

CS/CoE 536

Reconfigurable System On Chip Design

Lecture 6 : Content-based Packet Processing in Reconfigurable Hardware

Washington University
Fall 2002

<http://www.arl.wustl.edu/~lockwood/class/cs536/>

Copyright 2002, John W Lockwood
Lockwood@arl.wustl.edu

The Challenge

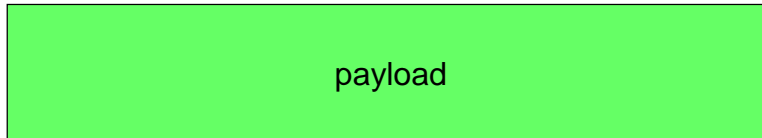
- Standard Firewalls do not protect networks from several types of traffic:
 - SPAM still flows through regular email ports
 - Applications that usually run on blocked ports can be switched to unblocked ports
 - Napster users operate on the web port (80)
 - Telnet traffic (23) can be moved to the ssh port (22)
 - Firewalls need to look at the packet payloads

IP Packet Header and Payload

- Header has fixed values in fixed locations

ver	IHL	service type	total length	
identification			flags	fragment offset
ttl	protocol		header checksum	
source address				
destination address				
options				padding

- Content may start at any byte in the payload



Example Payload

\M'	\A'	\K'	\E'
\ '	\M'	\O'	\N'
\E'	\Y'	\ '	\F'
\A'	\S'	\T'	\C'
\A'	\L'	\L'	\ '
\N'	\O'	\W'	

Content Matching Module

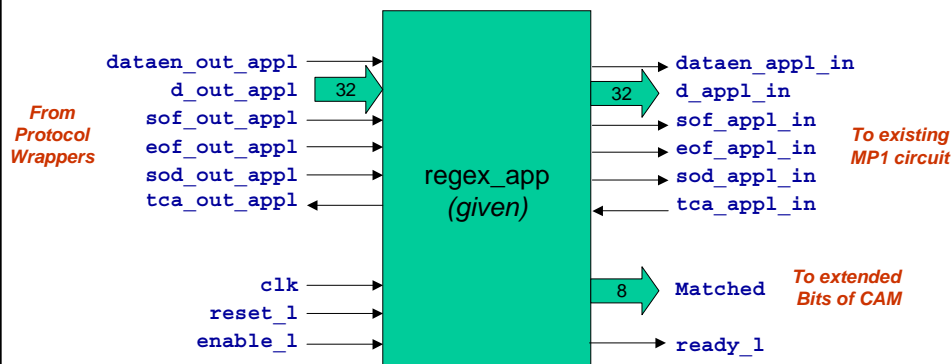
- A module has been developed that scans packets for regular expressions in FPGA hardware and report which content matches.
- You will use an instance of this module to implement Machine Problem 2
- This assignment builds upon the first Machine Problem to allow for filtering both on header and payload data.

Phrase Lists

- General Spam (Bit 0)
 - “amazing”
 - “CALL NOW”
 - “Limited Time Offer”
 - Save Money SPAM (Bit 1)
 - “Consolidate”
 - “full refund”
 - Fast Money SPAM (Bit 2)
 - “MAKE MONEY FAST”
 - “Work from home”
 - Chains and Forwards (Bit 3)
 - “Read this”
 - “FWD”
 - Jokes (Bit 4)
 - “Joke”
 - “walks into bar”
 - Work List (Bit 5)
 - “Homework”
 - “Machine problem”
 - “CS536”
 - “Lockwood”
 - “Washington University”
 - Personal List (Bit 6)
 - “Mom”
 - “Dad”
 - “Call Home”
 - Urgent (Bit 7)
 - “Urgent”
 - “Emergency”
- Note: Underscored letters are case-insensitive*

- The output of the content matching circuit is a vector called the “content match vector” that identifies which regular expressions were found
 - Multiple bits in the vector can be set if the content contains phrases from multiple lists
 - A decision to drop the packet can be made using results from any combination of header and payload matching results.

