

**2024-2-HU01-KA210-VET-000271388 numaralı
başvuru, Akıllı evlerin üretim ve işletiminde
Endüstri 4.0 teknolojilerinin uygulanması.**



İçerik

1.	Proje tanıtımı.....	4
1.1.	Giriş – Proje ve eğitim ortamı.....	4
1.2.	Kılavuzun amacı ve rolü.....	4
1.3.	Projenin kısa tanıtımı ve hedefleri	5
1.4.	“Akıllı ev / akıllı çiftlik” projesinin temel fikri.....	6
1.5.	Entegre Endüstri 4.0 eğitim mimarisi ve uluslararası kalkınma ortamı.....	6
2.	Proje yapısı ve uygulama mantığı	7
2.1.	Proje aşamaları ve bunların birbirine bağımlılığı.....	7
2.2.	Katılımcılar ve roller.....	8
2.3.	Geliştirilecek öğrenme kazanımları ve yetkinlikler	9
3.	Akıllı Çiftlik Kitinin Detaylı Sunumu ve İşlenmesi.....	9
3.1.	Akıllı Çiftlik Kitinin öğrenme sürecindeki rolü.....	9
3.2.	Akıllı Çiftlik Kitinin Ana Bileşenleri.....	9
3.2.1.	Kontrolör ve temel modüller.....	10
3.2.2.	Giriş Elemanları – Sensörler.....	11
3.2.3.	Çıkış elemanları – Aktüatörler ve ekranlar.....	14
3.2.4.	Aksesuarlar ve montaj aletleri.....	17
3.2.5.	Eğitimde bileşenlerin rolü	17
3.3.	Fabrika montaj talimatlarının rolü	17
3.3.1.	Mekanik ve elektronik montaj ve modüler tasarım	18
3.3.2.	Montaj deneyimine dayalı mühendislik değerlendirmeleri.....	19
3.4.	Akıllı Çiftlik Kitinin Programlanması	20
3.4.1.	Blok tabanlı programlama ortamı.....	20
3.4.2.	Metin tabanlı programlama ortamlarına bağlanma.....	21
3.4.3.	Projenin sonraki aşamalarıyla bağlantı.....	22
3.4.3.1.	Proje 1 – Aydınlatma Kontrolü (Aydınlatma Sistemi).....	22
3.4.3.2.	Proje 2 – Işık Kontrol Sistemi	26
3.4.3.3.	Proje 3 – Mesafe Algılama ve Otomatik Müdahale (Akıllı Besleme Sistemi).....	29
3.4.3.4.	Proje 4 – Sıcaklık Kontrol Sistemi	33
3.4.3.5.	Proje 5 – Otomatik Sulama Sistemi.....	38
3.4.3.6.	Proje 6 – WiFi kontrollü Akıllı Çiftlik sistemi.....	42
3.4.3.7.	Özet pedagojik yorum.....	45
3.5.	Bölümün pratik uygulaması.....	47
4.	Kendi akıllı ev/akıllı çiftlik modelinizi dijital olarak tasarlayın (Fusion 360).....	47
4.1.	Başlangıç noktası: Akıllı Çiftlik Kitinin fiziksel tasarımının analizi.....	48
4.2.	3B Baskının Sınırlamaları ve Tasarım Etkileri.....	49
4.3.	Fiziksel kavram sistem açıklaması olarak.....	50
4.4.	Fusion 360 tasarım ortamı – temel bilgiler ve yaklaşım.....	50
4.4.1.	Tuval – Mekansal Hareket Öğrenimi	52
4.4.2.	Temel geometrik ve tasarım kavramları – somut bir şekilde.....	54

4.5.	3D baskıya hazırlık – son pratik odak noktası	56
4.6.	Dijital tasarım sürecinin sonuçları – öğrenci modellerinin sunumu ve yorumlanması	57
4.7.	Tamamlanan modellere dayalı öğrenme çıktıları özeti.....	60
4.8.	Bölümün pratik uygulaması.....	60
5.	3D baskı ve üretim deneyimleri.....	61
5.1.	3D yazıcı ve Bambu Studio'yu tanıtıyoruz.....	61
5.1.1.	Uygulanan prosedürün kısa açıklaması.....	61
5.1.2.	Donanım yapısı ve çalışması.....	61
5.1.3.	AMS filament besleme sistemi.....	62
5.1.4.	Dilimleme prensipleri	63
5.1.5.	Malzemeler (PLA, PETG, ABS)	64
5.2.	Baskı işlemindeki adımlar.....	64
5.2.1.	Model hazırlama ve doğrulama.....	64
5.2.2.	Baskı ve son işlem	65
5.2.3.	Kalite kontrolü ve hata düzeltme	66
5.3.	Beden ve uyum sorunları.....	67
5.3.1.	Üretim hatalarının ele alınması	67
5.3.2.	Tasarım Yinelemeleri.....	67
5.3.3.	Süreçlerin, öğrenci deneyimlerinin ve öğrenilen derslerin eğitimsel önemi.....	67
6.	Entegre kontrol sistemi – kavramsal temeller.....	69
6.1.	Sistem tasarımının amacı.....	69
6.2.	Dağıtılmış mimarinin teknik avantajları	71
6.2.1.	İletişim modeli ve yaklaşımı	71
6.2.2.	Saha kontrol düğümü – bağımsız örnekleme ünitesi	72
6.3.	Program yapısı fonksiyonel birimlere ayrılmıştır	73
6.3.1.	Ağ genişletmeli saha kontrol düğümü – WiFi + Modbus TCP.....	76
6.3.2.	Denetim Seviyesi – Python Tabanlı HMI / Denetim Döngüsü	82
6.4.	Raspberry Pi tabanlı izleme sisteminin hazırlanması ve devreye alınması.....	83
6.5.	Python istemcisinin rolü	86
6.6.	Sensör verilerinin SQL veritabanına kaydedilmesi.....	89
6.7.	Web yönetim katmanı – sistemdeki rolü.....	91
6.8.	Öğrencilerin ekip çalışmasıyla web HMI arayüzü geliştirmesi	95
6.9.	Özet – sistem entegrasyonu ve öğrenme çıktıları	97
7.	Özet	99

1. Proje sunumu

1.1. Giriş – Proje ve eğitim ortamı

Bu kılavuz, Endüstri 4.0 ve Nesnelerin İnterneti (IoT) teknolojilerine dayalı uluslararası işbirlikçi bir eğitim projesinin profesyonel ve metodolojik dokümantasyonudur. Proje, akıllı ev/akıllı çiftlik modeline odaklanmaktadır ve bu modelin geliştirilmesine dijital tasarım ve eklemeli üretimden elektronik entegrasyonuna ve yazılım geliştirmeye, karmaşık, web tabanlı bir bina yönetim sisteminin oluşturulmasına kadar çeşitli alanlardan öğrenciler katılacaktır.

Proje, sensörler, aktüatörler ve IoT tabanlı kontrol hakkında bilgi edinmek için iyi bir temel sağlayan, piyasada bulunan ticari bir eğitim kiti (Akıllı Çiftlik Kiti) ile başladı. Ancak geliştirme süreci fabrika çözümlerinin uygulanmasıyla sınırlı kalmadı: katılımcılar kritik analizler gerçekleştirdi, sistem sınırlamalarını belirledi ve ardından endüstriyel bir modeli takip ederek kendi ölçeklenebilir sistem mimarilerini geliştirerek bunlara yanıt verdi.

Bu kılavuzun amacı, yalnızca belirli bir proje uygulamasını belgelemek değil, aynı zamanda diğer kurumlar ve öğrenme grupları tarafından, hazır unsurlar kullanılmadan bile, izlenebilecek ve daha da geliştirilebilecek açık, uyarlanabilir bir eğitim modeli sunmaktır.

1.2. Kılavuzun amacı ve rolü

Bu kılavuzun amacı, Endüstri 4.0 ve IoT teknolojilerinin pratik uygulaması yoluyla öğrencilerin dijital, teknik ve mühendislik yetkinliklerinin geliştirilmesini destekleyen bir eğitim projesi için kapsamlı mesleki ve metodolojik rehberlik sağlamaktır. Belge, yalnızca hazır çözümler sunmakla kalmaz, aynı zamanda problem çözme, analiz ve yinelemenin önemli bir rol oynadığı bir öğrenme ve geliştirme sürecini de ortaya koyar.

Bu kılavuzun yaklaşımı, eğitimin pasif bilgi aktarımına değil, aktif yaratıcı faaliyete dayalı olduğu yönündedir. Proje süresince öğrenciler sadece kullanıcı değil, aynı zamanda karmaşık bir teknik sistemin tasarımcıları, uygulayıcıları ve geliştiricileridir. Belge, tasarım, üretim ve programlamadan sistem entegrasyonuna kadar bu yaklaşımı desteklemektedir.

Bu kılavuzun temel işlevi, bir öğretim aracı olarak kullanılmaktır. İçeriği şunlar için uygundur:

- sınıf içi ve sınıf dışı proje çalışmalarına temel teşkil eder,
- Mesleki eğitim, yetenek yönetimi veya proje tabanlı eğitim programlarına uygun,
- Öğretmenlere karmaşık teknik ve bilişim teknolojisi projelerinin uygulanmasında metodolojik destek sağlamak.

Metodolojik yaklaşım, proje tabanlı ve keşifsel öğrenmeye odaklanmaktadır. Öğrenciler, daha basit alt görevlerden karmaşık sistem anlayışına doğru kademeli olarak, ardışık adımlarla ilerlerler. Buna göre, kılavuz tek bir "doğru" çözüm sunmamaktadır.

Bu, yalnızca bir çözüm kaydetmekle kalmaz, aynı zamanda alternatifleri, karar noktalarını ve geliştirme fırsatlarını da sunar.

Bu belge çeşitli hedef gruplar için önemlidir:

- Öğrenciler için: Bu kılavuz, geliştirme süreçlerini öğrenci dostu bir mantıkla sunarak bağımsız denemeler yapma, hata yapma ve yeniden tasarlama fırsatı sağlar. Proje süresince öğrenciler mühendislik düşüncesinin temellerini deneyimlerler: bir problemi tanımlarlar, bir çözüm tasarlarlar, test ederler, değerlendirirler ve geliştirirler.
- Eğitimciler için: Bu kılavuz, ders organizasyonunu, farklılaştırılmış ödevleri ve heterojen grupların yönetimini destekler. Farklı geçmişlere sahip öğrencileri ortak bir projeye nasıl dahil edeceğiniz ve teknik içeriği pedagojik olarak yapılandırılmış ancak esnek bir şekilde nasıl işleyeceğiniz konusunda yardımcı olur.
- Kurumlar için: Proje, yerel müfredatlara, proje haftalarına veya uluslararası işbirliklerine entegre edilebilen uyarlanabilir bir model sunmaktadır. Açık kaynaklı yapısı, kurumların uygulamayı kendi ekipmanlarına, altyapılarına ve eğitim profillerine uyarlamalarına olanak tanır.

1.3. Projenin kısa tanıtımı ve hedefleri

Teknik ve ortaöğretim bilişim eğitimiyle bağlantı: Proje başlangıçta, endüstriyel bilişim, mekatronik ve ilgili teknik eğitim alanlarında temel teknolojik bilgiye sahip öğrencilerin bulunduğu teknik bir ortamda başladı. Ancak uluslararası iş birliği sırasında, daha önce teknik deneyimi olmayan teorik bir lise de programa katıldı. Bu durum hem bir zorluk hem de pedagojik bir fırsat oluşturdu.

Avantajlar:

- Farklı geçmişlere sahip öğrencilerin işbirliği, açıklayıcı ve sistematik düşünme becerilerini güçlendirdi.
- Teknik okul öğrencileri mentörlük rolünü üstlenerek kendi bilgilerini daha da derinleştirdiler.
- Teorik lise öğrencileri için bu proje, soyut bilişim bilgisi ile fiziksel sistemler arasında bir köprü oluşturdu.

Olası zorluklar:

- Temel teknik bilgi eksikliği, ilk ilerlemeyi yavaşlatmış olabilir.
- Temel kavramların (elektronik, sensörler, kontrol) adım adım tanıtılmasına daha fazla önem verilmesi gerekiyordu.
- Eğitimcilerin rolü daha çok kolaylaştırıcı ve destekleyici bir hal aldı.

Proje deneyimlerine dayanarak, sunulan modelin sadece teknik okul ortamında işlevsel olmadığı söylenebilir. Uygun bir metodolojik yapı ve farklılaştırma ile, ortaöğretim, teknik olmayan kurumlarda da başarıyla uygulanabilir.

1.4. "Akıllı ev / akıllı çiftlik" projesinin temel fikri

Projenin temel fikri, akıllı sistemlerin işleyişini somut bir şekilde gösteren fiziksel ve dijital bir model oluşturmak ve modern teknolojilerin uygulamasını öğrenciler için anlaşılabilir ve gerçek dünya sorunları üzerinden sunmaktır. Akıllı ev/akıllı çiftlik modeli, teknik, bilişim ve çevresel yönlerin eğitimde birleşik bir sistem olarak ortaya çıkması için bir fırsat yaratmaktadır.

Model, aşağıdaki unsurların aynı anda sunulması için uygundur:

- Çevresel koşulların izlenmesi ve kontrolünde sensörlerin ve aktüatörlerin rolü,
- Veri toplama, veri işleme ve geri bildirim süreci,
- Yerel ve uzaktan ortamlarda kontrol ve denetimin mantığı,
- Enerji ve kaynak verimliliği sağlayan bir araç olarak web tabanlı, uzaktan erişim seçenekleri.

Bu projede, akıllı ev ve akıllı çiftlik sadece teknolojik bir gösteri olarak değil, aynı zamanda çevre bilincine sahip bir yaklaşımın taşıyıcısı olarak da karşımıza çıkıyor. Sensör destekli ölçüm ve otomatik müdahale yoluyla öğrenciler, akıllı kontrolün enerji tüketiminin optimizasyonuna, çevresel kaynakların bilinçli kullanımına ve sürdürülebilir işletmeye (örneğin aydınlatma, havalandırma, sulama kontrolü) nasıl katkıda bulunabileceğini öğreniyorlar.

Bu model nihai bir ürün değil, bilinçli olarak açık, sürekli gelişen ve tasarım, deney, değerlendirme ve yeniden tasarım döngüleri aracılığıyla sistem düşüncesinin gelişimini destekleyen bir öğrenme aracıdır. Bu yaklaşım, öğrencilerin yalnızca teknik çözümlere hakim olmalarına değil, aynı zamanda bunların çevresel, ekonomik ve sosyal bağlamlarını anlamalarına da yardımcı olur.

1.5. Entegre Endüstri 4.0 eğitim mimarisi ve uluslararası kalkınma ortamı Proje, Endüstri

4.0 yaklaşımının ilkelerine uygun olarak yürütülmektedir:

- dağıtılmış sistem mimarisi,
- Veriye dayalı operasyon,
- Ağ bağlantılı cihazların kullanımı,
- Standart iletişim protokollerini kullanarak.

Nesnelerin interneti (IoT) teknolojilerinin kullanımı, öğrencilerin birbirinden bağımsız cihazlarla çalışmak yerine, donanım ve yazılım bileşenlerini tutarlı bir sistemin parçası olarak yorumlamalarına olanak tanır.

Akıllı Çiftlik Kiti, bilinçli olarak bir başlangıç noktası olarak tasarlandı. Temel unsurlarından biri, fabrika sistemini analiz etmek, sınırlamalarını belirlemek ve ardından endüstriyel bir modeli izleyen kendi ölçeklenebilir sistem mimarisini geliştirerek bunlara yanıt vermektir. Bu süreç, eleştirel düşünme ve yinelemeli geliştirmenin teknolojik ilerlemenin temelleri olduğunu vurguladı.

Yaklaşım tamamen proje tabanlıdır. Öğrenciler önceden belirlenmiş adımları izlemezler, bunun yerine sorunları çözerler, kendi kararlarını alırlar ve sonuçlarını analiz ederler. Öğrenme sürecinde hatalar başarısızlık olarak değil, gelişimin doğal bir parçası olarak görülür.

Fiziksel modeller, anında geri bildirim ve görünür sonuçlar doğal olarak öğrenci motivasyonunu artırır. Zorluklar, kilometre taşları ve ortak hedefler gibi oyunlaştırma unsurları, proje boyunca yapı oluşturmak ve ilgiyi sürdürmek için yer alır.

Karmaşık bir şekilde dijital yetkinlikler geliştirir: programlama, veritabanı yönetimi, ağ bilgisi, dijital tasarım ve dokümantasyonun hepsi mevcuttur. Ayrıca mühendislik düşüncesini, sistem düşüncesini ve fonksiyonel birimler arasındaki ilişkilerin kavranmasını güçlendirir.

Proje, uluslararası iş birliğiyle hayata geçirildi ve bu da mesleki içeriğe ek olarak yetkinlik geliştirme ve pedagojik içerik açısından önemli bir katma değer sağladı. Katılımcı kurumlar farklı ülkelerde, eğitim sistemlerinde ve kurumsal profillerde faaliyet gösterdiğinden, ortak çalışma gelecekteki işgücü piyasası durumlarını iyi bir şekilde modelleyen gerçek bir iş birliği ortamı yarattı.

Ortak faaliyetler sırasında dil farklılıkları belirgindi. Birçok durumda, profesyonel iletişim İngilizce olarak gerçekleşti; bu da öğrencilerin yabancı dil becerilerini geliştirdi ve onlara doğru ve net teknik iletişimin önemini farkına varmalarını sağladı.

Çok kültürlü iş birliği, çalışma sürecinin doğal bir parçası haline geldi. Farklı eğitim geçmişine ve problem çözme stratejilerine sahip öğrencilerin ortak çalışması, esnekliği, uyum yeteneğini ve profesyonel diyalogu güçlendirdi.

Bu uygulamanın işleyişi, çok uluslu şirketlerin beklentilerini çeşitli açılardan yansıtıyordu: yabancı dillerde iletişim, ortak dokümantasyon yönetimi, ekip çalışması ve karmaşık sistemlerde sorumluluk. Kazanılan deneyimler, öğrencilerin teknik bilgilerini daha geniş, uluslararası bir bağlamda yorumlayabilmelerine katkıda bulundu.

2. Proje yapısı ve uygulama mantığı, proje

2.1. aşamaları ve bunların birbirine bağımlılığı.

Proje, bilinçli olarak birbirinin üzerine inşa edilmiş aşamalar halinde uygulandı. Bu yapı sadece teknik bir gereklilik değil, aynı zamanda pedagojik bir karardı: amaç, öğrencilerin kendi deneyimlerinden yola çıkarak, kademeli olarak bağımsız gelişim görevlerine ulaşmalarını sağlamaktı. Aşamalara ayırma, farklı geçmişlere sahip öğrencilerin katılımına ve her adımla ilgili yansıma ve değerlendirmeye olanak tanıdı.

İlk aşamada öğrenciler hazır, çalışan bir sistemle karşılaştılar. Amaç, temel kavramları, operasyonel ilişkileri ve sistem mantığını öğrenmekti. Vurgu, çözümü "kopyalamak" değil, sistemin ne yaptığını, neden yaptığını, hangi unsurlardan oluştuğunu ve bunların birbirleriyle nasıl bağlantılı olduğunu anlamak üzerineydi.

Bu aşama, özellikle teknik altyapısı daha az olan öğrenciler için son derece önemliydi; çünkü ortak bir başlangıç noktası oluşturarak daha sonraki, daha karmaşık görevlere giriş engelini düşürdü.

Alıştırma aşamasını, bilinçli analiz aşaması olan ikinci aşama izledi. Öğrenciler sadece sistemi kullanmakla kalmadılar, aynı zamanda operasyonel sınırlarını, güçlü ve zayıf yönlerini de incelediler. Bu adım, teknik bir çözümün her zaman uzlaşmaların sonucu olduğunu ve belirli amaçlar için optimize edildiğini anlamada kilit rol oynadı.

Kritik değerlendirme yoluyla, kendi gelişim ihtiyaçlarını formüle etmeyi ve mevcut bir sistemi daha da geliştirmenin veya yeniden düşünmenin ne zaman uygun olduğunu anlamayı öğrendiler. Bu düşünme biçimi, mühendislik ve BT problem çözme prensipleriyle doğrudan ilişkilidir.

Projenin üçüncü aşamasında öğrenciler, analitik deneyimlerinden yola çıkarak kendi geliştirme çalışmalarına başladılar. Bu aşamada süreç artık önceden belirlenmiş adımlarla değil, birlikte alınan kararlar, yinelemeli planlama ve sürekli test etme ile yönlendirildi.

Bu aşamada, proje yaklaşımının avantajı gerçekten belirginleşti: öğrenciler karmaşık bir sistemin bir anda "tamamlanmadığını", aksine kademeli olarak şekillendiğini ve her kararın tüm işleyişi etkilediğini deneyimlediler.

2.2. Katılımcılar ve roller

Projenin belirleyici özelliklerinden biri, katılımcılar arasında rollerin bilinçli bir şekilde paylaşılmasıydı. Öğrenciler, öğretmenler ve kurumlar farklı ancak birbirini güçlendiren rollerde birlikte çalışarak projenin esnekliğine ve sürdürülebilirliğine katkıda bulundular.

Katılımları sırasında öğrenciler pasif alıcılar değil, aktif geliştiricilerdi. Planlama, sorunları yorumlama, çözümleri test etme ve değerlendirme süreçlerinde iş birliği yaptılar. Alt görevlerin dağıtım sırasında sorumluluk almanın önemini ve çalışmalarının daha karmaşık bir sisteme nasıl uyduğunu deneyimlediler. Bu rol, bağımsızlığı, iş birliğini ve uzun vadeli düşünmeyi güçlendirdi.

Eğitmenlerin rolü öncelikle destekleyici ve yol göstericiydi. Hazır çözümler sunmaktan ziyade, rehberlik etmeye, soru sormaya ve öğrenme sürecini yapılandırmaya odaklanılmıştı.

Bu yaklaşım, öğrencilerin kendi deneyimleri aracılığıyla çözümler bulmalarına olanak sağlarken, onlara sağlam bir mesleki ve pedagojik altyapı da kazandırdı.

Uluslararası işbirliği, farklı eğitim geçmişine ve deneyimlerine sahip öğrencilerin birlikte çalıştığı karma öğrenme gruplarının oluşmasına yol açtı. Bu durum, iletişim becerilerini, uyum yeteneğini ve ortak problem çözme becerisini güçlendirdi.

Karma ekip çalışması tüm projeye eşlik etti ve geleneksel okul çerçevesinin ötesine geçen gerçek iş birliği durumları yarattı.

2.3. Geliştirilecek öğrenme kazanımları ve yetkinlikler

Tasarım sürecinde, öğrenme çıktılarını tek bir yetkinlik alanıyla sınırlandırmamak bilinçli bir hedefti. Geliştirme süreci, teknik, bilişsel ve sosyal becerilerin aynı anda geliştirilmesini destekledi.

Bu proje, dijital araçların bilinçli kullanımına, yapılandırılmış çalışmaya ve dijital ortamda yaratıcılığa katkıda bulundu. Öğrenciler, karmaşık bir sistemin nasıl anlaşılabilir ve yönetilebilir hale geldiğini deneyimlediler.

Çok aşamalı çözümler, planlama ve geri bildirim gerektiren karmaşık sorunlarla karşılaştılar. Bu durum, sistem düşüncesini, neden-sonuç ilişkilerini kavrama yeteneğini ve uzun vadeli planlama becerisini güçlendirdi.

Tamamlanan çalışmaları belgelemek ve sunmak projenin ayrılmaz bir parçasıydı. Öğrenciler, bir geliştirme sürecini şeffaf bir şekilde kaydetmeyi ve anlaşılabilir bir biçimde sunmayı öğrendiler. Ortaya çıkan portfolyolar ayrıca öğrenme sonuçlarının uzun vadeli kullanımını da mümkün kılmaktadır.

3. Akıllı Çiftlik Kitinin detaylı sunumu ve işlenmesi

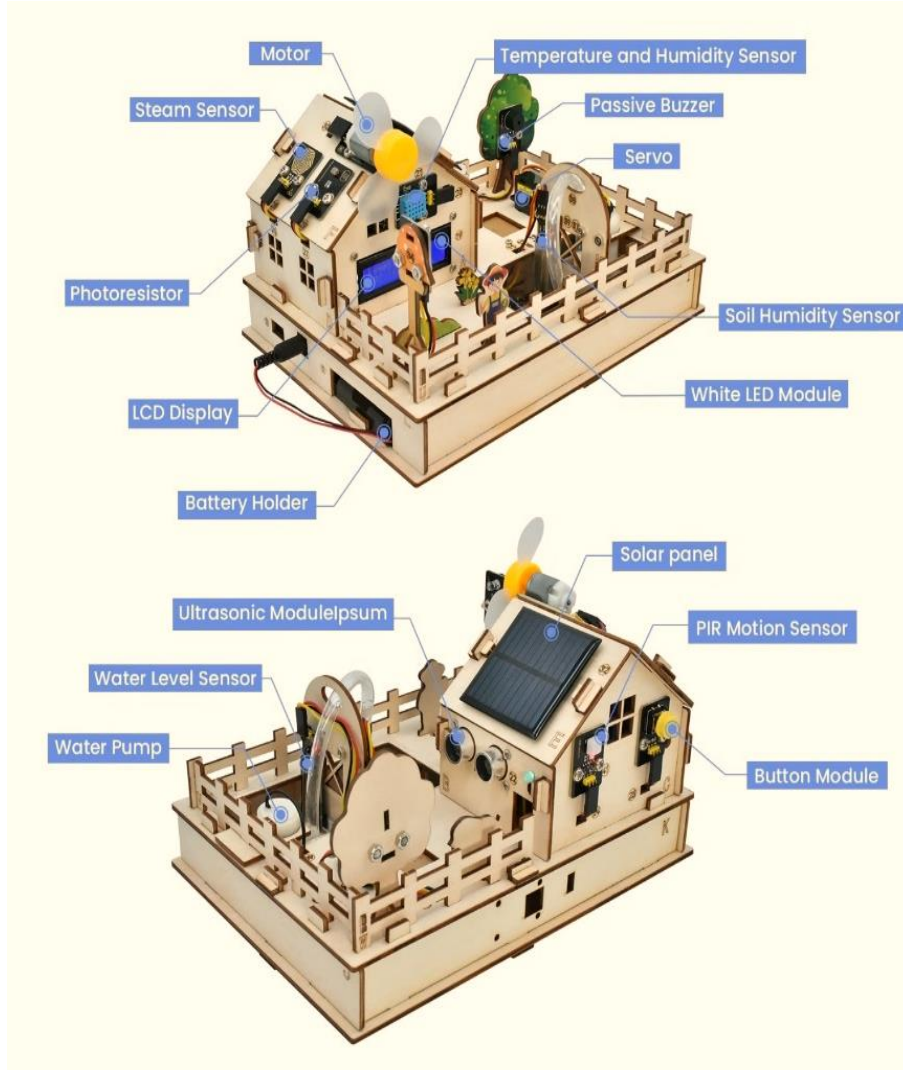
3.1. Akıllı Çiftlik Kitinin öğrenme sürecindeki rolü

Akıllı Çiftlik Kiti, akıllı sistemlerin temel prensiplerini somut bir model aracılığıyla gösteren, eğitim amaçlı tasarlanmış modüler bir IoT kitidir. Kit, öğrencilerin kendi çözümlerini geliştirmeye başlamadan önce gerçek, çalışan bir sistemle tanışmalarını amaçlamaktadır.

Bu projede, kit nihai bir ürün olarak değil, bir öğrenme aracı olarak sunulmaktadır. İşleme sürecinde, elemanların çalışma prensibini anlamaya, sistemin mantığını haritalamaya ve daha sonraki geliştirmeler için temel oluşturmaya odaklanılmaktadır.

3.2. Akıllı Çiftlik Kitinin Ana Bileşenleri

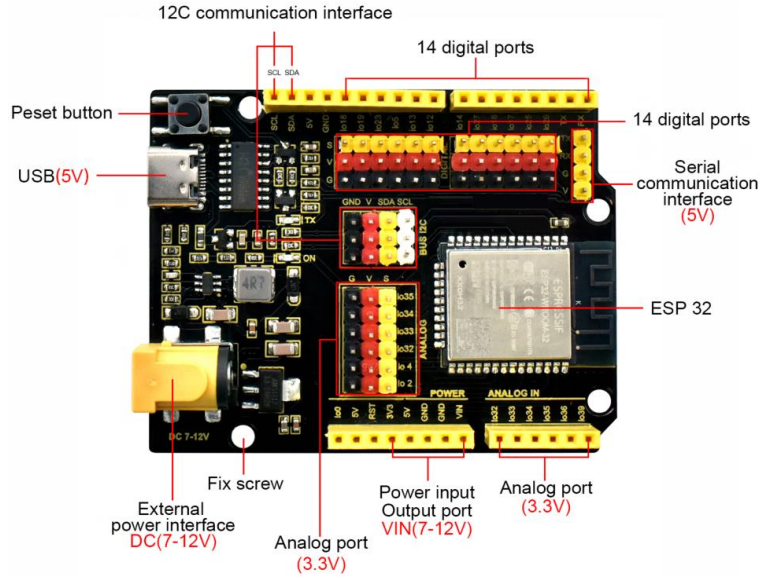
Akıllı Çiftlik Kiti'nin elektronik bileşenleri, çoklu giriş sensörleri, kontrol modülleri ve çıkış aktüatörleri de dahil olmak üzere karmaşık bir IoT sistemini göstermek üzere tasarlanmıştır. Liste, üreticinin kaynağına ve güvenilir ürün açıklamalarına dayanmaktadır ve kitin öğrenme programı sırasında kullanılan gerçek cihazları kapsamaktadır.



