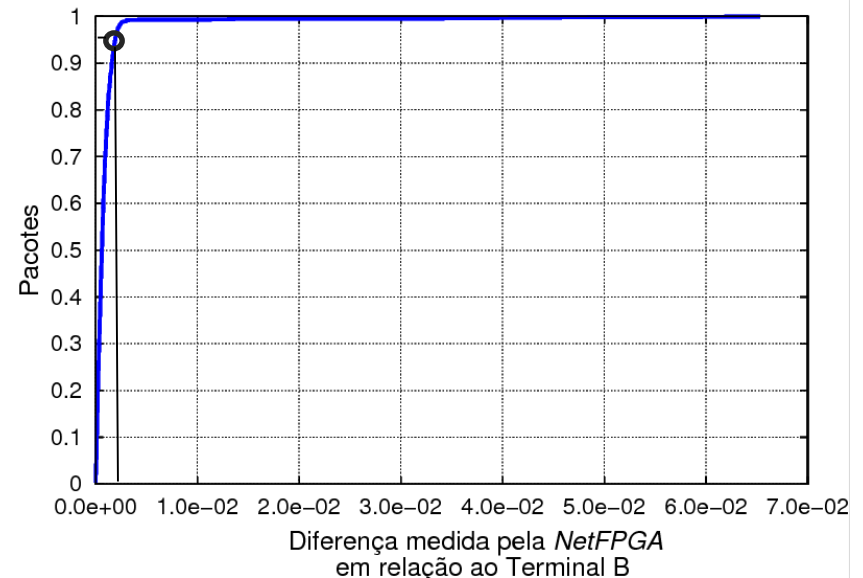
Resultados

- Experimento com atraso aleatório

Média	
Porc. de amostras	95%
Diferença relativa	0,0071%
Intervalo de confiança	[3,169:3,432] x 10 ⁻⁵



Variância *	
Porc. de amostras	95%
Diferença relativa	0,19%
Intervalo de confiança	[7,9642:12,4338] x 10 ⁻⁴



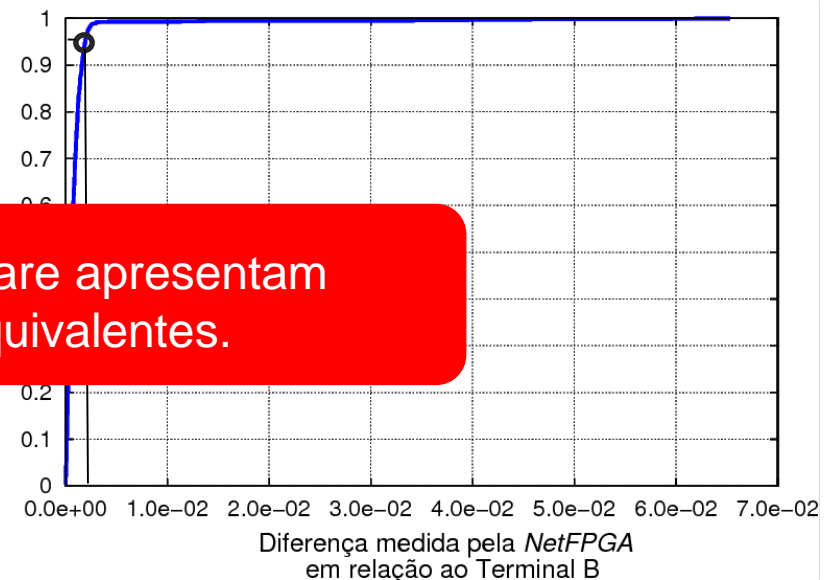
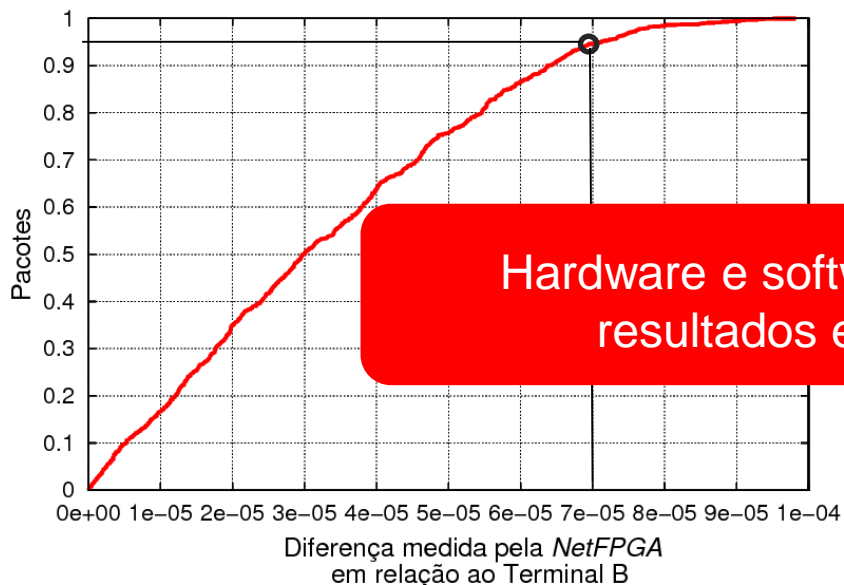
* No gráfico da variância seis amostras ficaram entre 3% e 6%, mas as diferenças absolutas foram menores que 10⁻⁴.

