Yapay zekâ bir robotun hareketlerini nasıl yönlendirir?

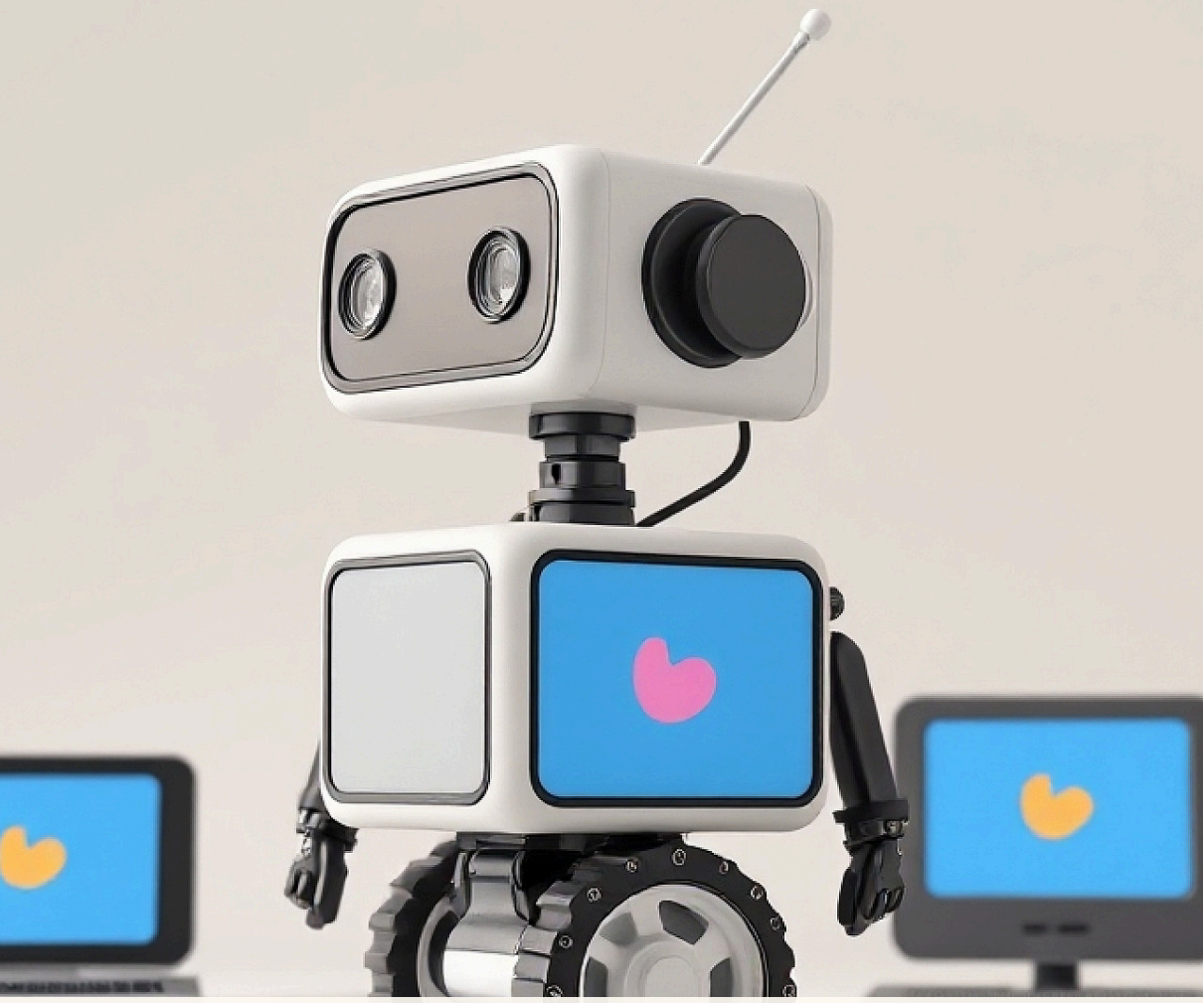


www.yzokul.com.tr

Yapay zekâ bir robotun hareketlerini nasıl yönlendirir?



Yapay zekânın, yani YZ'nin bir robotun hareketlerini nasıl yönlendirdiğini bilmek istiyorsun. Kısa yanıt: kameralar ve sensörler çevre hakkında veri toplar, yazılım robotun konumunu, engelleri ve hedefi değerlendirir, bir hareket planı yapılır ve motorlara komutlar gönderilir. YZ bu adımların bazılarında yardımcı olabilir, ama mutlaka hepsinde değil. Bu kitap her parçayı adım adım açıklıyor.