Şekil 1: Akıllı Çiftlik Kif'in ana parçaları (Kaynak: <https://www.keyestudio.com>)

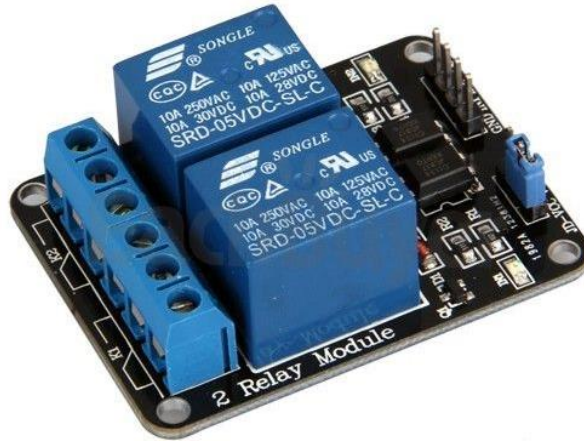
3.3. Kontrolör ve temel modüller

- ESP32 mikrodnetleyici:
 - Yapı: Wi-Fi ve Bluetooth modülüyle entegre edilmiş mikrodnetleyici
 - İşlevi: Kitin "beyni" olup sensör verilerini, mantığı ve iletişimi yönetir.
 - Eğitimsel önemi: Ağ iletişimi, paralel olay işleme ve giriş/çıkış mantığının incelenmesi.



Şekil 2: ESP32 mikrodenetleyici (Kaynak: <https://docs.keyestudio.com>)

- Röle modülü:
 - Yapı: 5 V kontrollü opto-izole röle
 - İşlev: Daha yüksek tüketimli cihazların (örneğin pompa) anahtarlanması.
 - Eğitimsel amaç: Aktüatörlerin izolasyonu ve güvenli kontrolü.



Şekil 3: Röle modülü (Kaynak: <https://docs.keyestudio.com>)

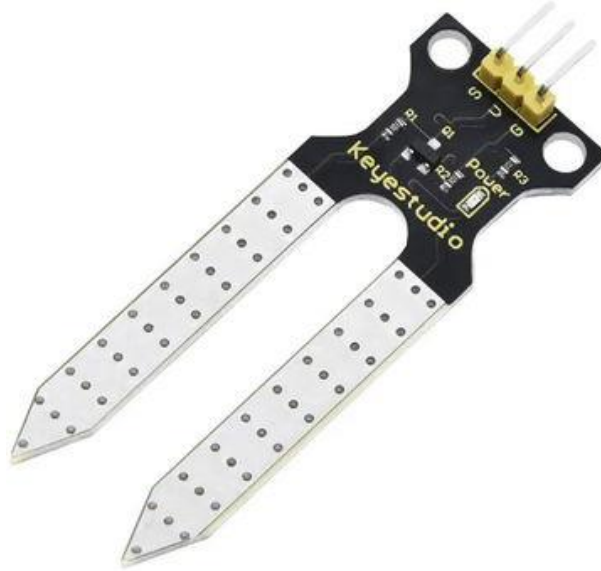
3.4. Giriş elemanları – Sensörler

- Foto direnç: O
 - Yapı: ışığa duyarlı yarı iletken direnç
 - Çalışma prensibi: Direnç, ışık yoğunluğuna bağlı olarak değişir (analog sinyal).
 - Eğitim amaçlı kullanım: Işık koşullarını ölçmek, analog girişleri kullanmak.



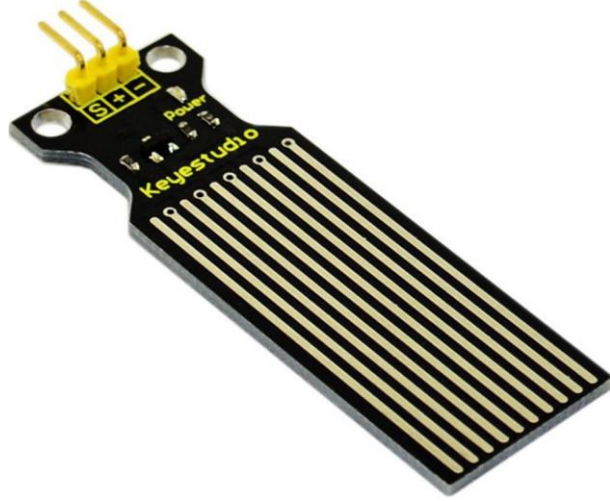
Şekil 4: Fotorezistör (Kaynak: <https://docs.keyestudio.com>)

- Toprak nem sensörü: O
 - Yapı: iki elektrot ve dönüştürücü
 - Çalışma prensibi: Topraktaki su direnci azaltır → sinyal değeri yükselir. Eğitimsel
 - amaç: Analog ölçüm teknikleri ve çevresel verilerin yorumlanması.



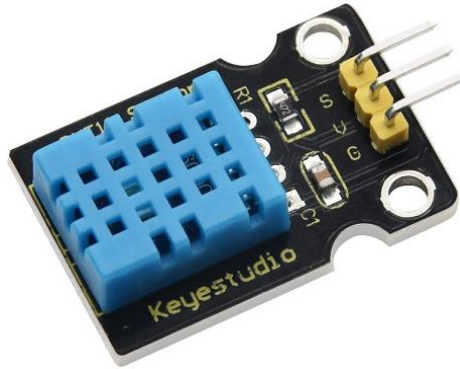
Şekil 5: Toprak nem sensörü (Kaynak: <https://docs.keyestudio.com>)

- Su seviyesi sensörü:
 - Yapı: çok elektrotlu temas
 - Çalışma prensibi: Sıvı seviyesine bağlı olarak kapanma/
 - açılma sinyali. Eğitim amacı: Dijital olay odaklı ölçüm.



Şekil 6: Su seviyesi sensörü (Kaynak: <https://docs.keyestudio.com>)

- DHT11 sıcaklık ve nem sensörü: O
 - Yapı: Üniteye entegre edilmiş dijital sensör
 - Fonksiyon: Entegre bir algoritma kullanarak sıcaklık ve nemi hesaplar.
 - Eğitimsel amaç: Dijital veri iletişimi ve fiziksel ortamın ölçümü.



Şekil 7: DHT11 sıcaklık ve nem sensörü (Kaynak: <https://docs.keyestudio.com>)

- SR01 V3 ultrasonik mesafe sensörü: O
 - Yapı: verici ve alıcı piezoelektrik elemanlar.
 - Çalışma prensibi: ultrasonik darbe → yansıma → zaman ölçümünden mesafe hesaplama
 - Eğitimsel amaç: Fiziksel dalgaları ve zamana dayalı ölçümü anlamak.



Şekil 8: SR01 V3 Ultrasonik Mesafe Sensörü (Kaynak: <https://docs.keyestudio.com>)

- PIR hareket sensörü:

- Yapı: pasif kızılötesi sensör
- İşlev: Görüş alanındaki ani sıcaklık değişimlerini algılar. Eğitimsel
- amaç: Olay odaklı mantık ve gerçek zamanlı algılama.



Şekil 9: PIR hareket sensörü (Kaynak: <https://docs.keyestudio.com>)

- Pasif zil:

Yapı: piezo hoparlör

- Çalışma prensibi: Gerilim darbelerine ses çıkararak yanıt verir. Eğitimsel amaç: Çıkış
- modülleri ile pratik sinyalizasyon sistemi arasında bağlantı kurmak.

3.5. Çıkış elemanları – Aktüatörler ve ekranlar

- Beyaz LED modülü:

- Yapı: önceden monte edilmiş LED
- Çalışma prensibi: Dijital çıkış, aydınlatmayı kontrol eder.
- Eğitim amacı: Basit mantık için görsel geri bildirim.



Şekil 10: Beyaz LED modülü (Kaynak: <https://docs.keyestudio.com>)

- SG90 9G servo motor:

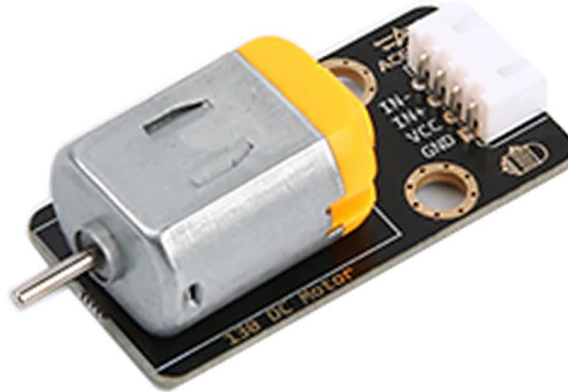
- Yapı: Konum geri beslemeli mikro servo motor. Çalışma
- prensibi: PWM sinyaline dayalı konum kontrolü.

OEğitimsel amaç: Aktüatörlerin ve mekanik etkilerin hassas kontrolü.



Şekil 11: SG90 9G servo motor (Kaynak: <https://docs.keyestudio.com>)

- DC motor/vantilatör: O
 - Yapı: Küçük fanlı DC motor
 - O İşlev: Ortam havasını hareket ettirmeye yarayan çıkış cihazı. Eğitimsel
 - O amaç: Motor kontrolü ve performans izleme.



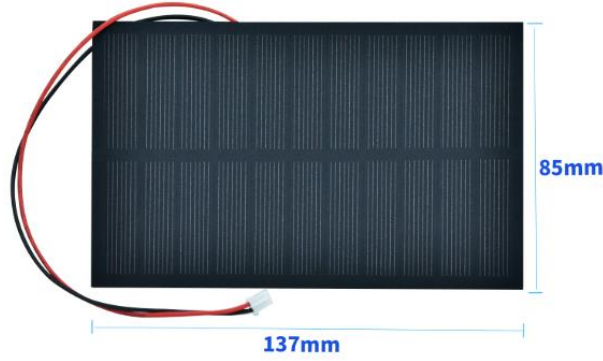
Şekil 12: DC motor/vantilatör (Kaynak: <https://docs.keyestudio.com>)

- I2C 1602 LCD ekran:
 - O Yapı: 2 satır × 16 karakter, I2C iletişimi
 - O İşlev: Mikrodenetleyici mesajlarından karakter verilerini görüntülemek.
 - O Eğitimsel amaç: Veri görüntüleme ve iletişim protokollerini öğrenmek.



Şekil 13: I2C 1602 LCD ekran (Kaynak: <https://docs.keyestudio.com>)

- Güneş paneli:
 - Yapı: Küçük fotovoltaik panel Çalışma prensibi: Fotovoltaik etki
 - → elektrik Üretim amacı: Nesnelerin İnterneti (IoT)
 - sistemlerinde yenilenebilir enerji kaynakları.



Şekil 14: Güneş paneli (Kaynak: <https://docs.keyestudio.com>)

- Su pompası:
 - Yapı: küçük DC pompa
 - Çalışma prensibi: Röle kontrolü ile sıvı transferi
 - Eğitimsel amaç: Aktüatör kontrolü ve mekanik çalışma kavramlarını anlamak.



Şekil 15: Su pompası (Kaynak: <https://docs.keyestudio.com>)

3.6. Aksesuarlar ve montaj aletleri

- şeffaf akrilik elemanlar ve talaş kaplama
- güç kabloları, DuPont telleri, vidalar, bağlantı elemanları
- pil taşıyıcı veya pil tutucu
- İletişim ve güç kaynağı için USB kablosu.

3.7. Eğitimde bileşenlerin rolü

Kitin içinde yer alan 17 elektronik bileşen, giriş niteliğindeki Nesnelerin İnterneti (IoT) ve otomasyon müfredatını iyi bir şekilde kapsamaktadır:

- Sensörler, fiziksel ortamdan sinyal iletimi sağlar.
- Kontrol ünitesi yorumlar ve karar verir.
- Aktörler çevrede bir şeyler sergiler veya çevreye müdahale ederler.

Bu yapı, gerçek bir IoT sisteminde elemanların nasıl birlikte çalıştığına ve programlama, ölçüm mantığı ve kontrol ile nasıl bağlantı kurulabileceğine dair sistematik bir genel bakış sunar.

Akıllı Çiftlik Kitinin montajı, öğrencilerin karmaşık, çok bileşenli bir fiziksel sistemle çalıştığı projenin ilk adımıdır. Fabrika dokümanı, kullanıcıyı mekanik ve elektronik elemanların montajı konusunda adım adım yönlendirir; bu da bilinçli bir öğrenme süreci olarak yorumlanabilir.

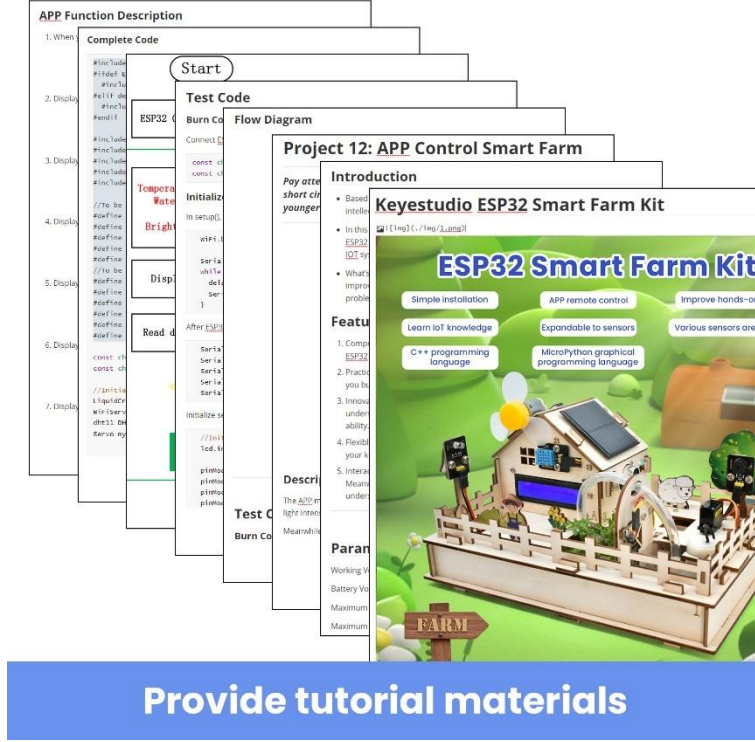
Montaj, yalnızca teknik bir görev değil, aynı zamanda daha sonraki bağımsız tasarım ve üretim çalışmalarının temelini oluşturan bir deneyimdir.

3.8. Fabrika montaj talimatlarının rolü

Akıllı Çiftlik Kiti fabrika kılavuzu, montaj sırasında birincil referans kaynağıdır. Kılavuz şunları içerir:

- Her bir montaj adımının sırasını belirler.
- Parçaların doğru konumunu ve sabitlenmesini gösterir.
- Elektronik bileşenlerin güvenli bir şekilde bağlanmasına yardımcı olur.
- Görsel açıklamalarla anlamayı destekler.

Öğrenciler bu kılavuzu kullanırken, bir mühendislik sistemini bir araya getirmenin doğaçlama değil, yapılandırılmış, dokümantasyona dayalı bir süreç olduğunu deneyimleyeceklerdir.

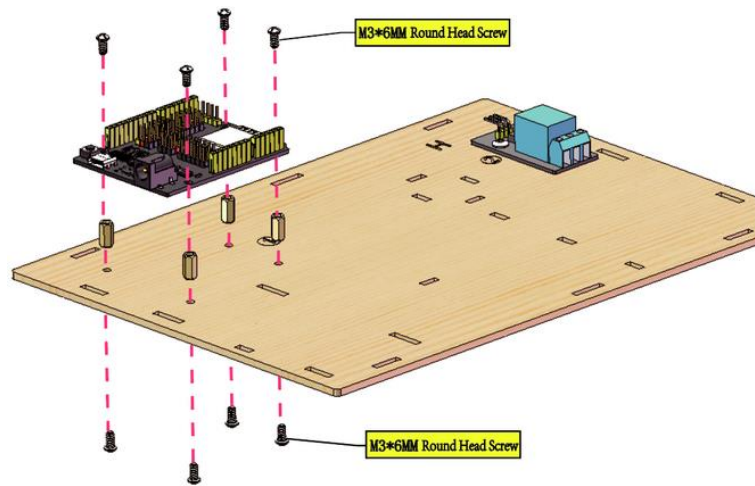


Şekil 16: Fabrika montaj talimatları (Kaynak: <https://www.keyestudio.com>)

3.9. Mekanik ve elektronik montaj ve modüler tasarım

Montaj sırasında öğrenciler, kitin modüler mekanik tasarımıyla tanışacaklardır. Her bir eleman, standart bağlantı mesafelerine göre önceden tasarlanmış montaj noktalarına sahiptir; bu da bileşenlerin değiştirilmesine ve yerinin değiştirilmesine olanak tanır.

Bu modüler yaklaşım önemli bir mühendislik prensibini ortaya koymaktadır: sistemin genişletilebilirliği ve bakım kolaylığı fiziksel tasarım seviyesinde belirlenir.

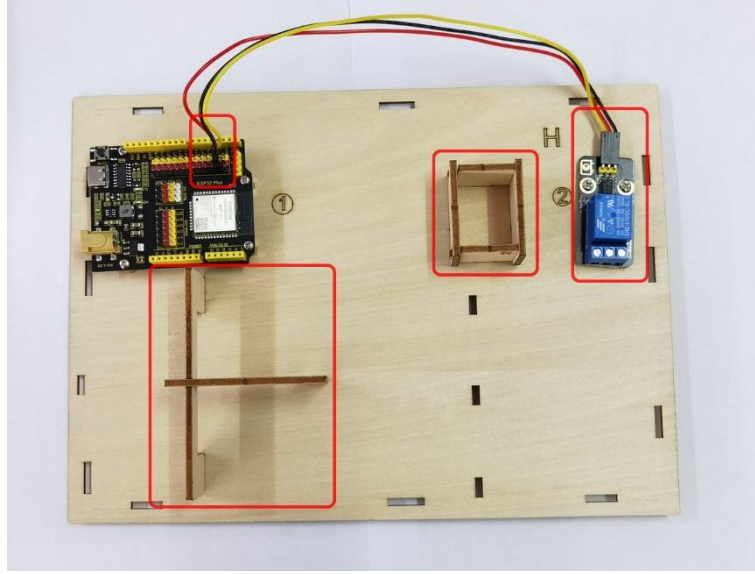


Şekil 17: Akıllı ev anakartı (Kaynak: <https://docs.keyestudio.com>)

Elektronik montajı sırasında öğrenciler sensörlerin ve aktüatörlerin bağlantı noktalarını belirler, güç kaynağı ve sinyal kablolarının rolünü öğrenir ve fabrika kablolama şemalarını takip ederler.

Bağlantılar sırasında, kutupların doğru şekilde kullanılmasına, dijital ve analog girişler arasındaki ayrımın yapılmasına ve kabloların mekanik olarak korunmasına ve netliğine özel önem verilir.

Bu adım, daha sonraki sorun giderme ve sistem teşhisine yönelik düşünme süreçlerinin temelini oluşturur.



Şekil 18: Akıllı ev anakart bağlantı noktaları (Kaynak: <https://docs.keyestudio.com>)

Montaj, tek seferlik, kapalı bir süreç değil, tekrarlayan bir faaliyettir. Montajın her aşamasından sonra öğrenciler:

- Mekanik sağlamlığı kontrol edin,
- Elektrik bağlantılarını test ediyorlar.
- Her bir elemanın çalışması kısa test programlarıyla doğrulanır.

Bu yaklaşım, daha sonra ortaya çıkabilecek karmaşık hataları önler ve kademeli doğrulamanın önemine dikkat çeker.

3.10. Montaj deneyimine dayalı mühendislik değerlendirmeleri

Montaj işlemi sırasında, daha sonra kendi tasarımımız olan elemanların oluşturulmasında doğrudan kullanılacak çeşitli mühendislik yönleri gözlemlenebilir:

- Geometrik uyum, yani bileşenlerin boyutlarının ve montaj noktalarının doğruluğu, montajı temelden etkiler.
- Kablolama güzergahlarının tasarlanması ve tellerin düzenlenmesi, şeffaflığı, bakım kolaylığını ve estetiği etkiler.
- Erişilebilirlik ve servis kolaylığı, sensörlerin ve kontrollerin yerleştirilmesi, daha sonraki değişiklik ve genişletme olanaklarını etkiler.

- Modülerlik ve değiştirilebilirlik açısından, fabrika çözümü, sistemin bireysel elemanlarının bağımsız olarak çalıştırılabileceği şekilde nasıl inşa edilebileceğini vurgulamaktadır.

Bu deneyimler, öğrencilerin fiziksel tasarım ve işlevsel operasyonun birbirinden ayrılmaz bir bütün oluşturduğunun farkına varmalarını sağlar.

Montaj aşaması sona erdiğinde, öğrenciler artık sadece sistemin kullanıcıları değil, aynı zamanda analistleri de olurlar. Burada edindikleri deneyimler onları doğrudan şunlara hazırlar:

- sonraki geometrik ölçümler,
- kendi 3D tasarım görevleriniz,
- Bireysel bağlantı elemanlarının ve gövde elemanlarının tasarımı,
- Fabrika çözümlerinin daha da geliştirilmesi veya değiştirilmesi.

Dolayısıyla Akıllı Çiftlik Kitinin montajı bir son nokta değil, daha üst düzey, bağımsız bir mühendislik yaratıcı sürecinin başlangıç noktasıdır.

3.11. Akıllı Çiftlik Kitinin Programlanması:

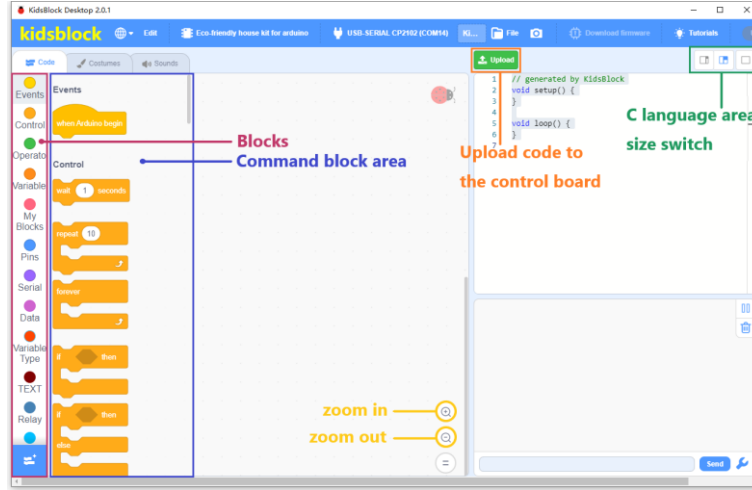
3.12. Blok Tabanlı Programlama Ortamı

Blok tabanlı programlama, sistemleri anlamada ve diğer programlama modellerine geçişte önemli bir rol oynar. Akıllı Çiftlik Kiti'nin grafiksel programlama ortamı (KidsBlock), Scratch tabanlı mantığı izler. Programlama, gerçek, yürütülebilir kontrol talimatlarıyla desteklenen görsel blokların bir araya getirilmesiyle yapılır. Bu, öğrencilerin sistemin gerçek çalışmasını kontrol ederken programlama dillerinin sözdizimsel ayrıntılarıyla uğraşmak zorunda kalmamalarını sağlar.

Eğitimsel açıdan bakıldığında, bu ortam birbirini güçlendiren çeşitli işlevlere hizmet etmektedir:

- Bu yöntem, mantıksal düşünmeyi dilsel zorluklardan ayırarak öğrencilerin kontrol süreçlerine, neden-sonuç ilişkilerine ve durum değişikliklerine odaklanmalarını sağlar.
- Programda yapılan değişikliklerin ardından fiziksel sistemin çalışması doğrudan gözlemlenebildiği için anında geri bildirim sağlar.
- Bu, yinelemeli deneyleri, yani deneme-yanılma-geliştirme şeklindeki doğal öğrenme döngüsünü destekler.
- Bu, sizi daha sonraki metin tabanlı programlamaya hazırlar; çünkü olay işleme, koşul testi, döngüler ve durum yönetimi gibi temel yapılar blokların ardında yer alır.

Bu projede blok tabanlı programlama bir son nokta değil, çok adımlı bir programlama öğrenme yolunda bilinçli olarak seçilmiş bir giriş seviyesidir.



Şekil 19: Blok tabanlı programlama arayüzü, ana blok kategorilerine genel bakış (Kaynak: <https://docs.keyestudio.com>)

3.13. Metin tabanlı programlama ortamlarına bağlanma

Akıllı Çiftlik Kitinin temel kontrol ünitesi olan ESP32 mikrodenetleyici, yalnızca grafiksel bir ortamda programlanabilir olmakla kalmaz. Proje ayrıca, Arduino tabanlı bir ortam gibi metin tabanlı bir programlama dilinde aynı işlevleri uygulama fırsatı da sunar.

Arduino ortamı:

- C/C++ tabanlı sözdizimi kullanır.
- Düşük seviyeli donanım yönetimine olanak tanır.
- Programın yürütülmesine, değişkenlere ve bellek kullanımına ilişkin doğrudan bilgi sağlar.

Pedagojik açıdan bu değişim:

- Blok tabanlı öğelerin arkasında belirli bir kodun çalıştığını anlamaya yardımcı olur.
- Soyutlama yeteneğini geliştirir (blok ↔ kod satırı karşılığı),
- Bu, sizi daha profesyonel gelişim ortamlarının kullanımına hazırlar.

Uygulama sırasında, blok tabanlı ve metin tabanlı programlama birbirini dışlayan araçlar olmaktan ziyade tamamlayıcı araçlar olarak ortaya çıkar.

Akıllı Çiftlik projesinin eğitimsel değeri, öğrencilerin öğrendiği kontrol mantığının endüstriyel otomasyonda kullanılan programlama prensipleriyle yakından ilişkili olması gerçeğiyle daha da artmaktadır.

Açık kaynaklı OpenPLC yazılım platformu, IEC 61131-3 programlamayı yalnızca klasik PLC'lerde değil, Arduino ve ESP32 tabanlı kontrolcülerde de mümkün kılarak mesleki fırsatları daha da genişletmektedir. OpenPLC ortamı, IEC 61131-3 dilinin tam desteğini sağlayarak düşük maliyetli donanımlarda endüstriyel kontrol mantığının uygulanmasına olanak tanır ve özellikle Modbus ve Modbus TCP standartları olmak üzere endüstriyel iletişim protokollerini destekler.

Bu yaklaşım, eğitim amaçlı mikrodenetleyici geliştirme ile gerçek dünya endüstriyel otomasyon sistemleri arasındaki boşluğu doldurmaktadır. Öğrenciler bu sayede mikrodenetleyici geliştirme hakkında bilgi edinebilirler.

Endüstriyel programlama ile, teknolojik olarak daha basit ancak yaklaşımları itibarıyla profesyonel sistemlerle tamamen uyumludur.

3.14. Projenin sonraki aşamalarına bağlantı

Akıllı Çiftlik projesinin sonraki aşamalarında endüstriyel iletişim kilit rol oynayacaktır.

Buna göre:

- Modbus / Modbus TCP protokolünün kullanımı, sistem entegrasyonunun temelini oluşturmaktadır.
- Arduino tabanlı C dili programlama, mikrodenetleyici kontrolünde hala kullanılmaktadır.
- Blok tabanlı ve IEC 61131-3 yaklaşımı öncelikle bir düşünme ve tasarım çerçevesi görevi görür.

Bu karar, öğrencilerin esnek, eğitime uygun bir geliştirme ortamında çalışırken sektör standartları ve beklentileri hakkında bilgi edinmelerini ve daha sonra PLC tabanlı veya endüstriyel IoT sistemleriyle çalışmaya hazırlanmalarını sağlar.

Akıllı Çiftlik projesi, blok tabanlı programlamayı tek başına değil, daha geniş bir endüstriyel bağlam içinde kullanmaktadır. IEC 61131-3 standardının, OpenPLC platformunun/projesinin ve Modbus tabanlı iletişimin 도입 edilmesi, öğrencilerin modern, çok uluslu endüstriyel ortamın beklentilerine doğrudan uyan orta düzeyde bir sistem yaklaşımı edinmelerini sağlarken, eğitimin güvenli, kademeli ve motive edici doğasını da korumaktadır.

3.15. Proje 1 – Aydınlatma Sistemi

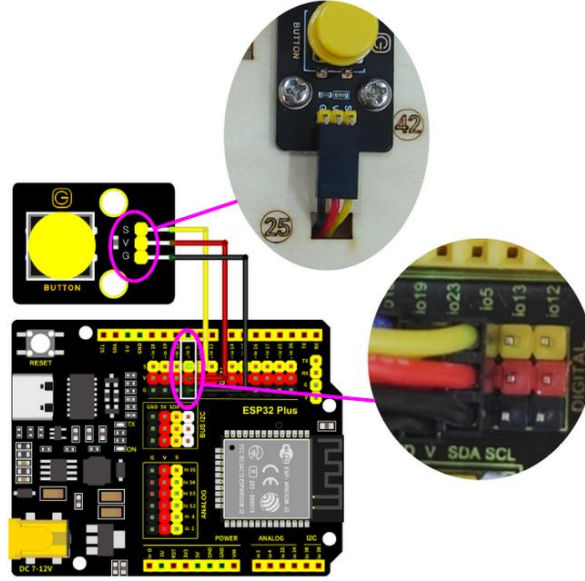
Aydınlatma kontrolü, Akıllı Çiftlik Kitinin en basit ancak eğitim açısından önemli örnek projelerinden biridir. Bu görevde, dijital giriş → karar mantığı → dijital çıkış kontrol zinciri ilk kez eksiksiz, kapalı bir ünite olarak ortaya çıkmaktadır. Projenin amacı, aydınlatmayı bir fonksiyon olarak anlamak değil, otomasyonun temel mantığını anlamaktır.

Dijital giriş olarak basma düğmesi

Basmalı düğme, kontrol cihazına iki farklı durumu iletebilen en basit dijital giriş aygıtıdır:

- Basılmamış durum → mantıksal DÜŞÜK (0),
- basılı durum → mantıksal YÜKSEK (1).

Kontrol ünitesi (ESP32) için, basma düğmesi "fiziksel bir nesne" değil, belirli bir giriş pininde görünen dijital bir sinyaldir. Girişin durumu genellikle dahili veya harici bir yukarı çekme veya aşağı çekme direnci (pull-up/pull-down) ile stabilize edilir, böylece kontrol ünitesi her zaman net bir YÜKSEK veya DÜŞÜK değer algılar.



Şekil 20: Dijital giriş olarak basmalı düğme bağlantı şeması (ESP32 GPIO) (Kaynak: <https://docs.keyestudio.com>)

Basmalı düğme, dijital sinyal kavramını tanıtmak için mükemmel bir araçtır; çünkü "ara" bir değer yoktur, durum her zaman açıkça yorumlanabilir ve bilgi, değerini büyüklüğünden değil, durumdaki değişimden gelir.

