



AN1340

APPLICATION NOTE

Longest Prefix Match Using the M7010 and M7020 Search Engines

INTRODUCTION

This note describes how to perform longest prefix match using the M7010 and M7020 Search Engines. We use the example of IPv4 address lookups to illustrate this application.

Format of IP addresses

IPv4 addresses are 32-bit binary numbers commonly represented in dotted decimal notation. A 3-digit decimal number between 0 and 255 represents each group of 8 bits. The 3-digit groups are separated by a dot (e.g., xxx.xxx.xxx.xxx). The examples described here deal with IPv4 addresses only. A similar approach can be used for other applications that use the longest prefix match concept.

IP prefixes and masks

Routers interpret the IPv4 address as being composed of a network address and a host address within the network. The IP prefix refers to the network address portion of the IP address. For example, a machine may have an IP address of 192.168.104.100. To understand IP prefixes it is more appropriate to use the binary representation, which would be 11000000.10101000.01101000.01100100. Let's say that the first 18 bits represent the network address, i.e. network address = 11000000.10101000.01. Then the remaining 14 bits (011000.01100100) represent the host address within the network. A router would use a prefix mask consisting of 18 ones followed by 14 zeros (11111111.11111111.11000000.00000000) to indicate that the network address (prefix) is the first 18 bits of the IP address. The length of the prefix mask is not fixed but can vary from network subnet to subnet.

ROUTERS AND IP LOOKUPS

One of the functions of a router is to maintain a routing table so that when an IP address is presented as a lookup key it retrieves associated data that would include among other things the route port. Routing table entries will contain 32 bit IP addresses but the network address portions can be of varying lengths, represented by the leading ones in the prefix mask.

The Longest Prefix Match Concept

Applications, which require the longest prefix match, do not consider all 32 IP address bits for the lookup, but only the prefix as indicated by the prefix mask. The addresses and prefixes can be stored in the M7010 and M7020 as pairs in the data and the mask arrays.

A "1" in the prefix mask specifies the bit position where a match is enabled for a lookup operation. It is possible that when the prefix mask is applied, more than one entry in the table matches a given address. This can be seen in the following set of 4 entries with addresses A, B, C, D and their respective masks (see Table 1).

AN1340 - APPLICATION NOTE

Table 1. Set of 4 Entries with Addresses A, B, C, and D

11111111.11111111.11111111.00000000 - prefix mask for A 11000000.10101000.01101000.01100100 - IP address A
11111111.11111111.11111110.00000000 - prefix mask for B 11000000.10101000.01101001.01100100 - IP address B
11111111.11111111.11111100.00000000 - prefix mask for C 11000000.10101000.01101010.01100100 - IP address C
11111111.11111111.11111111.10000000 - prefix mask for D 11000000.10101000.01101000.11100100 - IP address D
11000000.10101000.01101000.01111111 - IP address being looked up L

When IP address L is presented as a lookup key, entry A, B, C will all match L as a result of the address mask. Entry D would not match due to the mismatch in bit 7. Longest prefix match specifies that the lookup operation should resolve multiple matches by reporting only matched entry which has the longest prefix mask. Therefore a longest prefix match would report a match for entry A.

The M7010 and M7020 are ideally suited to implementation of longest prefix match operations. The data arrays in the M7010 and M7020 are used to store entry values. The mask arrays in the devices are used to store the prefix mask. The prefix masking operation is built in and occurs automatically in the M7010 and M7020. The table management software sorts entries in order of decreasing prefix length, so that the entry with the longest prefix has the lowest address. The lowest address location has the highest priority, and therefore in the above example IP address A has a lower address and therefore a higher priority when compared to IP Addresses B,C, and D. The M7010 and M7020 are designed so that when multiple matches occur they output the index from the lowest address location. This way, the longest prefix match is easily implemented.

The M7010 and M7020 are both flexible and fast. They will work with any length of subnet mask, up to the word width, and are capable of generating a longest prefix match result at the maximum data rate.

In summary, the steps to building table for longest prefix match are as follows:

1. Write the entry values so that they are sorted in order of decreasing prefix length.
2. Write the corresponding mask values.
3. Set the Global Mask Register to all "1's" for writes and searches so the global mask does not modify any data bits on a write and enables a search for all 32 bit positions.