Bir robot; gövdesi, onu hareket ettiren motorları ve onu denetleyen bir bilgisayarı olan bir makinedir. Kendine ait bir zihni yoktur; yaptığı her şey talimatlardan ve verilerden gelir. Bilgisayar, robota ne yapacağını söyleyen programlar bütünü olan yazılımı çalıştırır. Motorlar hareket üretmek için tekerlekleri veya eklemleri döndürür. Yazılım ve veri olmadan robot hareket edemez.



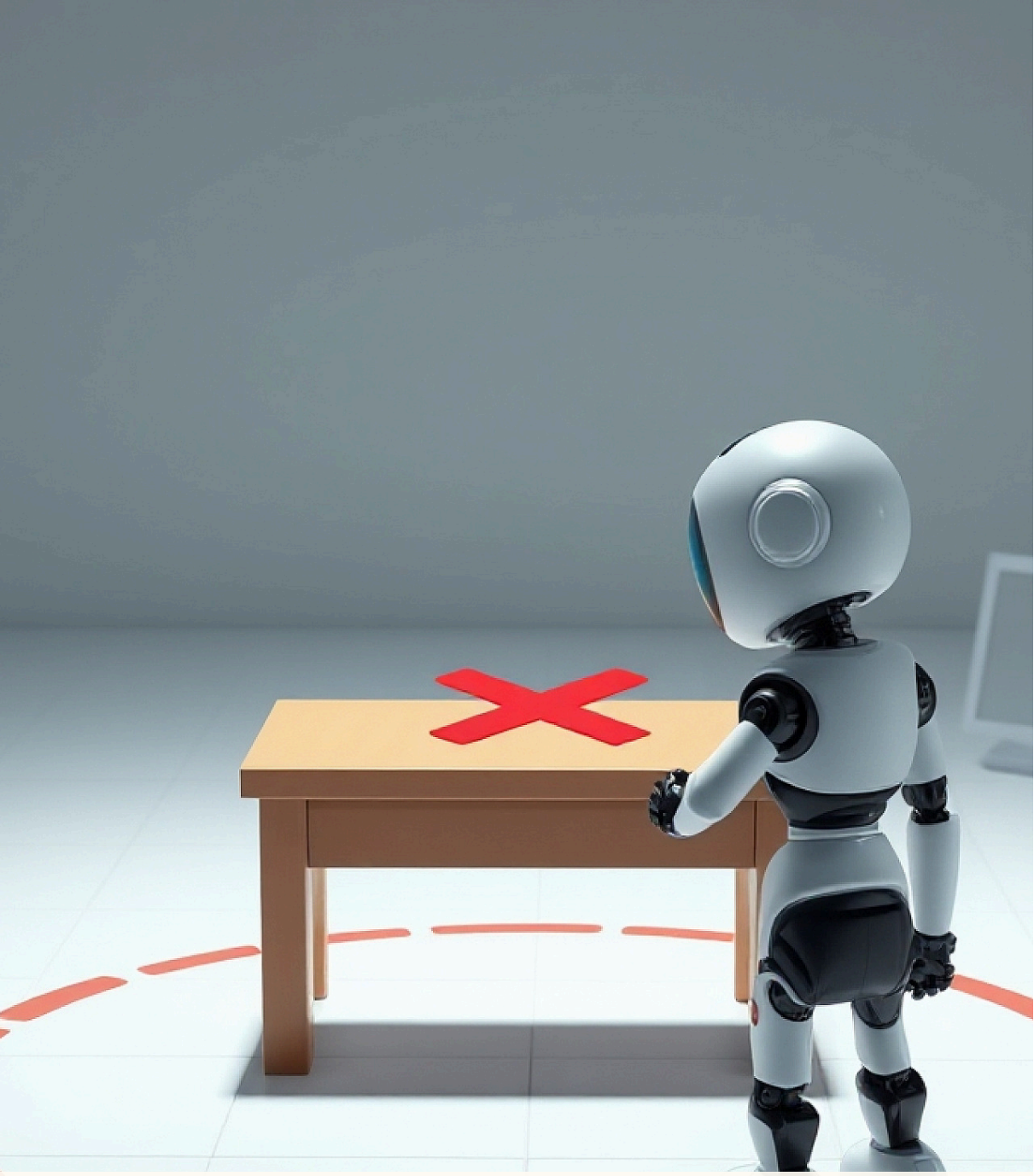
Kameralar ve sensörler robotun gözleri ve kulaklarıdır. Çevre hakkında, örneğin yakında ne olduğu ve ne kadar uzakta olduğu gibi bilgileri toplarlar. Bir kamera görüntüler yakalar, mesafe sensörleri gibi sensörler ise nesnelerin ne kadar yakın olduğunu ölçer. Bu veriler işlenmek üzere robotun bilgisayarına gönderilir. İyi veri, her hareketin temelidir.