Bu yaklaşım, sinyalin sürekli olduğu ve sayısal yorumlama gerektiren analog sensörlerle tam bir zıtlık oluşturmaktadır. İkisi arasındaki farkı anlamak, otomasyon sistemlerini öğrenmede çok önemlidir.

Basma düğmesi, birçok dijital girişten sadece biridir. Proje süresince – veya projenin daha da geliştirilmesiyle – aşağıdaki cihazlar da benzer bir prensiple çalışır:

- Reed rölesi (manyetik kontak): Manyetik alanın etkisi altında bir devreyi kapatır veya açar; tipik uygulaması kapı ve pencere algılamadır.
- Limit şalteri: Mekanik hareketin limit konumunu tespit etmek için kullanılır.
- PIR hareket sensörünün dijital çıkışı: Hareket algılandığında YÜKSEK sinyal verir, aksi halde DÜŞÜK durumda kalır.
- Dijital röle kontak geri beslemesi: Endüstriyel sistemlerde yaygın bir geri besleme biçimi.

Bu cihazların ortak özelliği, ölçülen bir miktar değil, durum bilgisi sağlamalarıdır.

Dijital girdilerin – özellikle de basmalı düğmelerin – kullanımının eğitimsel önemi, çeşitli temel düşünme kalıplarını geliştirmesinde yatmaktadır:

- Olay odaklı düşünme (bir şey olur → tepki gelir),
- İnsan-makine arayüzünün temel kavramı,
- Fiziksel eylemin dijital bilgiye dönüştürüldüğü soyutlama süreci,
- Sonraki durum makineleri ve kontrol mantıklarının anlaşılması.

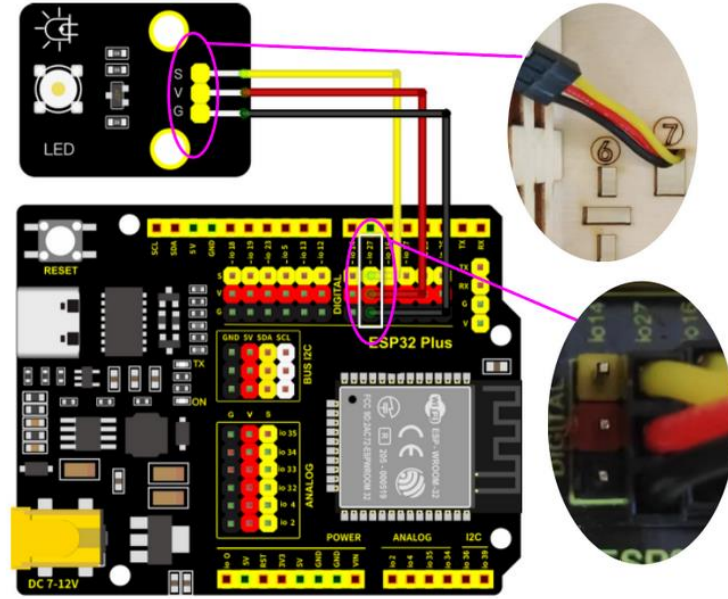
Bu aşamada öğrenciler, otomatik bir sistemin "sürekli olarak izleme" yapmadığını, aksine olaylara tepki verdiğini, bunun da endüstriyel kontrol sistemlerinin temel bir prensibi olduğunu ilk kez anlarlar.

Dijital çıkış – LED kontrollü aktüatör olarak

LED (Işık Yayan Diyot), Akıllı Çiftlik Kiti'ndeki en basit dijital çıkış cihazlarından biridir ve kontrol cihazının kararlarının görsel bir temsilini sağlar. İki durumda çalışır:

- Çıkış DÜŞÜK → LED kapalı,
- Çıkış YÜKSEK → LED yanar.

Kontrol ünitesinin (ESP32) dijital çıkış pini, genellikle aşırı akımdan koruyan bir akım sınırlayıcı direnç üzerinden LED'i çalıştırır. Burada öğrenciler, kontrol ünitesinin herhangi bir yükü doğrudan çalıştıramayacağı, ancak elektronik ortamı dikkate alarak tasarlanması gerektiği gerçeğiyle ilk kez karşılaşır.



Şekil 21: Dijital çıkış için LED bağlantı şeması (ESP32 GPIO + akım sınırlayıcı direnç) (Kaynak: <https://docs.keyestudio.com>)

Dijital çıktı kavramı, girdi kavramından temelde farklıdır. Girdiler sistem hakkında bilgi sağlarken, çıktılar aktif müdahaleyi temsil eder; çünkü sistem sadece algılamakla kalmaz, aynı zamanda çevresini de etkiler.

Bu aşamada LED, aydınlatma amacıyla değil, durum göstergesi olarak kullanılmaktadır. Öğrenciler, LED'in durumunun programın iç mantığının bir temsili olduğunu, çünkü çıktının her zaman bir karar sürecinin sonucu olduğunu anlarlar. Fiziksel geri bildirim, anlama ve hata ayıklamada önemli bir rol oynar.

LED'i kontrol ederken, öğrenciler tüm kontrol zincirini ilk kez deneyimliyolar:

1. Giriş olayı (örneğin, bir düğmeye basma),
2. İşleme / karar (koşul kontrolü, durum kontrolü),
3. Çıkış müdahalesi (LED'i açma veya kapatma).

Bu zincir, daha sonraki daha karmaşık sistemler (motor kontrolü, röleleme, otomasyon) için soyut bir model görevi görür.

Eğitimsel açıdan bakıldığında, dijital bir çıktı olarak LED, aşağıdaki cihazların anlaşılmasına zemin hazırlar:

- röle modülleri (yüksek güçlü cihazların anahtarlanması),
- zil / sesli uyarıcılar,
- trafik ışıkları (kırmızı-sarı-yeşil),
- Dijital ekranlar için kontrol hatları.

Öğrenciler böylece LED'in aslında basitleştirilmiş bir model olduğunu ve arkasında endüstriyel aktüatörlerde olduğu gibi aynı kontrol prensibinin yattığını anlayacaklardır. Uygulaması şu şekilde gelişir:

- neden-sonuç düşüncesi,
- Sistem yaklaşımı (girdi → işleme → çıktı),
- Görsel geri bildirim yorumlamak,
- Soyutlama yeteneği, daha sonra fiziksel cihazlardan vazgeçmemizi ve simüle edilmiş sistemleri anlamamızı sağlar.

Bu aşamada öğrenciler, otomatik bir sistemin "düşünmediğini" ancak önceden tanımlanmış kurallara göre tepki verdiğini ve her durumda çıktının programın mantıksal durumunun bir sonucu olduğunu anlarlar.

Aydınlatma kontrolü – giriş ve çıkışın bağlanması

Projenin bir sonraki adımı, basmalı düğmeyi ve LED'i basit ama eksiksiz bir kontrol mantığıyla birbirine bağlamaktır. Bu görev, girdi-işleme-çıkıttı düşünme modelini birleşik bir sistem olarak sunan ilk adımdır.

Aydınlatma kontrol mantık modeli aşağıdaki unsurlardan oluşmaktadır:

- Olay: basma düğmesinin durumundaki değişiklik,
- Karar: LED'in mevcut durumunu kontrol etmek,
- Müdahale: LED'i açın veya kapatın.

Bu yapı, teknik olarak uygulaması basit olsa bile, eksiksiz bir otomasyon döngüsünü zaten tanımlamaktadır.

Öğrencilere, görevin kendisinin bir mikrodenetleyici gerektirmediği konusunda bilgi verilmelidir. Bir basma düğmesi ve bir LED, basit bir kablolu devre olarak bağlanabilir; burada basma düğmesi devreyi doğrudan kapatır ve LED fiziksel anahtarlamaya anında tepki verir. Ancak bu otomasyon değil, doğrudan donanım işlemidir.

Mikrokontrolcünün kullanılmasının amacı görevi "çözmek" değil, tüm etki zincirini modellemektir; yani fiziksel bir olayı dijitalleştirmek, yazılımda mantıksal kararlar almak ve programlanmış müdahaleleri gerçekleştirmektir. Bu ayırım, öğrencilerin sistemi anlamalarını geliştirmede kilit öneme sahiptir.

Blok tabanlı bir ortamda programlama öğeleri

Blok tabanlı programlama sırasında aşağıdaki öğeler ortaya çıkar:

- Olay tetikleyici bloklar (düğmeye basma veya durum değişikliği),
- dijital çıkış kontrol blokları (LED açma/kapama),
- LED'in mevcut durumunu saklayan isteğe bağlı durum değişkeni.

Değişkenin eklenmesi teknik bir kısıtlama değil, olayı (ne oldu) ve durumu (sistemin mevcut durumu nedir) birbirinden ayıran bilinçli bir tasarım kararıdır.



Şekil 22: Değişken kullanımına sahip bir aydınlatma kontrol bloğu programının tam yapısı (Kaynak: <https://docs.keyestudio.com>)

Görev sırasında öğrenciler, kontrol cihazının önceki durumları otomatik olarak "hatırlamadığını", bunun yerine durumu korumanın bir programlama sorumluluğu olduğunu, yani olay ve durumun ayrı ayrı ele alınması gereken kavramlar olduğunu hızla fark ederler.

Bu deneyim, çok durumlu otomasyon, zamanlanmış kontrol sistemleri ve ağ tabanlı, eşzamansız sistemler için doğrudan bir hazırlıktır.

Eğitimsel yansıma

Basitliğine rağmen, aydınlatma kontrol projesi olağanüstü bir eğitim değerine sahiptir.

Öğrenciler, dijital sistemlerin düşünmediğini, aksine çalıştığını, yani sistemin çalışmasının tamamen tasarlanmış mantığa bağlı olduğunu ve bu nedenle görünüşte önemsiz bir görevin bile bilinçli mühendislik kararlarına dayandığını deneyimlerler.

Kablolu bir çözüm ile kontrolcü tabanlı bir uygulamayı karşılaştırmak, öğrencilerin otomasyonun araçlarla değil, zihniyetle ilgili olduğunu anlamalarına yardımcı olur.

Projenin bu bölümü, daha sonraki sensör tabanlı, koşullu ve ağ tabanlı kontrol sistemlerini anlamak için sağlam bir temel oluşturuyor ve daha karmaşık IoT çözümlerine doğal bir geçiş sağlıyor.

3.16. Proje 2 – Işık Kontrol Sistemi

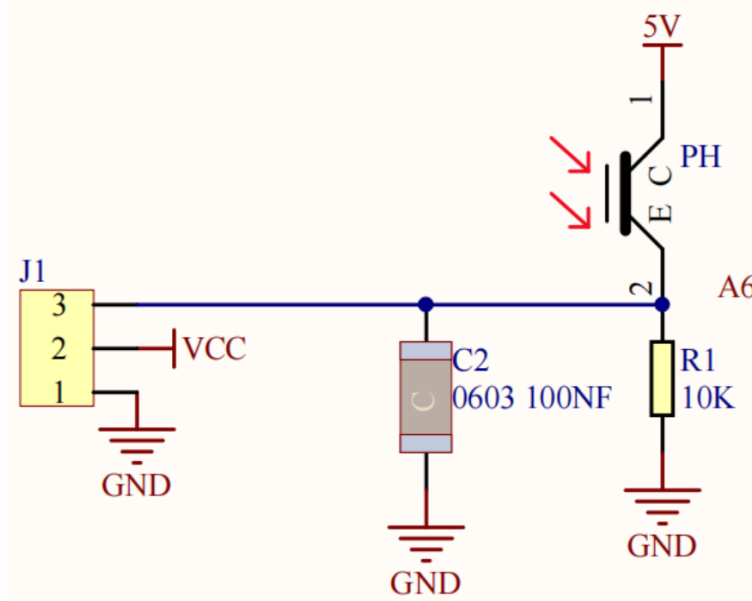
Bu proje, tamamen dijital mantığa dayalı önceki çalışmalardan temel bir farklılık göstererek, analog niceliklerin ölçülmesini ve işlenmesini ilk kez sunmaktadır. Burada öğrenciler artık iki durumlu sinyallerle çalışmak yerine, sürekli değişen fiziksel nicelikleri yorumlayıp kontrol kararlarına dönüştürmektedirler.

Işık sensörü (LDR) çalışması – analog giriş

Bir LDR (Işığa Bağımlı Direnç), üzerine düşen ışığın şiddetine bağlı olarak direnç değeri değişen, yarı iletken tabanlı bir dirençtir:

- güçlü aydınlatma → daha düşük direnç,
- Düşük ışık → daha yüksek direnç.

Bu değişim sürekli olduğundan, doğası gereği analogdur. ESP32 parlaklığı doğrudan "görmez", ancak analog girişindeki LDR ile oluşturulan voltaj bölücü devresinde oluşan voltajı ölçer.



Şekil 23: LDR karakteristiği ve gerilim bölücü bağlantı şeması (ESP32 analog girişi) (Kaynak: <https://docs.keyestudio.com>)

Analog giriş:

- Yüksek veya düşük bir değer vermez.
- ancak sayısal bir aralık içinde (örneğin 0–4095) bir ölçüm sonucu,
- Bu, ışığın yoğunluğuyla orantılıdır.

Projeye katılan öğrenciler, ölçüm değerinin mutlak değil, bağlama bağlı olduğu, sensör ve elektroniklerin doğruluğunun sınırlı olduğu ve ölçülen verilerin yorumlanmasının her zaman bağlam gerektirdiği gerçeğiyle karşı karşıya kalmaktadır.

Karar verme süreci ve müdahale – analog işleme Işıqla

kontrol edilen bir sistemin en basit mantıksal yapısı:

1. Işık yoğunluğu ölçümü (analog giriş),
2. Bir limit değerle karşılaştırma,
3. Bir müdahale gerçekleştirin (aydınlatmayı açın veya kapatın).

Bu model hala analog bir girdiye dayanarak dijital bir karar üretiyor.



Şekil 24: Analog giriş için koşullu kontrol bloğu yapısı (Kaynak: <https://docs.keyestudio.com>)

Analog ve dijital geçişin eğitimsel önemi

Bu adım, analog sinyalin sayısallaştırılmasına, sınır değerinin seçilmesi sorununa ve gürültülü, dalgalanan ölçüm verilerinin işlenmesine odaklanmaktadır.

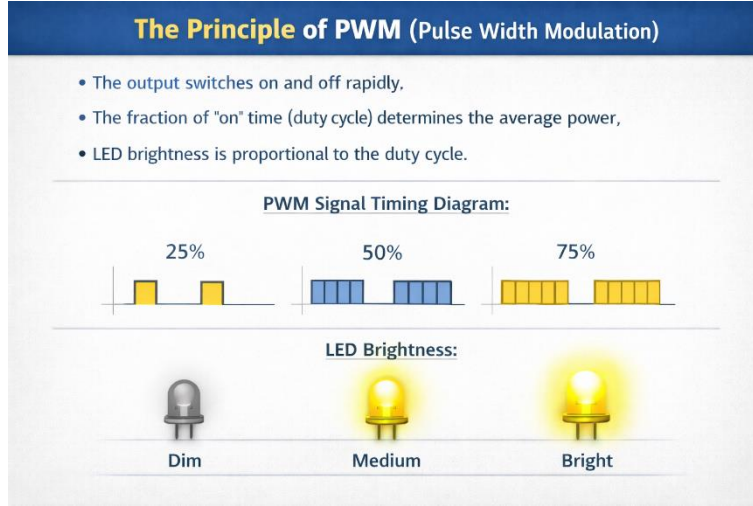
Öğrenciler, çok düşük veya çok yüksek bir limitin kararsız çalışmaya neden olabileceğini, bu nedenle histerezis veya filtreleme gerekebileceğini deneyimleyecek ve böylece mühendislik kararlarının sistem davranışını etkilediğini anlayacaklardır.

Analog çıkış – PWM tabanlı karartma

Projenin bir sonraki aşamasında, müdahale artık ikili değil, sürekli. LED'in parlaklığı sadece açılıp kapatılmakla kalmaz, orantılı olarak düzenlenir.

PWM (Darbe Genişliği Modülasyonu), dijital çıkışla analog bir etki elde etmemizi sağlayan bir tekniktir:

- Çıkış çok hızlı bir şekilde açılıp kapanıyor.
- Çalışma süresinin oranı (görev döngüsü), ortalama performansı belirler.
- LED'in parlaklığı doluluk oranıyla orantılıdır.



Şekil 25: PWM sinyal zamanlama diyagramı ve LED parlaklık değişimi (Kaynak: Yapay zeka tarafından oluşturulmuştur)

Bu projede öğrenciler orantılı bir ilişki kuracaklar:

- Giriş: Ölçülen ışık yoğunluğu (analog değer),
- İşleme: ölçeklendirme, dönüştürme,
- Çıkış: PWM sinyali ile kontrol edilen LED parlaklığı.

Bu yaklaşım zaten bir süreç kontrolü bakış açısını yansıtıyor ve endüstriyel düzenlemelerin doğrudan öncüsü niteliğinde.

Eğitimsel yansıma

Işık kontrollü sistem projesi, öğrencilerin düşünme becerilerinin geliştirilmesinde önemli bir rol oynamaktadır:

- Dijital ve analog nicelikler arasında ayırım yapmak,
- Gerçek dünyanın sürekli, kontrolörlerin ise ayrık olduğunu anlayın.
- Orantılı düzenleme ilkesini tanımak,
- Daha iyi bir işletme için daha fazla planlama gerektiğini deneyimlediler.

Bu bölüm, daha sonraki sıcaklık kontrolü, motor kontrolü ve enerji optimizasyon sistemlerine doğal bir köprü oluşturmakta ve mühendislik düşüncesini derinleştirmede önemli bir adım teşkil etmektedir.

3.17. Proje 3 – Mesafe Algılama ve Otomatik Müdahale (Akıllı Besleme Sistemi)

Akıllı Besleme Sistemi, öğrencilerin zaman tabanlı bir sensör ve mekanik bir aktüatörü birleştirerek çalışan bir otomasyon oluşturdukları entegre bir projedir. Basit ama gerçek hayattaki bir durumu modellemektedir: bir nesnenin yakınlığını algıladığında, mekanik bir işlem gerçekleştirilir.

Bu, otomatik sistemlerin temel çalışma zincirini eksiksiz olarak kapsamaktadır:



Akıllı Besleme Sisteminin işlevsel amacı, bir nesnenin (örneğin elin) yaklaşımını algılayan, ölçülen mesafeye göre karar veren ve bir servo motor kullanarak mekanik bir hareket (örneğin kapağı açma/kapama) gerçekleştiren otomatik bir besleyicinin çalışmasını sağlamaktır.

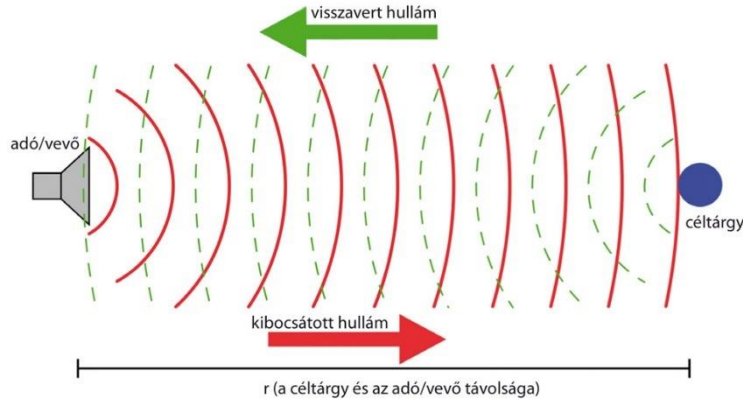
İşlevi anlamak basit olsa da, aynı zamanda karmaşık teknik ve mühendislik düşünme biçimi gerektiriyor.

Ultrasonik mesafe sensörü

Bu sensörün görevi, zaman ölçümü kullanarak bir nesneye olan mesafeyi belirlemektir. Sensörün çalışması dijital kontrol (tetikleme/yankı) prensibine dayanır ve mesafe değeri hesaplanır.

Eğitimsel önemi:

- Yeni bir ölçüm prensibi hakkında bilgi edinmek,
- Ham ölçüm verilerinin ve yorumlanan bilgilerin ayrılması.



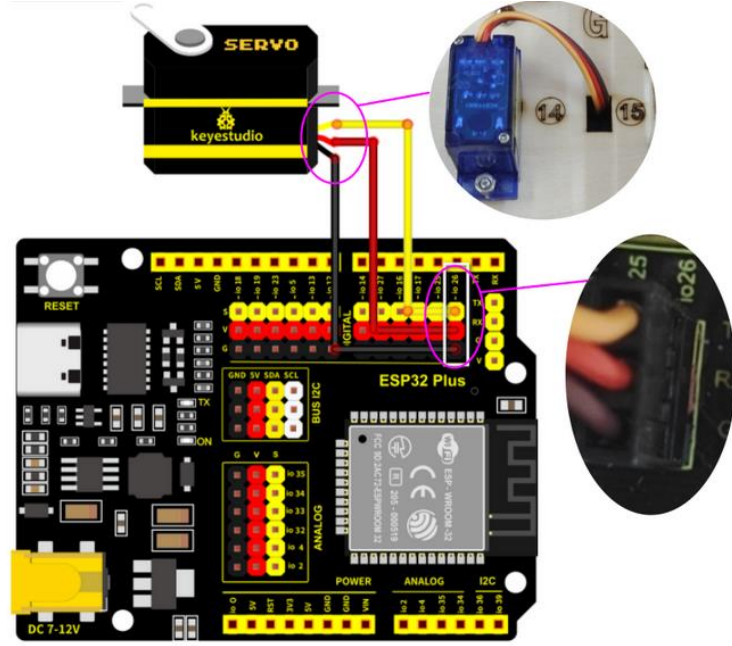
Şekil 26: Ultrasonik mesafe ölçümünün şematik diyagramı (Kaynak: <https://docs.keyestudio.com>)

Servo motor, mekanik aktüatör olarak

Servo motor, kontrol kararını verir. fiziksel uygulama için cevap.
PWM sinyali ile kontrol edilebilir, belirli açışal pozisyonlara ayarlanabilir ve hassas ve tekrarlanabilir hareket sağlar.

Eğitimsel önemi:

- Elektronik ve mekaniği birbirine bağlamak,
- Kontrolün doğrudan fiziksel etkisini deneyimlemek.



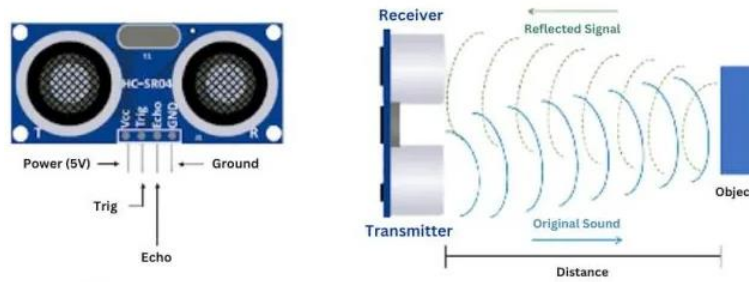
Şekil 27: Servo motor yapısı ve kontrol prensibi (Kaynak: <https://docs.keyestudio.com>)

Ultrasonik mesafe ölçümünün prensibi

Ultrasonik ölçüm süreci:

1. Sensör bir ses darbesi yayar.
2. Ses nesneden yansır
3. Kontrol cihazı geçen süreyi ölçer.
4. Mesafe hesaplama yoluyla belirlenebilir.

Bu, analog bir voltaj ölçümü değil, dijital sinyallere dayalı, zaman tabanlı bir ölçümdür ve hesaplama adımı gerektirir.

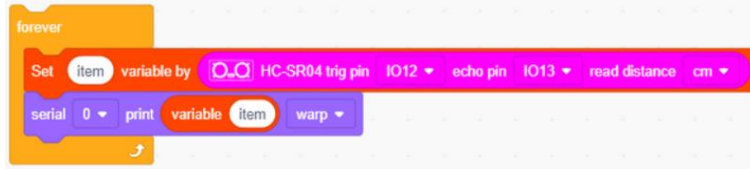


Şekil 28: Ultrasonik ölçüm süreci (Kaynak: <https://docs.keyestudio.com>)

Sensör verilerinin işlenmesi ve karar mantığının oluşturulması sürecine yardımcı olmak için, ölçülen mesafe kontrol kararının temelini oluşturur. Mantıksal süreç şu şekildedir:

1. mesafeyi ölçmek
2. Limit değerle karşılaştırma
3. karar vermek
4. bir müdahale başlatmak

Bu yapı zaten tamamen otomatikleştirilmiş bir süreci tanımlıyor.



Şekil 29: Blok bazında mesafe verilerinin işlenmesi (Kaynak: <https://docs.keyestudio.com>)

Servo motor kontrolü

Servo motor, darbe genişliğinin açısal konumu belirlediği ve kontrolün güç değil konum kontrolü yaptığı bir PWM sinyali ile kontrol edilir.

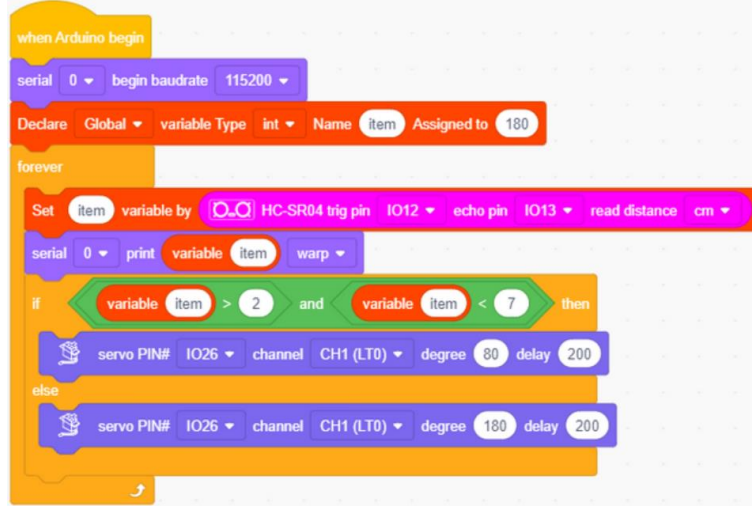
Bu, LED'lerin PWM karartmasıyla karşılaştırıldığında önemli bir farktır.



Şekil 30: PWM sinyali ile servo pozisyonu arasındaki ilişki (Kaynak: <https://docs.keyestudio.com>)

Akıllı Besleme Sistemi – entegre çalışma

Akıllı Besleme Sisteminin çalışması, birleşik bir mantıksal bütünlük oluşturur: ölçülen mesafe eşik değer içindeyse, sistem bir karar verir, servo motor hareket eder ve mekanik bir işlem gerçekleşir.



Şekil 31: Akıllı Besleme Sistemi tam blok programı (Kaynak: <https://docs.keystudio.com>)

Eğitimsel yansıma

Akıllı Beslenme Sistemi projesi, özellikle aşağıdaki amaçlarla öğrenciler için uygundur:

- Sistem düzeyinde kontrolün işleyişini görün,
- Zaman bazlı ölçümün özelliklerini anlamak,
- Karar mantığı ile mekanik etki arasındaki ilişkiyi kavrayın,
- Bilinçli mühendislik düşüncesini geliştirmek.

Bu proje, otomatik sistemlerin işleyişinin tek tek unsurlarda değil, bunların birbirleriyle olan bağlantılarında ve tasarlanmış mantıkta yattığını açıkça göstermektedir.

3.18. Proje 4 – Sıcaklık Kontrol Sistemi

Sıcaklık Kontrol Sistemi, endüstriyel otomasyon ve bina yönetim sistemlerinde temel olan klasik bir kontrol görevini modellemektedir. Proje, çevresel bir niceliğin sürekli ölçümüne, görüntülenmesine ve ardından bir müdahale cihazının otomatik kontrolüne odaklanmaktadır.

Projenin tüm operasyonel zinciri:



Bu yapı, öğrencilerin anlamasını kolaylaştırırken aynı zamanda gerçek endüstriyel düzenleme mantığını da yansıtır.

Sıcaklık ve nem sensörü (DHT sensörü)

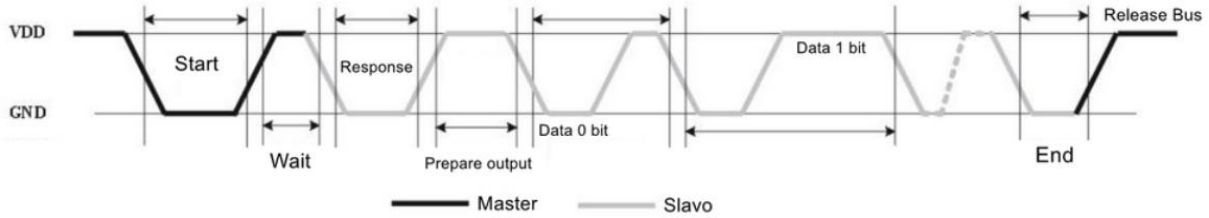
DHT sensörü, sıcaklığı ve bağıl nemi ölçen ve verileri dijital veri paketi şeklinde kontrol ünitesine ileten dijital bir çevre sensörüdür.

Elektronik özellikler:

- tek bir veri hattı üzerinden iletişim kurar,
- Ölçüm sonucu analog bir voltaj değil, işlenmiş bir dijital değerdir.
- Sensör, dahili bir zamanlama ve kontrol mekanizması kullanır.

Eğitimsel önemi:

- Analog ölçümü dijital veri iletiminden ayırır.
- Sensörlerin bazılarının zaten "işlenmiş" veri sağladığına dikkat çekiyor.
- Örnekleme ve güncelleme zamanı kavramlarını tanıtır.



