



**Gamze AKSU**

**171180005**

**GAZİ ÜNİVERSİTESİ MÜHENDİSLİK FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ**

**BM 311 BİLGİSAYAR MİMARİSİ**  
**M.ALİ AKÇAYOL**

**Hafıza İçi Hesaplama (In-Memory Computing)**

**OCAK 2021**

| <b>İÇİNDEKİLER</b>   | <b>Sayfa</b> |
|--|--------------|
| İçindekiler.....   | 1            |
| Özet.....  | 2            |
| Hafıza İçi Hesaplama.....  | 3            |
| 1.Giriş.....   | 3            |
| 2. Hafıza İçi Veri İzgarası (IMDG).....                                    | 5            |
| 2.1. IMDG Mimarisi.....  | 6            |
| 3. Hafıza İçi Hesaplamanın Temel İlkeleri.....                             | 7            |
| 4. Hesaplamalı Hafıza Teknolojileri (Computational Memory Technologies)... | 9            |
| 4.1. Dirençli RAM (Resistive RAM).....                                     | 9            |
| 4.2. Faz Değiştirme Belleği (Phase-Change Memory).....                     | 10           |
| 4.3. Manyeto-dirençli RAM (Magnetoresistive RAM).....                      | 10           |
| 4.4. Ferroelektrik RAM (Ferroelectric RAM).....                            | 11           |
| Sonuç.....   | 12           |
| Kazanımlar.....  | 13           |
| KAYNAKÇA .....   | 14           |

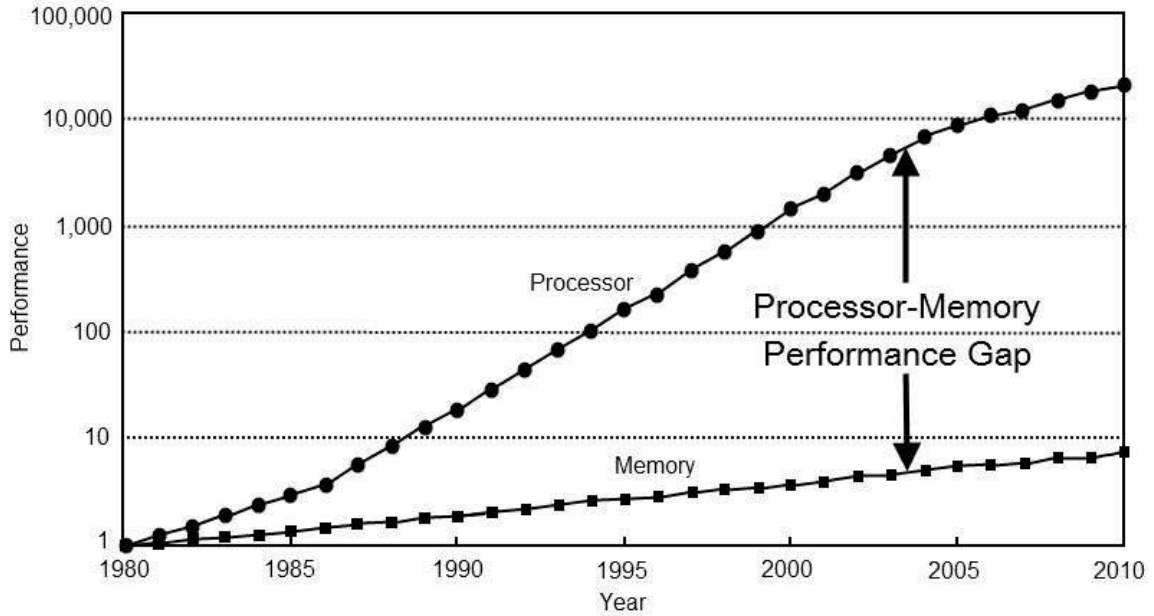
## ÖZET

Hafıza içi hesaplama (in-memory computing) genel olarak von Neumann mimarisindeki bir sorun olan von Neumann darboğazı olarak adlandırılan bu sorunun üstesinden gelmek için geliştirilmiş bir teknolojidir. IoT ve Big Data yaygınlaşmaya başladıkça daha çok veri işlemek durumunda kalınca darboğaz daha büyük bir sorun haline gelmiştir. Hafıza içi hesaplama öz mantığı olarak hesaplama ve depolama birimlerinin bir arada olması denilebilir. Bu araştırma ödevinin birinci bölümünde von Neumann mimarisinden bahsedilmiştir ve hafıza içi hesaplama tanımı yapılmıştır. İkinci bölümünde Hafıza İçi Veri Izgarası (IMDG) anlatılmıştır. Ayrıca IMDG mimarisinden de bahsedilmiştir. Üçüncü bölümde ise hafıza içi hesaplamanın 4 temel ilkesinden bahsedilmiştir. Bunlar yüksek hız ve ölçeklenebilirlik, gerçek zamanlı öngörüler, geniş kullanım durumları ve teknolojiyi etkinleştirmektir. Dördüncü bölümde ise hafıza içi hesaplamanın genel olarak gerçekleştirildiği RAM türlerinden 4 tanesinden bahsedilmiştir. Bunlar dirençli RAM (RRAM), faz değiştirme belleği (Phase-Change Memory PRAM), manyetoresistif RAM (MRAM) ve ferroelektrik RAM (FeRAM) dir.

