

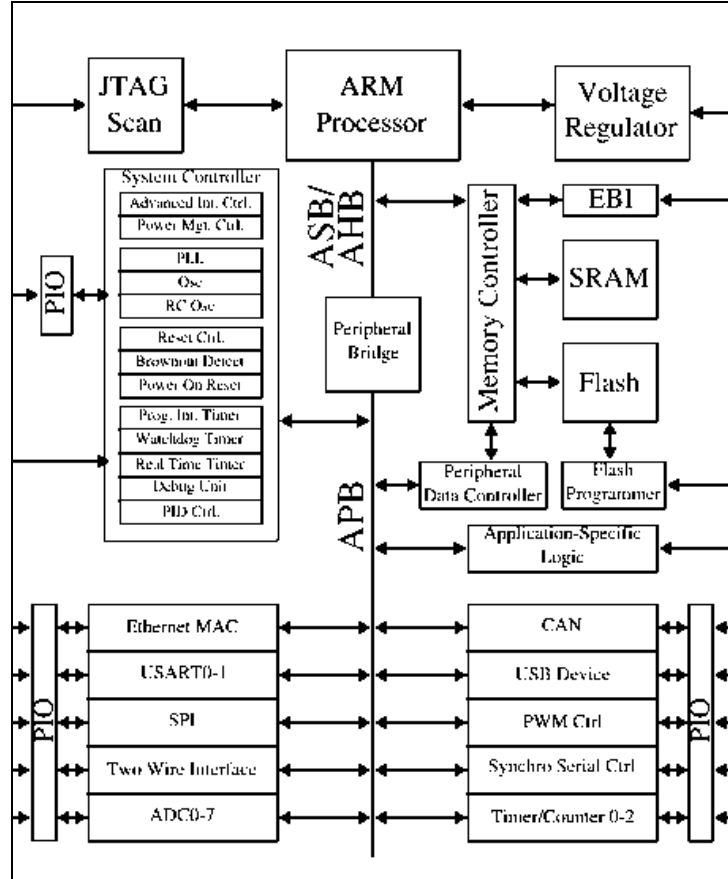
YONGA ÜSTÜ İLETİŞİM AĞI (Network-on-Chip / NoC)

Tevfik YAPICIOĞLU
Bilişim Enstitüsü
İstanbul Teknik Üniversitesi
Maslak, 34469, İstanbul,
Türkiye
yapicioglu@itu.edu.tr

1. Giriş

Yonga tasarımında dört farklı alan vardır: İşlem, Bellek, İletişim ve Giriş/Çıkış (G/Ç). İşlem gücünün artması ve veriyi yoğun olarak kullanan uygulamaların yaygınlaşmasıyla birlikte tek yongalı SoC (System-on-Chip / Yonga üstü Sistem) mimarisi içinde iletişim önemli bir araştırma konusu haline gelmiştir. Bu yapıya NoC (Network-on-Chip / Yonga Üstü İletişim Ağı) adı verilmiştir.

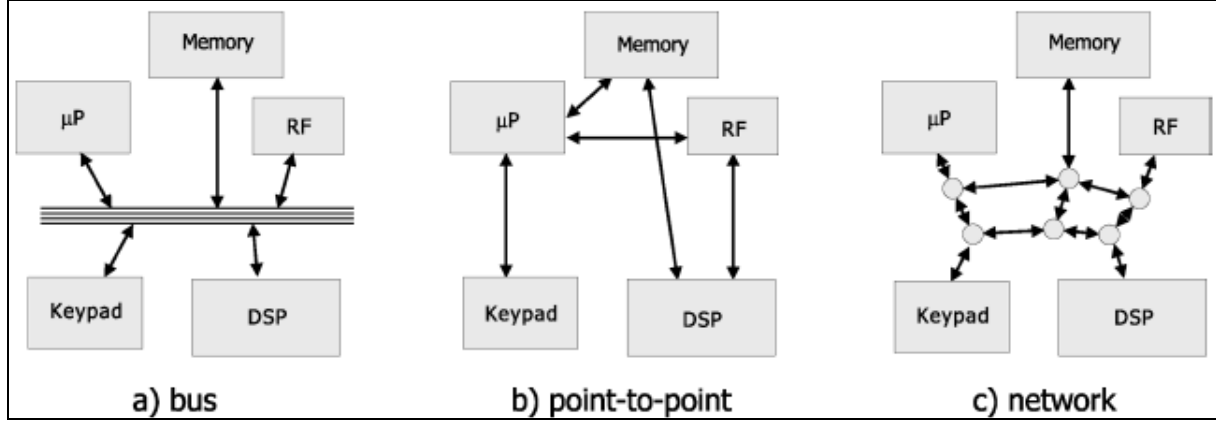
SoC, bir bilgisayarın veya herhangi elektronik sistemin tüm bileşenlerinin tek bir yonga üzerinde bütünleştirildiği yapıyı ifade etmektedir [1]. Bu yapı, özellikle gömülü istemlerde yaygınlaşmıştır. Bir SoC, genelde şu bileşenlerden oluşur: Bir veya daha çok mikroişlemci/denetleyici, bellek blokları, saatler, dış sistemlere bağlanmak için arayüzler, güç yönetimi ile ilgili devreler. SoC yapısında tüm bileşenler ortak yol üzerinden haberleşirler (Şekil 1).