Content Matching Module



wrapper_module.vhd

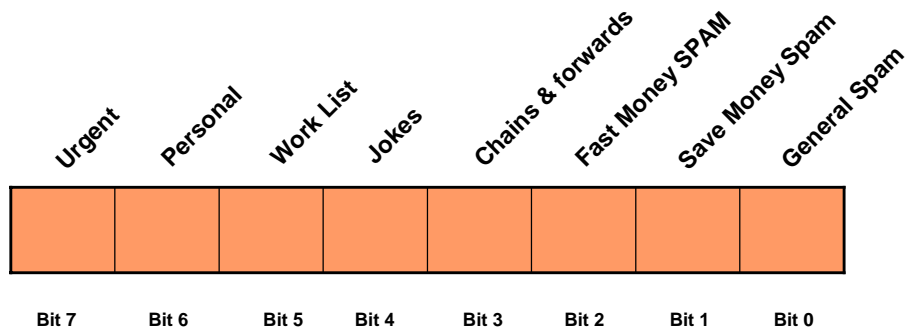
Content matching module (Given)

```
entity regex_app is
  port (clk          : in  std_logic;
        reset_l     : in  std_logic;
        enable_l    : in  std_logic;
        ready_l     : out std_logic; -- AND with other ready signals

        dataen_out_appl : in  std_logic; -- Signals coming from wrapper
        d_out_appl      : in  std_logic_vector(31 downto 0);
        sof_out_appl    : in  std_logic;
        eof_out_appl    : in  std_logic;
        sod_out_appl    : in  std_logic;
        tca_appl_in     : in  std_logic; -- Pass through

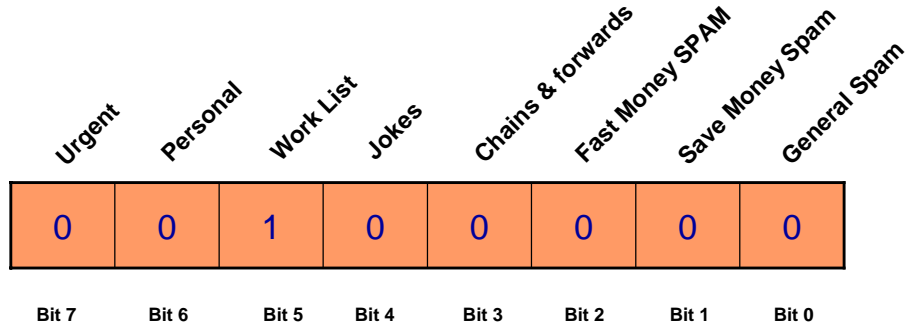
        dataen_appl_in : out std_logic; -- Signals going to MP1 circuit
        d_appl_in       : out std_logic_vector(31 downto 0);
        sof_appl_in     : out std_logic;
        eof_appl_in     : out std_logic;
        sod_appl_in     : out std_logic;
        tca_out_appl    : out std_logic;
        matched         : out std_logic_vector(7 downto 0)); -- 1 on match
end regex_app;
```

Content Match Vector



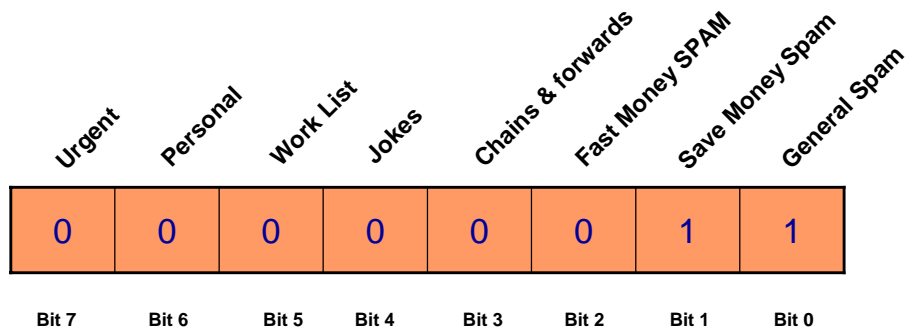
- Bit is set (1) if any phrase in a category is found anywhere in payload
- Bit is clear (0) if none of the phrases in the category appear in the payload

Sample Content Match Vector



- “I can’t wait to work on my CS536 assignment!”

