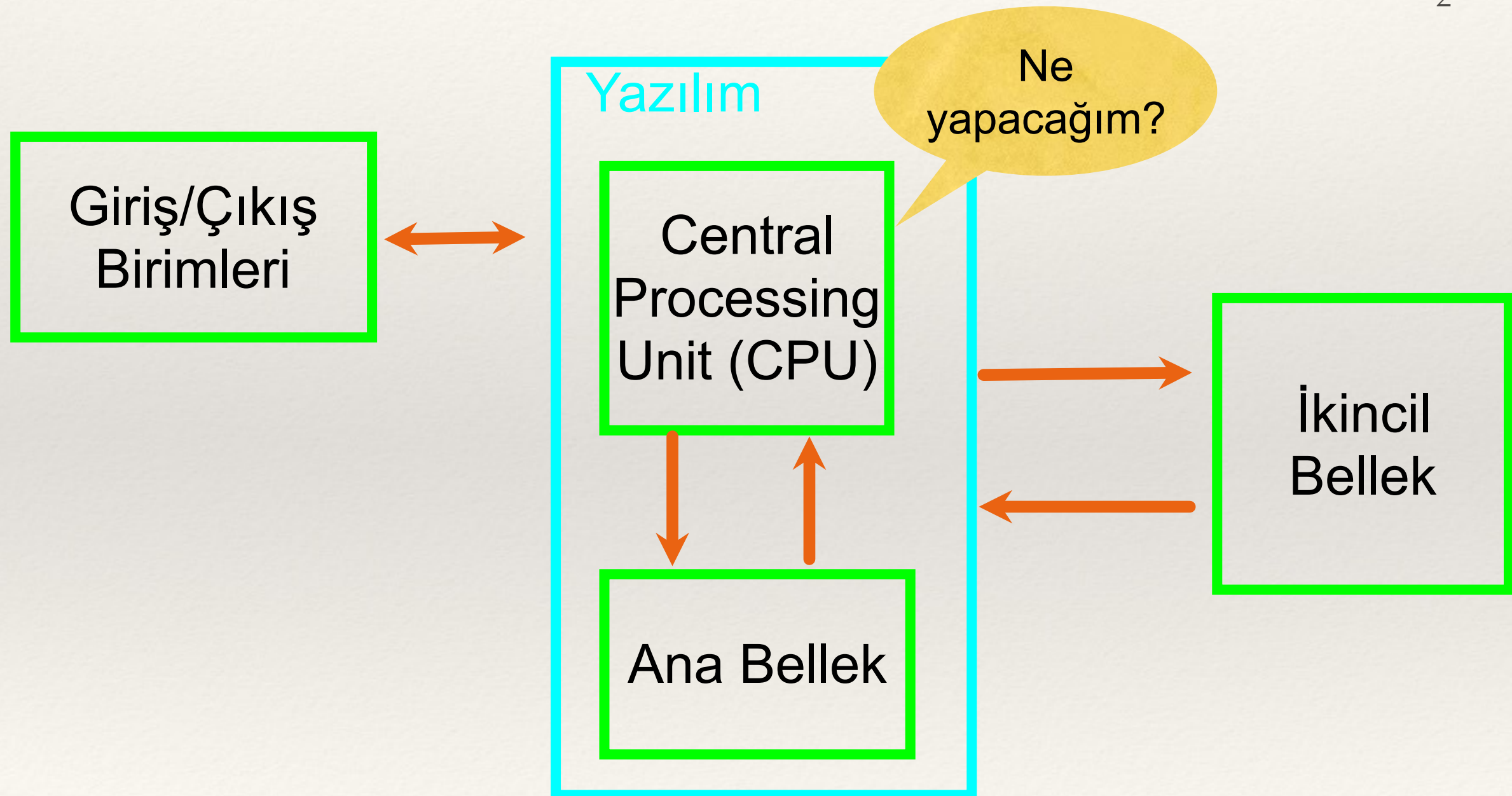

Bilgisayar Programlama 1

DERS 6: DOSYA İŞLEMLERİ (YAZMAK)