Şekil 1 – Mikrodenetleyici temelli SoC

Yonga teknolojisindeki gelişmelerle birlikte büyük ölçekli SoC yapısı içinde iletişimin önemi artmıştır. Eskiden iletişim gücü işlem gücünden daha pahalıyken artık iletişim işlem gücünden daha ucuzdur. Bir SoC içindeki eleman sayısı arttıkça iletişim yol üzerinden iletişim daha güç hale gelmektedir. Bununla birlikte global senkronizasyonu sağlamak da daha zor olup bunu sağlamak için gereken elemanlar güç tüketimini artırmaktadır.

SoC yapısı içinde iletişim farklı şekillerde sağlanmaktadır (Şekil 2):



Şekil 2 - SoC içi iletişim biçimleri

SoC yapısının ortaya çıktığı 1990'lerden beri, yonga içi iletişim, genellikle yol temelli ve noktadan noktaya iletişim biçimlerinin karışımı şeklinde sağlanmaktaydı [2]. Yol üzerinden iletişim kolay modellenebilmesine karşın çok bağlantılı bir sistemde bir iletişim darboğazı oluşturabilir. Bununla birlikte birim iletişim maliyeti de artar. Noktadan noktaya özgül bağlar, bant genişliği, gecikme, güç tüketimi gibi ölçütlere göre iyi bir başarıma sahip olmasına rağmen gerekli bağ sayısının eleman sayısına göre üstel olarak artması sonucu yonga alanı yetersizliği gibi bir soruna neden olabilir.

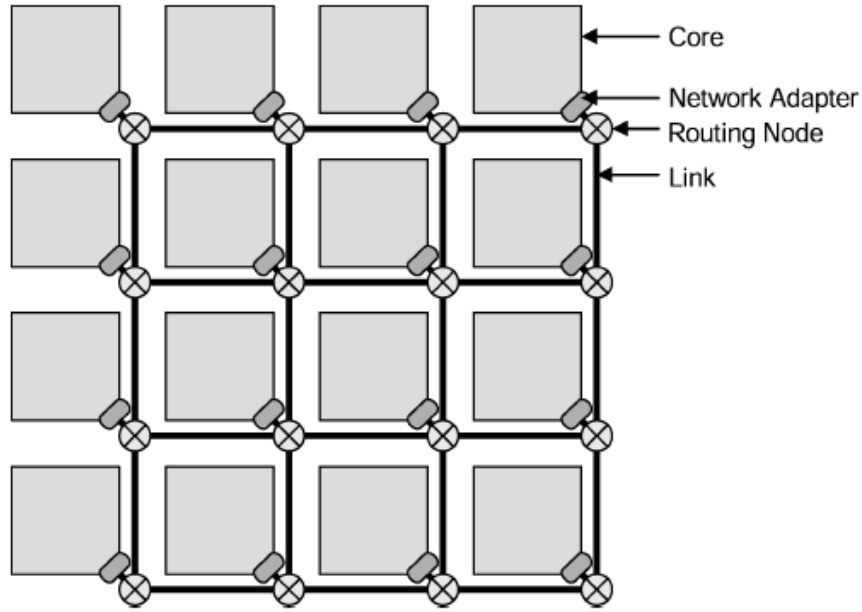
Büyük ölçekli SoC içi iletişim sorunlarını çözmek için bilgisayar ağlarında kullanılan çözümlerin yonga içi iletişimde kullanılması düşünülmüştür. Amaç basit bir ağ yapısıyla SoC içerisinde küresel iletişimi sağlamaktır. NoC adıyla bilinen bu yöntemde iki temel amaç vardır:

- Hesaplama merkezli tasarımdan, iletişim merkezli tasarıma geçiş
- Ölçeklenebilir iletişim mimarilerinin gerçekleştirilmesi

NoC, mimari olarak SoC'tan tamamen farklı bir kavram ortaya koymayıp SoC'a bir eklenti olarak tasarlanmıştır. Amaç, bilgisayar ağlarında kullanılan katmanlı yapı yaklaşımını yonga içi iletişime uyarlamaktır. Bu da tasarımda, bilgisayar ağlarında kullanılan çözümlerin sınırlı güç ve alan gibi farklı kısıtlamalara göre uyarlanmasını gerektirmiştir.

2. NoC Mimarisi ve Bileşenleri

NoC, temel olarak dört bileşenden oluşur (Şekil 3):



Şekil 3 - Örnek NoC topolojisi

Çekirdek (IP blok): Yonga üstündeki elemanlar. IP (Intellectual Property / Fikri Mülkiyet) bloğu olarak da adlandırılırlar.