# Hafıza İçi Hesaplama

## 1. Giriş

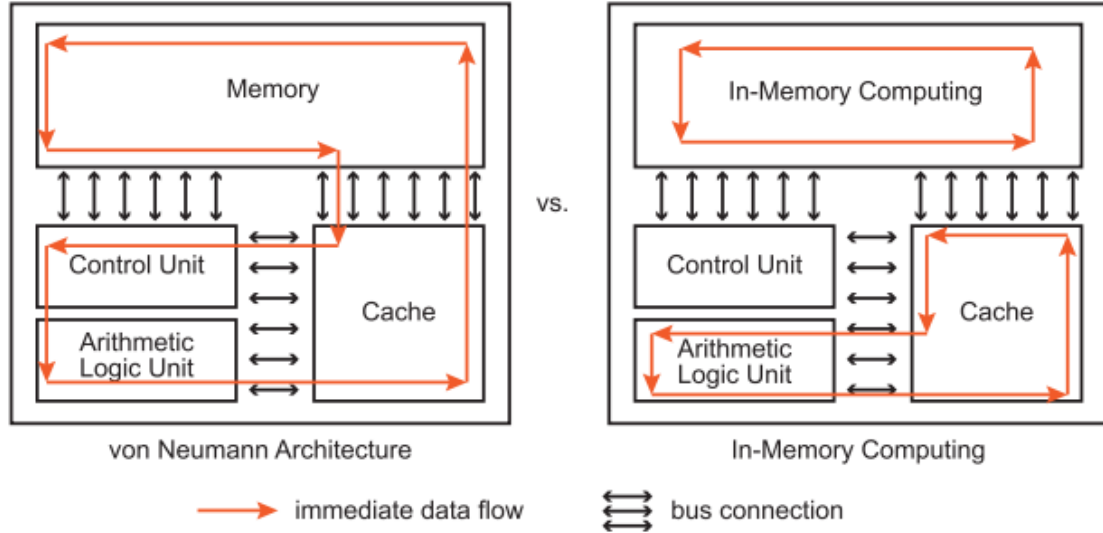
John von Neumann mimarisi çok eski yıllardan (1946) günümüze kadar gelmiştir. Hala modern bilgisayarlar da Von Neumann mimarisine dayanarak tasarlanır. Bu mimari günümüze kadar gelmiş olmasına rağmen beraberinde bazı sorunları da getirmiştir. Bu sorunlardan biri Von Neumann Darboğazı (Von Neumann Bottleneck) olarak adlandırılır. Von Neumann darboğazı von Neumann mimarisinin genel yapısından kaynaklanır. Von Neumann mimarisinde hesaplama ve depolama birimleri fiziksel olarak ayrı yerlerde. Bu yapı gereği işlemci işleyeceği veriyi hafızadan teker teker çağırıp işler ve hafızaya geri gönderir. Burada işlemci ve hafıza arasında adres ve data buslar tarafından iletişim sağlanır. Ancak son 25 yılda işlemcilerin hızı yıllık olarak yaklaşık %55 oranında artarken bu oran hafızalarda yıllık olarak %10'larda kalmıştır [1]. Bu şekilde hafıza ve işlemci arasındaki hız farkı gittikçe artmıştır. Bu soruna von Neumann darboğazı denir. İşlemcinin yüksek işlem yapma kapasitesi ile birlikte hafıza birimlerinden veri akışının yavaşlığı sonucu işlemcinin veriminin kısıtlanacağı ve aradaki farkı aşağıdaki şemayı inceleyerek de görebiliriz.



Şekil 1: İşlemci ile hafıza arasındaki performans grafiği

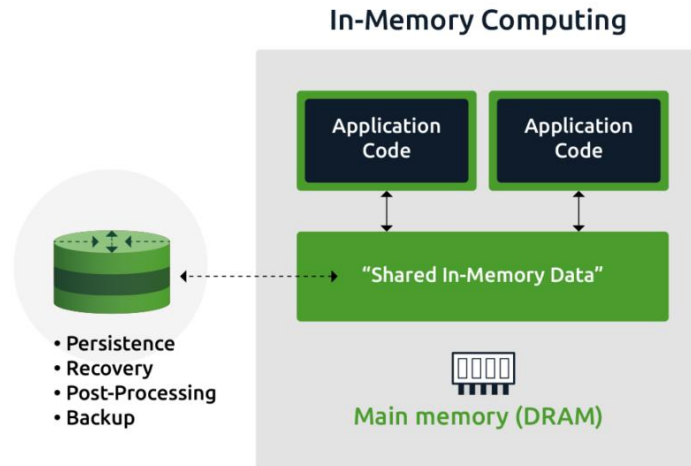
Von Neumann bilgisayarlarındaki darboğaz sorununu aşmak için bazı çözümler geliştirilmiştir. Bunlardan bazılarına örnek olarak cache bellekler verilebilir. Günümüzde Nesnelerin İnterneti (IoT) ve büyük veri (big data) gibi uygulamaların ortaya çıkması ile birlikte

büyük miktarda veriyi gerçek zamanlı olarak işleme ve geri alma ihtiyacı da ortaya çıktı. Cache bellekler gibi küçük bir depolama alanına sahip teknoloji ile bu sorunun üstesinden gelinemezdi. Bu durumda, mevcut bellek hiyerarşisine yönelik daha fazla optimizasyon artık bir çözüm olamazdı. Daha devrimci bir yaklaşım ile hafıza içi hesaplama (in-memory computing) fikri ortaya çıktı. Cache bellek yapısında da olduğu gibi hafıza içi hesaplama teknolojisinde de bellek işlemciye ne kadar yakın olursa o kadar hızlı olur mantığı işlenmiştir.



Şekil 2: von Neumann ile bellek içi bilgisayar mimarisi karşılaştırması.