Şekil 32: DHT sensörünün çalışma prensibi ve veri iletim süreci (Kaynak: <https://docs.keyestudio.com>)

LCD 1602 ekran - ölçüm verilerinin gösterimi

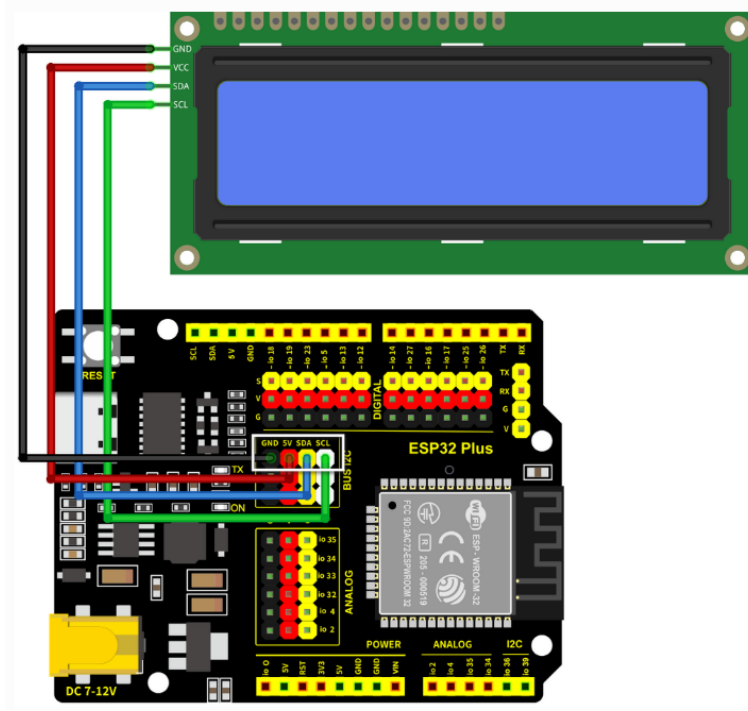
Sıcaklık kontrol sisteminin önemli bir unsuru kullanıcı geri bildirimidir. Bu rolü, ölçülen verileri sayısal biçimde görüntüleyen ve sistemin mevcut durumunun sürekli olarak izlenmesine olanak sağlayan LCD 1602 ekran üstlenir.

İşlevsel rol:

- Sıcaklık ve nem göstergesi,
- Çalışma durumu geri bildirim (örneğin fan aktif/pasif),
- Teşhis bilgilerini görüntüle.

Eğitimsel önemi:

- Ölçüm ve gösterimin birbirinden ayrıştırılmasını vurgular.
- Bu durum, sistemin işleyişinin "görünmez" olmadığını göstermektedir.
- Hata ayıklamayı ve sistem anlayışını destekler.



Şekil 33: LCD 1602 bağlantı şeması (Kaynak: <https://docs.keyestudio.com>)

Müdahale unsuru olarak fan

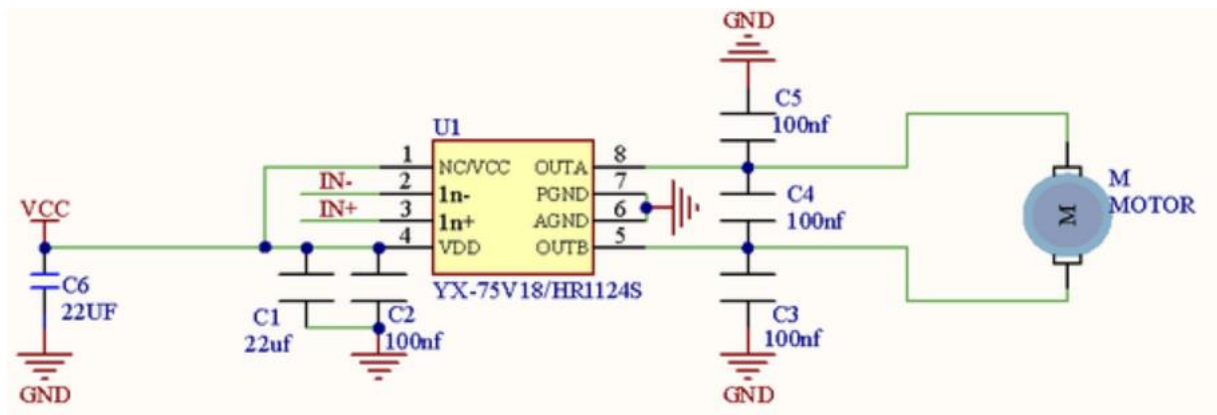
Sistemde, çevresel koşulları değiştirmek için kullanılan aktif bir aktüatör olan fan, kontrol ünitesi tarafından doğrudan veya dijital çıkış üzerinden bir sürücü vasıtasıyla çalıştırılır.

Kontrol mantığı:

- Ölçülen sıcaklık belirlenen limit değerini aşarsa → fan devreye girer.
- Sıcaklık limit değerinin altına düştüğünde → fan kapanır.

Eğitimsel önemi:

- Basit iki durumlu kontrolü gösterir,
- Limit tabanlı kontrolün avantajlarını ve sınırlamalarını anlamak,
- Bu, ilerleyen aşamalarda histerezis ve PID kontrolünün anlaşılmasına hazırlık niteliğindedir.



Şekil 34: Fan kontrolü kablolu şeması (Kaynak: <https://docs.keyestudio.com>)

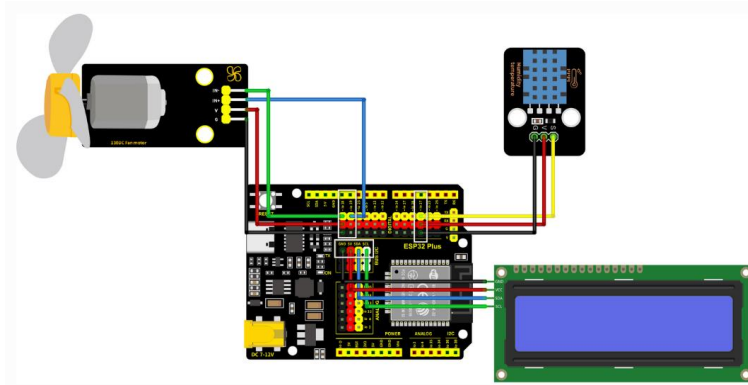


Şekil 35: Fan kontrol bloğu mantık diyagramı (Kaynak: <https://docs.keyestudio.com>)

Sıcaklık Kontrol Sisteminin Entegre Çalışması

Sıcaklık Kontrol Sistemi projesinin tamamı aşağıdaki mantığa göre çalışmaktadır:

1. DHT sensörü bir ölçüm gerçekleştirir.
2. Veriler işlenir
3. Ölçülen değer LCD ekranda görüntülenir.
4. Kontrolör, değeri limit değeriyle karşılaştırır.
5. taraftar durumu buna göre değişir



Şekil 36: Sıcaklık Kontrol Sistemi bağlantı şeması (Kaynak: <https://docs.keyestudio.com>)

```
when Arduino begin
  init lcd I2C address 0x27
  clear lcd
  set lcd back light on
  set lcd cursor position x: 0 y: 0
  lcd print Temp:
  set lcd cursor position x: 0 y: 1
  lcd print Hum:
  init dht11 1 pin IO17 mode dht11
  Declare Global variable Type int Name temp Assigned to 0
  Declare Global variable Type int Name hum Assigned to 0
  forever
    Set temp variable by dht11 1 read temperature
    Set hum variable by dht11 1 read humidity
    set lcd cursor position x: 5 y: 0
    lcd print variable temp
    set lcd cursor position x: 5 y: 1
    lcd print variable hum
    if variable temp > 29 or variable hum > 80 then
      fan INA# IO18 State HIGH INB# IO19 analogWrite 70
    else
      fan INA# IO18 State LOW INB# IO19 analogWrite 0
    wait 1 seconds
  wait 0.5 seconds
```

Şekil 37: Bir sıcaklık kontrol sisteminin tam blok programı (Kaynak: <https://docs.keyestudio.com>)

Bu yapı, gerçek bir düzenleyici sistemin basitleştirilmiş bir modelidir.

Eğitimsel yansıma

Sıcaklık Kontrol Sistemi projesi öğrencilere şu fırsatları sunmaktadır:

- Otomasyonda çevresel verilerin rolünü anlamak,
- Ölçüm, görselleştirme ve müdahale arasındaki farkı ayırt etmek,
- Limit tabanlı düzenlemenin işleyişini ve sınırlamalarını anlamak,
- Görünüşte basit bir görevi sistematik bir şekilde düşünün.

Bu, akıllı ev ve akıllı çiftlik konseptiyle iyi örtüşüyor ve daha sonraki, daha karmaşık kontrol ve ağ tabanlı çözümleri anlamak için sağlam bir temel oluşturuyor.

3.19. Proje 5 – Otomatik Sulama Sistemi

Otomatik Sulama Sistemi projesi, öncelikle kontrol ünitesinin elektriksel kapasitesinin aktüatörü doğrudan çalıştırmak için yetersiz olduğu mühendislik sorununu vurgulamaktadır. Görevin amacı, karar verme ve fiziksel uygulama arasında güvenli, endüstriyel tarzda bir arayüzün sağlandığı otomatik bir sulama sistemi modeli oluşturmaktır.

Projenin operasyonel zinciri:



Bu yapı, tüm gerçek endüstriyel otomasyon sistemleri için temeldir.

Mikrokontrolcü çıkışlarının sınırlamaları

Akıllı Çiftlik Kiti kontrol ünitesi, dijital çıkışları düşük voltaj seviyelerinde (tipik olarak 3,3 V veya 5 V DC) çalışan ve yalnızca küçük akımlar (birkaç on mA mertebesinde) sağlayabilen mikrodenetleyici tabanlı bir sistemdir.

Bu, LED'leri, daha küçük elektronik devreleri veya mantık sinyallerini kontrol etmek için yeterlidir.

Ancak, motorları, pompaları, vanaları veya şebeke gerilimiyle çalışan cihazları (örneğin 230 V AC) doğrudan çalıştırmak için yeterli değildir.

Eğitimsel açıdan bakıldığında, bu çok önemli bir nokta: öğrenciler kontrol ve performansın aynı kavram olmadığını anlıyorlar.

Gerilim seviyeleri ve güç farklılıkları

Otomatik sulama, bir sistemde birden fazla voltaj seviyesinin bir arada bulunmasına iyi bir örnektir:

- Kontrol tarafı: düşük voltaj (5 V DC),
- Aktüatör tarafı: daha yüksek voltaj veya güç (örneğin 12 V DC, 24 V DC, 230 V AC'ye kadar).

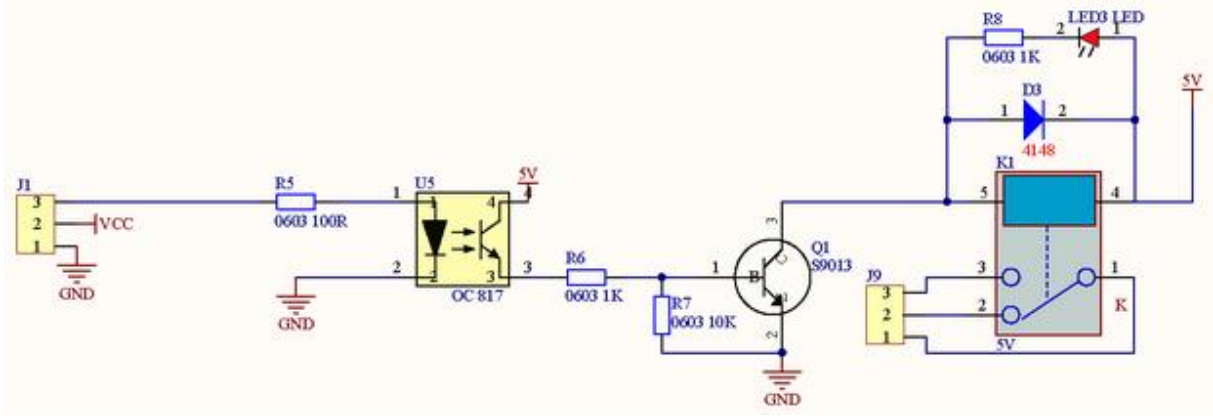
Proje boyunca, bir motorun neden doğrudan mikrodenetleyici çıkışına bağlanamayacağına, neden bir güç eşleştirme elemanına ihtiyaç duyulduğuna ve iki dünyayı nasıl güvenli bir şekilde ayırabileceğimize vurgu yapılmaktadır.

Röle modülü, güç eşleştirme elemanı olarak görev yapar.

Otomatik sulama sisteminin temel elemanı, mikrodenetleyiciden gelen düşük güçlü bir sinyalin daha yüksek güçlü bir devreyi kontrol etmesini sağlayan röle modülüdür.

Rölenin çalışma prensibine göre, kontrol ünitesi bir bobini anahtarlar, bobinin manyetik alanı kontakları mekanik olarak kapatır veya açar ve kontaklar ayrı bir devrede çalışır.

Bu, galvanik izolasyon sağlayarak kontrol elektroniğini korur, sistemin çalışma güvenilirliğini artırır ve endüstriyel otomasyon ilkelerine uygundur.



Şekil 38: Röle modülü kontakları ve çalışma mantığı (Kaynak: <https://docs.keyestudio.com>)

Endüstriyel Paralel – PLC Çıkış Çözümleri

Röle kullanımı bir "hobi çözümü" değil, endüstriyel bir prensiptir. Programlanabilir mantık kontrolcülerinin (PLC) çıkışları da tipik olarak şöyledir:

- röle çıkışları,
- transistör (PNP/NPN) çıkışları,
- Triyak çıkışları alternatif akım için daha az sıklıkla kullanılır.

Röleli PLC çıkışlarının avantajları:

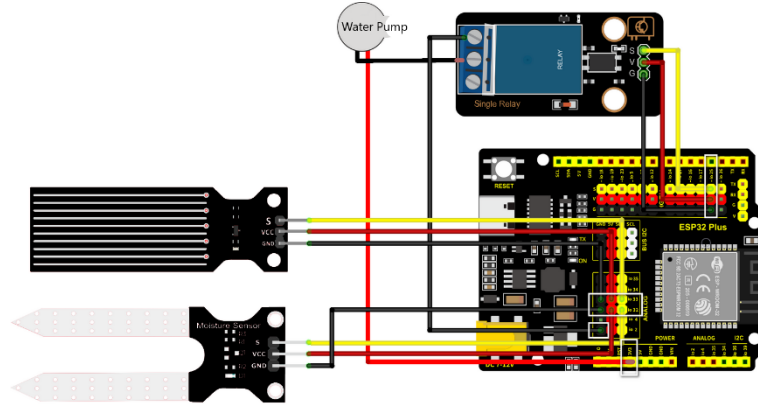
- geniş voltaj aralığı,
- AC ve DC yüklerin anahtarlanması,
- Elektriksel izolasyon.

Akıllı Çiftlik projesi, kullanılan araç eğitim amaçlı olsa bile, şu an itibarıyla endüstriyel bir zihniyeti model alıyor.

Otomatik Sulama Sisteminin Kontrol Mantığı

Sistemin çalışma prensibi basit olmasına rağmen, tam otomasyon sağlıyor:

1. Kontrolör giriş koşullarını değerlendirir.
2. bunları bir sınır değeriyle karşılaştırır,
3. Sulama ihtiyacına karar verir
4. Röleyi dijital çıkış aracılığıyla etkinleştirir
5. Röle, sulama ünitesini devreye sokar.



Şekil 39: Otomatik Sulama Sisteminin Bağlanması (Kaynak: <https://docs.keyestudio.com>)

```
when Arduino begin
  init lcd I2C address 0x27
  clear lcd
  set lcd back light on
  Declare Global variable Type int Name waterlevel Assigned to 0
  Declare Global variable Type int Name soilhum Assigned to 0
  forever
    Set waterlevel variable by read the value of water level IO33
    Set soilhum variable by read the value of soil moisture IO32
    clear lcd
    set lcd cursor position x: 0 y: 0
    lcd print WaterLevel:
    set lcd cursor position x: 0 y: 1
    lcd print SoilHum:
    set lcd cursor position x: 11 y: 0
    lcd print variable waterlevel
    set lcd cursor position x: 11 y: 1
    lcd print variable soilhum
    if 700 > variable waterlevel then
      Tone PIN# IO16 frequency NOTE_E5 duration 131
      Tone PIN# IO16 frequency NOTE_A5 duration 131
      Tone PIN# IO16 frequency NOTE_B5 duration 131
      NoTone IO16
    if 1200 > variable soilhum then
      Tone PIN# IO16 frequency NOTE_D3 duration 131
      Tone PIN# IO16 frequency NOTE_F3 duration 131
      Tone PIN# IO16 frequency NOTE_A3 duration 131
      NoTone IO16
    if variable soilhum < 1200 and variable waterlevel > 700 then
      relay pin IO25 output HIGH
      wait 0.4 seconds
      relay pin IO25 output LOW
      wait 0.7 seconds
      wait 0.5 seconds
```

Şekil 40: Otomatik Sulama Sistemi blok tabanlı kontrolü (Kaynak: <https://docs.keystudio.com>)

Bu yapı, endüstriyel proses kontrol döngülerine doğrudan eşlenebilir.

Eğitim ve mühendislik dersleri

Otomatik sulama projesi sırasında öğrenciler şunları yapacaklardır:

- Mikrokontrolcülerin fiziksel sınırlamalarını anlamak,
- Performans uyumunun gerekliliğini kabul etmek,
- galvanik izolasyon kavramıyla karşılaşmak,
- Endüstriyel otomasyon prensiplerine benzer çözümler kullanıyorlar.

Bu proje, özellikle kontrol kararlarının fiziksel sonuçları konusunda farkındalığı artırma, PLC tabanlı düşünmeye hazırlık yapma ve daha sonra ağ bağlantılı ve endüstriyel iletişim kullanan sistemleri anlama temelini atma konusunda oldukça güçlüdür.

Bu bölüm, basit IoT otomasyonu dünyası ile profesyonel endüstriyel kontrol sistemleri dünyası arasında doğal bir geçiş oluşturmaktadır.

3.20. Proje 6 – WiFi kontrollü Akıllı Çiftlik Sistemi

WiFi kontrollü Akıllı Çiftlik projesi, yerel olarak çalışan otomasyondan ağ tabanlı, uzaktan izlenen sistemlere geçişi temsil etmektedir. Bu aşamada öğrenciler artık sadece sensörleri ve aktüatörleri kontrol etmekle kalmıyor, aynı zamanda eksiksiz bir IoT tabanlı izleme modeli uyguluyorlar.

Projenin temel fikri, sistemin şu şekilde çalışmasıdır:

- ağa bağlanır,
- Uzak bir istemciye veri sağlar,
- HMI web arayüzü üzerinden kontrol edilebilir.

Bu işlevsellik, basit gömülü sistemlerin ötesine geçerek sizi gerçek endüstriyel ve akıllı cihaz çözümlerine daha da yaklaştırıyor.

IoT sisteminde WiFi bağlantısının rolü

Smart Farm Kit ESP32 kontrol ünitesi, kablosuz ağa doğrudan bağlantı sağlayan dahili bir WiFi modülüne sahiptir. Bu projede WiFi bağlantısı sadece teknik bir ekleme değil, sistemik bir bakış açısı değişikliğidir: kontrol ve izleme, yerel, kablolu ortamdan çıkarak ağ bağlamına girer.

Sisteme ağ üzerinden bağlanarak, sistem izole bir kontrol ünitesi gibi çalışmaz; müdahaleler doğrudan fiziksel girdiye değil, ağ isteklerine bağlı olduğundan, sensör verilerine bir tarayıcı aracılığıyla uzaktan erişilebilir.

Eğitimsel açıdan bakıldığında, Nesnelerin İnterneti (IoT) kavramı burada somut hale geliyor: cihaz kendi başına var olmuyor, ancak ağa bağlı bir sistemde aktif bir düğüm görevi görüyor.

Öğrenciler, WiFi bağlantısı kurarken Erişim Noktası gibi temel ağ kavramlarıyla tanışacaklardır. ESP32, erişim noktası görevi gören mevcut bir WiFi ağına (örneğin okul veya ev yönlendiricisine) bağlanır; böylece kontrol cihazının istemci olarak bağlandığını ve ağın iletişim için ortam sağladığını anlarlar.

Ağa bağlı bir ESP32, ağda cihazı tanımlayan, tarayıcıdan erişime olanak sağlayan ve daha sonraki istemci yönetim sistemi iletişimi için temel oluşturan benzersiz bir IP adresi alır.

Öğrenciler ilk kez fiziksel bir cihazın, diğer bilgisayarlar veya gözetim sistemlerine benzer şekilde ağ üzerinde bir "adrese" sahip olduğunu deneyimliyorlar.

Proje sırasında, ESP32'nin, kullanıcının bilgisayarının veya mobil cihazının aynı ağda bulunduğu, yani iletişimin ancak bu cihazların ağ düzeyinde "birbirlerini görebilmeleri" durumunda mümkün olduğu anlaşılmıştır. Kontrol ve veri alma işlemleri ağ mesajları şeklinde gerçekleşir.

Eğitimsel önemi

WiFi bağlantısının kullanıma sunulması sırasında öğrenciler:

- Temel ağ düşünme becerisine sahipler.
- Modern sistemlerde IP tabanlı erişimin rolünü anlamak,
- Yerel ve ağ tabanlı kontrol arasındaki farkı kavrayın.
- IoT sistemlerinin temel çalışma koşullarından birini deneyimleyin.

Bu bilgi sizi doğrudan şunlara hazırlar:

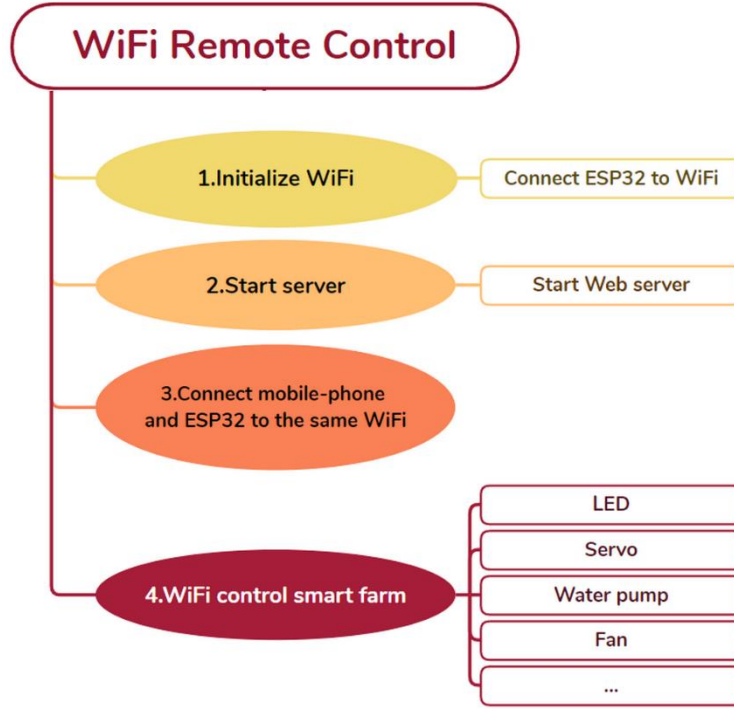
- Web yönetim arayüzünü anlamak,
- Sonraki istemci denetim sistemi mimarilerinin işlenmesi,
- ve endüstriyel ve dağıtık sistemlere ağ tabanlı bir yaklaşımın benimsenmesi.

Gömülü web izleme sistemi işletimi

Fabrika ortamında kullanılan Akıllı Çiftlik çözümünde, web yönetim arayüzü doğrudan ESP32 üzerinde çalışan web yönetim sistemi kullanılarak uygulanmaktadır. Bu, kontrol ünitesinin kendisinin web sitesini sağladığı, dolayısıyla sistemde harici bir yönetim sistemi veya bilgisayar bulunmadığı, bunun yerine istemcinin (tarayıcının) doğrudan ESP32'ye bağlandığı anlamına gelir.

Bu çözüm, tek bir araç içinde yer aldığı için eğitim açısından son derece sezgiseldir:

- veri toplama,
- işleme,
- ekran,
- kontrol.

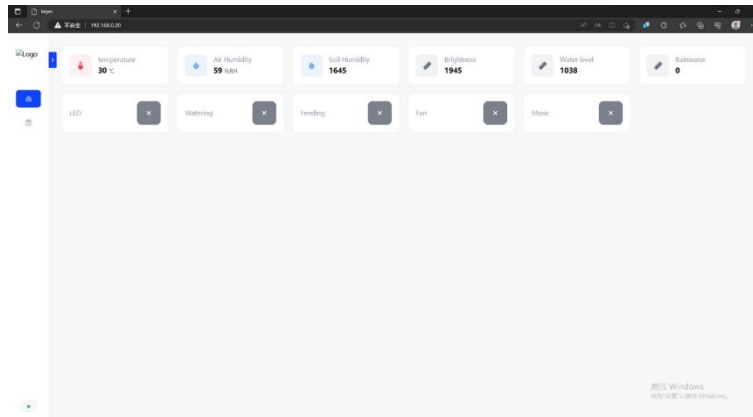


Şekil 41: ESP32 üzerinde çalışan bir web izleme sisteminin iletişim modeli (Kaynak: <https://docs.keyestudio.com>)

Web yönetim sayfası, sistemin kullanıcıya açılan "penceresidir". Fabrika kılavuzuna göre, arayüz tipik olarak aşağıdaki işlevleri yerine getirir:

- Mevcut sensör verilerini görüntüle,
- Çıktı durumuna ilişkin geri bildirim,
- Basit kontroller (düğmeler, anahtarlar) sağlamak.

Arayüz, grafiksel karmaşıklık açısından güçlü değil, ancak fiziksel sistemin durumunu gerçek zamanlı olarak yansıtma yeteneği açısından güçlüdür.



Şekil 42: Akıllı Çiftlik web yönetim sayfasının görünümü (Kaynak: <https://docs.keyestudio.com>)

Önemli bir eğitim unsuru, web izleme sisteminin sınırsız kaynaklarla çalışmadığını anlamaktır. ESP32 örneğinde, öğrenciler aşağıdaki faktörlerle karşı karşıya kalırlar:

- sınırlı bellek,
- sınırlı depolama alanı,
- Sınırlı sayıda eş zamanlı bağlantıyı yönetmek,

- Basit HTML yapıları kullanarak.

Bu durum, web sitesinin işlevselliğinin kasıtlı olarak basit tutulmasına ve dolayısıyla görüntülenen veri miktarının sınırlı olmasına yol açar; bu da ESP32'nin karmaşık web uygulamalarını çalıştırmak için uygun olmadığı anlamına gelir.

Bu farkındalık, mühendisliğin temel derslerinden biridir: tüm sistemler uzlaşmalar üzerine kuruludur.

HMI web arayüzünde görüntülenen kontrol elemanları, kullanıcının sisteme bir komut göndermesine olanak tanır; kontrol ünitesi bunu dijital bir talimat olarak yorumlar ve aktüatörler ağdan gelen sinyale yanıt verir.

Bu işlem, kullanıcının fiziksel olarak orada bulunmadığı, ancak kontrolün anında ve geri bildirim odaklı olduğu modern IoT sistemlerinin temel prensibini göstermektedir.

WiFi kontrollü Akıllı Çiftlik projesi kapsamında öğrenciler şunları yapacaklardır:

- Ağ tabanlı kontrolün özünü anlamak,
- Gömülü sistemlerin kaynak sınırlamalarını kabul etmek,
- Müşteri denetim sistemi modelinin temellerini deneyimleyin,
- Kontrol ve görüntüleme görevlerini birbirinden ayırırlar.

Projenin güçlü yönü şudur:

- Minimum cihaz gereksinimleriyle gerçek bir IoT deneyimi sunar.
- Fabrika çözümlerinin sınırlarını açıkça ortaya koymaktadır.
- Bu, daha sonraki, dışsal, sistematik, endüstriyel mimarileri anlamak için doğal bir hazırlık sağlar.

Bu bölüm, fabrika akıllı çiftlik sisteminin olanaklarının haritasını çıkarmayı tamamlıyor ve aynı zamanda kaynaklara göre ayrılmış ve endüstriyel bir modeli izleyen kendi sistem mimarinizin geliştirilmesi için temel oluşturuyor.

3.21. Özet pedagojik yorum

Blok tabanlı programlamaya dayanan Akıllı Çiftlik projesi bölümü, bağımsız bir öğrenme hedefi olarak değil, proje boyunca bilinçli olarak planlanmış bir pedagojik geçiş olarak karşımıza çıkmaktadır. Bu aşamada öğrenciler, daha karmaşık, endüstriyel tip sistemleri anlamak ve geliştirmek için daha sonra gerekli olan temel kavramları, ilişkileri ve çalışma prensiplerini edinirler.

Sistem düşüncesini teşvik etmek amacıyla, öğrenciler proje boyunca sürekli olarak aynı temel modelle çalışırlar:



Bu etki zinciri projenin her aşamasında ortaya çıkar:

- dijital girişler (ör. buton, anahtar),
- analog girişler (ör. LDR, potansiyometre),
- Zamana dayalı ölçümler (ultrasonik sensör),
- Dijital ve PWM tabanlı çıkışlar (LED, fan, servo motor),
- röleli güç anahtarlama (sulama sistemi),
- Ağ iletişimi ve web kontrolü (WiFi).

Öğrenciler, bireysel unsurların tek başına yorumlanamayacağını, aksine daha büyük bir sistemin işlevsel parçaları olduğunu yavaş yavaş fark ederler.

Analog ve dijital dünyalar arasındaki bağlantı

Önemli bir pedagojik kazanım, öğrencilerin aşağıdakiler arasında net bir ayrım yapabilmeleridir:

- dijital sinyaller (iki durumlu, olay odaklı),
- Analog nicelikler (sürekli değerler),
- Analogdan dijitale dönüştürmenin rolü,
- Limit değerleri, oransal kontrol ve PWM modülasyonu arasında.

Ölçüm gürültüsü kavramı, karar eşiğinin seçilmesinin önemi ve "ölçüm \neq bilgi" yaklaşımı, ışık kontrolü, sıcaklık kontrolü ve fan kontrolü sırasında ortaya çıkar.

Röle modülü ve motorlu aktüatörler kullanılırken, müdahale ve güç yönetimi anlayışı nedeniyle öğrenciler, mikrodenetleyicilerin çıkışlarının sınırlı akım ve güç iletimi kapasitesine sahip olduğu gerçeğiyle karşı karşıya kalırlar; bu nedenle kontrol ve güç iletimi ayrılır, böylece düşük voltajlı mantık sinyalleri (3,3–5 V DC) daha yüksek güçlü devreleri (örneğin 12 V DC, 230 V AC) kontrol edebilir.