Ağ bağdaştırıcısı: Çekirdeklerin NoC'a bağlanmasını sağlarlar. Çekirdeklerdeki işlem ile ağ üzerindeki iletişimi birbirinden soyutlarlar.

Yönlendirici düğüm: Verilerin seçilen protokole göre yönlendirilmesini sağlarlar.

Bağ: Düğümler arası fiziksel bağlantıyı oluştururlar. Bir veya daha fazla kanaldan oluşabilirler.

NoC mimarisinde, başarıyı artırmak için çeşitli düzenlemeler yapılabilir. Trafik türü ve miktarına göre çekirdekler uygun şekilde kümelenebilirler. Konum temelli bu kümeleme biçimi "Fiziksel kümeleme" olarak anılır [3]. Bu sayede toplam iletişim maliyeti ile başarıyı ve bunlara bağlı olarak güç tüketimine iyileşme sağlanabilir.

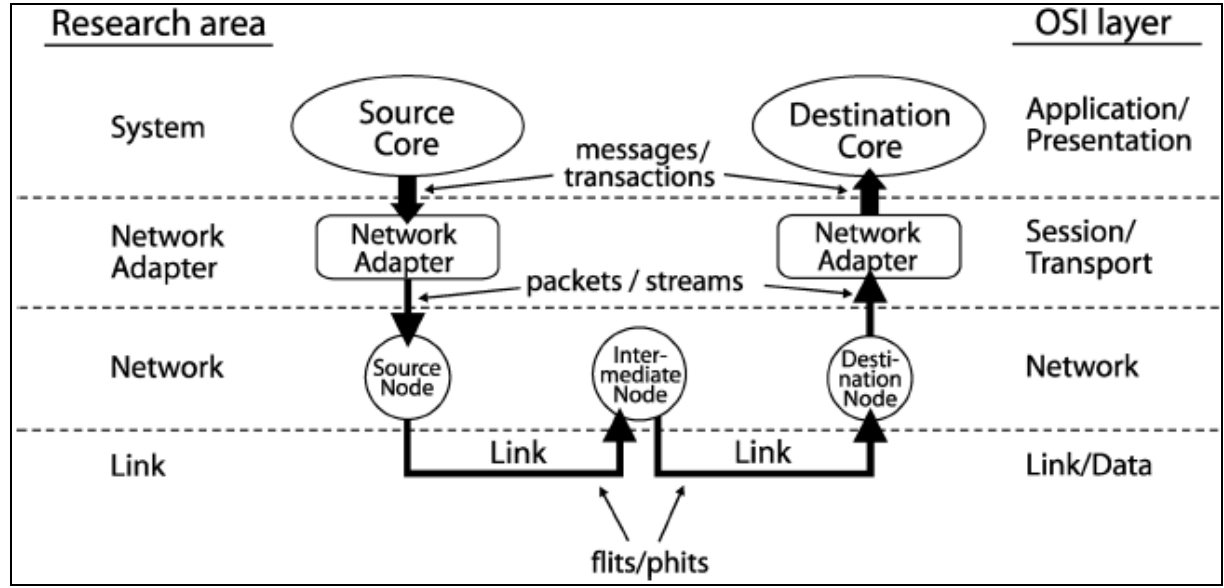
Katmanlı OSI mimarisini NoC mimarisi için uyarlandığı zaman [4] NoC araştırmaları 4 düzeyde sınıflanabilir: Sistem, Ağ bağdaştırıcısı, Ağ, Bağ. (Şekil 4)

Sistem düzeyi, uygulamaları ve genel mimariyi kapsar. İletişim ağından tamamen soyutlanmıştır.

Ağ bağdaştırıcı düzeyi, uçtan uca akış denetimini sağlar, iletişim ağıyla ilişkili ilk katmandır.

Ağ düzeyi, yönlendirme protokolleri, topoloji, düğümden düğüme akış denetimi ile ilgili araştırmaları kapsar.

Bağ düzeyi en küçük iletişim birimlerinin olduğu düzeydir. Düşümden düşüme fiziksel iletimin kotarılmasını sağlar.