Hafıza içi hesaplama teknolojisinde hafıza birimi harddisk ya da SSD yerine sadece RAM veya flash bellek kullanılmaktadır. Bir veya birden fazla bilgisayarın aynı RAM üzerindeki veriler üzerinde çalışması ile gerçekleştirilebilmektedir. Bu teknolojiye RAM’de işlem yapılırken aynı zamanda RAM’deki kayıtlar da von Neumann mimarilerindeki gibi bilgi aktarımı sırasında vakit kaybı olmadan otomatik güncellenmektedir.

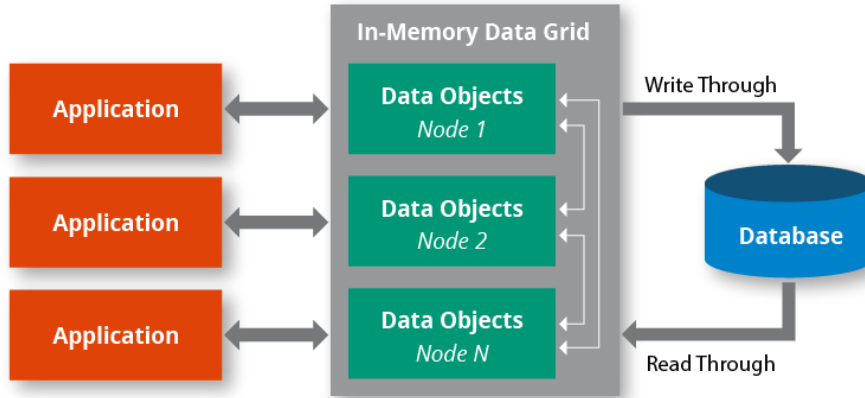


Şekil 3: Hafıza içi hesaplama ve veri paylaşımı

## 2. Hafıza İçi Veri İzgarası (IMDG)

Hafıza içi hesaplama teknolojisinde bir veya birden çok bilgisayarda verilerin yönetildiğinden bahsedilmiştir. Bilgisayarların birden çok olması durumunda hesaplama daha küçük parçalara bölünür ve her bilgisayarda paylaştırılıp paralel olarak çalıştırılır. Hafıza içi hesaplama Oracle, DB2 veya Microsoft SQL Server gibi geleneksel veri tabanı aracılığıyla ya da Hazelcast, Infinispan, Oracle Coherence veya ScaleOut Software gibi hafıza içi veri ızgarası (in-memory data grid IMDG) sunan NoSQL üzerinden gerçekleştirilebilir [2].

Bir hafıza içi veri ızgarası (IMDG), gerçekleştirilecek olan işlem ve uygulamalarda aynı ağ üzerinde bulunan işlemcilerin ortak bir RAM üzerinde çalışarak oluşturdukları kümelenmiş yapıdır. IMDG sistemlerinde sadece depolama işlemi için değil depolama işlemi ile birlikte yüksek hızda işlem yapmak için geliştirilmişlerdir. Yüksek RAM ihtiyacı duyulan, yani tek bilgisayar sunucunun RAM'inin yetersiz olacağı uygulamalarda, kullanılmaktadır. Büyük verilerin hızlı bir şekilde işlenmesi gerektiği durumlarda kullanılmaktadır. Böylelikle yüksek uygulama performansı sağlanmış olunur. IMDG ile işlem yapmak, fazladan işlevsellik için performansı düşüren ACID gibi gelişmiş işlevselliğe sahip ilişkisel veri tabanlarından üç kat daha hızlı gerçekleşir. ACID atomiklik, tutarlılık, izolasyon, dayanıklılık (atomicity, consistency, isolation, durability) kelimelerinin baş harflerinden oluşan bir veri tabanı işlemlerinin özelliğidir [3].



Şekil 4: IMDG diyagramı

IMDG'ler hafıza içi veri tabanı ile karıştırılmamalıdır. IMDG'lerde tablolar, indeksler, triggerlar gibi yapılar yoktur sadece düz depolama yapısıdır. Buna karşılık, bazı IMDG ürünleri, verilerin RDBMS ile senkronize halde çalışabilmesine olanak tanır. IMDG'de kullanılan veri modeli anahtar-değer çiftleridir. Bir anahtar, veri tabanındaki bir tablonun dizinine veya birincil

anahtarına eşit olabilir. Anahtar/değer çiftlerinin değerleri, sayı veya dize gibi basit veri türlerinden karmaşık nesnelere kadar ulaşan her şey olabilir.

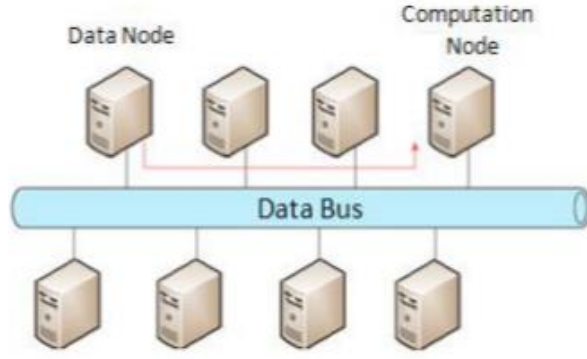
IMDG'ler bellekteki verilerin bütünlüğünü korurken aynı zamanda verilerin ölçeklenebilirliklerini ve performanslarını artırır. Bunlara ek olarak da kullanıcıların büyük hacimli verilerle uygulamaları çalıştırma maliyetlerini azaltmalarına yardımcı olur. IMDG'lerin yapıları genel olarak şu maddelerde açıklanmıştır. [4]

- Veriler birkaç sunucuya (bilgisayara) dağıtılmıştır. Bu sunuculara data fabric de denilir.
- Veriler sunucuların RAM'lerinde depolanır.
- IMDG ortamındaki tüm sunucular verilerini aktif modda çalıştırır.
- Bellek miktarını artırmak için sunucular eklenebilir veya kaldırılabilir.
- Bir veri modeli genellikle nesne tabanlıdır ve ilişkisel değildir.