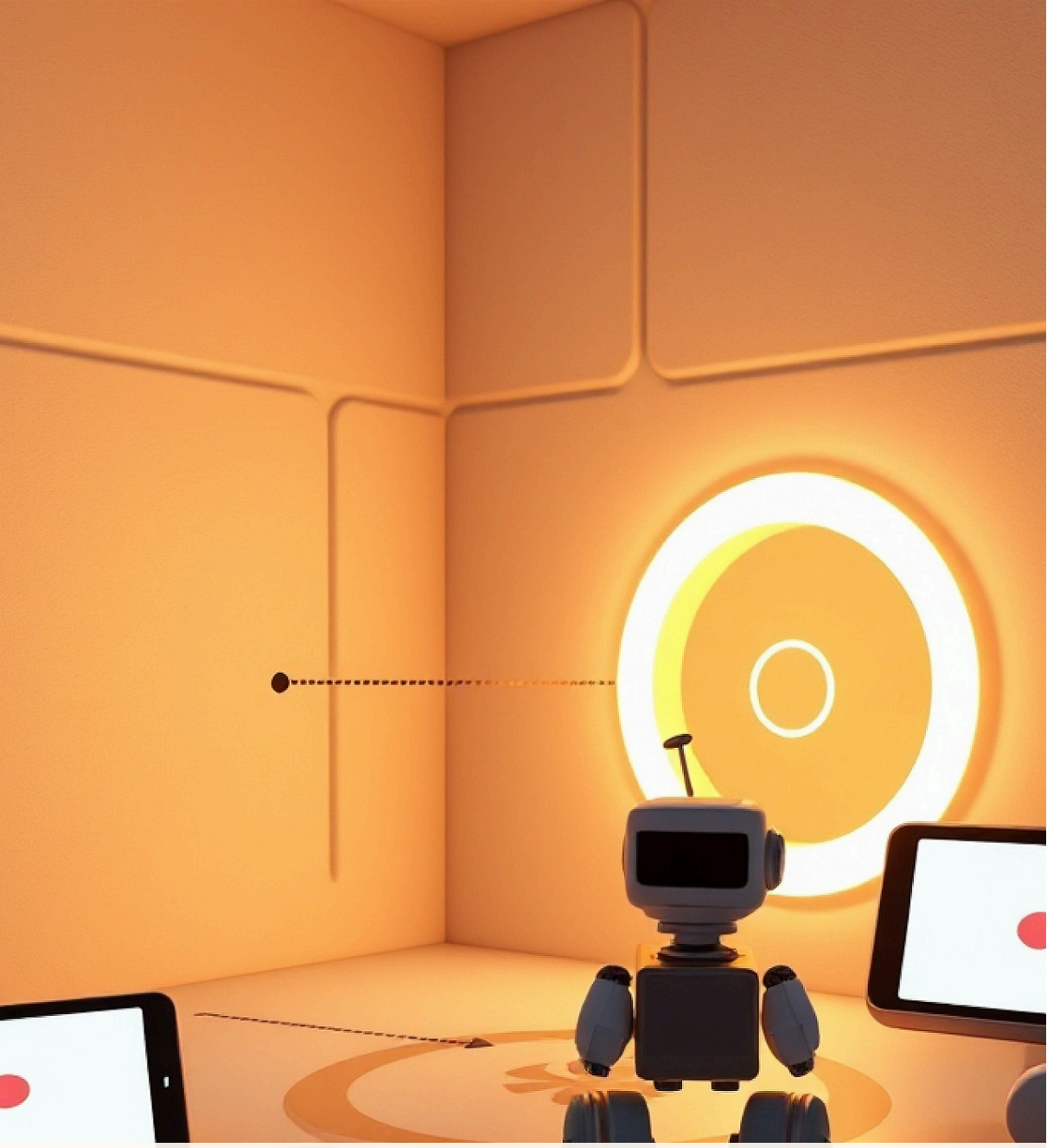
Ham sensör verisi, robotun çevresindeki dünyanın basit bir resmine dönüştürülür. Bu resim, robotun yolundaki bir masa da dahil olmak üzere nesnelerin nerede olduğunu gösterir. Yazılım, ortamın bir haritasını veya modelini oluşturmak için veriyi işler. Bu iç resim, robotun çevresini anlamasına yardımcı olur. Bir fotoğraf değil, nesnelerin ve konumlarının bir temsilidir.