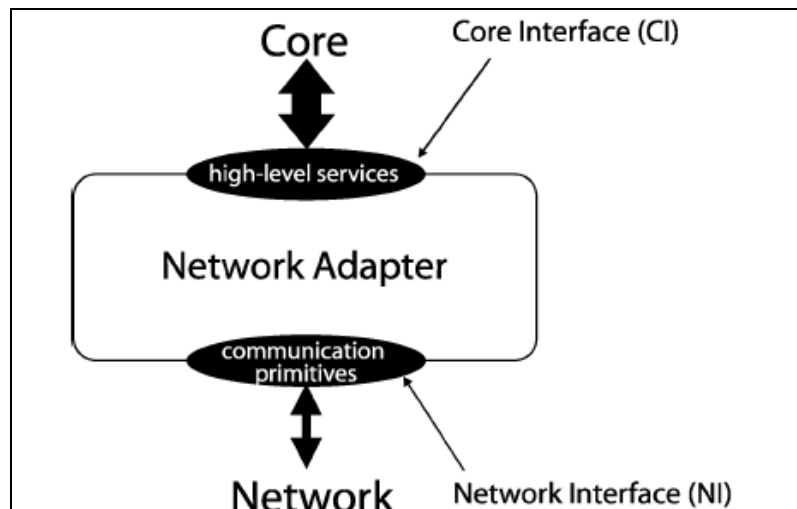


Şekil 4 - Verinin NoC bileşenleri boyunca kaynaktan hedefe iletimi

NoC'ta katmanlar, bir bilgisayar ağının katmanlarına göre birbiriyle daha ilişkilidir. NoC'un bilgisayar ağına göre avantajı daha statik bir ağ yapısına sahip olmasıdır. Çekirdeklerin bağlanma biçimleri, trafik özellikleri gibi bilgiler önceden bilinerek tasarım yapılır. Bu nedenle soyutlama klasik bilgisayar ağlarındaki düzeyde değildir.

3. Ağ bağdaştırıcısı düzeyi

Ağ bağdaştırıcısı çekirdek ve ağ katmanlarıyla iletişim için iki arayüze sahiptir. (Şekil 5)



Şekil 5 - Ağ bağdaştırıcısının arayüzleri

Ağ bağdaştırıcısı, işlem ile iletişim arasında bir sınır görevi görür. Çekirdek ile ağı birbirinden soyutlar. Bu soyutlamanın düzeyi tasarıma göre değişebilir. Yüksek düzeyde soyutlama çekirdeklerin yeniden kullanılabilirliğini artırır. Öte yandan daha az miktarda soyutlamayla ağ yapısıyla ilgili bilgi sahibi olan bir çekirdek, ağ kaynaklarının kullanımını en iyi seviyeye çıkarabilir.

Çekirdeğin tasarımında bağlı olduğu tek şey ağ bağdaştırıcısının çekirdek arayüzünde kullanılan soket yapısıdır. Yaygın olarak kullanılan bir soket yapısı OCP (Open Socket Protocol / Açık Soket Protokolü)'dir [5]. OCP, çekirdek arayüzü için esnek protokoller tanımlar. OCP'nin 3 temel hedefi vardır:

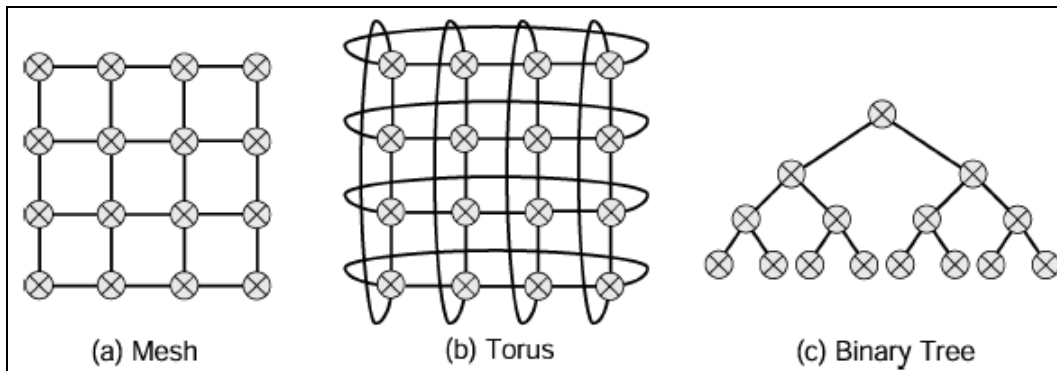
- Mimariden bağımsız yeniden kullanılabilirlik
- Özniteliğe özgü soket uygulaması
- Sistem doğrulamayı ve sınamayı kolaylaştırma

Çekirdek arayüzünde OCP kullanımının maliyeti ihmal edilebilecek düzeyde değildir. [6]'ya göre OCP, mesajların iletiminde %50 yavaşlamaya neden olmaktadır. Ancak OCP kullanımı ile mevcut yapılardan NoC yapısına geçiş çekirdekte bir değişiklik yapılmadan gerçekleştirilebilir.

4. Ağ düzeyi

Ağın görevi mesajları kaynak düğümden hedef düğüme yönlendirilmesini sağlamaktır. İyi tasarlanmış bir ağ, düğümlere mantıksal bir tel gibi görünmelidir. Bir yonga üstü ağ, temelde, topolojisi ve kullanılan protokolüyle tanımlanır. Topoloji, çekirdeklerin yerleşimi ve bağlantılarıyla ilgilendir. Protokol, düğümlerin ve bağlantıların kullanım biçimini düzenler.