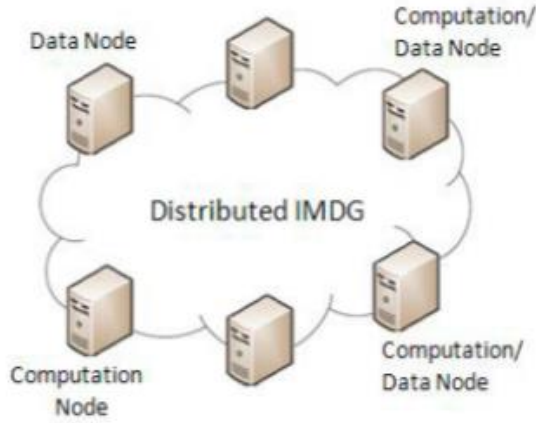
## 2.1. IMDG Mimarisi

IMDG aslında tek bir mantıksal parça gibi çalıştırılan ama birden fazla donanımı birbirine bağlayarak kapasitesini arttırabilen bir mimariye sahiptir. Buna yatay ölçeklenebilirlik mimarisi denir. RAM'leri hafıza olarak kullanmanın önündeki sorunlardan biri olan ölçeklenebilirlik sorunu bu şekilde aşılmış oldu. Bir diğer sorun olan güvenilirlik sorununu ise dağıtılmış bir yapısal tasarım kullanarak çözülür.

IMDG tabanlı dağıtılmış veri işleme mimarisi ile veri yolu tabanlı dağıtılmış veri işleme mimarisini karşılaştırdığımızda tipik bir veri yolu tabanlı dağıtılmış veri işleme mimarisinde veriler Veri Dğümleri (Data Nodes) üzerinde depolanır. Hesaplama yapılmak için veriler Veri Yolu (Data Bus) ile Hesaplama Dğümlerine (Computational Nodes) gönderilir. Bu tipik bir "veri taşıma (moving data)" dağıtılmış veri işleme stratejisidir. Şekil 5'da gösterilmiştir. Bu işlem IMDG tabanlı olduğunda ise bir dizi bilgisayar düğümü bir IMDG oluşturur ve veriler, IMDG düğümlerinde depolanır. Hesaplama yapılacağı zaman ise probleme özgü algoritmalar kendi veri işleme görevlerini yerine getirmek için verilerin bulunduğu düğüme gönderilir. Bu şekilde "algoritma taşıma (moving algorithm)" dağıtılmış veri işleme stratejisine ulaşılır. Şekil 6'de gösterilmiştir. Böylece genel hesaplama verimliliğini artırmak için hesaplama sürecinde veri hareketi ve disk I/O işlemleri en aza indirilir. Bu, hafıza içi hesaplamanın özüdür.



Şekil 5: Veri yolu tabanlı dağıtılmış veri işleme mimarisi.



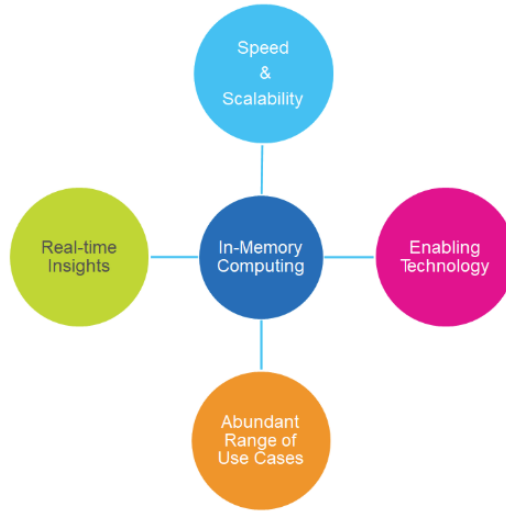
Şekil 6: IMDG tabanlı dağıtılmış veri işleme mimarisi.

### 3. Hafıza İçi Hesaplamanın Temel İlkeleri

Hafıza için hesaplama 4 ilke altında incelenebilir. Bunlar yüksek hız ve ölçeklenebilirlik, gerçek zamanlı öngörüler, geniş kullanım alanları ve teknolojiyi etkinleştirmedir. [5]

**Yüksek Hız ve Ölçeklenebilirlik (High Speed and Scalability):** Hafıza içi hesaplama RAM ve indeksleme tabanlı olduğu için daha fazla hıza sahiptir. Bu şekilde diğer çözümlerden 100 kat daha hızlı olabilmektedir. Ölçeklenebilirlik tanımı bir sistemin artan veri karşısındaki işleme yeteneği veya sürekli artan veri karşısında genişleyebilme potansiyeli olarak verilebilir. Özellikle big data için gerekli olan bu özellik için hafıza içi hesaplamada paralelleştirilmiş dağıtılmış işleme kullanılır. Tüm paylaşılan bilgisayarlara veri işleme olanağı sunularak bir bilgisayar ağı oluşturulur.