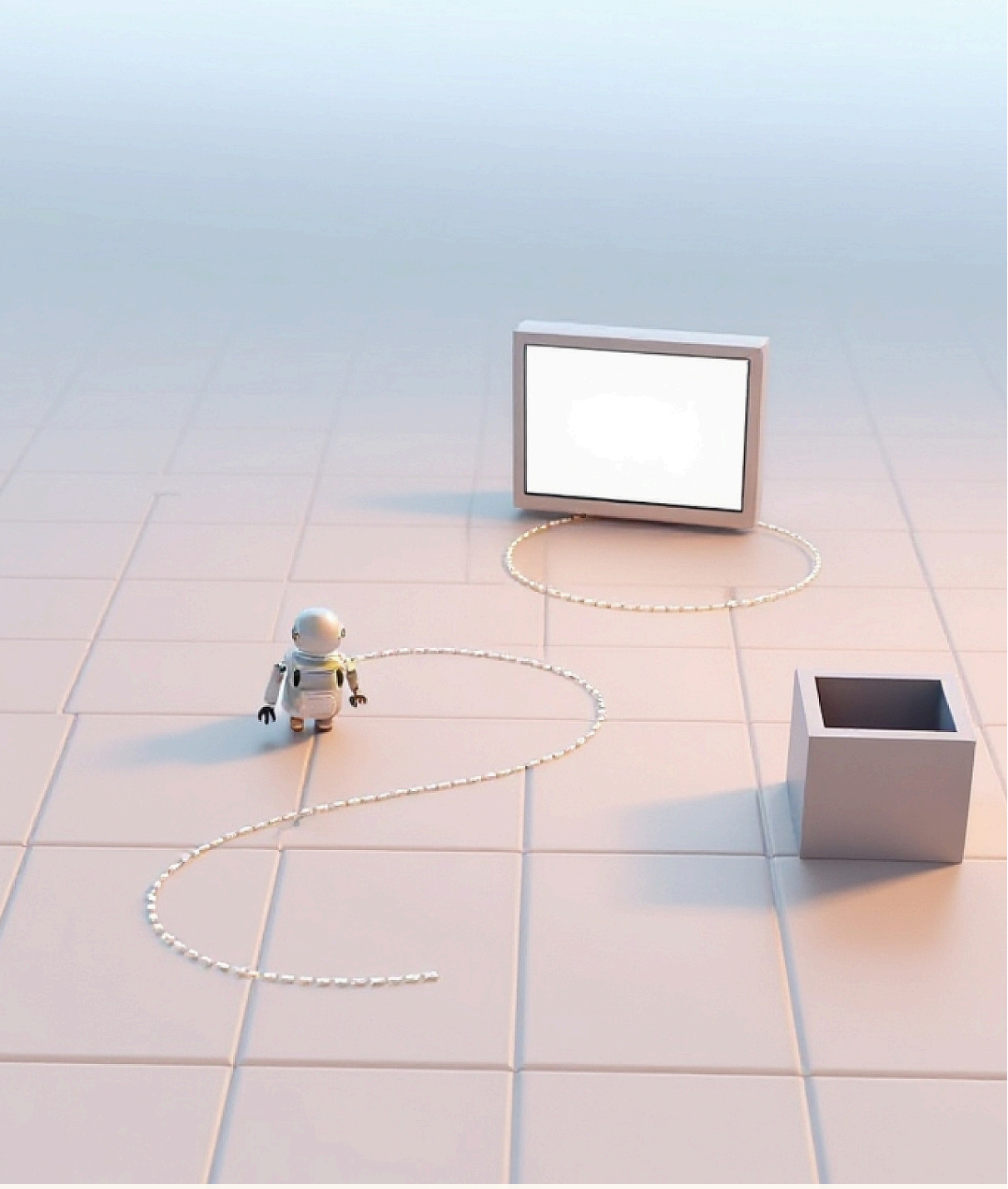
Yazılım, robotun kendisinin nerede durduğunu hesaplar. Kendi konumunu bilmeden robot hangi yöne gideceğine karar veremez. Bilinen işaret noktalarına veya başlangıç noktasına göre konumunu tahmin etmek için sensör verisini kullanır. Bu sürece konum belirleme denir. Konumunu bilmek, hedefe giden bir yolu planlamak için gereklidir.



Yazılım, resmi masa gibi engeller açısından denetler. Bunları robotun çarpmaması gereken yerler olarak işaretler. Engeller, mesafe ölçen sensörlerle veya kamera görüntüleri analiz edilerek saptanabilir. Yazılım, bunların boyutunu ve konumunu belirler. Engellerden kaçınmak güvenli hareket için çok önemlidir.



Robotun ayrıca bir hedefe, örneğin odanın karşısındaki bir noktaya ihtiyacı vardır. Yazılım, hedefi robotun mevcut konumuyla karşılaştırır. Hedef bir insan tarafından verilebilir veya bir program tarafından belirlenebilir. Hem hedefi hem de robotun konumunu bilmek, yazılımın hangi yöne hareket edeceğini belirlemesini sağlar. Hedef, planlama sürecine yön verir.