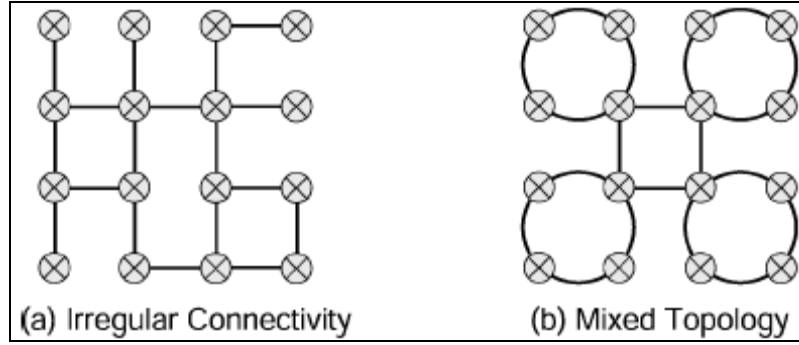
Düzenli topolojilerde (Şekil 6), yonga alanı kullanımı ve güç tüketimi topolojinin boyutu büyüdükçe doğrusal olarak tahmin edilebilir miktarda artar. Bu da NoC tasarımlarında düzenli topolojilerin daha çok kullanımına neden olmaktadır.



Şekil 6 - Düzenli topolojiler

Örgü topolojiler genellikle bağ kullanımını iyileştirirken, ağaç topolojileri trafiğin yöreselliğinden faydalanmak için kullanılabilir.

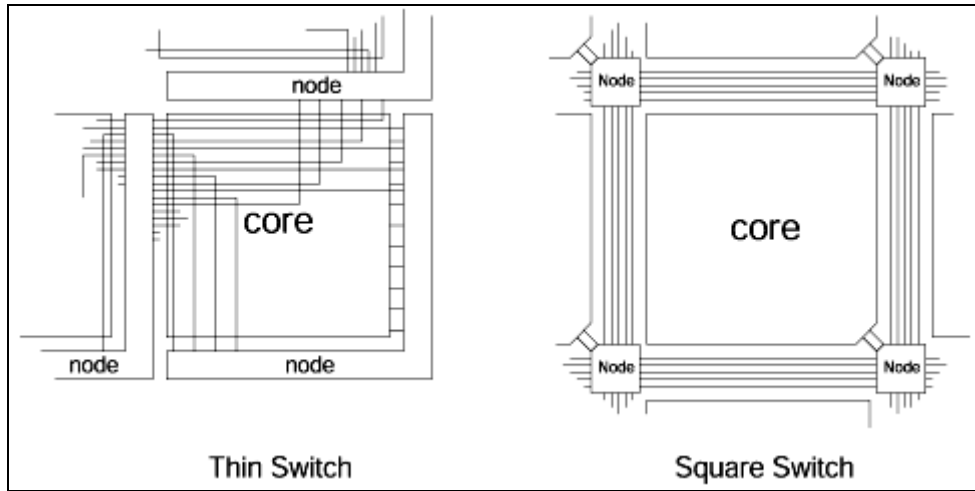
Düzensiz topolojiler, düzenli topoloji biçimlerinin hiyerarşik, melez veya asimetrik bir biçimde karıştırılmasıyla elde edilir (Şekil 7). Düzensiz topolojilerde yonga alanı kullanımı ve güç tüketimi doğrusal olarak ölçeklenemez. Bu yapılar genellikle kümelemeyle birlikte kullanılır. [7]'de kümelemenin enerji tüketiminde, trafiğin yöreselliğine bağlı olarak, %20 ila %40 arası azalma sağladığını gösterilmiştir.



Şekil 7 - Düzensiz topolojiler

Tüm yönlendirici düğümlerin en az bir çekirdeğe bağlı olduğu ağlara "Doğrudan ağ" denir. Bazı ağlarda ise bazı düğümler sadece diğer düğümler arası bağlantıyı sağlar, hiçbir çekirdeğe bağlı değildir. Bu tür ağlar "Dolaylı ağ" olarak adlandırılır.

Yönlendirici düğümlerin yerleşimi de ağ başarımında önemli bir yere sahiptir. Şekil 8'de, düğümlerin iki farklı yerleşimi görülmektedir [8].



Şekil 8 - Düğümlerin dizilimi: İnce anahtar ve Kare anahtar

Kare anahtar daha fazla bant genişliği ve genel olarak daha iyi başarımlar sağlarken, ince anahtar görece daha az yonga alanı kaplar. Sınırlı yonga alanı nedeniyle NoC yapısının gerektirdiği alan kullanımının en küçük olması istenir. Bu nedenle, ince veya kare anahtar kullanımı, NoC tasarımında bir ödünleşim gerektirir.

Ağ protokolü, verinin NoC boyunca yönlendirilme stratejisiyle ilgilenir. Anahtarlama verinin bir noktadan diğer noktaya iletimini sağlarken yönlendirme, verinin üzerinden iletileceği noktaları belirler.