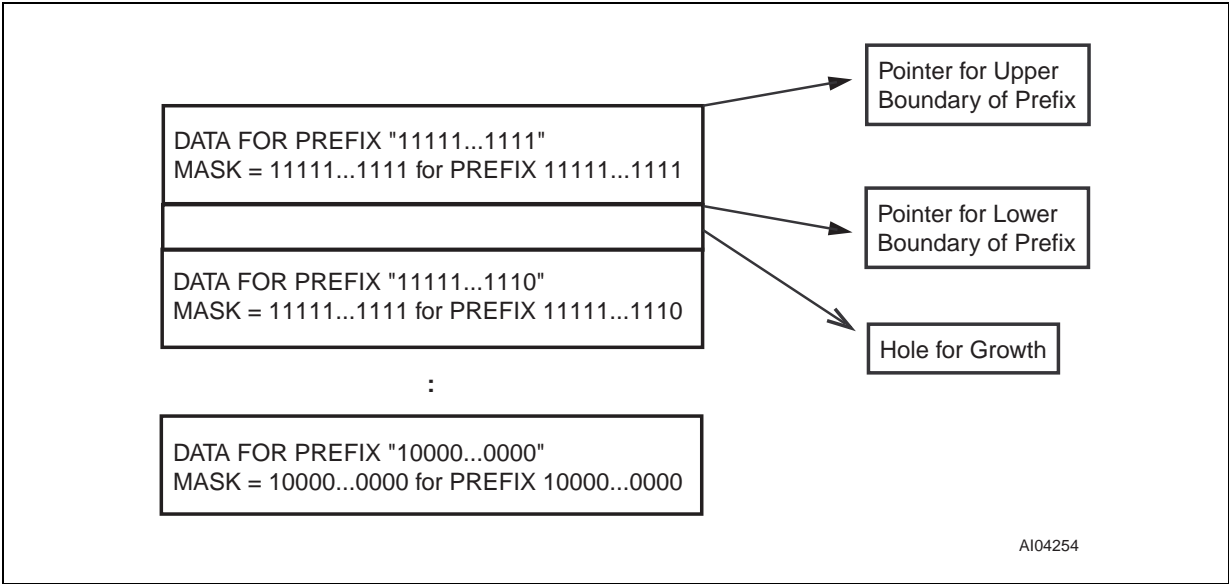
Table Maintenance for longest prefix match

Assume that there are 31 possible prefixes (see Figure 1). The table is then divided into 31 regions arranged in order of decreasing prefix length. The size of the regions need not be equal. Depending on knowledge of application space, the user may expect certain prefix lengths to occur more frequently than others. The user can then allocate a bigger region size for the more frequently occurring prefixes. The system maintaining the table needs to keep track of 31 pointers each marking the start of a region. In addition, each region also needs a pointer to the next free location in that region. To add a new entry with a certain prefix length, one can write to the next free location in the corresponding region.

As entries get added and deleted over time, regions can become fragmented and will have to be defragmented. For example, one can leave a "hole" between the prefixes based on frequency of certain prefixes thereby allowing the prefix to expand. When one reaches the boundary of the next prefix, entries from the next prefix set can be relocated to the bottom of its own space, thereby creating room for the prefix above to push downwards. In a dynamic environment it is possible for some regions to get filled up. Region sizes can be adjusted by relocating entries, and changing the region pointer to reflect the new sizes. By invoking burst read and burst write support, one can achieve very high throughput in moving blocks of data.

Note: These pointers are all maintained by table management software; the M7010 or M7020 themselves do not maintain these pointers, however, the on-chip burst counters for read and write help in data movement.

Figure 1. Arranging the Data Array in a Sorted Order of Prefixes



CONCLUSION

This application note describes how to use the M7010 and M7020 for Longest Prefix Match applications. Although this note uses a simple IPv4 example to convey the concept, the M7010 and M7020 can be used on other similar applications requiring a form of Longest Prefix Match. Table management was also discussed.

CONTACT INFORMATION

If you have any questions or suggestions concerning the matters raised in this document, please send them to the following electronic mail addresses:

- apps.nvram@st.com* (for application support)
- ask.memory@st.com* (for general inquiries)

Please remember to include your name, company, location, telephone number, and fax number.

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is registered trademark of STMicroelectronics
All other names are the property of their respective owners.

© 2001 STMicroelectronics - All Rights Reserved

STMicroelectronics GROUP OF COMPANIES
Australia - Brazil - China - Finland - France - Germany - Hong Kong - India - Italy - Japan - Malaysia - Malta - Morocco -
Singapore - Spain - Sweden - Switzerland - United Kingdom - U.S.A.

www.st.com