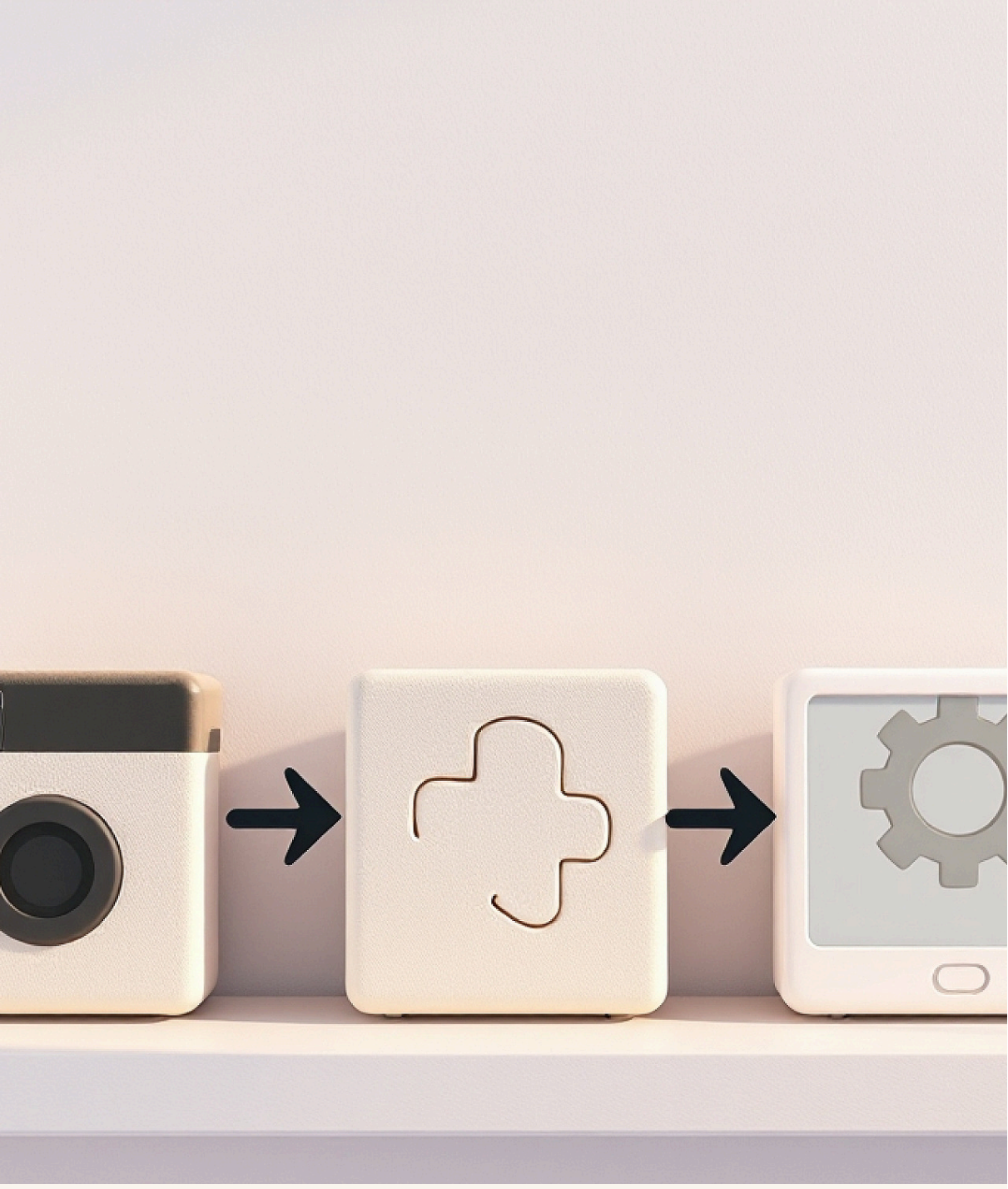
Yazılım basit bir plan oluşturur: robottan hedefe, engellerden kaçınan bir yol. Bu plan, nasıl hareket edileceğine dair karardır. Düz bir çizgi veya bir dizi dönüş olabilir. Planlama adımı, robotun boyutunu ve mevcut alanı dikkate alır. Sonuç, bir dizi ara nokta veya yöndür.