NoC tasarımında ağ katmanında karar verilmesi gereken diğer noktalar şunlardır [3]:

- *Devre anahtarlama / Paket anahtarlama*

Devre anahtarlama, iletim o bağlantıya adanmış bir devre üzerinden gerçekleştirilir. Paket anahtarlama ise her paket bağımsız olarak yönlendirilir.

- *Bağlantılı iletişim / Bağlantısız iletişim*

Bağlantılı iletişimde, veri iletime başlamadan önce, mantıksal bir bağlantı kurulur. Bu bağlantı, veri iletimi bitince sonlandırılır. Bağlantısız iletim ise kaynak ve hedef arasında hiçbir ön düzenleme olmadan dinamik olarak kotarılır. Bu nedenle devre anahtarlama bağlantılı iletişim gerektirirken paket anahtarlama bağlantılı veya bağlantısız çalışabilir.

- *Deterministik yönlendirme / Adaptif yönlendirme*

Deterministik yönlendirmede iletim yolu yalnızca kaynak ve/veya hedef düğümler tarafından belirlenir. Bu türün NoC'taki yaygın uygulamaları, kaynak yönlendirme ve X-Y yönlendirme. Kaynak yönlendirmede hedefe giden yolun tamamı kaynak çekirdekte belirlenir. X-Y yönlendirmede ise paket önce X eksenini boyunca gerekli yolu kat ettikten sonra Y eksenini boyunca gereken ilerlemeyi yaparak hedefe ulaşır. Adaptif yönlendirme düzenekleri ise koşullara bağlı (bağ yoğunluğu vs.) olarak dinamik karar verir. Bu yöntem düğümün karmaşıklığını artırırken eşit yük dağılımını sağlar.

- *Minimal yönlendirme / Minimal olmayan yönlendirme*

Minimal yönlendirmede her zaman en kısa yol iletim yolu olarak seçilir. Minimal olmayan yönlendirmede ise yolun uzunluğunun yanında diğer etkenlerde yönlendirme kararında göz önüne alınır.

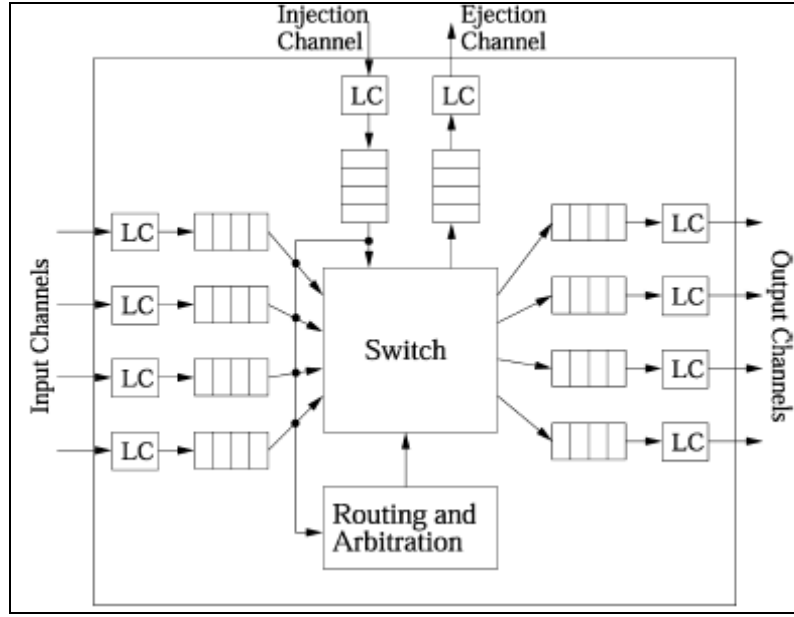
- *Gecikmeli model / Kayıplı model*

Gecikmeli modelde mesajlar asla yok edilmez, en kötü ihtimalle geciktirilir. Kayıplı modelde ise fazla gecikme hoş görülmez ve paket düşürülür. Bu durumda paketin yeniden iletilmesi gerekir. Bu düzenek maliyeti biraz artırsa da ağ tıkanıklıklarını engellemede etkili olur.

- *Merkezi denetim / Dağıtık denetim*

Merkezi denetimde yönlendirme kararları merkezi olarak verilir. Örnek: Yol hakemi. Dağıtık denetimde ise yönlendirme kararları yerel olarak verilir.

Bir yönlendirme birimi genel olarak şu bileşenlerden oluşur: tamponlar, anahtar, yönlendirme ve karar verme birimi, bağ denetleyicisi (Şekil 9).



Şekil 9 - Genel yönlendirici modeli

Anahtar, giriş tamponlarını çıkış tamponlarına bağlarken yönlendirme ve karar verme birimi bu bağlantıya karar verir. Anahtar tasarımında önemli olan nokta yönlendiricinin verdiği hizmettir. [9]'da katı başarımların garantilerini sağlamak için tıkanmaz bir anahtar kullanılmıştır.