Şekil 7: Bellek İçi Hesaplamanın Temel İlkeleri

**Gerçek Zamanlı Öngörüler (Real-time Insights):** Hafıza içi hesaplama birden fazla kaynaktan alınan verilerin bir araya getirilmesini sağlar. Hafızada aynı anda büyük güncel ve geçmiş verinin akışı vardır. Hafıza ve işlemci birlikte konumlandırıldığından anlık iç görüler için gerçek zamanlı gelişmiş analitik ve makine öğreniminin çalıştırılması desteklenir. Bunlara ek olarak saniyede milyonlarca veriyi işleme ve elde edilen verileri analiz etme yeteneğinden yararlanan sürekli tahmin analizi ile ekipman arızaları, müşteri arızaları, siber saldırılar gibi istenmeyen olaylar önlenir.

**Geniş Kullanım Durumları (Abundant Range of Use Cases):** Hafıza içi hesaplama özellikle perakende, finansal hizmetler, sigorta, ulaşım, telekomünikasyon ve kamu hizmetleri gibi tüketici verileri çok büyük boyutlarda olan şirketlerde kullanılması gerekir. Hafıza içi hesaplamanın kullanım alanlarına örnek verilecek olursa bankalarda / finans kurumlarında risk ve işlem yönetimi, ödemelerde ve sigortada dolandırıcılık tespiti, tüketici ürünleri şirketlerinde ticari promosyon simülasyonları ve gerçek zamanlı veya kişiselleştirilmiş reklamlar verilebilir.

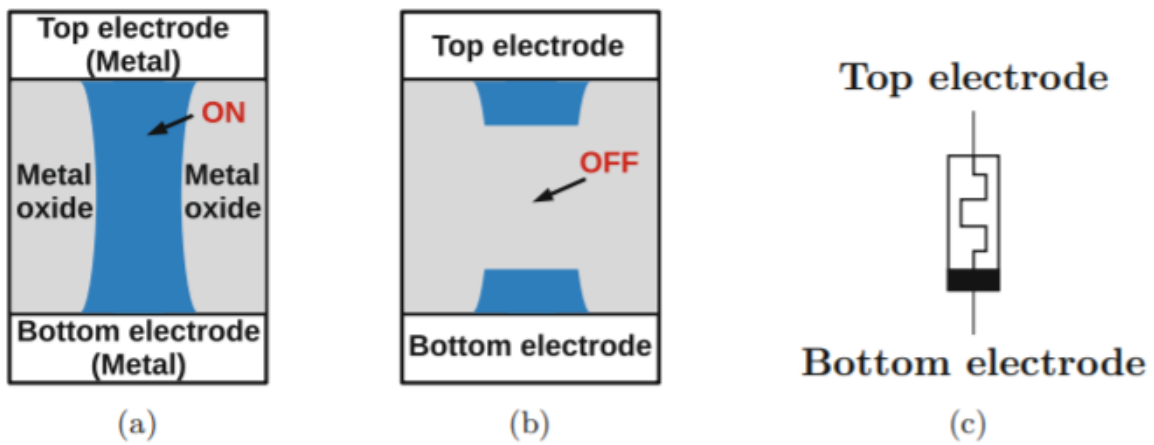
**Teknolojinin etkinleştirilmesi (Enabling Technology):** Hafıza içi hesaplama olmadan günümüz teknolojilerinin çoğu olmazdı. Verilerin büyüklüğü arttıkça onları hızlı bir şekilde işlememize yarayacak kolaylaştırıcı bir teknoloji olmadan bu verileri günlerce işlemek zorunda kalırdık. Bu durumda bu kadar hızlı bir şekilde ilerleyemezdik. Gerçek zamanlı talimatlar içeren uygulamaların çoğu yapılamazdı.

#### 4. Hesaplamalı Hafıza Teknolojileri (Computational Memory Technologies)

Hafıza içi hesaplama genel olarak yukarıda belirtilen ilkeleri gerçekleştirebilmesi için hızlı, yüksek yoğunluklu, düşük güçte çalışabilen ve ölçeklenebilir cihazlara ihtiyaç duyar. Bu bellekler arasında Dirençli RAM (RRAM), faz değiştirme belleği (Phase-Change Memory PRAM), manyetoresistif RAM (MRAM) ve ferroelektrik RAM (FeRAM) bulunur. Bu belleklerden bazıları piyasada bulunan ticari teknolojilere yol açmış olsa da, bunlar hala çok yavaştır, sınırlı veri bant genişliğine sahiptir veya von Neumann darboğazını çözmeye önemli ölçüde katkıda bulunmak için çok pahalıdır. [6]

##### 4.1. Dirençli RAM (Resistive RAM)

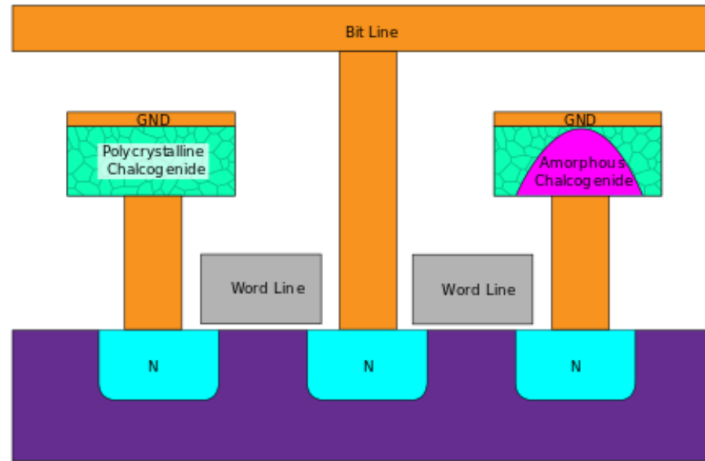
Dirençli RAM (Resistive RAM) verileri elektrik direnci biçiminde depolayan uçucu olmayan bir teknolojidir. Uçucu olmayan hafıza (non-volatile memory) güç kaynağı kesildikten sonra bile bilgileri saklamaya devam edebilen bir bellek türüdür. Dirençli RAM'lerin OxRAM, ReRAM ve RRAM gibi kısaltmaları mevcuttur. Bu araştırma ödevinde RRAM kısaltması kullanılacaktır. Bir RRAM hücresi genellikle bir metal oksit malzeme ile ayrılmış bir üst ve alt elektrottan oluşur. İç direnci düşük ve yüksek olarak iki farklı durum arasında değişiklik gösterir. Bunları binary (ikili) olarak 0 ve 1 temsil eder. Bu farklı durumlar arasında geçiş yapmak RRAM cihazlarının içinde mantıksal işlemler yapılmasını sağlar. Bu cihaza uygun voltaj verilerek mümkün kılınır. [7]