Resultados

- Experimento com atraso aleatório

Média	
Porc. de amostras	95%
Diferença relativa	0,0071%
Intervalo de confiança	[3,169:3,432] x 10 ⁻⁵

Variância *	
Porc. de amostras	95%
Diferença relativa	0,19%
Intervalo de confiança	[7,9642:12,4338] x 10 ⁻⁴



Hardware e software apresentam resultados equivalentes.

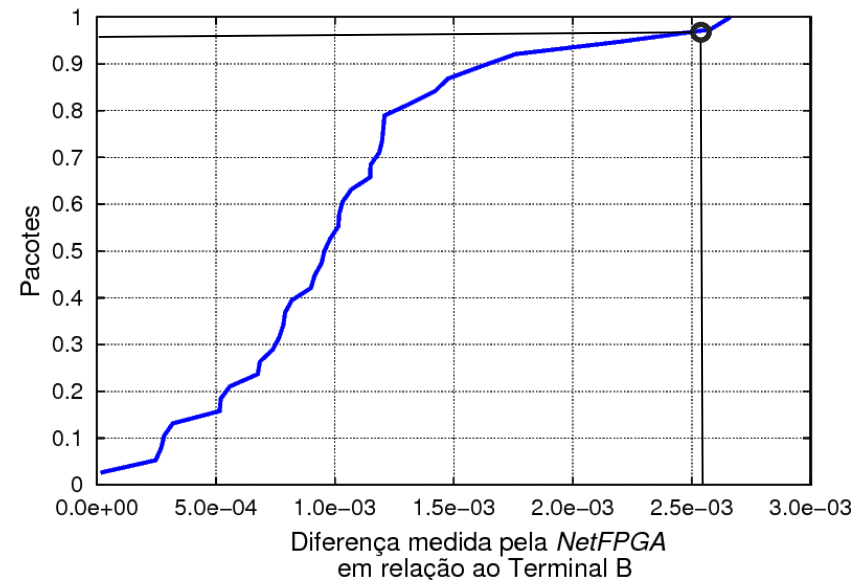
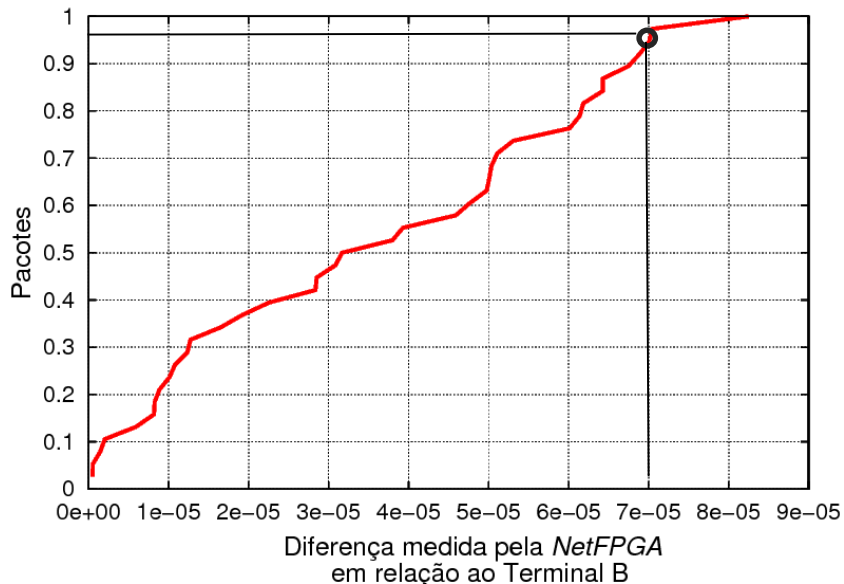
* No gráfico da variância seis amostras ficaram entre 3% e 6%, mas as diferenças absolutas foram menores que 10⁻⁴.