[10]'da bağlantılı ve bağlantısız iletişimlerin başarımlarını karşılaştırılmıştır. Bağlantısız iletimde uçtan uca gecikmenin, video kod çözümünde, bağlantılı iletime göre, daha düşük olduğu görülmüştür. Bununla birlikte bağlantısız iletimde, bir akışa yanlış hizmet sınıfını atandığı zaman daha kararlı bir başarımlar sağlamaktadır.

Bağlantılı veya bağlantısız iletişim tercihi hizmet gerekliliklerine göre değişmektedir. "Hizmet garantisi" olan durumlarda bağlantılı iletişim tercih edilirken, "elden gelenin en iyisi" türü hizmet verildiğinde bağlantısız iletişim kullanılır, çünkü kotasının daha kolaydır.

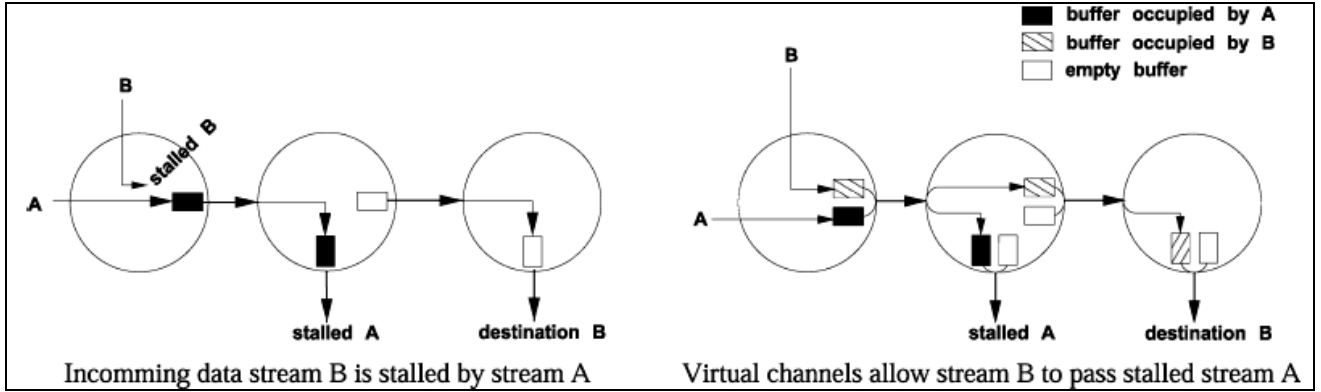
Deterministik ve adaptif yönlendirme protokolleri karşılaştırıldığında, deterministik protokollerin hem kararlı başarımlarını hem de gerçekleştirme kolaylığı nedeniyle NoC'ta daha iyi bir seçim olduğu gösterilmiştir [11].

NoC'ta yaygın olarak kullanılan anahtarlama yöntemi paket anahtarlama değildir. Bununla birlikte gecikme tabanlı model temel alınmıştır. Bunun nedeni kaybedilen paketlerin bilgisini tutup paketi tekrar yollamak NoC için masraflı bir çözüm olmasıdır.

Düğümünden düğüme yönlendirme 2 farklı şekilde yapılır: Sakla-ve-ilet, solucan deliği. Sakla-ve-ilet kipinde, paketin tamamı alınana kadar saklanır, daha sonra başlıktaki bilgiye göre iletilir. Solucan deliği kipinde, paketin başlığı alınıp

hedef düğüm belirlenir belirlenmez, paket iletmeye başlanır. Böylece paket anahtarlama kullanılırken devre anahtarlama düzeyinde bir gecikmeyle iletim sağlanır. Bilgisayar ağlarında sakla-ve-ilet kipi daha yaygın bir şekilde kullanılırken, NoC'larda baskın olan yönlendirme türü solucan deliğidir. Bunun nedeni daha düşük gecikme ve daha az tampon kullanımına bağlı olarak daha az yonga alanı kullanımınıdır. Solucan deliği yönteminde önemli bir sorun, hata durumunda paketin gecikmesi ve yolu kilitlemesi durumudur. Bu durum gerçekleşirse ters yönde bir sonlandırma paketi yollanarak yol iptal edilir ve paket yeniden iletilir [12].

NoC'larda akış denetimini sağlamak için kullanılan bir yapı sanal kanallardır. Akış denetiminin amacı ağ kaynaklarını düzgün kullanımını sağlayıp kararlı başarımını korumaktır. Sanal kanalda fiziksel bağlar, mantıksal olarak birbirinden bağımsız çok sayıda kanala bölünür. Sanal kanalları uygulamak güç tüketimi ve kullanılan yonga alanı bakımından masraflıdır, ancak bunun ölümcül kilitlemeleri engelleme, tel kullanımını optimize etme, başarımı artırma, sınıflandırılmış hizmet sağlama gibi getirileri vardır. Şekil 10'da sanal kanalların ölümcül kilitlemeyi nasıl engellediği görülmektedir [3].