Şekil 8: Bir RRAM cihazının yapısı: (a) düşük direnç durumu (açık), (b) yüksek direnç durumu (kapalı). (c) RRAM cihazının sembolü

#### 4.2. Faz Değişirme Belleği (Phase-Change Memory)

Faz değişim bellekleri uçucu olmayan bir RAM türüdür. PCM, PCME, PRAM, PCRAM, OUM ve CRAM (kalkojenit RAM) olarak da adlandırılırlar. Bu araştırma ödevinde PRAM kısaltması kullanılacaktır. PRAM başta Hewlett Packard tarafından geliştirilmiştir. Hatırlatıcı (memresitor) olarak bilinen bir tekniğe dayanmaktadır. PRAM'ler cihazın üretildiği maddenin durumunu değiştirerek verileri depolar. Bunun için kalkojenit cam olarak adlandırılan bir maddenin özelliklerinden yararlanır. Kalkojenit camının, bir hücreden geçerken ısı üreten akımın geçişiyle polikristalin ve amorf olmak üzere iki durum arasında değiştiği gerçeğini kullanır. Bu, madde iki durum veya faz arasında değiştikçe isim faz değişikliğine yol açar. Hız açısından flash belleklerden daha hızlıdır ve daha dayanıklıdır. Aynı zamanda verilerin silinmeden üstüne yazma da yapılabilir. Üç boyutlu bir fiziksel yapıya sahip olduğundan sabit boyutlu bir çipte bulanabilen kapasitör sayısı artırılabilir. Bu şekilde ise daha az güç kullanılarak daha hızlı bir çalışma mümkün kılınır. [8, 9]

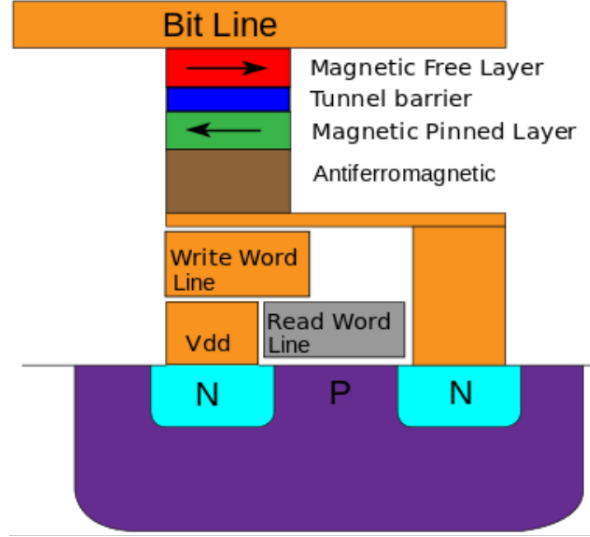


Şekil 9: İki PRAM bellek hücresinin enine kesiti. Bir hücre düşük dirençli kristal haldedir, diğeri ise yüksek dirençli amorf haldedir.

#### 4.3. Manyeto-dirençli RAM (Magnetoresistive RAM)

Manyeto-dirençli RAM (Manyeto-Resistive RAM) uçucu olmayan bir tür RAM'dir. Veriler DRAM'deki gibi elektrik yükü veya akımı olarak değil manyetik depolama elemanları ile depolanmaktadır. Bir metalin Manyeto-Dirençli olarak tanımlanması için manyetik alana yerleştirildiğinde elektrik direncinde hafif bir değişiklik olması gerekir. MRAM'ler ince yalıtım katmanları ile ayrılmış iki ferro-manyetik katmandan oluşur. Bu katmanlardan biri kalıcı mıknatıstır ve diğer katmanın mıknatıslanması ile veriler hafızada tutulmuş olur. MRAM'lerin çalışması manyetik tünel bağlantısı (Magnetic Tunnel Junction MJT) olarak adlandırılmaktadır.

MRAM'lerin avantajlarını sıralayacak olursak. İlk olarak, MRAM bellekler uçucu değildir. Güç kaynağı kesilse bile verilerini koruyabilir. İkinci avantaj ise hız konusundadır. Flash ve diğer bazı teknolojilere kıyasla daha yüksek okuma yazma hızına sahiptir. Diğer RAM türlerine kıyasla nispeten daha az güç tüketir. Son olarak da MRAM'ler sınırsız uzun kullanım ömrü sağlamaktadır. [10]