Bu durum, PLC'lerin röle, transistör veya yarı iletken çıkışları aracılığıyla harici aktüatörleri kontrol ettiği endüstriyel kontrol sistemlerinin çalışma şekline doğrudan bir paralellik oluşturur.

Ağ düşüncesi ve IoT bakış açısı

WiFi tabanlı projede öğrenciler şunları deneyimleyecekler:

- Müşteri denetim sistemi modelinin temelleri,
- Gömülü web izleme sisteminin çalışması,
- Fiziksel kaynak kısıtlamalarının (bellek, depolama) sistem tasarımına etkisi,
- Uzaktan izleme ve kontrolün olanakları ve sınırlamaları.

Bu deneyim, endüstriyel bir ortamda görevlerin genellikle birbirinden ayrıldığını (saha düğümü kontrolörü – merkezi izleme – arka uç hizmetleri) anlamamızı sağlıyor.

Blok tabanlı programlama, bu bağlamda, programlama yaklaşımındaki bir değişikliğe hazırlanmaya yardımcı olur:

- Kontrol mantığını sözdizimsel zorluklardan ayırır.
- Devlet yönetimini, olayları ve koşulları görselleştirmek,
- Endüstriyel programlama paradigmalarına (örneğin IEC 61131-3, fonksiyon blokları, durum makineleri) kolayca uyarlanabilir.

Öğrenciler "Scratch öğrenmiyorlar", bunun yerine daha sonra C tabanlı mikrodenetleyici programlamada, endüstriyel PLC'lerde, Modbus tabanlı iletişimde ve dağıtık sistem mimarilerinde uygulanabilecek kontrol düşünme becerisini öğreniyorlar.

Özet

Bu proje aşaması, kendi tasarladığımız donanım bileşenlerinin geliştirilmesi, metin tabanlı programlamaya geçiş, endüstri odaklı, dağıtık mimarilerin anlaşılması ve ardından istemci-alan düğümü, arka uç-ön uç yapısına sahip sistemlerin geliştirilmesi için istikrarlı bir temel oluşturmaktadır.

Blok tabanlı ortam bu nedenle bir son durak değil, öğrencileri gerçek mühendislik ve endüstriyel sorunları ele almaya doğru kademeli olarak yönlendiren, bilinçli olarak inşa edilmiş bir öğrenme merdivenidir.

3.22. Bölümün pratik uygulaması

Akıllı Çiftlik Kitinin montajı ve devreye alınması, öğrenciler için ilk gerçek anlamda "uygulamalı" deneyim oldu. Fabrika kitinin çalışmasını gözlemlemek hızlı bir başarı hissi verdi, ancak görünüşte basit otomasyonların ardında karmaşık mantıksal ve elektronik bağlantıların olduğu kısa sürede anlaşıldı.

Fabrika dokümanlarının işlenmesi, özellikle yabancı dildeki açıklamaların yorumlanması ciddi bir zorluktu. Kısmi Macarca çeviri ve eğitimsel uyarlama sırasında öğrenciler, içerik yorumlamasına aktif olarak katılarak anlayışlarını önemli ölçüde derinleştirdiler. Blok tabanlı programlama ilk başta kolay görünse de, durum yönetimi ve koşullu kontrol daha ciddi düşünmeyi gerektiriyordu.

4. Kendi akıllı ev/akıllı çiftlik modelinizin dijital tasarımı (Fusion 360)

Bu bölüm, projenin belirleyici bir dönüm noktasıdır: daha önce öğrenilen sensör, kontrol ve IoT mantığı burada somut, elle tutulur bir fiziksel yapıya dönüşür. Bu aşamada öğrenciler, elektronik ve yazılım düşünme dünyasından, her kararın geometrik, montaj ve üretim teknolojisi açısından sonuçları olduğu mühendislik tasarım alanına geçerler.

Bu bölümün amacı sadece 3 boyutlu bir model oluşturmak değil, 3 boyutlu baskı için tasarlanmış fiziksel bir modelin destekleyici bir yapı olduğunu anlamaktır: kontrol ünitesi, sensörler, aktüatörler ve kablolama için alan sağlamalı, aynı zamanda montaj, ölçüm faaliyetleri ve daha sonraki değişiklikleri desteklemelidir. Bu anlamda model, nihai bir ürün değil, bir öğrenme aracıdır.

Öğrenme yolu kasıtlı olarak birkaç adımdan oluşacak şekilde yapılandırılmıştır. Dijital tasarımdan önce, Akıllı Çiftlik Kitinin fabrika tasarımını analiz eden fiziksel bir kavram aşaması gelir.

Bu süreç, işlevsel unsurların belirlenmesine ve eklemeli imalatın sınırlamalarının anlaşılmasına dayanmaktadır. Bunu, Fusion 360 tasarım ortamı hakkında bilgi edinme, ardından parametrik, bileşen tabanlı bir dijital model geliştirme ve nihayetinde üretime hazır bir duruma ulaşma takip eder.

Bu süreçte öğrenciler şunları deneyimler:

- İyi tasarım yazılımla başlamaz,
- Biçim her zaman işlevin bir sonucudur.
- Parametrik düşünme, yineleme ve yeniden tasarıma olanak tanır.
- Dijital model, üretim ve montaj sorunlarını önlemeye yardımcı olan bir "sanal prototip" görevi görür.

Bu bölüm, elektronik-bilişim sistemlerinin anlaşılması ile fiziksel uygulaması arasında bir köprü oluşturmakta ve 3D baskı ve sistem entegrasyonu ile ilgili projenin sonraki aşamaları için sağlam bir temel sağlamaktadır.

4.1. Başlangıç noktası: Akıllı Çiftlik Kitinin fiziksel tasarımının analizi

Dijital tasarım sürecindeki ilk adım CAD yazılımı kullanmak değil, mevcut fiziksel çözümleri bilinçli olarak analiz etmektir. Akıllı Çiftlik Kiti bu projede bir referans sistemi görevi görür: iyi işleyen, eğitimsel olarak optimize edilmiş kendi modelinizi oluşturmanın imkansız olduğu bir başlangıç noktasıdır.

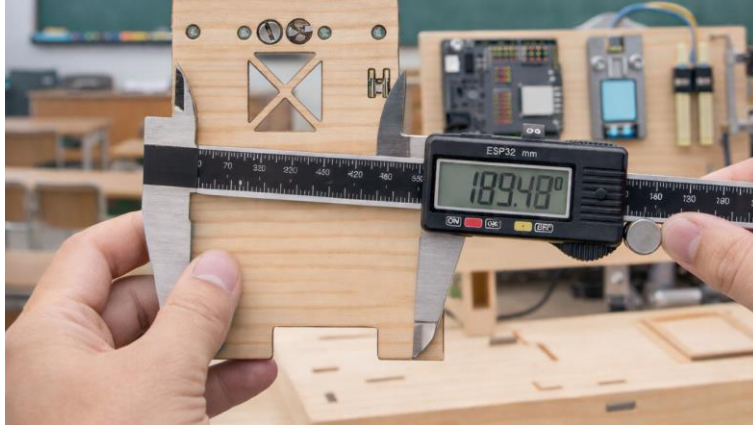
Akıllı Çiftlik Kitinin fiziksel tasarımı, kopyalanacak bir model olarak değil, analiz edilecek bir örnek olarak sunulmaktadır. Öğrencilerin amacı şunları keşfetmektir:

- Hangi unsurların gerçek bir işlevi vardır?
- Elektronik bileşenlerin fiziksel alana nasıl bağlandığı,
- Hangi çözümler eğitim amaçlıdır ve hangileri üretim teknolojisi veya estetik kaygılarla geliştirilmiştir?

Bu yaklaşım, işlev odaklı tasarımın temel prensibini bu erken aşamada ortaya koymaktadır: biçim, işlevin bir sonucudur, başlangıç noktası değil. Öğrenciler, hazır çözümleri eleştirel bir şekilde incelemeyi ve gerekli unsurları bağlamsal veya isteğe bağlı ayrıntılardan ayırt etmeyi öğrenirler.

Kavramsallaştırmanın temeli, gerçek boyutları öğrenmektir. Öğrenciler, herkesin erişebileceği basit araçlarla fabrika elemanlarını incelerler. Bir kumpas kullanarak çevre boyutlarını belirlerler, ardından grafik kağıdına orantılı çizimler yaparlar ve açıklıkların, montaj noktalarının ve sensör konumlarının yerlerini kaydederler.

Pedagojik açıdan bakıldığında, bu aşamada standart teknik çizimlerin üretilmemesi önemlidir; çünkü asıl amaç biçimsel doğruluk değil, mekansal ilişkilerin ve oranların anlaşılmasıdır.



Şekil 43: Akıllı Çiftliğin fiziksel unsurlarının ölçülmesi (Kaynak: Yapay zeka tarafından oluşturulmuştur)

Bu adım, fiziksel gerçeklik ile soyut tasarım arasındaki bağlantıyı güçlendirirken, daha sonraki parametrik düşünme süreçlerine doğrudan hazırlık niteliği taşır.

4.2. 3B Baskının Sınırlamaları ve Tasarım Etkileri

Öğrenciler, fabrika bileşenlerini analiz ederken, eklemeli imalatın pratik sınırlamalarıyla karşılaşacaklardır. İnce duvarlar, kapalı boşluklar ve girintiler gibi bazı geometrik detayların masaüstü 3D yazıcılarla yeniden üretilemeyeceğini, bu nedenle tam geometrik doğruluk yerine bilinçli bir basitleştirmenin gerekli olduğunu fark edeceklerdir.

Bu farkındalık, üretim için tasarım yaklaşımını ortaya koymakta ve tasarım kararlarının her zaman seçilen üretim teknolojisiyle ilişkili olduğunu vurgulamaktadır.

Fiziksel analizin bir sonraki adımı bilinçli karar vermektir: Kendi modelinizi geliştirirken neleri dahil edip neleri dışarıda bırakacağınız.

Öğrenciler, sensörlerin, aktüatörlerin ve kontrolörlerin yerleştirilmesi için hangi elemanlara ihtiyaç duyulduğunu, bunların bağlantı noktalarının nerede olması gerektiğini ve öğrenme hedeflerinden ödün vermeden hangi parçaların çıkarılabileceğini birlikte belirlerler.

Kararlar her zaman işlevselliğe ve eğitilebilirliğe dayanır, fabrika çözümüne biçimsel benzerliğe değil. Kavramsallaştırmanın amacı, aşağıdaki özelliklere sahip fiziksel bir yapı oluşturmaktır:

- Gereksiz boşluklar içermez.
- Malzeme tasarrufu sağlayan ve hızlı baskı imkanı sunan bir ürün.
- Kurulumu ve değiştirilmesi kolay,
- Şeffaf bir eğitim ortamı sağlar.

Bu adım, baskı süresini, malzeme kullanımını ve yineleme döngülerinin hızını doğrudan etkiler.

4.3. Fiziksel kavram, sistem açıklaması olarak

Bu aşamada öğrenciler modeli artık bir "ev" veya "kutu" olarak değil, sensörlerin yerleştirileceği bir nokta sistemi, aktüatörler için bir hareket aralığı sağlayan bir yapı ve kontrol ve kablolama için bir taşıyıcı olarak yorumluyorlar.

Bu yaklaşım, tasarımın çok erken aşamada biçimsel ayrıntılara boğulmamasını, ancak sistemik düzeyde yorumlanabilir kalmasını sağlamaya yardımcı olur.

Parametrik tasarım için kavramsal hazırlık

Bu bölümde Fusion 360 kullanımına henüz değinilmemiş olsa da, parametrik tasarımın temel fikri tanıtılmaktadır. Boyutlar izole sayılar değil, birbirleriyle ilişkili parametrelerdir; burada bir değişiklik birden fazla öğeyi etkileyebilir ve tasarımın amacı uyarlanabilirlik ve yeniden tasarlanabilirliktir.

Bu düşünce yapısı, fiziksel kavramın dijital, parametrik bir modele dönüştürüldüğü bir sonraki aşamaya doğrudan geçişi sağlar.

Pedagojik özet

Bu hazırlık ve kavramsallaştırma aşaması, daha sonraki Fusion 360 tabanlı tasarımın yalnızca bir yazılım kullanım egzersizi değil, bilinçli mühendislik kararlarının dijital bir temsili olmasını sağlar.

Öğrenciler, iyi tasarımın bilgisayar açılmadan önce başladığını, sadeleştirme ve soyutlamanın mühendislik erdemleri olduğunu ve fiziksel ve dijital dünyaların yakın bir bütünlük oluşturduğunu deneyimleyeceklerdir.

4.4. Fusion 360 tasarım ortamı - temel bilgiler ve yaklaşım

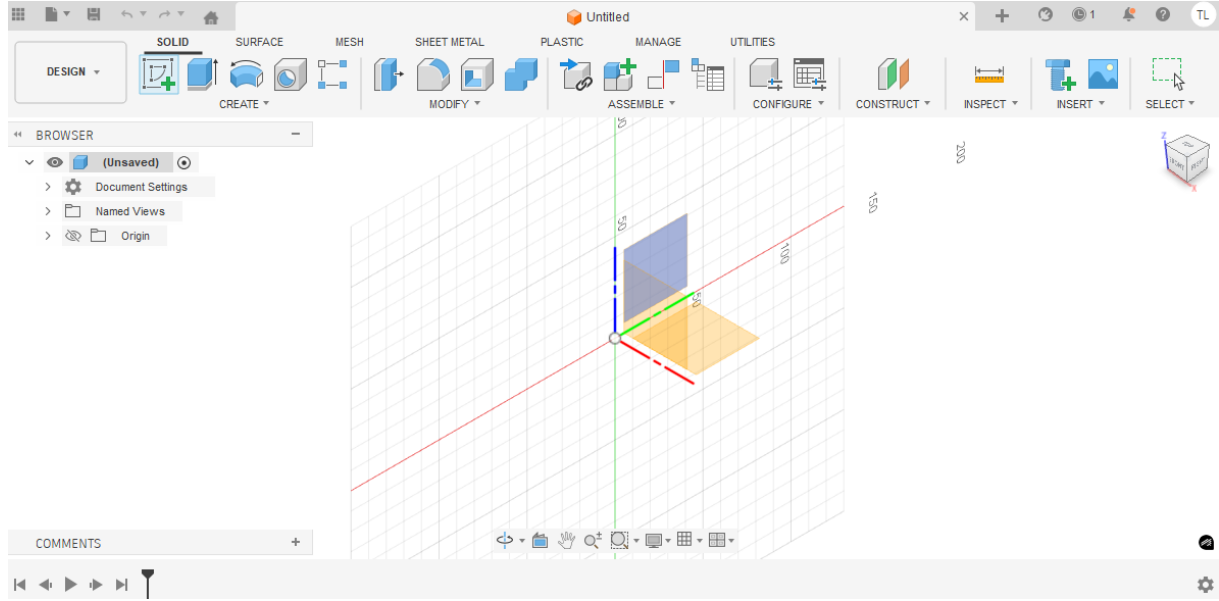
Bu aşamada öğrenciler artık dijital tasarımı teorik düzeyde konuşmak yerine, fiziksel bir kavramın dijital, değiştirilebilir bir modelle nasıl eşleştirilebileceğini somut işlemler yoluyla deneyimliyorlar.

Öğrenciler, Autodesk'in resmi yayını olan "Autodesk Fusion 360 Eğitimi: Şeyler Yapmanın Geleceği Katılımcı Kılavuzu"nu kullanarak Fusion 360'ın temellerini ve kullanımını öğreneceklerdir.

İşlenmiş eğitim materyaline dayanarak, öğrenme süreci işlev listelerini öğrenmeye değil, temel tasarım rutinlerini geliştirmeye dayanır; örneğin:

- Yeni bir modeli nasıl piyasaya süreriz?
- 3 boyutlu uzayda nasıl yönümüzü buluyoruz?
- Basit ama istikrarlı geometri nasıl oluşturulur?

Eğitimdeki amaç, öğrencilerin CAD tasarımının görsel bir imge yaratmak değil, bir etkinlik olduğunu anlamalarını sağlamaktır.



Şekil 44: Fusion 360 tasarım ortamı – temel bilgiler ve yaklaşım (Kaynak: Kendi düzenlemem)

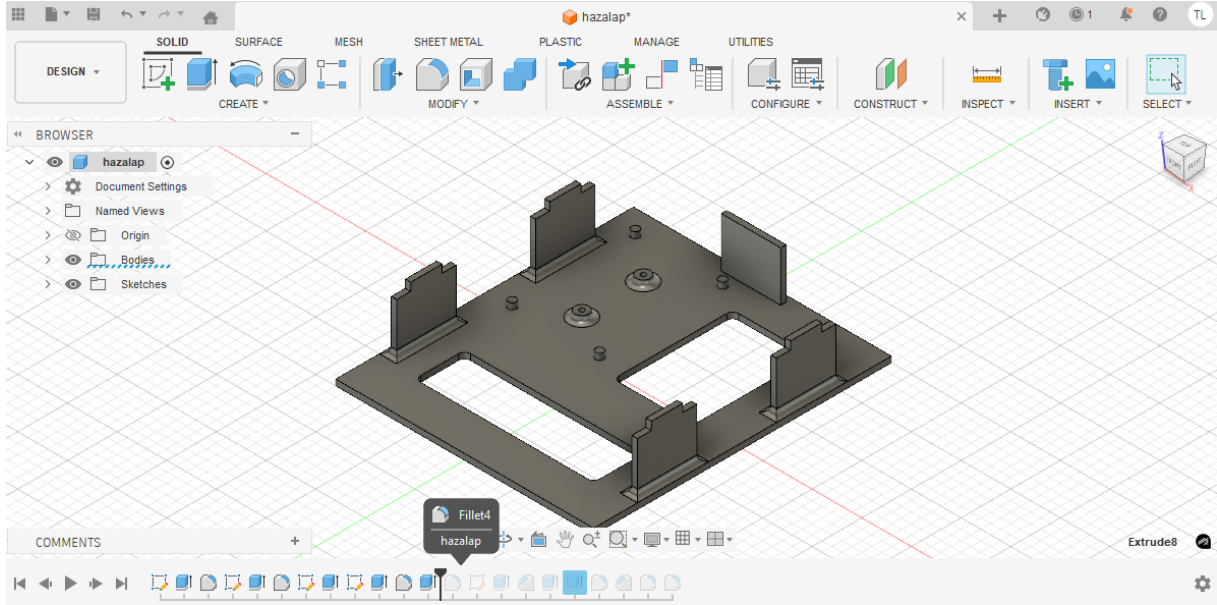
Fusion 360, eğitimde "düşünme yüzeyi" olarak sunulmaktadır. Belge, başlangıç seviyesinde bile, model oluşturma, değiştirme, önceki durumlara geri dönme ve değişikliklerin etkisini gözlemlenme gibi tam bir tasarım döngüsünün tamamlanabileceğini vurgulamaktadır. Bu yaklaşım, modelin statik bir nesne değil, bir sürecin sonucu olduğu parametrik modelleme ve hikaye tabanlı tasarım dünyasına onları tanıtır.

Öğrencilere özel etkinlikler:

- Yeni Bir Tasarım Oluşturun
- Projeye kaydet (bulut tabanlı çalışma prensibini anlama)
- Zaman çizelgesinde önceki bir duruma geri dön

Belgedeki temel terimler:

- Parametrik Modelleme
- Tarih Temelli Tasarım
- Tahribatsız Düzenleme



Şekil 45: Yeni bir tasarım oluşturma – akıllı ev tabanı (Kaynak: Kendi düzenlemem)

Pedagojik odak noktası: Öğrenciler, modelin tarihinin, mevcut durumu kadar önemli olduğunu deneyimlerler.

Autodesk Fusion 360 Eğitim materyali, arayüzü özellikle yeni başlayanlar için, işlevlerden ziyade kullanım mantığına dayalı olarak tanıtmaktadır.

4.5. Tuval – mekansal hareketin ustalaşması

Mekansal düşünme becerisinin geliştirilmesine özel bir önem verilir. Öğrenciler, uzayda güvenle gezinmek için temel görünümünün yanı sıra döndürme, yakınlaştırma, taşıma gibi görünüm manipülasyon işlemlerini aktif olarak kullanırlar. Bu beceri, daha sonra doğru ve düşünülmüş modeller oluşturmak için bir ön koşuldur.

Öğrenci alıştırmaları:

- model dönüşü (Yörünge),
- Yakınlaştır/Uzaklaştır,
- Tava,
- Temel görünümleri kullanarak (Üst, Ön, Sağ).

Pedagojik amaç: Çizime başlamadan önce bile mekansal düşünme becerisini geliştirmek.

Tarayıcı – modelin “iç haritası”

Sistemin "dahili haritası" olan Tarayıcı, modelin yapısını anlamada önemli bir rol oynar. Öğrenciler çizimleri, gövdeleri ve bileşenleri yönetmeyi öğrenir ve iyi yapılandırılmış bir modelin yalnızca daha şeffaf olmakla kalmayıp, daha sonra değiştirilmesinin de daha kolay olduğunu fark ederler.

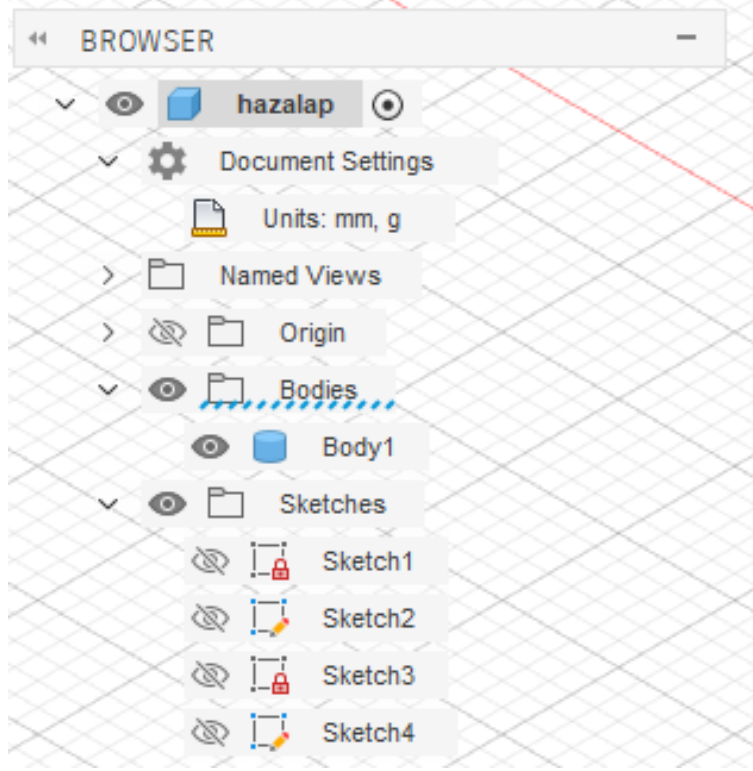
Öğrenci etkinlikleri:

- Çizimi gizle/göster,
- Bedenlerin ayrılması,

- Öğeleri adlandırma.

Anahtar terimler:

- Çizim Klasörü
- Cesetler Klasörü
- Bileşenler



Şekil 46: Modelin "iç haritası" (Kaynak: Kendi düzenlemem)

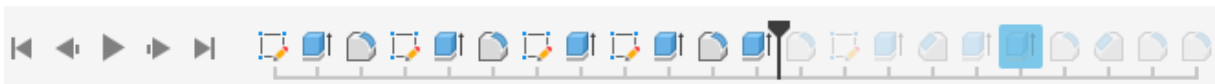
Buna paralel olarak, Zaman Çizelgesi kullanımı tasarım sürecinin nedensel sistemini görünür kılar: önceki bir adımı değiştirmek tüm modeli etkiler ve bu da sistem düşüncesinin geliştirilmesine yardımcı olur.

Bu belge, Zaman Çizelgesi'ni bir öğrenme aracı olarak yoğun bir şekilde kullanmaktadır.

Öğrenci uygulaması:

- Ekstrüzyon adımını değiştirin,
- Modelin tamamı üzerindeki sonuçları gözlemlemek.

Anahtar Kelime: Tasarım Tarihi



Şekil 47: Zaman Çizelgesi Kullanımı (Kaynak: Kendi düzenlemem)

4.6. Temel geometrik ve tasarım kavramları – somut bir şekilde

Eskiz – bir çizim değil, bir kurallar bütünü

Çizim kavramını anlamak önemlidir; çizim bir resim değil, bir kurallar bütünüdür. Öğrenciler, bir çizimin basit bir çizim değil, matematiksel olarak tanımlanmış bir sistem olduğunu deneyimlerler. Boyutların ve geometrik kısıtlamaların kullanımı, modelin istikrarlı ve belirsiz olmamasını sağlar. "Eksik tanımlanmış" çizimlerin sorunları hızla görünür hale gelir, böylece öğrenciler doğru tasarımın önemini kendi deneyimlerinden anlarlar.

Autodesk Fusion 360 Eğitimine göre, yeni başlayan tasarımcıların yaptığı en yaygın hata, tanımlanmamış bir çizim oluşturmaktır.

Öğrenci uygulaması:

- dikdörtgen çizmek,
- boyutlandırma ekleme,
- Kısıtlamaların uygulanması.

Anahtar terimler:

- Tamamen Kısıtlanmış Çizim
- Serbestlik Dereceleri
- Geometrik Kısıtlamalar

Pedagojik anlayış: Tanımlanmayan şey daha sonra sorunlara yol açacaktır.

Gövde ve Bileşen – hangisi ne zaman kullanılır?

Katı cisimler ve bileşenler arasındaki ayrım da önemli bir mühendislik kararı olarak ortaya çıkmaktadır. Öğrenciler, katı cisimlerin hızlı tasarım için uygun olduğunu, bileşenlerin kullanımının ise özellikle daha karmaşık modeller için yapısal düşünmeyi gerektirdiğini fark ederler. Bu ayrım aynı zamanda onları daha sonraki montaj görevlerine de hazırlar.

Bu belge, pratik örnekler aracılığıyla şunları göstermektedir:

- Vücut = hızlı şekillendirme,
- Bileşen = yapısal düşünme.

Öğrenci karar verme durumu:

- Tek bir bedenin yeterli olduğu durumlarda,
- Ayrı bir bileşenin gerekçelendirildiği durumlarda.

Temel kavram: Gövdeden Bileşen Oluşturma

Parametreler tasarım araçları olarak

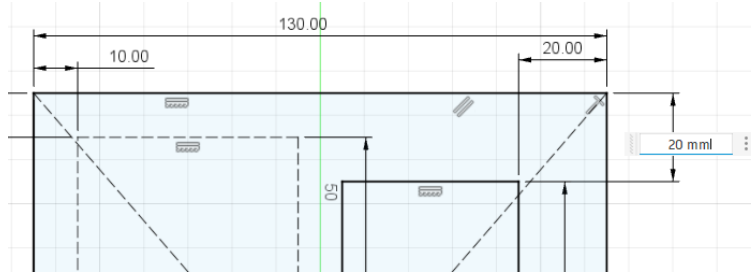
Parametrik tasarımın tanıtımı sırasında öğrenciler, tek bir boyutun değiştirilmesinin tüm modeli etkileyebileceğini deneyimlerler. Bu, yalnızca teknik bir değişiklik değil, aynı zamanda bir bakış açısı değişikliğini de temsil eder: model artık sabit bir form değil, tasarım ihtiyaçlarına uyum sağlayan değiştirilebilir bir sistemdir.

Autodesk Fusion 360 eğitimine dayanarak, öğrenciler şunları deneyeceklerdir:

- Bir boyutu değiştirmek,
- Modelin tamamının dönüşümü.

Anahtar terimler:

- Parametreler
- Parametreleri Değiştir İletişim Kutusu



Şekil 48: Parametrelerin kullanımı (Kaynak: Kendi düzenlemem)

Vücut modelleme - emin bir el ile temel işlemler

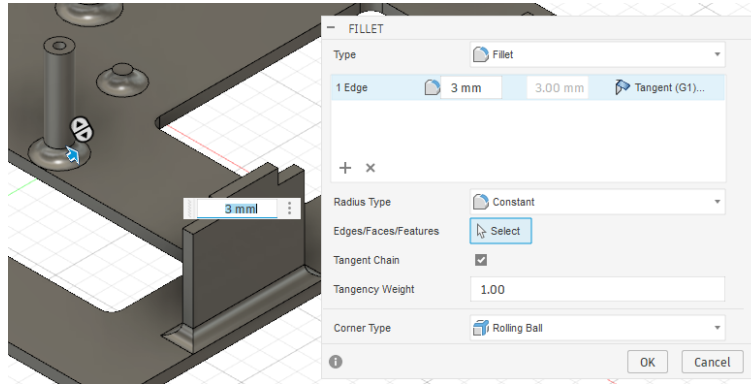
Ekstrüzyon, yuvarlama veya tekrarlama gibi temel katı modelleme işlemleri, birbirinden bağımsız olarak sunulmaz, her zaman belirli tasarım durumlarıyla ilişkilendirilir. Bu sayede öğrenciler komutları öğrenmek yerine, bu komutların modelin bütünü üzerindeki rolünü ve sonuçlarını anlarlar.

Autodesk Fusion 360 Eğitimine dayalı, öğretilebilir özel işlemler:

- Ekstrüzyon (Birleştirme / Kesme),
- Köşe yuvarlatma ve pah kırma,
- Desen (Doğrusal tekrar).

Anahtar terimler:

- Birleştirme mi, Kesme mi?
- Kenar Dolgusu
- Dikdörtgen Desen



Şekil 49: Vücut modellemesi (Kaynak: Kendi düzenlemem)

4.7. 3D baskıya hazırlık – son pratik odak noktası

Autodesk Fusion 360 Eğitiminin 62-65. sayfaları özellikle "bitmiş model" durumuna odaklanmaktadır.

Tasarım sürecinin son aşamasında, odak noktası dijital modelin fiziksel uygulanabilirliğine kayar. Öğrenciler, iyi tasarlanmış bir modelin otomatik üretime uygun olmayabileceğini, bu nedenle 3D baskının gereksinimlerine özel dikkat gösterilmesi gerektiğini öğrenirler.