Şekil 10 - Ölümcül kilitleme ve sanal kanal çözümü

Tamponlamanın NoC'ta doğrudan etkisi yonga üzerinde alan gereksinimidir. Bu nedenle tampon kullanımı NoC'ta en az olmalıdır. Tampon kullanımı başarımı artırır da güç ve yonga alanı kullanımından dolayı tercih edilen bir yöntem değildir. Bunun için NoC'ta DAMQ (Dynamically Allocated Multiqueue / Dinamik Tahsis Edilen Çoklu Kuyruk) yöntemi geliştirilmiştir [13]. Bu modelde herhangi bir pakete sırasız olarak erişilip iletilebilirken gelen paketler de boş yerlere yerleştirilebilir.

NoC'un uygulamalarından önemli olan bazıları aşağıda sıralanmıştır [3]:

- SPIN (Scalable Programable Integrated Network) – 2000
- ETHEREAL – Philips, 2001
- NOSTRUM – KTH, 2002
- CHAIN (CHip Area INterconnect) – 2002
- XPIPES – 2003
- MANGO (Message passing Asynchronous Network-on-chip providing Guaranteed Services over OCP interfaces) – 2005

7. Sonu

NoC'ta temel hedef, iřlem temelli bir yapıdan iletiřim temelli bir yapıya geiřtir. Bilgisayar ađları dzeyinde ađ teknolojileri ok geliřmiř olmasına karřın bu teknolojileri yonga dzeyinde iletiřimde olduđu gibi kullanmak olanaksıza yakındır. Bu nedenle yonga st iletiřim nemli bir arařtırma konusu haline gelmiřtir. Bir mimarinin yonga st iletiřime uygun olması iin basit ve kolay uygulanabilir olması gerekir. Yonga st iletiřim iin tasarlanan bir mimari dřk g tkietmeli ve fazla yer kaplamamalıdır. Bu nedenlerle NoC konusundaki alıřmalarda sistem gereksinimleriyle g tkietimi ve alan kullanımını arasında bir dnleřim yapılması gerekmektedir.

8. Kaynaklar

- [1] System-on-a-chip - Wikipedia, the free encyclopedia
<http://en.wikipedia.org/w/index.php?title=System-on-a-chip&oldid=173855786&printable=yes>
- [2] Lahiri, K., Raghunathan, A., Dey, S. 2001. Evaluation of the traffic-performance characteristics of system-on-chip communication architectures. In Proceedings of the 14th International Conference on VLSI Design. IEEE, 29–35.
- [3] Bjerregaard, T., Mahadevan, S., "A survey of research and practices of Network-on-chip". ACM Computing Surveys 2006 Vol. 38, Iss. 1, 2006, Article 1
- [4] Benini, L., Micheli, G. D. 2001. Powering network-on-chips. In The 14th International Symposium on System Synthesis (ISSS). IEEE, 33–38.
- [5] Haverinen, A., Leclercq, M., Weyrich, N., Andwingard, D. 2002. SystemC based SoC communication modeling for the OCP protocol. White paper.
<http://www.ocpip.org>.
- [6] Ost, L., Mello, A., Palma, J., Moraes, F., Calazans, N. 2005. MAIA—a framework for networks on chip generation and verification. In Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC). IEEE.
- [7] Pande, P., Grecu, C., Jones, M., Ivanov, A., Saleh, R. 2005. Effect of traffic localization on energy dissipation in NoC-based interconnect. In International Symposium on Circuits and Systems (ISCAS). IEEE, 1774–1777.
- [8] Kumar, S., Jantsch, A., Soininen, J.-P., Forsell, M., Millberg, M., Oberg, J., Tiensyrja, K., Hemani, A. 2002. A network-on-chip architecture and design methodology. In Proceedings of the Computer Society Annual Symposium on VLSI (ISVLSI). IEEE Computer Society, 117–124.
- [9] Bjerregaard, T, Sparso, J. 2005. A router architecture for connection-oriented service guarantees in the MANGO clockless network-on-chip. In Proceedings of Design, Automation and Testing in Europe Conference (DATE). IEEE, 1226–1231.
- [10] Harmancı, M., Escudero, N., Leblebici, Y., Ienne, P. 2005. Quantitative modeling and comparison of communication schemes to guarantee quality-of-service in networks-on-chip. In International Symposium on Circuits and Systems (ISCAS). IEEE, 1782–1785.
- [11] Neeb, C., Thul, M., Wehn, N. 2005. Network-on-chip-centric approach to interleaving in high throughput channel decoders. In International Symposium on Circuits and Systems (ISCAS). IEEE, 1766–1769.
- [12] Gaughan, P. T., Dao, B. V., Yalamanchili, S., Schimmel, D. E. 1996. Distributed, deadlock-free routing in faulty, pipelined, direct interconnection networks. IEEE Trans. Comput. 45, 6 (June) 651–665.
- [13] Tamir, Y., Frazier G. L. 1988. High-performance multiqueue buffers for VLSI communication switches. In Proceedings of the 15th Annual International Symposium on Computer Architecture. IEEE Computer Society, 343–354.