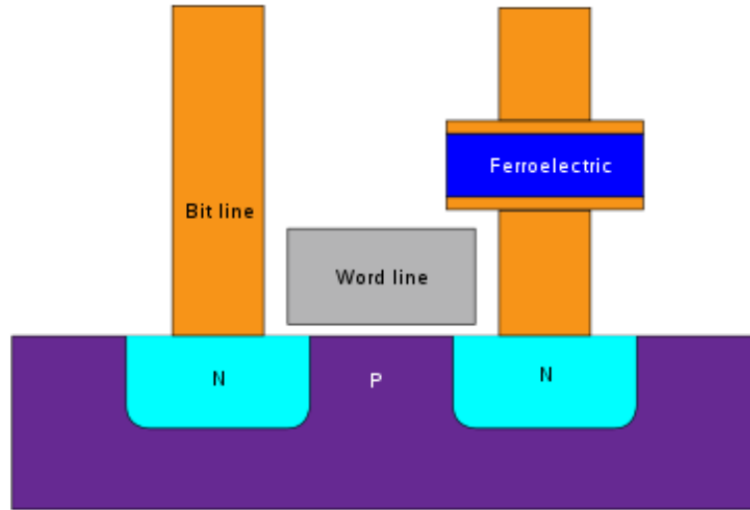
Şekil 10: MRAM hücresinin basitleştirilmiş yapısı

#### 4.4. Ferroelektrik RAM (Ferroelectric RAM)

Ferroelektrik RAM (Ferroelectric RAM) uçucu olmayan güç kesildiğinde bile verilerini koruyabilen bir tür RAM çeşididir. Uçucu olmama özelliğini kazanması için ferroelektirik katman kullanılır. FeRAM, F-RAM veya FRAM gibi kısaltmaları mevcuttur. Bu araştırma ödevinde FeRAM kısaltması kullanılacaktır. FeRAM'lerin +85 °C'de 10 yıldan fazla veri saklama kapasiteleri vardır. Daha düşük sıcaklıklarda bu sayı daha da artar. Flash bellek ile aynı işlevselliği sunar. Buna karşın, flash belleğe göre bazı avantajları şu şekilde sıralanabilir. [11]

- Flash bellekten daha az güç kullanımı sağlar.
- Flash bellekten daha fazla yazma hızına sahiptir.
- Flash bellekten daha dayanıklıdır, daha fazla okuma yazma yapılır.

FeRAM'lerin piyasa dezavantajları arasında ise flash belleklerden daha düşük depolama yoğunlukları, depolama kapasitesi sınırlamaları ve daha yüksek maliyetler verilebilir. Bir başka dezavantajı ise okuma süreci yıkıcıdır yani okunduktan sonra yazma mimarisine ihtiyaç duyar.



Şekil 11: FeRAM hücresinin yapısı

### Sonuç

Hafıza içi hesaplama konulu bu araştırma ödevimizin sonucu olarak bu teknolojilerin von Neumann dar boğazı problemi sonucu ortaya çıktığını söyleyebiliriz. Geleneksel mimari olan von Neumann mimarisinde Şekil 1’de de görüldüğü gibi işlemcini veri işleme kapasitesi ile veri erişimi arasında ciddi bir fark vardır. Bu durumun sonucu olarak büyük verilerin işlenmesi gerektiği yapay zeka uygulamaları ile birlikte RAM kapasitelerinin yetersiz olduğu durumlar ortaya çıkmıştır. Aynı zamanda gerçek zamanlı sonuçlar üreten sistemlere duyulan ihtiyaç sonucu büyük veriler çok hızlı bir şekilde işlenmelidir. Bu işlem birden fazla işlemcinin hafıza birimini paylaşarak paralel işlemlerle gerçekleştirebilmesiyle ve Şekil 2’de de gördüğümüz gibi hafıza birimlerinde yapılan hesaplamalarla da gerçekleştirilmektedir. Bu uygulamalarda büyük data’lar işleneceğinden hafıza içi işleme teknolojisi ile birlikte, tek bir mantıksal parça gibi çalıştırılan ama birden fazla donanımı birbirine bağlayarak kapasitesini arttırabilen bir mimariye sahip olan IMDG’ler kullanılmaktadır. IMDG tabanlı dağıtılmış veri işleme mimarisinde verileri taşımak yerine algoritmaları taşıma stratejisinden yararlanılır. Burada algoritmalar verilerin bulunduğu düğüme gönderilir. Böylece veri taşıma yükünden kurtarılmış olur. Bununla birlikte geliştirilmiş olan hafıza birimi cihazları arasında RRAM, PRAM, MRAM, FeRAM gibi hafıza birimleri bulunmaktadır. Bu birimler bazı algoritmaların işlenmesini desteklemekte verinin hafıza içinde işlenerek tahmin algoritmaları ile predict edilmesi sağlanmakta ve çok daha hızlı sonuçların elde edilmesi sağlanmaktadır. Bu araştırma ödevinde bu sonuçlara varılmakla birlikte bu RAM’ler ayrıntılı bir şekilde açıklanmaya çalışılıp veri üzerindeki işlem anlatılmıştır.

## **Kazanımlar**

Bu arařtırmada hafıza ii hesaplama teknolojisi, geliřtirilme sebepleri, zellikleri, kullanılan teknolojilerle birlikte ğrenilmiřtir. Bu teknolojinin temel ilkeleri olarak yksek hız ve leklenebilirlik, gerek zamanlı ngrler, geniř kullanım alanları ve teknolojinin etkinleřtirilmesi hakkında bilgi sahibi olunmuřtur. Aynı zamanda bu teknolojide kullanılan IMDG mimarisi detaylı olarak arařtırılıp ğrenilmiřtir. Btn bunlarla birlikte RRAM, PRAM, MRAM, FeRAM gibi cihazların mimarileri arařtırılmıřtır. Bu cihazların mekanik yapısı veri depolama ve iřleme yapıları ğrenilmiřtir.