DR. HÜSEYİN BAHTİYAR

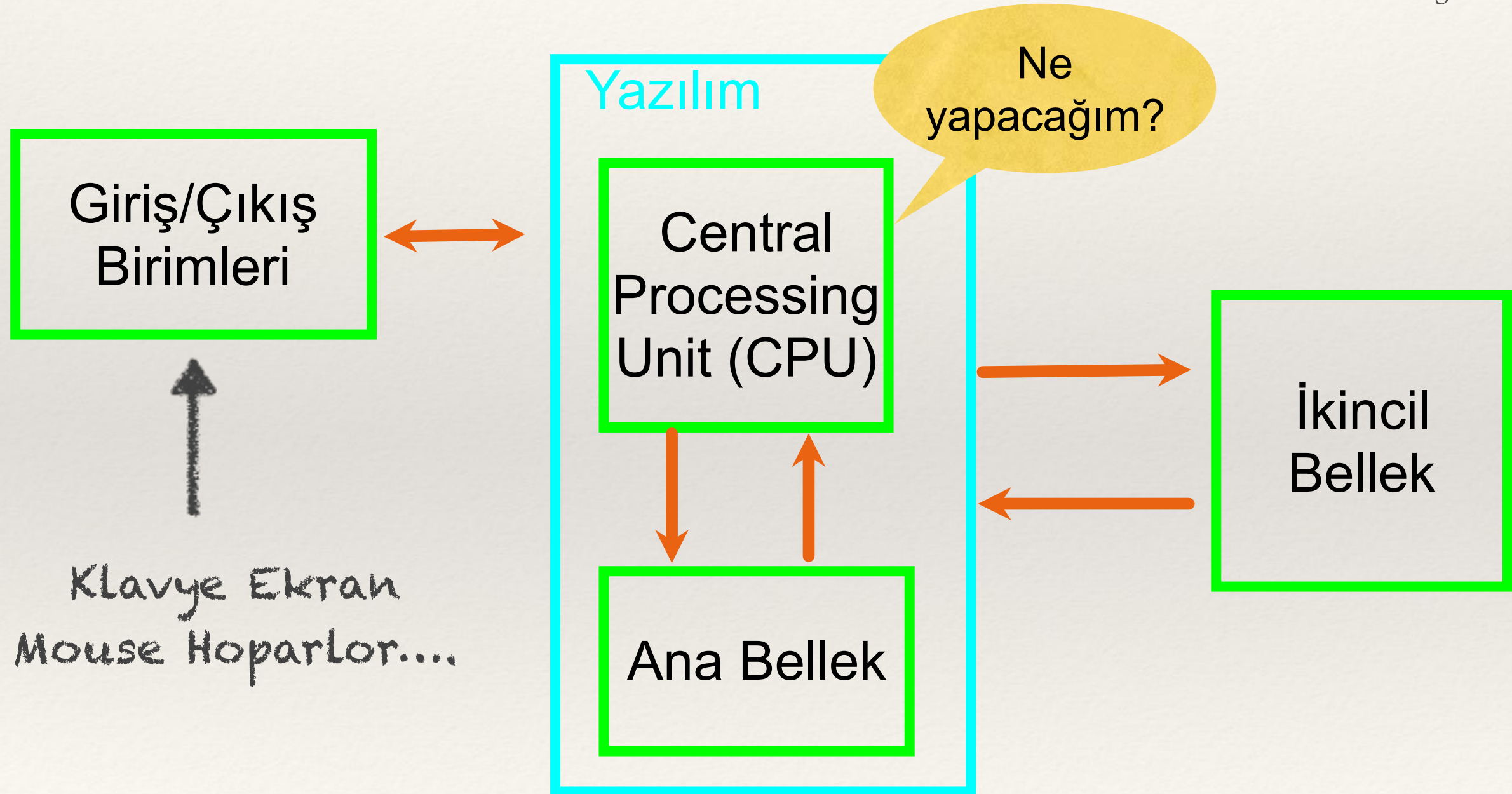
Donanım Mimaris

2



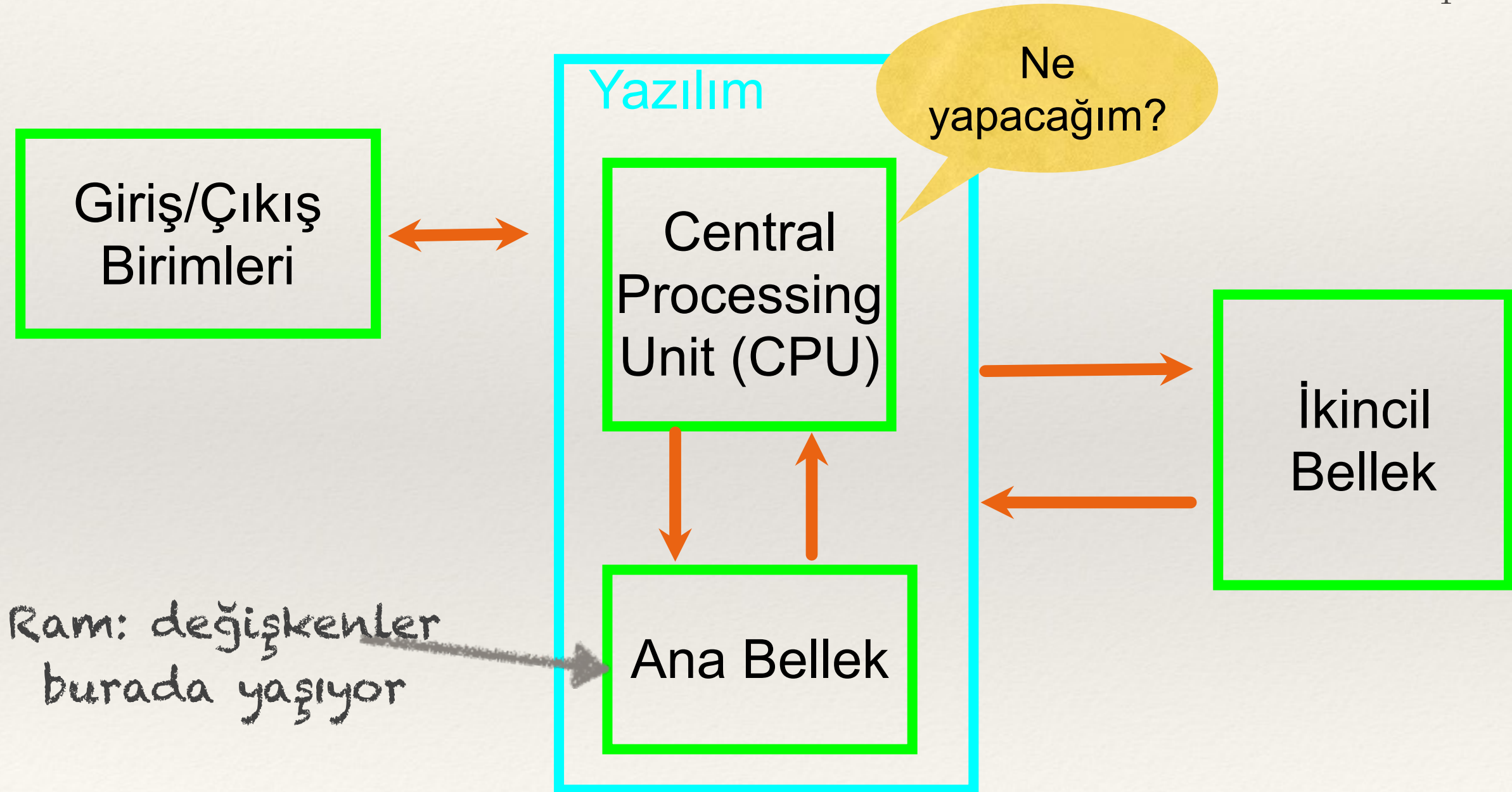
Donanım Mimarisi

3



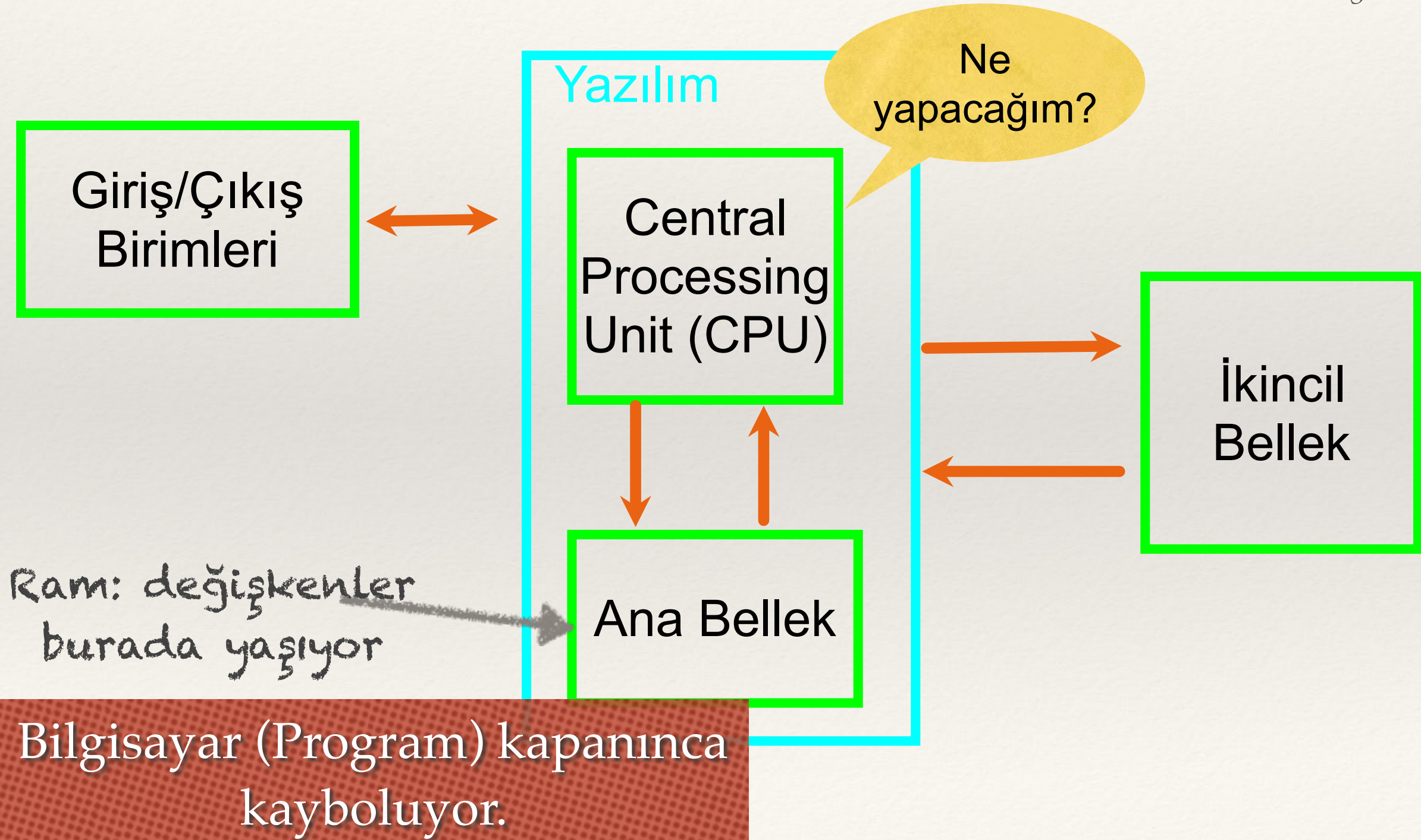
Donanım Mimarisi

4



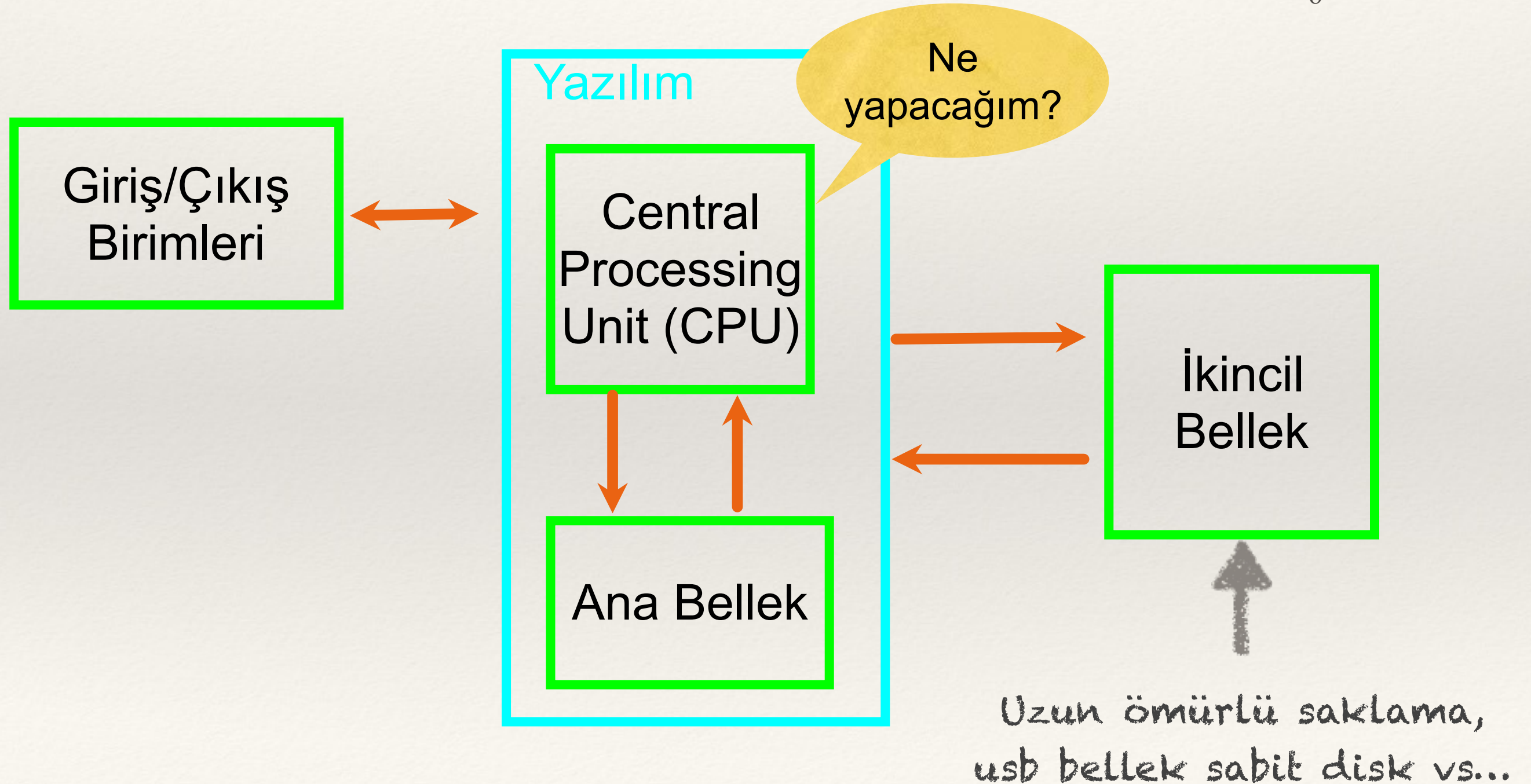
Donanım Mimaris

5



Donanım Mimarisi

6



Dosya işlemleri için

1. Öncelikle dosya kütüphanesini eklemeliyiz

```
#include<fstream>
```

2. Dosya işlemi yapacağımız değişkeni tanımlamalıyız

```
ofstream dosya_yaz;
```