Model kontrolü yapılırken, öncelikli husus kapalı geometrinin sağlanmasıdır, çünkü baskı ancak net tanımlanmış hacimlerle mümkündür. Buna ek olarak, duvar kalınlığı, detay boyutu ve geometri stabilitesi incelenmelidir. Öğrenciler, tasarım kararlarının üretim kalitesi ve başarısını doğrudan nasıl etkilediğini deneyimleyeceklerdir.

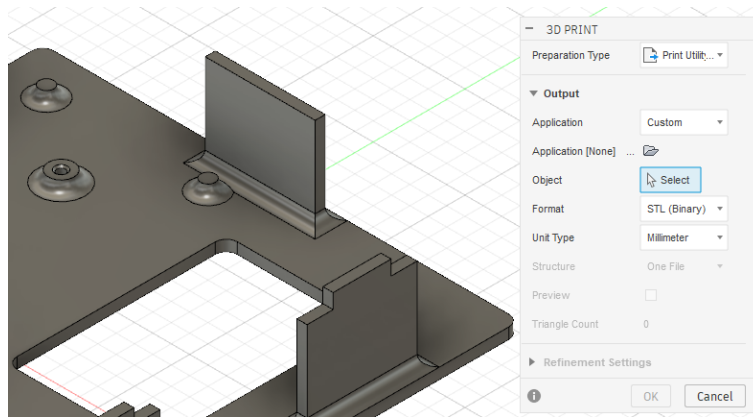
Modelin dışa aktarılması, üretim için gerekli formatları (STL veya 3MF gibi) ve CAD modelinden yazdırılabilir bir veri dosyası oluşturma sürecini başlatır. Bu adım, tasarım döngüsünü tamamlar ve aynı zamanda geri bildirim sağlar: tasarım hataları ve eksiklikleri, fiziksel modelin oluşturulması sırasında anında görünür hale gelir.

Öğrenci kontrol listesi:

- Kapalı gövde (su geçirmez),
- uygun duvar kalınlığı,
- STL/3MF formatında dışa aktarın.

Anahtar terimler:

- Ağ Önizlemesi
- İmalat için İhracat



Şekil 50: 3D baskı için hazırlık (Kaynak: Kendi düzenlemem)

Didaktik özet

Bu aşama, özellikle didaktik açıdan önemlidir, çünkü öğrencilere dijital tasarımın kendi başına bir faaliyet değil, gerçek fiziksel dünyada uygulanacak sistemlerin hazırlanması olduğu anlaşılır.

Tüm sürecin sonunda öğrenciler sadece bir model oluşturmakla kalmayacak, aynı zamanda tasarımın tüm döngüsünü de anlayacaklardır: fikirden parametrik modele ve üretilebilir forma kadar. Bu deneyim, tasarım ve uygulamanın ayrılmaz bir bütün oluşturduğu mühendislik yaklaşımını ortaya koymaktadır.

Projenin bu uygulama aşamasında öğrenciler bir yazılım programı kullanmayı öğrenmek yerine, adım adım ilk dijital modellerini tasarlamaya ve inşa etmeye başlarlar ve CAD'in düşüncenin dışı vurumu olduğunu fark ederler.

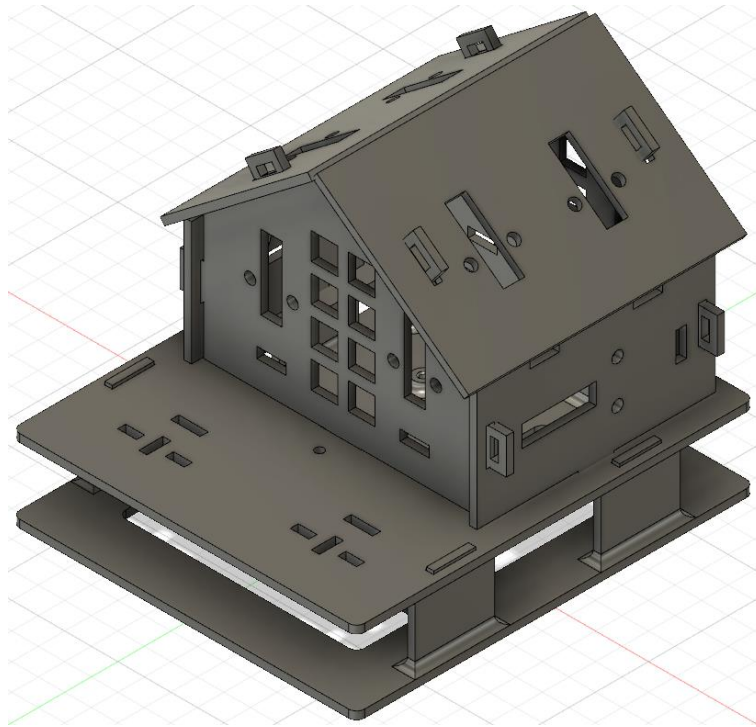
Bu, dijital modelin 3D baskı yoluyla gerçek bir fiziksel nesneye dönüştüğü bir sonraki aşama için sağlam bir temel oluşturur.

4.8. Dijital tasarım sürecinin sonuçları - öğrenci modellerinin sunumu ve yorumlanması

Bu alt bölümde proje süresince oluşturulan dijital modeller sunulmaktadır. Bu unsurlar önceden üretilmiş örnekler değil, öğrencilerin daha önce öğrendikleri tasarım prensiplerine dayanarak ve öğretmen rehberliğinde oluşturdukları 3 boyutlu modellerdir.

Modellerin oluşturulması sırasında öğrenciler, bilinçli olarak işlev odaklı tasarım ilkesini, parametrik düşünmeyi, modülerlik ve montaj unsurlarını ve eklemeli imalatın (3D baskı) geometrik sınırlamalarını uyguladılar.

Pedagojik açıdan bakıldığında, bu modeller öğrenme sürecinin somut izleridir: soyut tasarım ilkelerinin somut fiziksel biçimlerde nasıl tezahür ettiğini gösterirler.



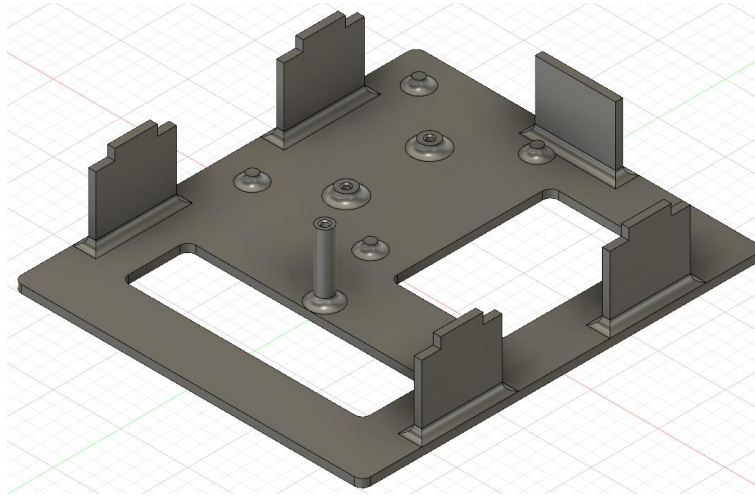
Şekil 51: Akıllı ev montaj modeli (Kaynak: Kendi düzenlemem)

Öğrenciler tarafından oluşturulan STL dosyaları, modüler bir akıllı ev/akıllı çiftlik modelinin fiziksel temelini oluşturmaktadır. Tasarım hedefi, estetik açıdan nihai bir nesne oluşturmak değil, şu işlevleri yerine getiren bir destekleyici yapı oluşturmaktır:

- Elektronik bileşenleri yerleştirmek için uygun,
- Sensörlerin ve aktüatörlerin kaydını destekler,
- Montaj, ölçüm ve değişiklik yapılmasına olanak tanır.
- Masaüstü 3D yazıcıların gereksinimlerini karşılar.

Böylece öğrenciler, dijital modelin bağımsız bir amaç değil, elektronik ve kontrol projeleri için fiziksel bir çerçeve olduğu gerçeğini anladılar.

Taban plakası ve alt yapısal elemanlar - öğrenci tasarım kararları



Şekil 52: Taban plakası ve alt yapısal elemanlar (Kaynak: Kendi düzenlemem)

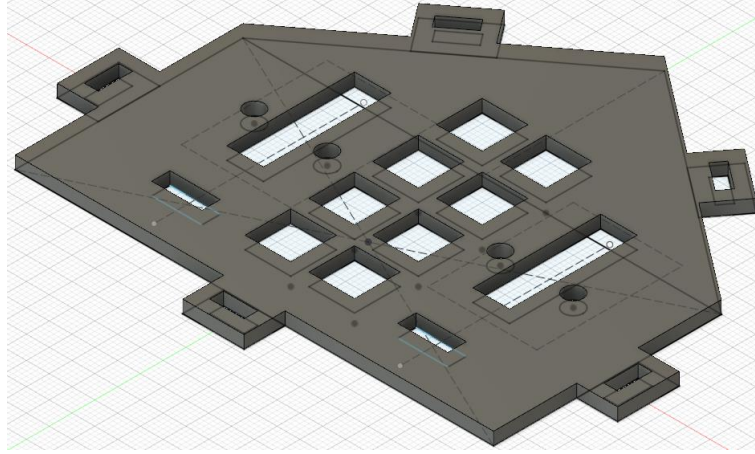
Öğrenciler, taban plakasını ve alt kapağı tasarlarken daha önce öğrendikleri mühendislik prensiplerini uyguladılar.

Tasarım sürecinde alınan kararlar:

- stabil, düz bir destek yüzeyi oluşturmak,
- Elektronik modüllerin montajı için hazırlık,
- Kabloların düzenli bir şekilde yönlendirilmesini sağlamak.

Öğretimsel önemi: Öğrenciler, temel yapının kalitesinin tüm sistemin kullanılabilirliğini belirlediğini ve görünüşte "basit" unsurların bile arkasında bilinçli bir mühendislik düşüncesi olduğunu deneyimlediler.

Yan paneller – işlevsellik ve erişilebilirlik



Şekil 53: Yan paneller (Kaynak: Kendi düzenlemem)

Yan panelleri tasarlarken öğrenciler, modelin kapalı bir yapı değil, eğitimsel hedefleri destekleyen "okunabilir" bir yapı olmasını sağlamaya bilinçli olarak özen gösterdiler.

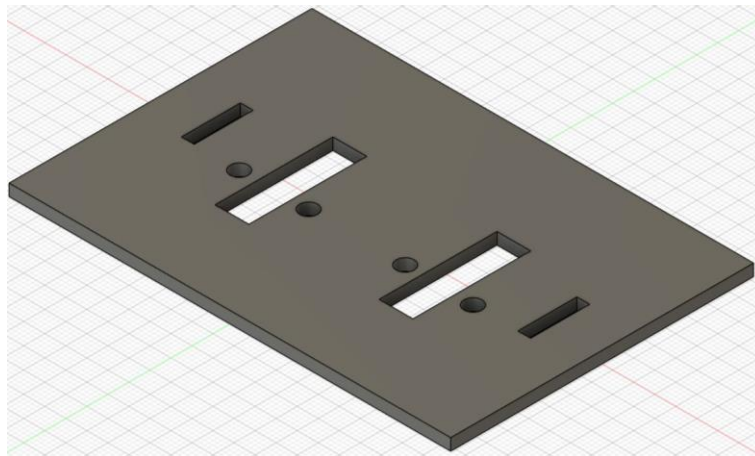
Tasarım aşamasında dikkate alınan hususlar arasında sensörlerin ve ekranların yerleşimi, montaj ve ölçüm için erişim kolaylığı ve iç mekanın görsel şeffaflığı yer almaktadır.

Pedagojik ders: Öğrenciler, eğitim modelinin kapağının gizlemediğini, aksine işleyişi anlaşılır kıldığını fark ettiler.

Çatı elemanları – koruma ve sökülebilirlik dengesi

Çatı elemanlarını tasarlarken öğrenciler, elektronik aksamı koruyan, sökülmeyi engellemeyen ve daha sonraki değişiklikleri destekleyen çözümler kullandılar.

Bu tasarım görevi, "kapanma"nın nihai bir sonuç anlamına gelmediğini, aksine koruma ve erişilebilirlik arasında bilinçli bir uzlaşma olduğunu açıkça ortaya koymuştur.



Şekil 54: Çatı elemanı (Kaynak: Kendi düzenlemem)

4.9. Tamamlanan modellere dayalı öğrenme kazanımlarının özeti

Proje kapsamında oluşturulan modeller, öğrencilere kendi tasarım kararlarını gözden geçirme ve bunların sonuçlarını anlama fırsatı sunmaktadır.

Modeller aracılığıyla yakalanan temel kavramlar:

- parametrik tasarım,
- modülerlik,
- kurulum,
- imalat için tasarım,
- Yinelemeli geliştirme.

Bu bölüm, öğrencilerin yalnızca yazılım kullanımını değil, aynı zamanda mühendislik yaklaşımını da öğrendiklerini açıkça ortaya koymaktadır; bu yaklaşım, sonraki bölümlerde 3D baskı ve fiziksel montaj ile tamamlanacaktır.

4.10. Bölümün pratik uygulaması

İşte bu aşamada, öğrencilerin dijital tasarımın fiziksel gerçekliğin doğru bir şekilde anlaşılmasına dayandığını kabul etmelerinin ne kadar zor olduğu gerçekten ortaya çıktı. Başlangıçta, ölçümlerin, orantılı çizimlerin ve mekansal ilişkilerin analizi onlara yavaş ve "daha az etkileyici" görünüyordu, ancak daha sonra modeldeki ve baskıdaki yanlışlıkların hemen geri yansıdığını kendileri fark ettiler.

Autodesk Fusion'ı tanıtırken, süreci kasıtlı olarak yavaşlatmak çoğu zaman gerekliydi. Öğrenciler parametrik düşünmeyi öğrenirken çizim yapmak istiyorlardı. Dönüm noktası, kendi deneyimlerinden yola çıkarak, kötü yapılandırılmış bir çizimin daha sonra ciddi sorunlara yol açtığını, iyi düşünülmüş bir yapının ise istikrarlı, kolayca değiştirilebilir bir modelle sonuçlandığını anlamaları oldu.

Kendi modelimizi tasarlarken, gerçek mühendislik kararları aldık: neyi basitleştireceğiz, nereye erişim sağlayacağız, ne gerçek bir işlev olarak kabul edilecek. Aşırı kalabalık yaygındı ve bunların bir arada tutulması gerekiyordu. Bir öğretmenin bakış açısından, en büyük sonuçlardan biri öğrencilerin düşünme biçiminin değişmesiydi: modeli bir nesne olarak değil, işleyen bir sistem olarak görmeye başladılar.

5. 3D baskı ve üretim deneyimi

Projenin kilit unsurlarından biri, dijital tasarım süreçlerinin fiziksel olarak uygulanmasıydı. 3D baskı kullanımı, öğrencilere bilgisayar modellemesi ile gerçek dünya üretimi arasındaki bağlantıları doğrudan deneyimleme ve üretim için tasarım ilkelerinin önemini anlama fırsatı sağladı. Günümüzde popüler olan ve sürekli artan uygulama alanlarına sahip 3D baskı, projenin bu aşamasının fiziksel olarak uygulanması için birçok açıdan ideal bir seçimdi.

Üretim aşamasında, dijital ortamda doğru görünen çözümlerin fiziksel uygulamada sıklıkla değişiklik gerektirdiği anlaşıldı. Bu deneyim, özellikle eğitimsel açıdan değerliydi; çünkü öğrenciler sorunları, analizlerini ve çözümlerini izole görevler olarak değil, tutarlı bir geliştirme sürecinin parçası olarak ele aldılar.

5.1. 3D Yazıcı ve Bambu Stüdyosuna Giriş:

5.2. Kullanılan Sürece Kısa Bir Giriş

Projede kullanılan eklemeli üretim süreci, günümüzde en yaygın ve kolay erişilebilir 3 boyutlu baskı yöntemlerinden biri olan FDM (Fused Deposition Modeling) teknolojisidir. Sürecin özü, yazıcının sürekli bir plastik ipliği (filament) eritmesi ve ardından önceden belirlenmiş yollar boyunca katman katman biriktirmesidir.

FDM teknolojisi, üretim sürecinin adımlarının kolayca takip edilebilmesi ve baskı parametrelerindeki değişikliklerin etkilerinin anında görülebilmesi nedeniyle özellikle eğitim ortamları için çok uygundur. Bu durum, öğrencilerin yalnızca nihai sonucu gözlemlemekle kalmayıp, tüm üretim sürecini anlamaları ve yorumlamaları için de bir fırsat yaratmıştır.

5.3. Donanım yapısı ve çalışması

Yazdırma işleri bir cihaza gönderilir. **Bambu Lab X1 Karbon** Projeyi kapalı tip masaüstü 3D yazıcı ile hayata geçirdik. Cihaz, gelişmiş bir sensör sistemine, otomatik tabla seviyeleme sistemine ve yüksek hassasiyetli hareket sistemine sahip olup, bu özellikler birlikte istikrarlı ve güvenilir bir çalışma sağlar. Projenin uygulanması sırasında bu yazıcıyı seçmemizin nedeni, piyasadaki alternatifler arasında sadece yüksek kalitede, çok sayıda kalibrasyon ve özelleştirme seçeneği sunan değil, aynı zamanda zaman açısından verimli bir şekilde nesnelere üretebilen bir cihaz arıyor olmamızdı. Yazıcının kullanımı, diğer cihazlara göre önemli ölçüde daha az arka plan kalibrasyon işlemi gerektirdi; bu da projenin daha önemli alanlarına daha fazla zaman ayırmamızı sağladı. Alternatif olarak bir Prusa i3 MK3S+ yazıcı da seçtik, ancak daha kolay kullanımı ve profesyonel tasarımı nedeniyle nihayetinde X1 Carbon'da karar kıldık.



Şekil 55: Bambu Lab X1 Karbon 3D yazıcı (Kaynak: <https://www.3djake.hu/bambu-lab/x1c>)

Kapalı baskı alanı, çevresel etkileri (sıcaklık, hava hareketi) en aza indirerek baskı hatalarının oluşumunu azalttığı için projede özellikle önemli bir rol oynadı. Bu, öğrencilerin teknolojik bağlama odaklanmalarını ve çevresel istikrarsızlıktan kaynaklanan sorunların deneyimlerine hakim olmasını engellemelerini sağladı.

5.4. AMS filament besleme sistemi



Şekil 56: AMS filament dağıtıcısı (Kaynak: <https://www.3djake.hu/bambu-lab/x1c>)

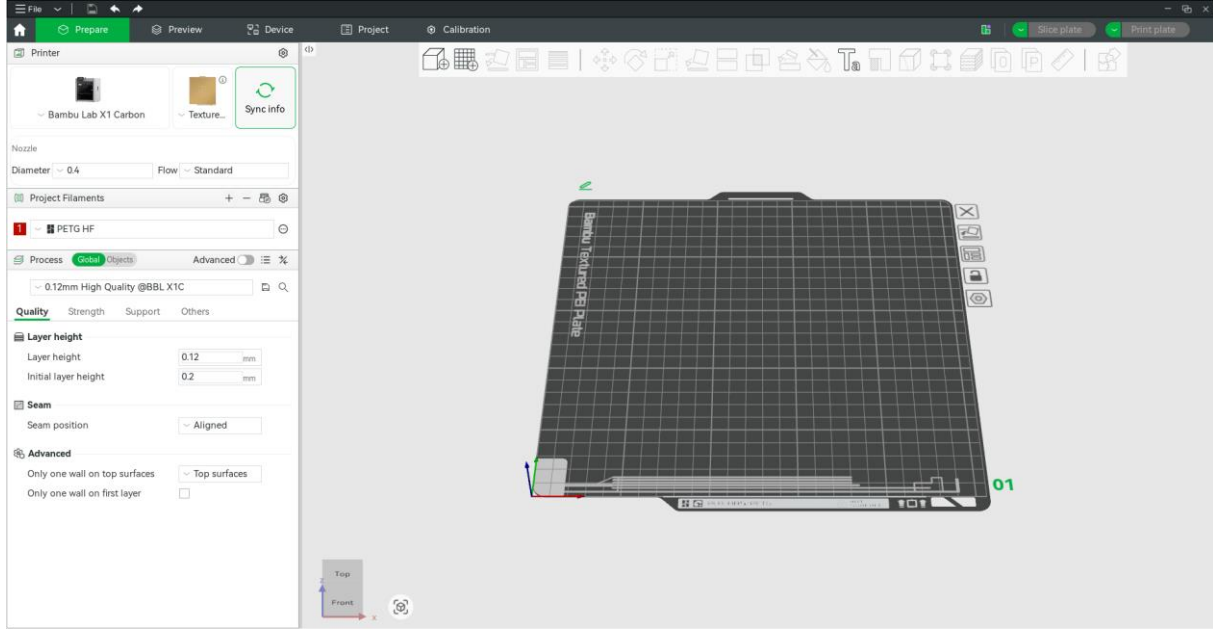
Yazıcıya bağlı AMS (Otomatik Malzeme Sistemi) otomatik filament besleme sistemi, aynı anda birden fazla malzeme rulosunu işleyebilmektedir. Proje ağırlıklı olarak tek renk ve tek tip malzemeden yapılmış parçalar üretmeyi amaçlasa da, AMS sisteminin kullanımı önemli teknolojik bilgiler sağlamıştır. Ayrıca, boş ruloları değiştirirken veya renkleri değiştirirken filamentleri yazıcıdan çıkarıp tekrar takmaya gerek kalmadığı için üretim sürecini kolaylaştırmıştır.

Öğrenciler, özellikle nemin etkisi ve depolamanın önemi üzerinde durularak, malzeme elleçlemenin ve hammaddelerin durumunun önemini öğrendiler; pasif ve aktif AMS (Yıllık Yönetim Sistemi) sistemleri vurgulandı.

Filament kurutma cihazları. Ayrıca, AMS'nin çalışması, daha sonra çok malzemeli ve çok renkli üretim yeteneklerinin 도입 edilmesi için iyi bir başlangıç noktası sağladı.

5.5. Dilimleme prensipleri

Dijital modellerin üretilebilir formlara dönüştürülmesi **Bambu Stüdyosu** bunu dilimleme yazılımı kullanarak yaptık. Bambu Studio, yazıcı üreticisinin (Bambu Lab) kendi açık kaynak yazılımıdır ve PrusaSlicer dilimleme programına dayanmaktadır ve tüm büyük masaüstü işletim sistemleri için mevcuttur.



Şekil 57: Bambu Studio dilimleme yazılımı (Kaynak: Kendi düzenlemem)

Dilimleme işlemi sırasında öğrenciler, başarılı bir baskı için yalnızca geometrinin yeterli olmadığı, doğru parametreleştirmenin en az onun kadar önemli olduğu gerçeğiyle karşı karşıya kaldılar.

Dilimleme ayarlarını değiştirerek, baskı süresi, malzeme tüketimi ve yapısal kararlılık arasındaki ilişki belirgin hale geldi. Bu deneyim, öğrencilerin üretim sürecini birbirinden bağımsız adımlar dizisi yerine karmaşık bir sistem olarak anlamalarına yardımcı oldu. Dilimleme işlemleri sırasında, yazılıma entegre edilmiş varsayılan baskı profilleri, yalnızca baskı kalitesini değil, aynı zamanda kullanılan malzemeden kaynaklanan optimum konfigürasyonu da ele alan iyi bir başlangıç noktası sağladı. Bu profillerin yardımıyla, bireysel ayarlar arasındaki fark daha da netleşti.

Baskı parametrelerinin belirlenmesinin yanı sıra, basılan modellerin yönlendirilmesi, renk ayarları ve tepsi hazırlığı da aynı derecede önemliydi. Modellerin optimum yapılaşmasına ve destek ihtiyacına özel dikkat gösterilmesi gerekiyordu.

5.6. Malzemeler (PLA, PETG, ABS)

Proje süresince, yaygın olarak kullanılan çeşitli 3D baskı malzemelerinin özelliklerini inceledik; ancak gerçek üretim için...**PETG** Malzemeyi biz seçtik. Karar hem mesleki hem de pedagojik değerlendirmelere dayanıyordu.

PETG, ABS'ye göre daha az deformasyona eğilimli olmakla birlikte, baskı kolaylığı ve mekanik direnç arasında iyi bir denge sağlar. Bu, özellikle daha uzun baskı sürelerine sahip fonksiyonel parçalar için faydalıydı. Bu sayede öğrenciler, gerçek bir mühendislik karar verme durumunda malzeme seçiminin önemini deneyimleme fırsatı buldular.

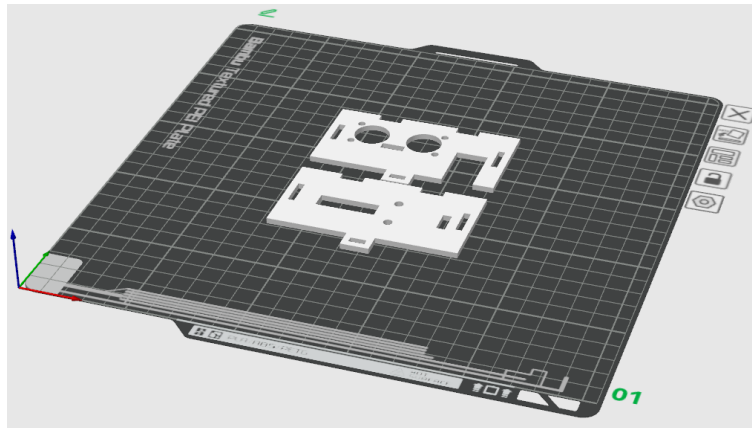
Projenin uygulanması sırasında, çoğu 3D baskı projesi için varsayılan malzeme seçeneği olarak kabul edilen PLA filamentinin de elimizde bulunduğunu belirtmek isteriz; ancak dayanıklılığı ve fiziksel darbelere karşı düşük direnci nedeniyle, onu kullanmaktan vazgeçtik ve yalnızca prototipleme ve test amaçlı kullandık. Bahsedilen üç malzeme türüne ek olarak, diğer filament türlerinin (örneğin TPU) kullanımını araştırmadık.

5.7. Baskı işlemindeki adımlar

3D baskı teknolojisinin pratik uygulaması, yapılandırılmış ve birbirine bağlı adımlar halinde gerçekleştirildi. Sürecin bilinçli yapısı, öğrencilerin bireysel aşamaları izole işlemler olarak değil, tutarlı bir üretim sisteminin parçası olarak yorumlamalarına olanak sağladı. Baskı aşamalarında, hazırlık, süreç izleme ve tamamlanmış parçaların değerlendirilmesine özel önem verildi.

5.8. Model hazırlama ve doğrulama

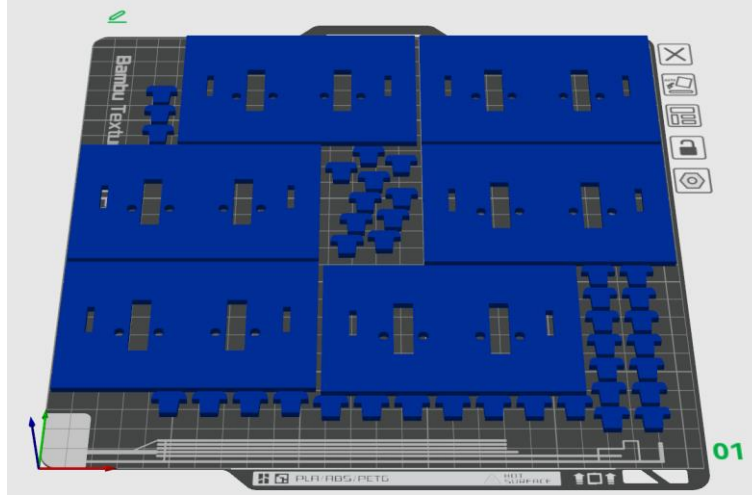
Her durumda, baskı işleminden önce, üretim hatalarını önlemek ve kaynakları verimli kullanmak amacıyla detaylı model hazırlığı yapıldı. Öğrenciler, özellikle kapalı gövdeler, duvar kalınlıkları ve kritik birleşme yüzeylerine dikkat ederek, dijital modellerin geometrik doğruluğunu kontrol ettiler.



Şekil 58: Model hazırlama ve doğrulama
(Kaynak: Kendi düzenlemem)

Hazırlık aşamasında, baskı yönünün seçimi de önemli bir rol oynadı. Öğrenciler, parçanın baskı alanındaki yerleşiminin doğrudan bir etkisi olduğunu deneyimlediler.

Yüzey kalitesi, mekanik dayanıklılık ve gerekli destek malzemesi ve destek miktarı gibi faktörler dikkate alınmalıdır. Bu karar verme durumu, mekansal algının geliştirilmesini ve üretim teknolojisi yönlerinin bilinçli bir şekilde uygulanmasını kolaylaştırmıştır. Elemanların en uygun şekilde düzenlenmesiyle, öğrenciler birden fazla modeli tek bir tepsiye yerleştirebildikleri için daha optimize edilmiş baskı süreçlerine başlama fırsatı bulmuşlardır.



Şekil 59: Yazdırma optimizasyonu
(Kaynak: Kendi düzenlemem)

Doğrulama sürecinin sonunda, dilimleme işleminden önce modeller valide edildi ve bu da hataların erken tespitinin, sonradan düzeltilmesinden önemli ölçüde daha etkili olduğu görüşünü doğruladı.

5.9. Baskı ve son işlem

Baskı işleminin başında, öğrenciler tüm üretimin başarısı için en kritik faktörlerden biri olan ilk katmanın doğru yapışmasına büyük önem verdiler. X1 Carbon yazıcıda, ilk katman yerleştirildikten sonra kısa bir süre baskıyı durduran ve ardından dahili kamera aracılığıyla inceleyen yerleşik bir yapay zeka asistanı bulunmaktadır. Bir sorun tespit ederse, bunu dilimleme programı (veya ayrı olarak yüklenebilen Bambu Handy mobil uygulaması) aracılığıyla kullanıcıya bildirir. Baskıya başladıktan sonra, katılımcı öğrencilerin baskı kalitesini kendi gözleriyle analiz etmelerinin daha bilgilendirici olacağını düşündük. İlk katmanı gözlemlmek, öğrencilere hazırlık ve ayarların doğruluğu hakkında anında geri bildirim alma fırsatı verdi.