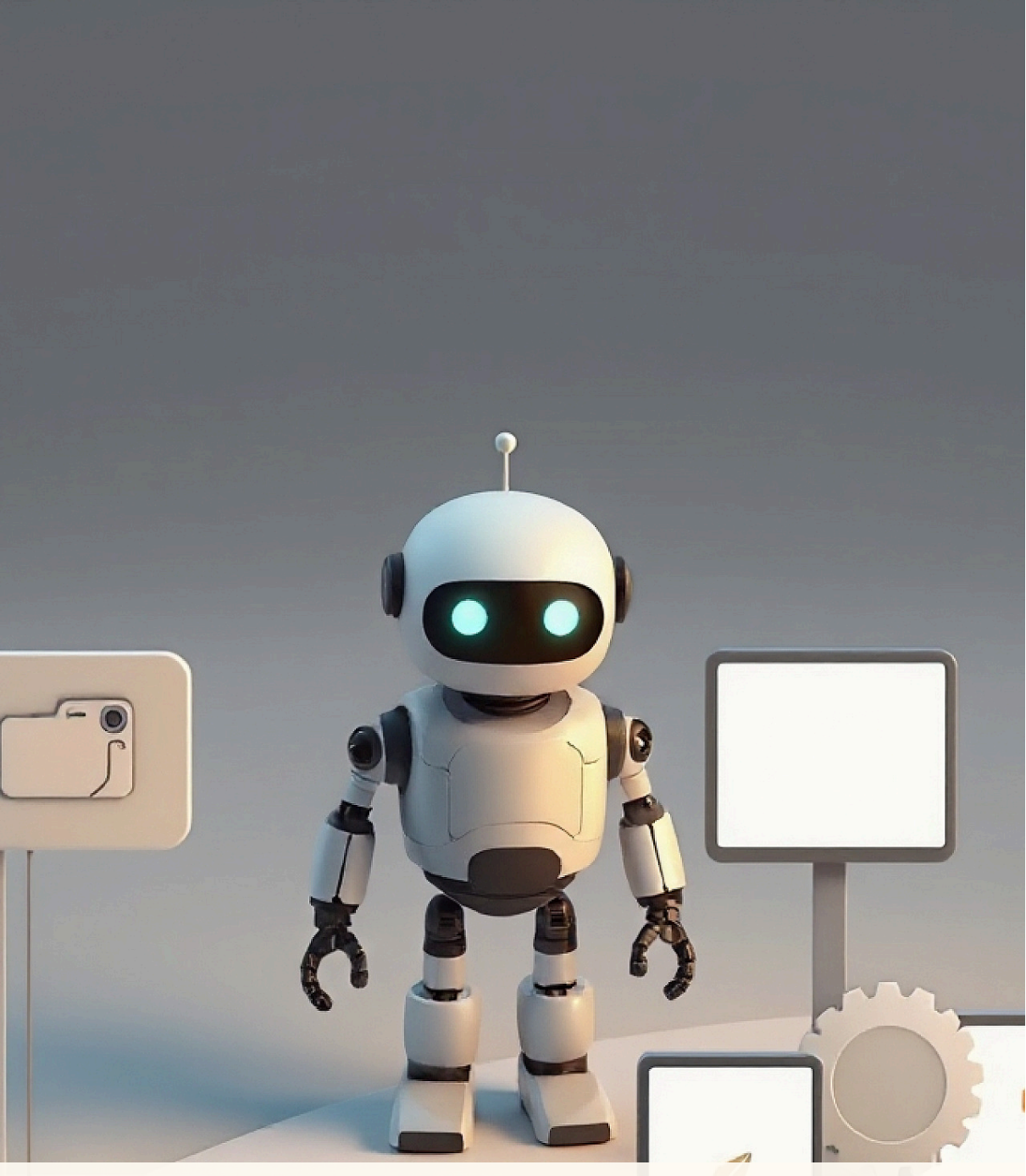
Plan, motorlara gönderilen komutlara dönüştürülür. Motorlar ardından tekerlekleri veya eklemleri döndürür ve robot hareket etmeye başlar. Bu komutlar, her motorun ne kadar hızlı ve hangi yönde döneceğini belirtir. Motor denetimi kısmı, planı elektrik sinyallerine çevirir. Bu, hareket planının uygulanmasıdır.