3. Bilgisayar akıllı bir alet olmadığından sadece dosya_yaz isminde bir değişken ile dosya ismi yapacağımızı, sanmaktadır. Dosyayı açmak için.

```
dosya_yaz.open("dosyaadi");
```

ifstream? fstream? ofstream?

Komut

İşlem

ifstream

okumak için (input)

ofstream

yazmak için (output)

fstream

hem okumak hem yazmak için

. operatörü?

- ❖ . operatörü kullandığımız sınıfa bağlı alt fonksiyonlara erişmemizi sağlayan operatördür.
- ❖ Kısacası `ofstream`'e özel fonksiyonlara erişmek için öncelikle `ofstream` yapısında bir değişken tanımlayıp `değişken_adi.fonksiyon()` şeklinde alt fonksiyonlarına erişebiliriz.

fx Public member functions

(constructor)	Construct object (public member function)
open	Open file (public member function)
is_open	Check if file is open (public member function)
close	Close file (public member function)
rdbuf	Get stream buffer (public member function)
operator= C++11	Move assignment (public member function)
swap C++11	Swap internals (public member function)

Dosya açmaya çalışalım

```
#include <iostream>
#include <fstream>
using namespace std;
int main() {
    ofstream myfile;
    myfile.open("deneme.txt");
    if(myfile.is_open()){
        cout<<"yazmak icin actik"<<endl;
    }
    else{
        cout<<"hataaaaaaa!"<<endl;
    }
    return 0;
}
```

fstream headerını çağırdık
buradaki fonksiyonları
kullanacağız dedik.

yazmak için (ofstream)
kullandık değişkenimiz myfile

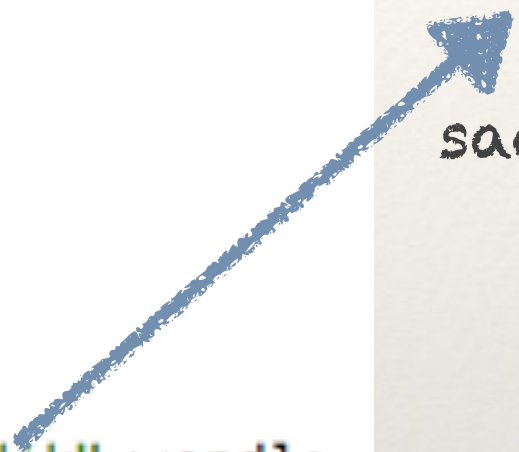
Eğer Dosya Yoksa

- ❖ ifstream ile aynı şekilde açıyoruz açma kısmında bir değişiklik yok.
- ❖ Eğer dosya yoksa deneme.txt isimli yeni bir dosya yaratacak.
- ❖ Şimdi içine birşeyler yazalım.

Dosya içerisine yazmak

```
#include <iostream>
#include <fstream>
using namespace std;
int main() {
    ofstream myfile;
    myfile.open("deneme.txt");
    if(myfile.is_open()){
        myfile<<"dosyanin icine yazdik"<<endl;
        myfile<<"bisey daha yazdik"<<endl;
    }
    else{
        cout<<"hataaaaaaaa!"<<endl;
    }
    return 0;
}
```

cout<< yapısına benzer
şekilde yazıyoruz
sadece cout<< yerine dosya
değişkenini yazıyoruz.



Örnek

- ❖ Şimdi bir dosya oluşturalım.
- ❖ İlk satırda adımız soyadımız yazsın, alt satırda ise numaramız yazsın.