Sample Content Match Vector



- “Consolidate your loans. CALL NOW”

Sample Content Match Vector



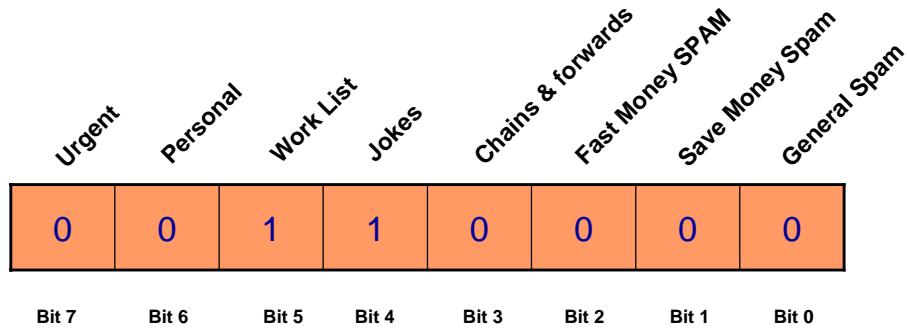
- “Subject: FWD : Read This !!!”

More complex matches



- “Urgent: Call Mom”

More complex matches



- “Lockwood walks into a bar ..”

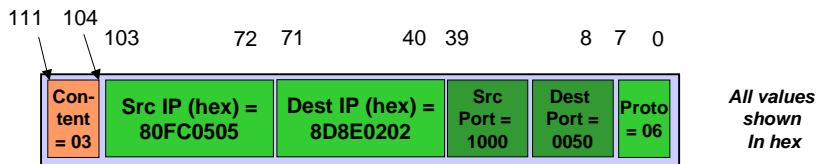
Using the the List Groups with CAMs

- Drop SPAM, pass others
 - Example: General SPAM or Chains and Forwards
 - Drop if Match = 1, CAM Value = 1, CAM Mask = 1
- Keep useful data, drop others
 - Example Filter: Work or Personal
 - Drop if Match = 0, CAM Value = 0, CAM Mask = 1
- Ignore the results of the content filters
 - Urgent or Jokes
 - CAM Mask = 0

Packet matching with Content Addressable Memory (CAM)

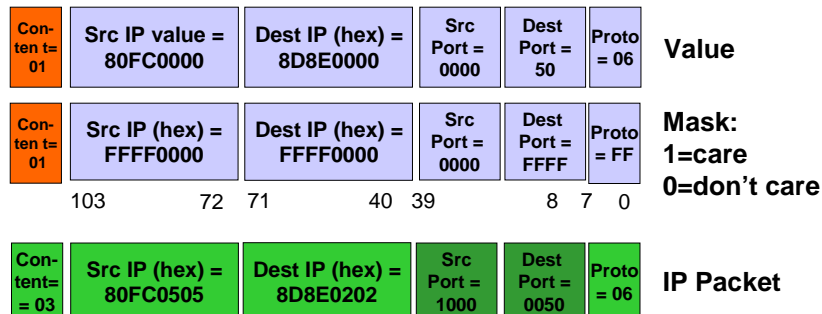
- **Sample Packet:**

- Source Address = 128.252.5.5 (*dotted.decimal*)
- Destination Address = 141.142.2.2 (*dotted.decimal*)
- Source Port = 4096 (*decimal*)
- Destination Port = 80 (*decimal*)
- Protocol = TCP (6)
- Payload = "Consolidate your loans. CALL NOW"
 - Payload Lists = { General SPAM (0), Save Money SPAM (1) }
 - Content Vector = "00000011" (*binary*) = x"03" (*hex*)



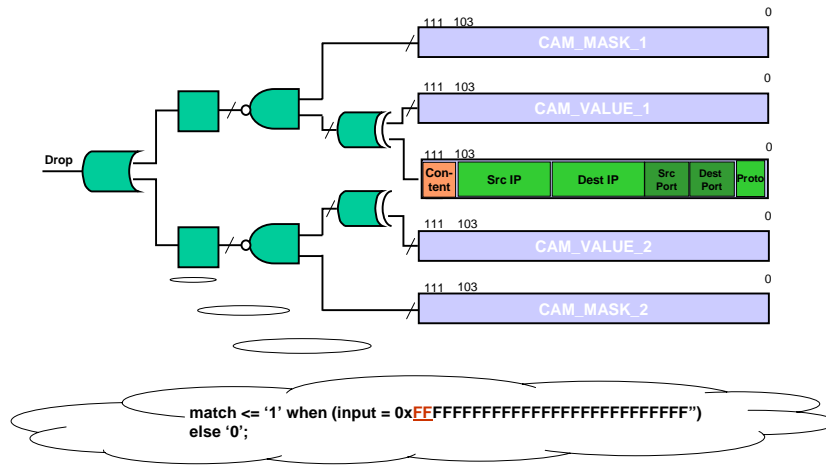
Sample Filter

- Source Address = 128.252.0.0 / 16
- Destination Address = 141.142.0.0 / 16
- Source Port = Don't Care
- Destination Port = 80
- Protocol = TCP (6)
- Payload includes general SPAM (List 0)