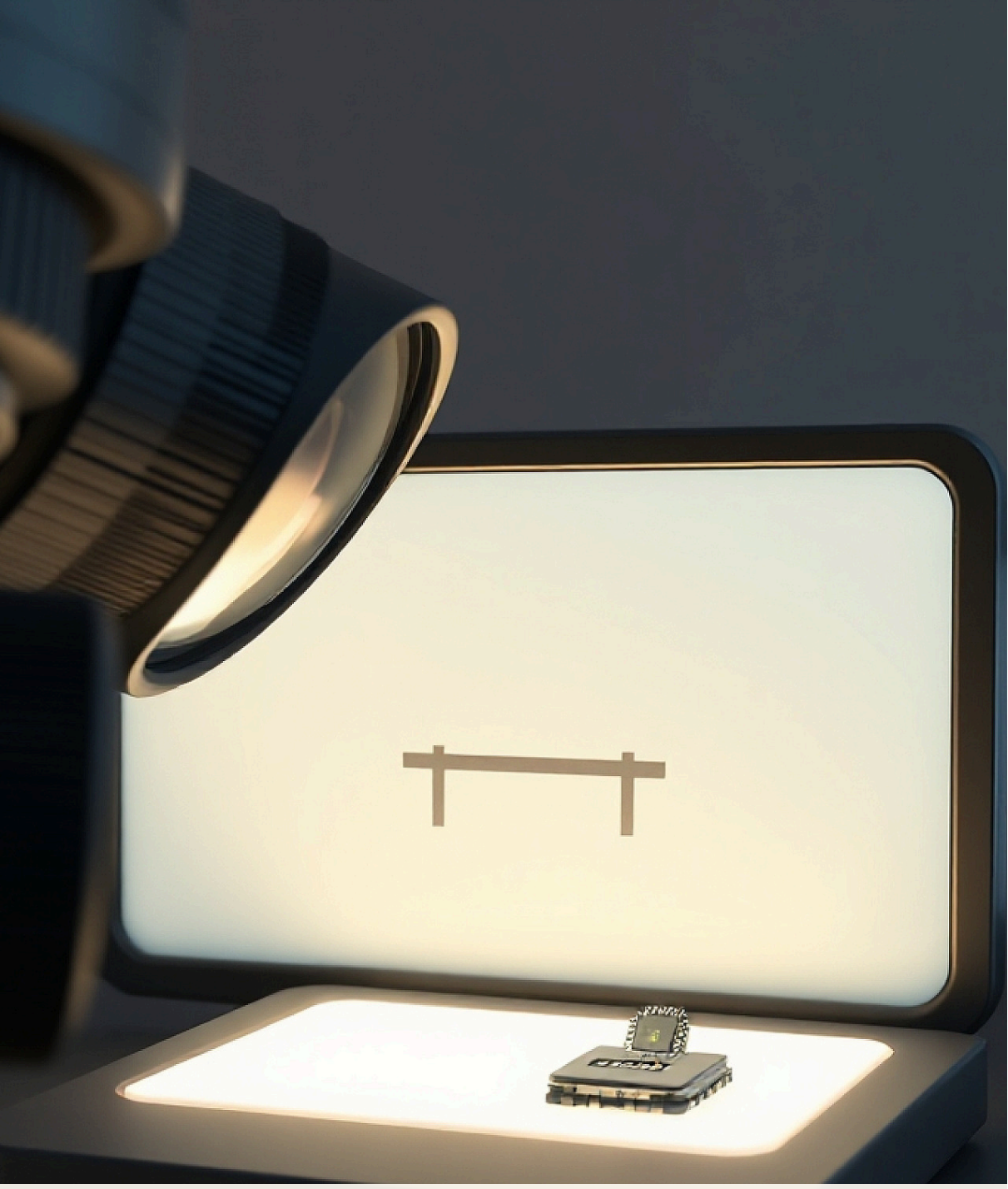
Yolun bir masa tarafından kapatıldığını düşün. Robotun sensörleri masayı algılar ve yazılım etrafından dolaşan bir yol planlar. Plan, biraz dönüp sonra ilerlemeyi söyleyebilir. Motor komutları buna göre ayarlanır: bir tekerlek diğerinden daha hızlı döner. Robot yön değiştirir ve masaya çarpmadan yanından geçer.



Algılama, planlama ve motor denetimi üç farklı iştir. Her iş, yazılımın kendi parçası tarafından yapılabilir. Algılama veri toplar, planlama yola karar verir ve motor denetimi komutları uygular. Bunları ayrı tutmak, sistemi tasarlamayı ve onarmayı kolaylaştırır. Bir sıra içinde birlikte çalışırlar.



Her aşamada yapay zekâ kullanmak zorunlu değildir. Bazı robotlar basit kurallarla iyi hareket eder ve YZ yalnızca üç görevin bazılarında yardımcı olabilir. Örneğin YZ, kamera görüntülerindeki engelleri tanımak için kullanılabilirken planlama basit bir algoritma kullanabilir. Seçim, görevin karmaşıklığına bağlıdır. YZ bir araçtır, bir zorunluluk değildir.