## KAYNAKÇA

1. Pekel I., (2007) Von Neumann Mimarisi <https://e-bergi.com/y/von-neumann-mimarisi/> 17.01.2021
2. Wikipedia contributors. (2020, Eylül 8). In-memory processing. In *Wikipedia, The Free Encyclopedia*. Ocak 17, 2021, [https://en.wikipedia.org/wiki/In-memory\\_processing](https://en.wikipedia.org/wiki/In-memory_processing)
3. Wikipedia contributors. (2021, Ocak 15). ACID. In *Wikipedia, The Free Encyclopedia*. Retrieved, Ocak 18, 2021, <https://en.wikipedia.org/wiki/ACID>
4. Guroob, Abdo. (2017). Big Data-based In-Memory Data Grid (IMDG) Technologies: challenges of implementation by analytics tools. *CSI Communications*. 41. 27-29.
5. Krivaa,K., (Kasım 24, 2019), In-Memory Computing: A Comprehensive Guide (2020), <https://www.gigaspace.com/blog/in-memory-computing/>
6. Ielmini, D., & Wong, H.-S. P. (2018). *In-memory computing with resistive switching devices. Nature Electronics*, 1(6), 333–343. doi:10.1038/s41928-018-0092-2
7. Shirinzadeh, S., & Drechsler, R. (2020). In-Memory Computing. doi:10.1007/978-3-030-18026-3
8. electronic-notes, Phase Change Memory, P-RAM, [https://www.electronic-notes.com/articles/electronic\\_components/semiconductor-ic-memory/pram-phase-change-memory-storage.php](https://www.electronic-notes.com/articles/electronic_components/semiconductor-ic-memory/pram-phase-change-memory-storage.php) 20.01.2020
9. Rouse, M., (Mayıs 2017), phase-change memory (PCM) <https://whatis.techtarget.com/definition/phase-change-memory-PCM> 21.01.2020
10. electronic-notes, What is MRAM Memory Technology [https://www.electronic-notes.com/articles/electronic\\_components/semiconductor-ic-memory/mram-magnetoresistive.php](https://www.electronic-notes.com/articles/electronic_components/semiconductor-ic-memory/mram-magnetoresistive.php) 21.01.2020
11. Wikipedia contributors. (2021, Ocak 16). Ferroelectric RAM. In *Wikipedia, The Free Encyclopedia*. Ocak 22, 2021 [https://en.wikipedia.org/wiki/Ferroelectric\\_RAM](https://en.wikipedia.org/wiki/Ferroelectric_RAM)
12. In-Memory Computing, Ocak 25, 2017, <https://www.techopedia.com/definition/28539/in-memory-computing> 18.01.2021
13. What is In-Memory Computation?, <https://hazelcast.com/glossary/in-memory-computation/> 18.01.2021
14. In-Memory Processing, <https://hazelcast.com/glossary/in-memory-processing/> 18.01.2021
15. What Is an In-Memory Data Grid?, <https://hazelcast.com/glossary/in-memory-data-grid/> 18.01.2021
16. Zhou, M., & Feng, D. (2018). Application of In-Memory Computing to Online Power Grid Analysis. *IFAC-PapersOnLine*, 51(28), 132–137. doi:10.1016/j.ifacol.2018.11.690
17. M.Mills,(2020). What Is In-memory Computing, is it the Future of Computing? retrieved by <https://itigic.com/what-is-in-memory-computing-is-it-future/>
18. Lapedus M. (2019). In-Memory Vs. Near-Memory Computing. <https://semiengineering.com/in-memory-vs-near-memory-computing/> 20.01.2021
19. R. Gauchi et al., "Memory Sizing of a Scalable SRAM In-Memory Computing Tile Based Architecture," 2019 IFIP/IEEE 27th International Conference on Very Large Scale Integration (VLSI-SoC), Cuzco, Peru, 2019, pp. 166-171, doi: 10.1109/VLSI-SoC.2019.8920373.

20. Yan, B., Li, B., Qiao, X., Xue, C.-X., Chang, M.-F., Chen, Y., & Li, H. (Helen). (2019). RRAM Based In-Memory Computing: From Device and Large-Scale Integration System Perspectives. *Advanced Intelligent Systems*, 1900068. doi:10.1002/aisy.201900068
21. Wikipedia contributors. (2021, Ocak 20). Resistive random-access memory. In *Wikipedia, The Free Encyclopedia.*, Ocak 21, 2021, [https://en.wikipedia.org/wiki/Resistive\\_random-access\\_memory](https://en.wikipedia.org/wiki/Resistive_random-access_memory)
22. Wikipedia contributors. (2021, Ocak 8). Non-volatile memory. In *Wikipedia, The Free Encyclopedia.*, Ocak 21, 2021, [https://en.wikipedia.org/wiki/Non-volatile\\_memory](https://en.wikipedia.org/wiki/Non-volatile_memory)
23. Wikipedia contributors. (2021, Ocak 21). Flash memory. In *Wikipedia, The Free Encyclopedia.*, Ocak 21, 2021 [https://en.wikipedia.org/wiki/Flash\\_memory](https://en.wikipedia.org/wiki/Flash_memory)
24. Wikipedia contributors. (2021, Ocak 11). Static random-access memory. In *Wikipedia, The Free Encyclopedia.*, Ocak 21, 2021, [https://en.wikipedia.org/wiki/Static\\_random-access\\_memory](https://en.wikipedia.org/wiki/Static_random-access_memory)
25. Wikipedia contributors. (2021, Ocak 19). Magnetoresistive RAM. In *Wikipedia, The Free Encyclopedia.*, Ocak 21, 2021, [https://en.wikipedia.org/wiki/Magnetoresistive\\_RAM](https://en.wikipedia.org/wiki/Magnetoresistive_RAM)