Örnek

- ❖ Şimdi bir dosya oluşturalım.
- ❖ İlk satırda adımız soyadımız yazsın, alt satırda ise numaramız yazsın.

```
#include <iostream>
#include <fstream>
using namespace std;
int main() {
    ofstream myfile;
    myfile.open("deneme.txt");
    if(myfile.is_open()){
        myfile<<"Huseyin Bahtiyar"<<endl;
        myfile<<"numaram yok :("<<endl;
    }
    else{
        cout<<"hataaaaaaa!"<<endl;
    }
    return 0;
}
```


Örnek Devam

- ❖ Dikkat ettiysek dosyanın içerisindeki önceki yazıklarımız silindi.
- ❖ Peki her ikisini nasıl koruyabiliriz?

```
#include <iostream>
#include <fstream>
using namespace std;
int main() {
    fstream myfile;
    myfile.open("deneme.txt", ios::out | ios::app);
    if(myfile.is_open()){
        myfile<<"Huseyin Bahtiyar"<<endl;
        myfile<<"numaram yok :("<<endl;
    }
    else{
        cout<<"hataaaaaaa!"<<endl;
    }
    return 0;
}
```

if/of stream yerine fstream kullanıyoruz

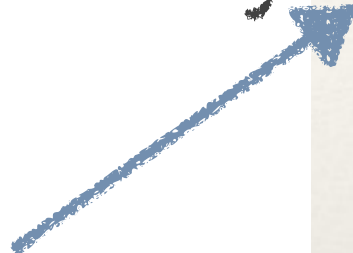
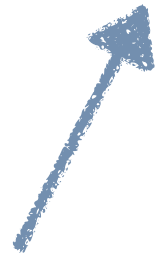
Girdi çıktı için (out) çıktı kısmını kullanıyoruz ve üzerine yazma (append) işlemini yapacağımızı söylüyoruz.

Hem okumak, hem yazmak

```
#include <iostream>
#include <fstream>
using namespace std;
int main() {
```

```
    fstream myfile;
    myfile.open("deneme.txt", ios::in| ios::out | ios::app);
    if(myfile.is_open()){
        myfile<<"Huseyin Bahtiyar"<<endl;
        myfile<<"numaram yok :("<<endl;
    }
    else{
        cout<<"hataaaaaaaa!"<<endl;
    }
    return 0;
}
```

Okuyacağım Yazacağım Dosya varsa üzerine yazar



Dosyayı Kapamak

- ❖ Şu ana kadar dosya işlemleri ile ilgili yaptığımız tüm örneklerde bir eksiklik vardı!
- ❖ İşlemimizi bitirdikten sonra dosyayı hiç kapamadık.
- ❖ Dosya kapama işlemi **.close()** ile yapılır.

```
#include <iostream>
#include <fstream>
using namespace std;
int main() {
    fstream myfile;
    myfile.open("deneme.txt", ios::in| ios::out | ios::app);
    if(myfile.is_open()){
        myfile<<"Huseyin Bahtiyar"<<endl;
        myfile<<"numaram yok :("<<endl;
    }
    else{
        cout<<"hataaaaaaa!"<<endl;
    }
    myfile.close();
    return 0;
}
```

Örnek

- ❖ Klavyeden ad, soyad, doğum tarihi bilgilerini girdi alıp bunu

ad: girilen isim soyad: girilen soyad

doğum tarihi: girilen doğum tarihi

- ❖ şeklinde dosyaya yazan program yazalım.

Örnek 2

- ❖ Şimdi bu dosyayı okuyup ekrana yazan program yazalım :)
- ❖ ister getline ile isterseniz >> operatörü ile yapabilirsiniz.

Kaynaklar

- ▶ <http://www.cplusplus.com/doc/tutorial/>
- ▶ www.py4e.com
- ▶ C++ How to program (Deitel)
- ▶ Ferhat Özok Bilgisayar Programlama I ders notları