Resultados

- Experimento com regra *wildcard*

Média	
Porc. de amostras	97%
Diferença relativa	0,007%
Intervalo de confiança	[2,788:4,349] x 10 ⁻⁵

Variância	
Porc. de amostras	97%
Diferença relativa	0,257%
Intervalo de confiança	[8,4223:12,072] x 10 ⁻⁴

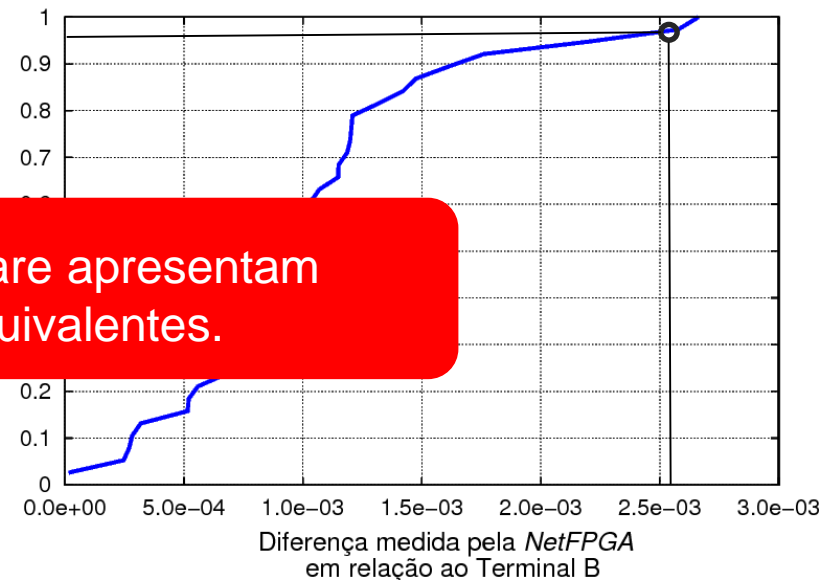
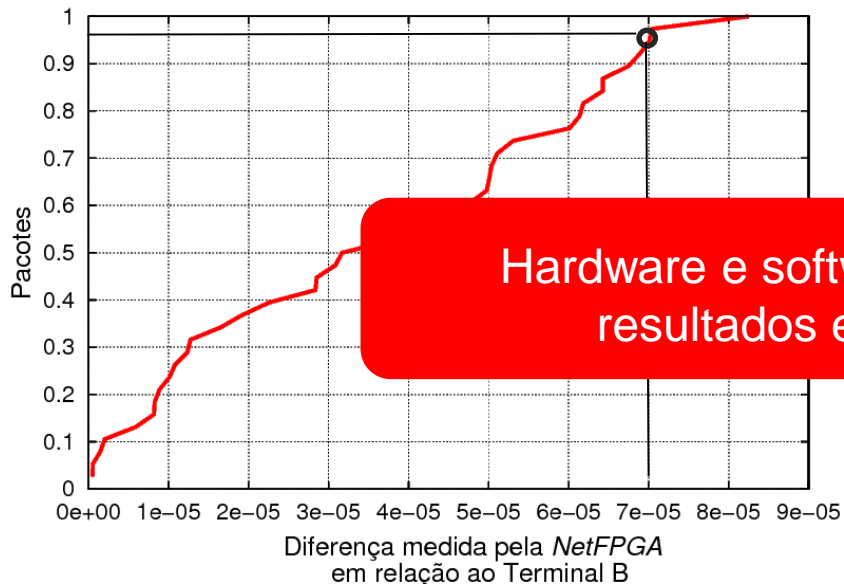


Resultados

- Experimento com regra *wildcard*

Média	
Porc. de amostras	97%
Diferença relativa	0,007%
Intervalo de confiança	[2,788:4,349] x 10 ⁻⁵

Variância	
Porc. de amostras	97%
Diferença relativa	0,257%
Intervalo de confiança	[8,4223:12,072] x 10 ⁻⁴



Hardware e software apresentam resultados equivalentes.

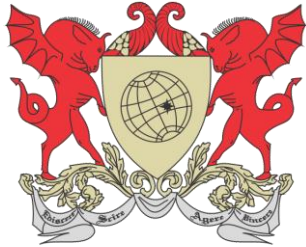
Conclusão

- **Protótipo de medição:**

- Medição do tempo de chegada entre pacotes utilizando o comutador OpenFlow;
- Validação do protótipo;
- Diferença relativa quase nula.

- **Trabalhos futuros:**

- Implementar as métricas tempo de chegada entre pacotes e latência utilizando estruturas hash (Bloom Filters).



Medição de desempenho de rede em hardware

Racyus Delano (UFV)
racyus.pacifico@ufv.br

Pablo Goulart (UFMG); Ítalo Cunha (UFMG); Marcos Vieira (UFMG);

Dorgival Guedes (UFMG); Alex Vieira (UFJF); José Nacif (UFV).

Dúvidas ?
Obrigado pela atenção !