Şekil 60. Yazıcının çalışma halii.

Daha uzun baskı işleri sırasında sürecin sürekli olarak izlenmesi de önemli bir öğrenme unsuruydu. Öğrenciler, baskının tamamen otonom bir işlem olmadığını, müdahale veya yeniden tasarım gerektirebilecek bir üretim süreci olduğunu deneyimlediler. Örneğin, bir nesne yerinden oynarsa veya filament AMS dağıtıcısından biterse bu durum söz konusudur. Öğrenciler ayrıca bu durumların nasıl ele alındığını da görme fırsatı buldular.

Baskı işleminden sonra, bitmiş parçalar son işleme tabi tutuldu. Bu işlem, yüzey düzensizliklerinin giderilmesini ve bağlantı noktalarının iyileştirilmesini içeriyordu. Son işlemler sırasında, 3D baskının bazen işlevsel kullanım için ek manuel çalışma gerektiren yarı mamul bir ürünle sonuçlandığı anlaşıldı. Bu durum, modellerin karmaşıklığı ve destek miktarı arttıkça artmaktadır.



Şekil 61: Yanlış baskının sonuçları (Kaynak: Kendi düzenlemem)

5.10. Kalite kontrolü ve hata düzeltme

Üretilen parçaların tümü detaylı kalite kontrolünden geçirildi. Kontrol sadece estetik yönleri kapsamakla kalmadı, öncelikle işlevsel uygunluğu ve montajı değerlendirdi.

Öğrenciler, basılan parçaların boyutlarını dijital model verileriyle karşılaştırdı ve herhangi bir tutarsızlığın nedenlerini analiz etti. Hataları belirlerken, dilimleme ayarlarının, malzeme özelliklerinin ve baskı ortamının etkilerini de dikkate aldılar. Bu nedenle, hata düzeltilmesi izole bir müdahale değil, tüm üretim süreci boyunca bir geri bildirim döngüsü oldu.

Bu yaklaşım, analitik düşünmenin gelişimini destekledi ve sonraki tasarım yinelemelerinin profesyonelce gerçekleştirilmiş bir şekilde yürütülmesinin temelini attı.

5.11. Beden ve uyum sorunları

Projeden öğrenilen en önemli derslerden biri, boyutlandırma ve uyum sorunlarının pratik önemidir. Öğrenciler, dijital tasarımda kullanılan nominal boyutların üretim sırasında her zaman doğru uyumu sağlamadığını, bu nedenle toleransların bilinçli bir şekilde yönetilmesinin şart olduğunu bizzat deneyimlediler.

5.12. Üretim hatalarının giderilmesi

3 boyutlu baskı sırasında oluşan hatalar, malzeme büzülmesi, katman yapısı özellikleri ve baskı yönü gibi çeşitli faktörlerin birleşiminden kaynaklanıyordu. Bu olgular, üretim teknolojisinin sınırlamalarını öğrenciler için somut hale getirdi ve modellerin boyutlandırılmasında minimal ayarlamalar yapılmasını gerektirdi.

Üretimdeki farklılıkları gidermek için öğrenciler, montaj boşluklarının kullanımını ve hareketli veya birbirine kenetlenen parçaların, katı parçalardan farklı bir tasarım yaklaşımı gerektirdiğini öğrendiler. Bu yaklaşım, teknik düşünme becerilerini derinleştirmelerine yardımcı oldu.

5.13. Tasarım yinelemeleri

Boyutlandırma sorunlarının çözümü genellikle birden fazla tasarım ve üretim döngüsü gerektiriyordu. Ayrıca, parçaların baskı işleminden ziyade tasarım hataları nedeniyle birbirine uymadığı durumlarla da başa çıkmak gerekiyordu. Öğrenciler, deneyimlerine dayanarak modelleri değiştirdiler ve etkilenen parçaları yeniden bastılar, böylece kısa sürede kararlarının doğruluğu hakkında geri bildirim aldılar.

Parametrik tasarımın avantajı, özellikle yinelemeli süreçte belirginleşti; çünkü boyutları hızlı bir şekilde değiştirme ve yeniden üretme yeteneği, verimli deneyler yapılmasına olanak sağladı. Bu yöntem, öğrencilerin statik çözümler yerine esnek bir geliştirme süreci açısından düşünmelerine yardımcı oldu.

5.14. Süreçlerin, öğrenci deneyimlerinin ve öğrenilen derslerin eğitimsel önemi

Ölçü ve uyum sorunlarının ele alınması, öğrencilerin gerçek mühendislik durumlarıyla karşı karşıya kalmaları nedeniyle olağanüstü bir eğitim değeri taşıdı. Sorunları analiz etme, hataları tanıma ve çözümler geliştirme yoluyla bağımsız düşünme ve sorumlu karar verme becerileri gelişti.

Öğrenciler, tasarım ve imalatın birbirine yakın bir bütün oluşturduğunu ve ikisinin de birbirleri olmadan anlaşılamayacağını açıkça kavradılar. Bu deneyimler, projeye sadece teknolojik bilgi aktarmakla kalmadı, aynı zamanda uzun vadede kullanılabilir bir yaklaşım geliştirmeye de katkıda bulundu.

Ayrıca, 3D baskıda kazanılan deneyim, diğer dersleri (matematik, fizik, teknik dersler) öğrenirken ve hatta ev ortamında kendi 3D yazıcınızı kullanırken bile kullanılabilir ek bilgi aktarım fırsatları sunar.



Şekil 62. Baskılı, monte edilmiş gövde

6. Entegre kontrol sistemi - kavramsal temeller

6.1. Sistem tasarımının amacı

Projenin elektronik-yazılım aşamasında bilinçli bir tasarım kararı alındı: sistem tek bir mikrodenetleyiciye dayalı monolitik bir cihaz değil, dağıtık bir kontrol mimarisi şeklinde uygulandı.

Şunu vurgulamak önemlidir ki, bu durum ESP32'nin "zayıflığından" kaynaklanmamaktadır. ESP32 mükemmel bir saha kontrolcüsü ve veri toplama ünitesidir: istikrarlı dijital G/Ç yönetimi sağlar, çoklu iletişim arayüzlerine (I²C, SPI, UART, WiFi) sahiptir ve bağımsız kontrol mantığı çalıştırmak için de uygundur. GPIO sayısı yeterli değilse, endüstri uygulamasına uygun olarak, G/Ç genişleticiler (örneğin I²C port genişleticiler) veya ek düğümler kullanılabilir - bu da dağıtılmış sistemlere işaret etmektedir.

Geliştirme sürecindeki gerçek sınırlama, giriş/çıkış veya iletişim alanında değil, izleme ve görüntüleme (HMI) fonksiyonlarının uygulanmasında ortaya çıktı. ESP32:

- Sınırlı RAM ve flash depolama alanına sahip.
- Web tabanlı olmayan kullanıcı arayüzleri, veritabanı yönetimi ve kalıcı günlük kaydı işlemleri için optimize edilmiştir.
- Karmaşık grafikler görüntülerken ve birden fazla istemciye hizmet verirken, kaynak sınırlarına hızla ulaşıyor.

Pratikte bu, ESP32'nin saha görevlerini (sensörler, aktüatörler, yerel kontrol) istikrarlı bir şekilde yerine getirebilmesine rağmen, sistemin izleme katmanının daha yüksek işlem gücüne sahip bir cihazın dahil edilmesini gerektirdiği anlamına geliyordu.

Ortaya çıkan mimari, sistemi açıkça birbirinden farklı katmanlara ayırır:

Seviye	İşlev	Cihaz
Saha seviyesi	Sensör verisi toplama, aktüatör kontrolü, yerel mantık	ESP32
Kontrol / veri yönetimi seviyesi	Döngüsel işlem, mantıksal kararlar, Modbus ana rolü	Raspberry Pi / PC (Modbus)
Denetim seviyesi (HMI)	Web ekranı, kullanıcı kontrolü, veri kaydı	Raspberry Pi / PC (Flask tabanlı sunucu)

Bu modelde, ESP 32 bir saha düğümü görevi görür ve endüstriyel bir modeli izleyerek, standart bir protokol (Modbus TCP) aracılığıyla üst düzey sisteme veri sağlar. Tamamen monolitik, tek cihazlı bir sistemde bu iletişim standardının gerekli olmayacağını belirtmek önemlidir; ancak dağıtılmış mimari bunu haklı çıkarır.

Dolayısıyla projenin amacı sadece çalışan bir akıllı çiftlik modeli oluşturmak değil, aynı zamanda şunları göstermekti:

- Bir mikrodenetleyicinin nasıl bir alan düğümüne dönüştüğü,
- Denetleyici sistem tarafındaki mantık ve HMI katmanının bunun üzerinde nasıl organize edildiği,
- ve endüstriyel otomasyonun katmanlı, ağ tabanlı mimarisinin eğitim ortamında nasıl modellenebileceği.

Bu yaklaşım, öğrencileri gerçek endüstriyel sistemlerin işleyişine daha da yaklaştırırken, sistem yeni sensörler, düğümler ve hizmetlerle kolayca genişletilebilir.

Dağıtılmış sistem kavramı

Projede geliştirilen kontrol modeli, bilinçli olarak iki seviyeli, katmanlı bir mimariyi takip etmektedir. Bu yapı sadece teknik bir karar değil, aynı zamanda monolitik, "her şey tek bir mikrodenetleyici üzerinde" yaklaşımından endüstriyel sistemlere özgü yapıya doğru atılmış bir adımdır.

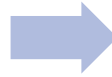
En alt seviye, sensörlerin ve aktüatörlerin doğrudan kontrol edildiği saha seviyesidir. Sistemde bu rolü ESP32 üstlenir. Görevleri arasında dijital ve analog sinyallerin işlenmesi, sensör verilerinin toplanması, temel ön işleme ve fiziksel çıkışların (örneğin LED, röle, servo) çalıştırılması yer alır. Bu seviye, gerçek fiziksel süreçle doğrudan bağlantının kurulduğu yerdir.

Üst katman, projede Raspberry Pi (veya PC) tabanlı bir sistemle uygulanan denetim seviyesidir. Önemli bir fark, bu seviyede merkezi kontrol mantığının çalışmaması, bunun yerine sistemin dijital durum görüntüsünün gösterilmesidir. Pi, saha seviyesindeki cihazlardan (ESP32) veri toplar, kaydeder ve bir HMI web arayüzünde görüntüler ve uzaktan müdahale olanağı sağlar.

Gerçek kontrol, saha seviyesinde, ESP32 üzerinde gerçekleştirilir. G/Ç işlemleri, sensör verisi işleme ve temel işletim mantığı yerel olarak çalışır, bu nedenle sistem çevrimdışı modda bile çalışır durumda kalır. İzleme katmanının arızalanması fiziksel süreci durdurmaz, yalnızca ekran ve uzaktan erişim etkilenir.

Bu yapı, aşağıdaki endüstriyel modelle daha doğru bir şekilde karşılaştırılabilir:

Uzaktan giriş/çıkış + entegre kontrol



Denetim sistemi (SCADA/HMI)

Projedeki karşılıkları şunlardır:

- ESP32, akıllı bir saha ünitesi olarak
 - Giriş/çıkış işlemlerini gerçekleştirir.
 - sensör verilerini toplar
 - yerel kontrol mantığını çalıştırır
 - Modbus denetim sistemi olarak iletişim kurar.
- Raspberry Pi + Python + Flask'ı yönetim katmanı olarak kullanan bir yapı.
 - Sistem durumunu yansıtan
 - verileri toplar ve görüntüler.

- Operatör müdahalesine izin verir ancak
- sürecin işleyişi için kritik değildir.

Gelecekte, Raspberry Pi seviyesinde aşağıdaki gibi ek işlevler uygulanabilir:

- veri kaydı ve uzun vadeli trend analizi
- grafik ekran
- kullanıcı yetkilendirme yönetimi
- bildirimler, alarm kayıtları
- uzaktan yapılandırma arayüzü

Bu işlevler yalnızca denetim ve bilgilendirme niteliğinde olmalıdır. Raspberry Pi üzerinde çalışan eklentilerin sistemin temel işlevselliğini veya güvenliğini etkilememesi bir tasarım prensibidir. Kritik kontrol mantığı ve temel durum yönetimi saha düzeyinde kalmalıdır.

6.2. Dağıtılmış mimarinin teknik avantajları

Sistemin anahtarlarından biri, güvenilirliği ve ölçeklenebilirliği doğrudan artıran fonksiyonel ayrımdır. ESP32 saha seviyesinde kalır: sensörler, girişler, çıkışlar ve temel kontrol mantığı yerel olarak çalışır, bu nedenle fiziksel sürecin çalışması sürekli bir ağ bağlantısına bağlı değildir. Bu, daha belirleyici G/Ç davranışı ve daha istikrarlı bir çalışma sağlar.

Üst katman, kritik kontrol yerine izleme, veri yönetimi ve görselleştirmeden sorumludur. Bu katmanlı mimari, saha cihazının, kontrol katmanının ve izleme sisteminin ayrı rollerde çalıştığı endüstriyel bir modeli takip eder.

Mimari yapının önemli bir avantajı ölçeklenebilirliğidir: ağa yeni ESP tabanlı düğümler eklenebilir, ek sensörler entegre edilebilir ve sistem, tek bir cihazın kaynaklarını zorlamadan işlevsel olarak genişletilebilir. Bu yatay genişleme, dağıtık sistemlerin temel bir özelliğidir.

Güvenilirlik açısından bakıldığında, temel prensip saha seviyesinin özerk kalmasıdır: denetim sisteminin veya ağın arızalanması yerel işlemleri durdurmaz, yalnızca izleme ve uzaktan müdahale etkilenir. Bu, endüstriyel otomasyon uygulamasına doğrudan karşılık gelir.

6.3. İletişim modeli ve yaklaşımı

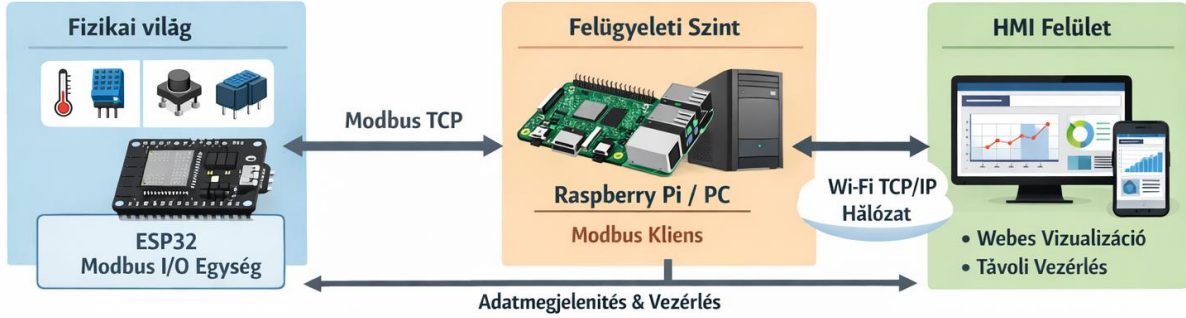
Katmanlar arasındaki bağlantı Modbus TCP'ye dayanmaktadır. Bu, standart bir kayıt yapısıyla durum tabanlı veri alışverişini uygular. Sensör verileri ve kontrol sinyalleri belirli "istek ve yanıtlar" şeklinde değil, iyi tanımlanmış adreslerde erişilebilen durumlar olarak hareket eder. Bu, tipik HTTP tabanlı, işlemsel yaklaşımdan farklıdır ve kontrolörün alan düzeyindeki durumları döngüsel olarak okuyup ardından çıkış kayıtlarını buna göre yazdığı endüstriyel iletişim modellerine daha yakındır.

Sistem çerçevesi - mantıksal yapı

Genel olarak, sistem veri yolu şu şekilde tanımlanabilir:

Fiziksel dünyadaki sensörler (düğme, DHT vb.) ve aktüatörler ESP32'ye bağlanır. ESP32, Modbus denetim sistemi olarak çalışır ve giriş ve çıkışların durumunu kayıt defterleri aracılığıyla kullanılabilir hale getirir. WiFi tabanlı TCP ağı üzerinden, denetim seviyesinde çalışan sistem – Raspberry Pi veya PC – bu verileri Modbus istemcisi olarak döngüsel olarak okur ve yazar. İşlenen durumlar ve ölçülen değerler daha sonra HMI web arayüzünde görüntülenir; burada görselleştirme ve uzaktan kontrol de uygulanır.

Bu modelde ESP merkezi bir kontrol ünitesi değil, akıllı bir saha giriş/çıkış ünitesidir; Raspberry Pi ise mantık ve izleme merkezi görevi görür.



Şekil 63: Sistem çerçevesi (Kaynak: Yapay zeka tarafından oluşturulmuştur)

Eğitimsel önemi

Uygulamalı mimari sadece teknik bir çözüm değil, aynı zamanda tutumları şekillendiren bir araçtır. Öğrenciler, saha ve denetim seviyelerinin ayrılmasının nedenini, kaynak tahsisinin rolünü ve endüstriyel otomasyon sisteminin katmanlı yapısının nasıl oluşturulduğunu anlarlar. Bu, PLC programlama, HMI tasarımı ve SCADA sistemlerini anlamak için doğrudan bir temel sağlar, çünkü projede kullanılan model bunların basitleştirilmiş ancak gerçekçi bir temsildir.

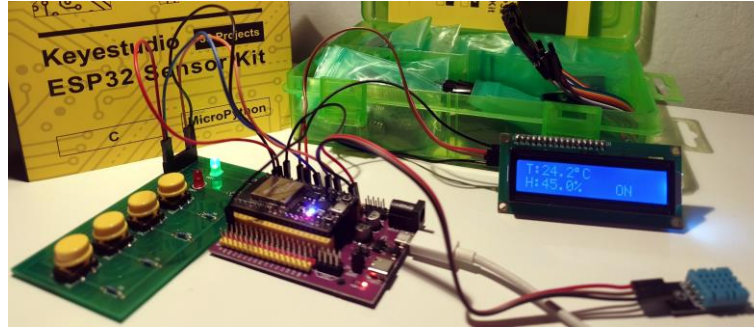
Sonraki bölümde, mimarinin özel yazılım uygulamasını sunacağız: ESP alan programı, izleme tarafındaki döngüsel işlem ve web arayüzünün çalışması.

6.4. Alan kontrol düğümü - bağımsız numune alma ünitesi

Projenin elektronik sisteminin alt seviyesi, ESP32 mikrodenetleyici tabanlı, kendi kendine yeten bir alan düğümü ile temsil edilmektedir. Bu ünite doğrudan fiziksel dünyaya bağlıdır: sensörleri okur, aktüatörleri kontrol eder ve yerel geri bildirim sağlar. Sunulan yapılandırma, sistemin temel mantığını gösteren ve nihai bir yapıyı temsil etmeyen, kasıtlı olarak basitleştirilmiş, eğitici bir örnektir.

Bu düğüm aşağıdaki öğeleri yönetir:

- Dijital giriş görevi gören bir basma düğmesi,
- dijital çıkış LED'i,
- bir DHT11 sıcaklık ve nem sensörü,
- I²C veri yoluna bağlı 16x2 karakterli bir LCD ekran.



Şekil 64: Test ortamı (Kaynak: Kendi düzenlemem)

Bu işlemin özü, cihazın temel görevlerini tamamen bağımsız olarak, ağ bağlantısı olmadan gerçekleştirmesidir. Kullanıcı, fiziksel düğme ile sistem durumuna (LED açma/kapama) müdahale edebilirken, çevresel veriler sürekli olarak ölçülür ve ekranda görüntülenir. Bu yaklaşım, saha seviyesinin izleme sisteminden bağımsız olarak, otonom olarak çalışabilmesi gerektiği endüstriyel prensibini modellemektedir.

Sunulan program, akıllı bir G/Ç biriminin davranışını göstermektedir. Yapı modülerdir: yeni sensörler, çıkış cihazları veya ekranlar, ek GPIO'lar veya iletişim veri yolları kullanılarak kolayca eklenebilir. Daha sonra, bu düğüm daha üst düzey sistemlerle (PLC, HMI, SCADA) de entegre edilebilir, ancak bu bölümde yalnızca yerel kaynakların yönetimine odaklanıyoruz.

6.5. Program yapısının fonksiyonel birimlere ayrılması,

kütüphaneler ve donanım tanımları.

```
1 // ===== KÖNYVTÁRÁK BETÖLTÉSE =====
2 // Ebben a szakaszban történik a felhasznált perifériákhoz szükséges könyvtárak betöltése.
3 // DHT könyvtár a szenzor kommunikációját valósítja meg.
4 // A Wire és hd44780_I2Cexp az I°C LCD vezérléséhez szükséges.
5 #include <DHT.h>
6 #include <Wire.h>
7 #include <hd44780.h>
8 #include <hd44780ioClass/hd44780_I2Cexp.h>
9
10
11 // ===== HARDVER KIOSZTÁS =====
12 //A #define direktívák rögzítik a fizikai bekötés és a program közötti kapcsolatot, így a
13 //hardverkonfiguráció egyértelműen dokumentált és könnyen módosítható.
14 #define BTN 17 // Fizikai gomb bemenet
15 #define LED 2 // LED kimenet
16 #define DHTPIN 4 // DHT adatvonal
17 #define DHTTYPE DHT11 //DHT típus megadása
```

Bu aşamada, kullanılan çevre birimleri için gerekli kütüphaneler yüklenir.

- DHT kütüphanesi sensör iletişimini uygular.
- I°C LCD'yi kontrol etmek için Wire ve hd44780_I2Cexp gereklidir.

#define yönergeleri, fiziksel bağlantı ile program arasındaki ilişkiyi kaydeder; böylece donanım yapılandırması açıkça belgelenir ve kolayca değiştirilebilir.

Nesneler ve global değişkenler

```
20 //Itt jönnek létre a szenzor- és kijelzőobjektumok. Ezek a program teljes futása alatt elérhetők, így
    bármelyik ciklusban használhatók.
21 // Szenzor objektum
22 DHT dht(DHTPIN, DHTTYPE);
23 // I2C LCD objektum
24 hd44780_I2Cexp lcd;
```

Szenzor ve ekran nesneleri burada oluşturulur. Bunlar programın tüm çalışma süresi boyunca kullanılabilir, dolayısıyla herhangi bir döngüde kullanılabilirler.

Düğme gecikmesi yönetimi

```
27 // ===== GOMB PRELLEGÉS SZÜRÉS =====
28 // A mechanikus gombok érintkezési záraskor rövid ideig „rezegnek” (prellegnek), ami több hamis jelváltást
    okozhat.
29 // Ez a logika időalapú szűréssel biztosítja, hogy csak a stabil állapotváltás legyen érvényes.
30
31 bool lastReading = false; // Legutóbbi nyers bemenet
32 bool stableState = false; // Szűrt, stabil állapot
33 unsigned long lastChangeTime = 0;
34 const unsigned long debounceTime = 50; // 50 ms stabilitási küszöb
```

Mekanik butonların kontakları, kapatıldıklarında kısa bir süre "titreşir" (sekme hareketi yapar), bu da birden fazla yanlış sinyal geçişine neden olabilir. Bu mantık, yalnızca kararlı durum geçişlerinin geçerli olmasını sağlamak için zamana dayalı filtreleme kullanır.

LED durumu ve ölçüm değişkenleri

```
37 // A LED állapota külön változóban tárolódik. A DHT mérések időzítéséhez és az aktuális értékek
    tárolásához is globális változók szükségesek.
38 bool ledState = false;
39 unsigned long lastDHTread = 0; // Utolsó olvasás időpontja
40 float temp = 0; // Hőmérséklet cache
41 float hum = 0; // Páratartalom cache
```

LED durumu ayrı bir değişkende saklanır. DHT ölçümlerinin zamanlaması ve akım değerlerinin saklanması için de global değişkenler gereklidir.

Başlatma – setup()

```
50 void setup() {
51
52     // Hardver inicializálás
53     pinMode(BTN, INPUT);
54     pinMode(LED, OUTPUT);
55
56     // DHT inicializálás
57     dht.begin();
58
59     // I2C inicializálás ESP32 alapértelmezett lábakkal
60     Wire.begin(21,22);
61
62     // LCD inicializálás
63     lcd.begin(16,2);
64     lcd.backlight();
65
66     lcd.setCursor(0,0);
67     lcd.print("Smart Farm Node");
68     delay(2000);
69     lcd.clear();
}
```

setup() fonksiyonu başlatma sırasında bir kez çalışır. İşte olanlar:

- GPIO yönlerini ayarlama,
- Sensörün ve I²C veri yolunun başlatılması,
- LCD'yi başlatıyorum.

Düğme kontrolü ve LED anahtarlama

Bu blok, düğmeye basıldığında LED'in yanıp sönmesini sağlar. LED yalnızca düğmeye sürekli basıldığında değişir.

```
74 //Ez a blokk valósítja meg a lenyomási élre történő LED-váltást.
75 //A LED csak akkor változik, amikor a gomb stabilan lenyomott állapotba kerül.
76 // Gomb olvasása
77 bool reading = digitalRead(BTN);
78
79 // Prellégés detektálása
80 if (reading != lastReading) {
81     lastChangeTime = millis();
82 }
83
84 // Stabil állapot vizsgálata
85 if ((millis() - lastChangeTime) > debounceTime) {
86
87     // Állapotváltozás kezelése
88     if (reading != stableState) {
89         stableState = reading;
90
91         // Lenyomás detektálás
92         if (stableState == true) {
93             ledState = !ledState; // LED kapcsolás
94         }
95     }
96 }
97
98 lastReading = reading;
```

DHT11 periyodik ölçümü

```
100 // DHT szenzor olvasás időzítetten
101 //A szenzor csak 2 másodpercenként kerül kiolvasásra, ami megfelel a DHT11 időzítési követelményeinek.
102 if(millis() - lastDHTread > 2000){
103     lastDHTread = millis();
104
105     float h = dht.readHumidity();
106     float t = dht.readTemperature();
107
108     // Hibás olvasás kizárása
109     if(!isnan(h) && !isnan(t)){
110         temp = t;
111         hum = h;
112     }
113 }
```

Szenzor yalnızca 2 saniyede bir okunuyor, bu da DHT11'in zamanlama gereksinimlerini karşılıyor.

LCD ekran

```
115 // LCD kijelzés
116 //A kijelző a mért adatokat és a LED állapotát jeleníti meg, így a rendszer állapota hálózat nélkül is ellenőrizhető.
117 lcd.setCursor(0,0);
118 lcd.print("T:");
119 lcd.print(temp,1);
120 lcd.print((char)223);
121 lcd.print("C ");
122
123 lcd.setCursor(0,1);
124 lcd.print("H:");
125 lcd.print(hum,1);
126 lcd.print("% ");
127
128 lcd.setCursor(11,1);
129 lcd.print(ledState ? "ON " : "OFF");
```

Ekranda ölçülen veriler ve LED durumu gösterilir, böylece ağ bağlantısı olmadan bile sistem durumu kontrol edilebilir.

Özet

Bu program, otonom bir saha kontrol ünitesinin çalışmasını göstermektedir:

- fiziksel girdi işleme
- aktüatör kontrolü
- sensör verisi toplama
- yerel görselleştirme

Kod yapısı, ayrı fonksiyonları iyi bir şekilde ayırıyor; bu da daha sonra genişletmeyi ve üst düzey sistemlere bağlanmayı kolaylaştırıyor.

6.6. Ağ genişletme özelliğine sahip saha kontrol düğümü - WiFi + Modbus TCP

Bu sürümde, daha önce tanıtilen bağımsız saha kontrol cihazına ağ özellikleri eklenmiştir.

Yerel işlemler (düğme → LED, DHT ölçümü, LCD ekran) değişmeden kalır, ancak cihaz artık şunları yapabilir:

- WiFi ağına bağlanın
- Modbus TCP izleme sistemi olarak görev yapın.
- Fiziksel giriş ve çıkışların durumunu kayıtlar aracılığıyla kullanılabilir hale getirin.
- Harici bir sistem (PLC / izleme yazılımı) aracılığıyla uzaktan okuma ve müdahale için

Mantığın saha cihazının dışına çıkmaması önemlidir: yerel kontrol özerk kalır, ağ yalnızca durum paylaşımı ve uzaktan müdahale için kullanılır.