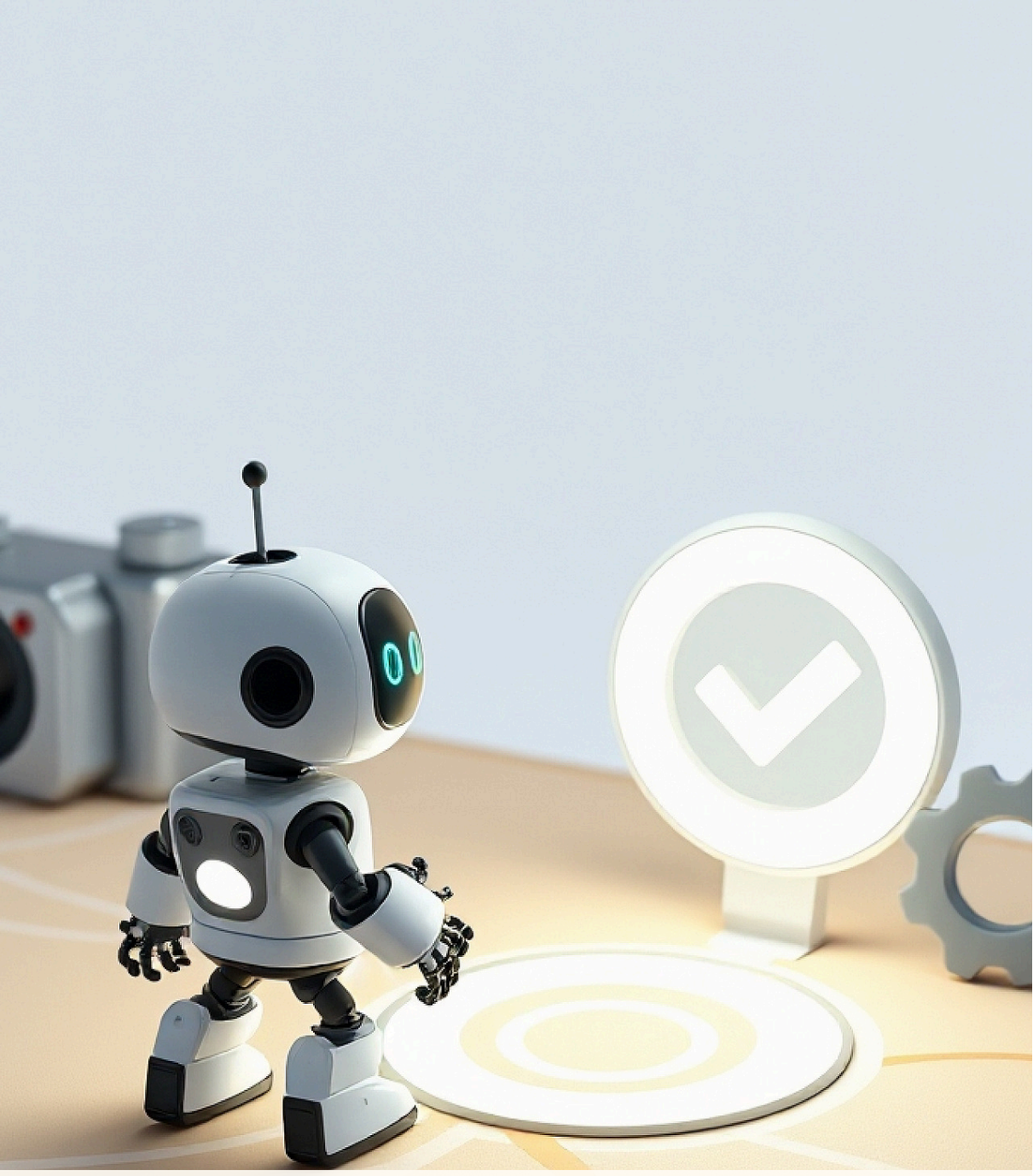
Yaygın bir yanılgı, YZ'nin kendisinin kameradan baktığını düşünmektir. Veriyi kamera ve sensörler toplar; YZ'li ya da YZ'siz yazılım bu veriyi kullanır. YZ veriyi yorumlamaya yardımcı olabilir, ama gören sensörlerdir. Yazılım, ham bilgiyi kullanılabilir bir biçime dönüştürür. YZ sensör değildir.



Bir başka yanılgı, YZ'nin motorları doğrudan döndürdüğüne inanmaktır. Plan komutlara dönüşür ve motor denetimi kısmı bunları uygular. YZ planın oluşturulmasına yardımcı olabilir, ama gerçek motor sinyalleri denetim yazılımı tarafından üretilir. Motorlar elektrik komutlarına yanıt verir, doğrudan YZ'ye değil. Motor denetimi ayrı bir adımdır.



Tek bir YZ'nin her şeyi aynı anda yaptığını düşünmek yanlıştır. Algılama, planlama ve motor denetimi, sırayla birlikte çalışan ayrı adımlardır. Her adımın kendi rolü vardır ve farklı yöntemler kullanabilir. YZ bir veya daha fazla adımda kullanılabilir, ama mutlaka hepsinde değil. Adımlar ayrıdır ve sıralıdır.



Özetle: sensörler veri toplar, yazılım konumu, engelleri ve hedefi değerlendirir, bir plan yapılır ve komutlar motorlara ulaşır. Bunların ayrı görevler olduğunu ve her aşamada YZ'nin gerekli olmadığını unutma. Robot bu sırayı izleyerek hareket eder. Her parçayı anlamak, YZ'nin hareketi nasıl yönlendirebileceğini görmene yardımcı olur.