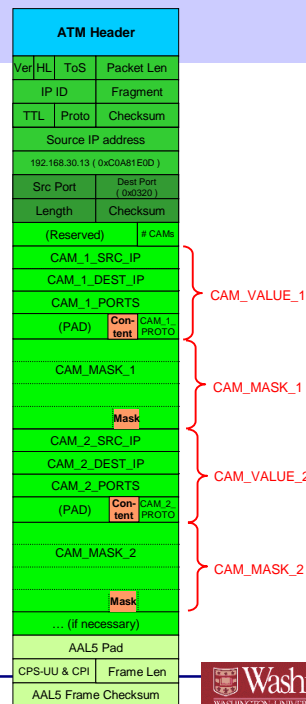
DROP the packet : It matches the filter

MP2 Packet classification hardware



CAM Update Datagram

- Additional fields for payload vector added to control packet, as shown
 - Content vector value1
 - Content vector mask1
 - Content vector value1
 - Content vector mask1



Homework: Creating Sample Inputs

- Generate a full text message (with normal sentences) that has:
 - Match Vector [7 downto 0] = “0000011”b
 - Match Vector [7 downto 0] = “00011000”b
 - Match Vector [7 downto 0] = “00110000”b
 - Match Vector [7 downto 0] = “01100000”b
- Determine the CAM setting to only preserve data in email (port 25) that contains work or personal data except for messages that originated from the network 192.168.30.0/255.255.255.0
- Determine the CAM settings to drop data in email (port 25) going to any machine at WashU from any machine in the Internet that contains all of: *General Spam, Save Money Spam, and Fast Money Spam.*
- What is the minimum number of CAM entries needed to match data above that contains *General Spam, Save Money Spam, or Fast Money Spam?*

Machine Problem 2: Implement the Firewall

- Instantiate and connect the Content Matching Module to the existing structure of MP1.
 - You will need to edit the file `wrapper_module.vhd`
 - Note: you are given the component entity for `regex_app` in MP2
 - Note: You are given an EDIF version of the content matching module called: `regex_app.edn`
- Extend CAM width by 8 bits to 112 bits
 - Bits [111 downto 104] used for payload vector
- Update state machine to accept values for the longer CAMs from control packets
- Generate your own test traffic to exercise the filters on the previous page. Submit your `INPUT_CELLS.TBP` file

Specifying Ranges in TCAMs

- In the best case, TCAMs can match multiple packets with one entry
 - Example: Match Address range of 141.142.2.16 - 141.142.2.31
 - In binary, Addresses range from appear as:
 - 10001101.10001110.00000010.00010000 --
 - 10001101.10001110.00000010.00011111
 - Note that all of the 4 Least Significant Bits change
 - 10001101.10001110.00000010.0001xxxx
 - Thus, one CAM can be set with:
 - Value = 0x8D8E0210
 - Mask = 0xFFFFFFFF

Specifying Ranges in TCAMs (continued)

- TCAMs usually require multiple entries to match a range
 - Example: Match Address range of 141.142.2.16 - 141.142.2.255
 - In binary, Addresses range from appear as:
 - 10001101.10001110.00000010.00010000 --
 - 10001101.10001110.00000010.11111111. Thus
 - Note that matching patterns include:
 - 10001101.10001110.00000010.0001xxxx
 - 10001101.10001110.00000010.001xxxxx
 - 10001101.10001110.00000010.01xxxxxx
 - 10001101.10001110.00000010.1xxxxxxx
 - Thus, four TCAM entries are needed

Specifying Multiple Entries in TCAMs

- In the worst case, individual TCAM entries are needed to match individual filters.
 - Example: Match Addresses:
 - 141.142.2.1,
 - 141.142.2.2
 - 142.142.2.4
 - 141.142.2.8
 - In binary, Addresses range from appear as:
 - 10001101.10001110.00000010.00000001
 - 10001101.10001110.00000010.00000010
 - 10001101.10001110.00000010.00000100
 - 10001101.10001110.00000010.00001000
 - Thus, four TCAM entries are needed