Fonksiyonel birimlere göre program yapısı,

kütüphaneler ve donanım tanımları

```
1 #include <WiFi.h> // ESP32 WiFi stack
2 #include <ModbusTCP.h> // Modbus TCP szerver kezelése
3 #include <DHT.h> // DHT szenzor magas szintű kezelése (protokollt elrejtí)
4 #include <Wire.h> // I2C kommunikáció (LCD miatt)
5 #include <hd44780.h> // LCD vezérlő
6 #include <hd44780ioClass/hd44780_I2Cexp.h> // I2C LCD expander kezelés
7
8 ModbusTCP mb; // Az ESP Modbus TCP szerverének létrehozása.
9
10
11 // ===== HARDVER KIOSZTÁS =====
12 // GPIO-k definiálása hardver absztrakcióként
13
14 #define BTN 17 // Fizikai gomb bemenet
15 #define LED 2 // Fő LED kimenet (vezérelt fogyasztó)
16 #define WIFI_LED 5 // WiFi státusz visszajelző LED (terepi hibajelzés)
17 #define DHTPIN 4 // DHT adatvonal
18 #define DHTTYPE DHT11 // Szenzor típusa
19
20 // Szenzor objektum létrehozása
21 DHT dht(DHTPIN, DHTTYPE);
22
23 // I2C LCD objektum létrehozása (automatikus I2C címfelismerés)
24 hd44780_I2Cexp lcd;
```

Ağ parametreleűtirmesi

```
27 // ===== WIFI HÁLÓZATI PARAMÉTEREK =====
28 // Ezek konfigurálják a hálózati kapcsolatot, valamint biztosítják,
29 // hogy az ESP mindig ugyanazon IP címen legyen elérhető.
30
31 const char* ssid = "plc";
32 const char* pass = "12345678";
33
34 IPAddress local_IP(192, 168, 137, 200);
35 IPAddress gateway(192, 168, 137, 1);
36 IPAddress subnet(255, 255, 255, 0);
37 IPAddress primaryDNS(8,8,8,8);
38 IPAddress secondaryDNS(8,8,4,4);
39
40
41 // ===== WIFI ÚJRACSATLAKOZÁSI LOGIKA =====
42 // A rendszer nem blokkoló reconnect stratégiát használ.
43 // Ha a kapcsolat megszakad, a program nem áll meg,
44 // hanem meghatározott időközönként újrapróbál csatlakozni.
45
46 unsigned long lastReconnectAttempt = 0; // Utolsó reconnect próbálkozás időpontja
47 const unsigned long reconnectInterval = 300000; // 5 perc újrapróbálkozási ciklus
48 bool wifiConnected = false; // Hálózati állapot logikai jelző
49
50 // WiFi csatlakozási rutin
51 // Ez a függvény bontja az előző kapcsolatot,
52 // majd új kapcsolatot kezdeményez a megadott SSID-re.
53 void connectToWiFi() {
54     WiFi.disconnect(true);
55     delay(100); // Rövid stabilizációs várakozás
56     WiFi.begin(ssid, pass);
57 }
58
59
60 // ===== WIFI HIBA LED IDŐZÍTÉS =====
61 // Amennyiben nincs WiFi kapcsolat,
62 // a WIFI_LED másodperces periódussal villog.
63 // A megvalósítás millis() alapú, tehát nem blokkolja a fő ciklust.
64
65 unsigned long lastBlinkTime = 0; // Utolsó LED állapotváltás
66 const unsigned long blinkInterval = 1000; // 1 másodperces villogási periódus
67 bool wifiLedState = false; // WiFi LED aktuális állapota
```

Dűgme titreűşim giderme, LED durumu ve ölçüm deęişkenleri

```
70 // ===== BELSŐ ÁLLAPOT RÉTEG =====
71 // A rendszer egyetlen "igazságforrást" használ a LED állapotára.
72 // A fizikai gomb, a Modbus és a fizikai LED is ezt az állapotot követi.
73
74 bool ledState = false; // A LED központi logikai állapota
75
76
77 // ===== GOMB PRELL SZŰRÉS =====
78 // Mechanikus gomb érintkezése záráskor pattog (prell).
79 // A stabil állapot detektálás millis alapú időszűréssel történik.
80
81 bool lastReading = false; // Utolsó nyers bemeneti érték
82 bool stableState = false; // Szűrt, stabil állapot
83 unsigned long lastChangeTime = 0;
84 const unsigned long debounceTime = 50; // 50 ms stabilitási küszöb
85
86
87 // ===== DHT MINTAVÉTELI IDŐZÍTÉS =====
88 // A DHT szenzor olvasása nem történhet túl gyakran,
89 // ezért 2 másodperces mintavételezési ciklust alkalmazunk.
90
91 unsigned long lastDHTread = 0;
92 float temp = 0;
93 float hum = 0;
```

Başlatma

```
97 // ===== SETUP =====
98 // =====
99 void setup() {
100
101     Serial.begin(115200);
102
103     // Hardver inicializálás
104     pinMode(BTN, INPUT);
105     pinMode(LED, OUTPUT);
106     pinMode(WIFI_LED, OUTPUT);
107
108     // ===== Stabil statikus IP konfiguráció =====
109     // Az ESP32 WiFi stack sajátossága miatt:
110     // - Station mód explicit beállítása szükséges
111     // - DHCP cache törlése
112     // - Statikus IP konfigurálása csatlakozás előtt
113
114     WiFi.mode(WIFI_STA);
115     WiFi.persistent(false);
116     WiFi.disconnect(true);
117     delay(100);
118
119     WiFi.config(local_IP, gateway, subnet, primaryDNS, secondaryDNS);
120     WiFi.begin(ssid, pass);
121
122     // Induláskori maximum 10 másodperces várakozás.
123     // Ha ezalatt nem jön létre kapcsolat,
124     // a program offline módban folytatódik.
125     unsigned long startAttemptTime = millis();
126     while (WiFi.status() != WL_CONNECTED &&
127           millis() - startAttemptTime < 10000) {
128         delay(500);
129     }
130
131     wifiConnected = (WiFi.status() == WL_CONNECTED);
132
133     // Szenzor és perifériák inicializálása
134     dht.begin();
135     Wire.begin(21,22);
136
137     lcd.begin(16,2);
138     lcd.backlight();
139     lcd.setCursor(0,0);
140     lcd.print("Smart Farm Node");
141     delay(2000);
142     lcd.clear();
```

Modbus kayıt tahsisi

```
144 // ===== MODBUS REGISZTERKIOSZTÁS =====
145 // Coil 0 → LED állapot
146 // Ists 0 → Gomb állapot
147 // Hreg 0 → Hőmérséklet (x10 skálázva)
148 // Hreg 1 → Páratartalom (x10 skálázva)
149
150 mb.server();
151 mb.addCoil(0);
152 mb.addIsts(0);
153 mb.addHreg(0);
154 mb.addHreg(1);
155
156 // Coil inicializálása a belső logikai állapotból
157 mb.Coil(0, ledState);
158 }
```

WiFi izleme, aktív bağlantılı Modbus hizmeti.

```
164 void loop() {
165
166     // ===== WIFI FELÜGYELET =====
167     // Nem blokkoló hálózati állapotellenőrzés.
168     // Ha nincs kapcsolat, 5 percenként újrapróbál.
169
170     if (WiFi.status() != WL_CONNECTED) {
171
172         wifiConnected = false;
173
174         if (millis() - lastReconnectAttempt >= reconnectInterval) {
175             lastReconnectAttempt = millis();
176             connectToWiFi();
177         }
178     }
179     else {
180         wifiConnected = true;
181     }
182
183     // ===== WIFI HIBA LED KEZELÉS =====
184     // Offline állapotban 1 Hz villogás.
185     // Online állapotban LED kikapcsolva.
186
187     if (!wifiConnected) {
188
189         if (millis() - lastBlinkTime >= blinkInterval) {
190             lastBlinkTime = millis();
191             wifiLedState = !wifiLedState;
192             digitalWrite(WIFI_LED, wifiLedState);
193         }
194     }
195     else {
196         digitalWrite(WIFI_LED, LOW);
197         wifiLedState = false;
198     }
199
200     // ===== MODBUS KISZOLGÁLÁS =====
201     // Csak aktív hálózat esetén kezel TCP kéréseket.
202     if (wifiConnected) {
203         mb.task();
204     }
205
206     // =====
207     // ===== MODBUS - BELSŐ ÁLLAPOT SZINKRON =====
208     // =====
209     // Ha külső rendszer írja a Coil 0-t,
210     // a belső állapot frissül.
211
212     bool coilState = mb.Coil(0);
213
214     if (coilState != ledState) {
215         ledState = coilState;
216     }
217 }
```

Düğme okuma, LED durum kontrolü ve uzaktan izleme

```
218 // =====  
219 // ===== GOMB - BELSŐ ÁLLAPOT =====  
220 // =====  
221 // Fizikai gombnyomás esetén a belső állapot toggolódik,  
222 // majd a Coil regiszter tükrözi azt.  
223  
224 bool reading = digitalRead(BTN);  
225  
226 if (reading != lastReading) {  
227     lastChangeTime = millis();  
228 }  
229  
230 if ((millis() - lastChangeTime) > debounceTime) {  
231  
232     if (reading != stableState) {  
233         stableState = reading;  
234  
235         if (stableState == true) {  
236             ledState = !ledState;  
237             mb.Coil(0, ledState);  
238         }  
239     }  
240 }  
241  
242 lastReading = reading;  
243  
244 // =====  
245 // ===== FIZIKAI LED VEZÉRLÉS =====  
246 // =====  
247 // A fizikai kimenet mindig a belső állapotot követi.  
248  
249 digitalWrite(LED, ledState);
```

Periodik ölçüm, DHT11'in uzaktan izlenmesi, Modbus üzerinden düğme durumu raporlaması

```
251 // =====  
252 // ===== DHT IDŐZÍTETT OLVASÁS =====  
253 // =====  
254  
255 if (millis() - lastDHTread > 2000) {  
256  
257     lastDHTread = millis();  
258  
259     float h = dht.readHumidity();  
260     float t = dht.readTemperature();  
261  
262     if (!isnan(h) && !isnan(t)) {  
263         mb.Hreg(0, (int)(t * 10));  
264         mb.Hreg(1, (int)(h * 10));  
265         temp = t;  
266         hum = h;  
267     }  
268 }  
269  
270 // Gomb státusz jelentése Modbuson  
271 mb.Ists(0, stableState);
```

LCD ekran

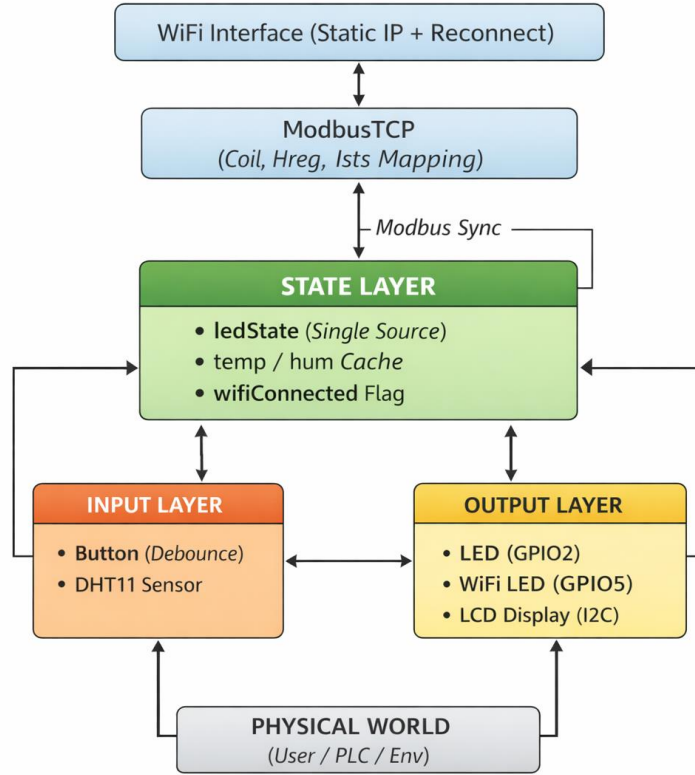
```
273 // =====  
274 // ===== LCD KIJELEZÉS =====  
275 // =====  
276 // Első sor: Hőmérséklet + Hálózati állapot  
277 // Második sor: Páratartalom + LED állapot  
278  
279 lcd.setCursor(0,0);  
280 lcd.print("T:");  
281 lcd.print(temp,1);  
282 lcd.print((char)223);  
283 lcd.print("C ");  
284 lcd.print(wifiConnected ? "N-ON " : "N-OFF");  
285  
286 lcd.setCursor(0,1);  
287 lcd.print("H:");  
288 lcd.print(hum,1);  
289 lcd.print("% ");  
290 lcd.print(ledState ? "L-ON " : "L-OFF");  
291  
292 delay(10); // CPU terhelés csökkentése  
293 }
```

Sistem blok diyagramı

Blok diyagramı, ESP32 tabanlı yazılımın katmanlı, PLC benzeri mimarisini göstermektedir. En üst seviyede, yalnızca ağ veri alışverişi ve kayıt alanı yönetiminden sorumlu olan WiFi arayüzü ve ModbusTCP iletişimi yer almaktadır. Sistemin merkezi, tüm dahili durum değişkenlerini (LED durumu, ölçülen sıcaklık, nem ve WiFi durumu gibi) tek bir doğruluk kaynağı olarak depolayan Durum Katmanıdır.

Giriş katmanı fiziksel girişleri, çıkış katmanı ise çıkışları yönetir ve her ikisi de doğrudan dahili durum katmanına bağlıdır. Kontrol tamamen ESP32 üzerinde, yani saha seviyesinde kalır, bu nedenle sistem ağdan bağımsız olarak otonom olarak çalışır. Modbus ve denetim katmanları yalnızca dahili durumu dijital bir gölge olarak yansıtır, ancak kontrol mantığını taşımaz. Bu yapı, deterministik, çevrimdışı çalışmayı ve endüstriyel sistemlere özgü sağlam mimariyi sağlar.

Firmware Architecture



Şekil 65: Sistem blok diyagramı (Kaynak: Yapay zeka tarafından oluşturulmuştur)

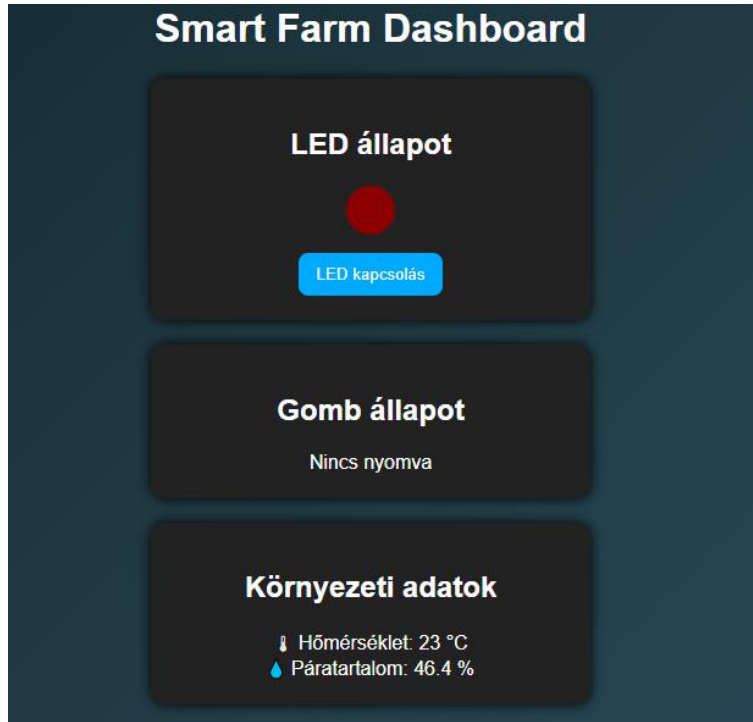
6.7. Denetim seviyesi - Python tabanlı HMI / denetim döngüsü

ESP32 saha düğümü, sensörlerin ve aktüatörlerin durumunu Modbus kayıtları aracılığıyla kullanılabilir hale getirir. Üst seviye, fiziksel kontrolü devralmaktan değil, sistemi dijital olarak haritalamaktan ve izlemekten sorumludur.

Projenin bu katmanı, HMI/SCADA tipi bir izleme sistemi olarak işlev gören, Python tabanlı, döngüsel bir uygulamadır.

Görevler:

- Alan verilerinin döngüsel sorgusu
- durumların saklanması ve işlenmesi
- uzaktan müdahale seçenekleri sunmak
- web görüntüleme hizmeti



Şekil 66: Python tabanlı HMI (Kaynak: Kendi düzenlemem)

6.8. Raspberry Pi tabanlı bir izleme sisteminin hazırlanması ve devreye alınması

Bir sonraki adım, yazılım izleme mantığını, endüstriyel ortamlarda da kullanılabilen, bağımsız ve düşük güç tüketimli bir donanım üzerinde çalıştırmaktı. Bu amaçla **Raspberry Pi** Ağ iletişimi ve web arayüzü içeren Python tabanlı uygulamalar için ideal bir platform sağlayan Raspberry Pi tabanlı bir bilgisayar kullandık. Raspberry Pi, iyi bir fiyat-performans oranına sahip, nispeten yüksek performanslı bir bilgisayardır ve çeşitli uygulamalara esnek bir şekilde uyarlanabilir. Varsayılan olarak, projenin gereksinimlerini tamamen karşılayan Debian tabanlı bir dağıtım olan Raspberry OS'yi çalıştırır.



Şekil 67: Raspberry Pi (Kaynak: <https://malnapc.hu/raspberry-pi-5-4gb>)

Bu mimarideki Raspberry Pi **bir izleme ve görselleştirme katmanı olarak** Bu, Modbus TCP protokolü aracılığıyla ESP32 tabanlı alt sistemle iletişim kuran bir kontrolör görevi görürken, kullanıcılara web tabanlı bir HMI arayüzü de sağlar.

İşletim sistemi ve temel sistem kurulumu

Cihaz hazırlığı, istikrarlı ve uzun vadeli destekli bir Linux tabanlı işletim sisteminin kurulumuyla başladı. Özel işletim sistemleri kurmak mümkün olsa da, Raspberry Pi OS uygun donanım desteği, sistem istikrarı ve Python geliştirme ortamına kolay erişim sağladı. Sistem, resmi Raspberry Pi Imager programı kullanılarak kuruldu ve temel yapılandırma ayarları belirtildikten sonra öğrencilere daha sonraki çalışmalar için hazır bir masaüstü ortamı sağlandı.



Şekil 68: Raspberry Pi OS Kurulumu (Kaynak: Kendi düzenlemem)

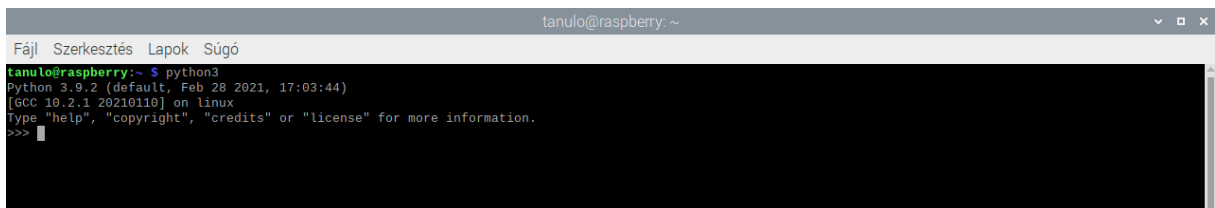
Sistem kurulumu sırasında aşağıdaki olaylar meydana geldi:

- Varsayılan kullanıcı hesabını yapılandırın,
- sistem güncellemesi,
- ve ağ bağlantısını kontrol etmek.

Ağ güvenilirliği son derece önemliydi, çünkü tüm izleme mantığı IP tabanlı iletişime dayanıyordu.

Python çalışma ortamının ve bağımlılıklarının hazırlanması

İzleme uygulaması Python dilinde yazıldığından, Raspberry Pi üzerinde birleşik ve bakımı kolay bir Python ortamı oluşturuldu. Sistem Python 3'ü içermektedir.



Sürüm 1.0, Flask çerçevesinin ve Modbus iletişimi için gerekli kütüphanelerin uyumlu çalışmasını sağladı.

Ortamın hazırlanması sırasında şu hususlara önem verildi:

- Python paket yöneticisini (PIP) kullanarak,
- Gerekli harici kütüphanelerin (web sunucusu, Modbus istemcisi) kurulumu,
- ve sürüm çakışmalarından kaçınmak.

```
tanulo@raspberrypi:~$ pip install flask
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Requirement already satisfied: flask in /usr/lib/python3/dist-packages (1.1.2)
tanulo@raspberrypi:~$ pip install pymodbus
Looking in indexes: https://pypi.org/simple, https://www.piwheels.org/simple
Collecting pymodbus
  Downloading https://www.piwheels.org/simple/pymodbus/pymodbus-3.8.6-py3-none-any.whl (164 kB)
    |#####| 164 kB 938 kB/s
Installing collected packages: pymodbus
  WARNING: The script pymodbus.simulator is installed in '/home/tanulo/.local/bin' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed pymodbus-3.8.6
tanulo@raspberrypi:~$
```

Bu adım, öğrencilerin dikkatini yazılım sistemlerinin tekrarlanabilirliği ve sürdürülebilirliğine çekti.

Ağ yapılandırması ve Modbus TCP bağlantısı

Sistemin çalışması için temel gereksinim, Raspberry Pi ile ESP32 arasında istikrarlı bir Modbus TCP bağlantısıdır. Bunu sağlamak için, Raspberry Pi öngörülebilir bir ağ ortamında sabit bir IP adresiyle çalıştırıldı, böylece izleme uygulaması alt sisteme her zaman ulaşabildi.

Öğrenciler, ağ iletişiminin güvenilirliğinin bir gözetim sistemi için hayati önem taşıdığını anladılar.

Bu bağlamda Modbus bağlantısı düşük seviyeli kontrol için değil, durum sorgulama ve komut iletimi için kullanılmıştır; bu da yazılım PLC döngüsü kavramıyla iyi örtüşmektedir.

Uygulama başlatma ve paralel işlem yönetimi

İzleme programı başlatıldığında, döngüsel durum güncellemelerini gerçekleştirme mantığı ayrı bir iş parçacığında çalışırken, Flask tabanlı web sunucusu kullanıcı isteklerini işler. Bu çözüm, web arayüzünün sunulmasının sistemin izleme işlevlerini engellememesini sağlar.

Raspberry Pi'nin kaynaklarının yeterli olduğu kanıtlandı:

- kesintisiz Modbus iletişimi için,
- döngüsel durum işleme için,
- ve web HMI'ya hizmet etmek için.

Bu işletim modeli, öğrencilere paralel işleme ile gerçek zamanlı olmayan ancak deterministik işleme arasındaki farkı iyi bir şekilde göstermiştir.

Eğitim ve sistem entegrasyonu deneyimi

Raspberry Pi tabanlı çalışma ortamının geliştirilmesi, projenin önceki aşamalarının ayrılmaz bir parçasıydı. Öğrenciler, sensörlerin ve aktüatörlerin fiziksel düzeyde yer aldığı, iletişimin endüstriyel protokoller aracılığıyla sağlandığı ve izleme ve kontrolün yazılım araçları kullanılarak uygulandığı eksiksiz bir sistemi çalışır halde gördüler.

Ayrıca, Debian tabanlı işletim sistemi öğrencilere Linux bilgilerini derinleştirme fırsatı verdi. Endüstriyel ortamda bu cihazların yapılandırması mutlaka grafik arayüz üzerinden yapılmadığından, öğrencilerin mümkün olduğunca komut satırı arayüzü üzerinden çalışmalarını hedefledik.

Ancak, uygulama sırasında öğrenciler ve ben Raspberry Pi'nin donanım sınırlarına ulaştık. Tüm öğrenci ekibi aynı anda web sunucularına bağlandığında, veri işlemede yavaşlama yaşandı. Ayrıca, ek yük, aktif soğutma sisteminin kurulmasını gerektirecekti, çünkü çip PC aşırı ısınma nedeniyle birkaç kez yeniden başlatıldı. Projenin ihtiyaçlarını karşılamış olsa da, endüstriyel düzeyde bir sistemde sunucu cihazının kaynaklarını önceden planlamanın ve beklenen yükü değerlendirmenin öncelikli olması gerektiği konusunda önemli bir ders oldu.

6.9. Python istemcisinin rolü

ESP Modbus izleme sistemi için geliştirilen Python programı, izleme mantığını ve dijital "gölgeyi" uygulamayı amaçlamaktadır.

ESP'ye bağlanma

```
1 # ===== FLASK + MODBUS TCP PLC RÉTEG =====
2 # Ez a program egy "szoftveres PLC ciklust" valósít meg.
3 # Feladata:
4 #   • ESP32 Modbus eszköz lekérdezése
5 #   • LED vezérlési logika kezelése
6 #   • Szenzoradatok továbbítása a webes HMI felé
7
8
9 # ===== MODULOK IMPORTÁLÁSA =====
10 from flask import Flask, render_template, jsonify
11 from pymodbus.client import ModbusTcpClient
12 import threading, time
13
14
15 # ===== WEB SZERVER OBJEKTUM =====
16 # Flask kezeli a HTTP kéréseket (HMI felület)
17 app = Flask(__name__)
18
19
20 # ===== MODBUS KLIENS (ESP32 FELÉ) =====
21 # Ez a sor létrehozza a TCP kapcsolatot az ESP Modbus felügyeleti rendszerével.
22 esp = ModbusTcpClient("192.168.0.200", port=502)
23
24
25 # ===== ÁLLAPOTVÁLTOZÓK LÉTREHOZÁSA =====
26 # Ezek a változók a rendszer állapotát tükrözik a szerver oldalon
27
28 led_state = False      # LED logikai állapot (kimenet)
29 btn_state = False      # Fizikai gomb állapota
30 prev_btn_state = False # Előző ciklus gomb állapota (élelérzékeléshez)
31
32 temperature = 0        # Hőmérséklet cache
33 humidity = 0           # Páratartalom cache
```

Denetim döngüsünün yapısı

```
37 # ===== PLC CIKLUS (IPARI LOGIKA MINTAJÁRA) =====
38 # Ez a függvény a felügyeleti alkalmazás ciklikusan futó központi része.
39 # Bár a neve „felügyeleti ciklus”, funkcionálisan nem valós idejű vezérlési ciklust, hanem egy
    állapotfrissítő, felügyeleti lekérdezési ciklust valósít meg.
40 # Ez egy végtelen ciklus, amely fix periódusban (200 ms) végzi az I/O beolvasást és a vezérlési logikát.
41
42 def plc_cycle():
43     global led_state, btn_state, prev_btn_state, temperature, humidity
44
45     while True:
46
47         # ===== DIGITÁLIS BEMENET OLVASÁSA (FIZIKAI GOMB) =====
48         # Modbus Input Status regiszter 0
49         # Az ESP az aktuális gombállapotot a Discrete Input 0 címre írja.
50         rr_btn = esp.read_discrete_inputs(address=0, count=1)
51
52         if not rr_btn.isError():
53             btn_state = rr_btn.bits[0]
54
55
56         # ===== ÉLÉRZÉKELÉS (EDGE TRIGGER) =====
57         # Ez a PLC logika:
58         #   # ha a gomb éppen most lett lenyomva – LED állapot vált,
59         #   # a parancs visszakerül az ESP-re
60         # Ipari PLC logikában ez "RISING EDGE DETECT".
61         # Csak akkor reagálunk, amikor a gomb 0-1 állapotba vált
62
63         if btn_state and not prev_btn_state:
64             led_state = not led_state # LED állapot váltás
65             esp.write_coil(0, led_state) # Fizikai LED frissítés Modbuson
66
67         prev_btn_state = btn_state
68
69
70         # ===== ANALÓG ADATOK OLVASÁSA =====
71         # Holding Register 0 – hőmérséklet
72         # Holding Register 1 – páratartalom
73         # Az ESP tizedes pontossággal küldi az adatot.
74
75         rr_temp = esp.read_holding_registers(address=0, count=2)
76
77         if not rr_temp.isError():
78             temperature = rr_temp.registers[0] / 10.0
79             humidity = rr_temp.registers[1] / 10.0
80
81
82         # ===== PLC CIKLUS IDŐZÍTÉS =====
83         # 200 ms ciklusidő (ipari PLC mintára)
84         time.sleep(0.2)
```

Web (HMI) ekranı

```
88 # ===== WEB OLDAL =====
89 # A HMI felület betöltése
90
91 @app.route("/")
92 def index():
93     return render_template("index.html")
94
95
96 # ===== ÁLLAPOT API =====
97 # A webes felület innen kapja az aktuális rendszerállapotot
98
99 @app.route("/status")
100 def status():
101     return jsonify({
102         "led": led_state,      # LED állapot
103         "btn": btn_state,     # Gomb állapot
104         "temp": temperature,  # Hőmérséklet
105         "hum": humidity       # Páratartalom
106     })
107
108
109 # ===== VIRTUÁLIS GOMB (WEB HMI) =====
110 # Ez a fizikai gomb funkcióját utánozza a weboldalon keresztül
111
112 @app.route("/toggle_led", methods=["POST"])
113 def toggle_led():
114     global led_state
115
116     led_state = not led_state      # LED állapot váltás
117     esp.write_coil(0, led_state)   # Parancs küldése az ESP-nek
118
119     return "OK"
120
121
122
123 # ===== PROGRAM INDÍTÁS =====
124 if __name__ == "__main__":
125
126     # PLC ciklus külön szálon fut, hogy a web szerver ne blokkolódjon
127     threading.Thread(target=plc_cycle, daemon=True).start()
128
129     # Flask webszerver indítása
130     app.run(host="0.0.0.0", port=5000)
131
```

6.10. Sensör verilerini SQL veritabanına kaydetme

İzleme sisteminin geliştirilmesindeki bir sonraki adım olarak, sensör verilerini web HMI arayüzünde gerçek zamanlı olarak görüntülemenin yanı sıra, yapılandırılmış ve aranabilir bir biçimde kaydetmeyi hedefledik. Bunu başarmak için, mevcut Python uygulamasını tamamlayacak şekilde SQL tabanlı bir veritabanı entegrasyonu oluşturuldu.

Bu gelişme, sistemin artık sadece izleme değil, aynı zamanda veri toplama işlevlerini de yerine getirerek daha sonra analiz edilmesini sağlamasıyla projeye yeni bir boyut kazandırdı.

Gözetim sisteminde veri kaydının rolü

Sensör verilerinin bir veritabanına kaydedilmesi, zaman içindeki değişikliklerin izlenmesine, uzun vadeli çalışma analizine ve verilerin daha sonra işlenmesine olanak tanır. Bu, öğrencilere gerçek sistemlerde yalnızca mevcut durumu göstermenin yeterli olmadığını, verilerin geçmişe dönük olarak saklanmasının temel bir gereklilik olduğunu açıkça göstermiştir.

Geliştirme sürecinde, veri toplamanın sürekli kontrol amacına hizmet etmediği, aksine karar destek ve analiz temeli oluşturduğu anlayışına önem verildi.

Veritabanı seçimi ve mantıksal yapısı

Eğitim ortamına uygun, kullanımı kolay ancak ilişkisel veri yönetiminin prensiplerini açıkça gösteren, SQL tabanlı basit bir veritabanı yapısı oluşturuldu. Veritabanı, ölçülen verileri tek, iyi tanımlanmış bir tabloda kaydeder. Veritabanı için MariaDB (MySQL tabanlı) veritabanı motoru kullanıldı.

Saklanan veriler tipik olarak (**sensör verileri**masa):

- kayıt tanımlayıcısı (**veri kimliği**(int, otomatik artan, birincil anahtar)
- zaman damgası (**ölçüm zamanı**, tarih ve saat, varsayılan geçerli zaman damgası)
- sıcaklık değerleri (**adımlamak**(çift),
- nem verileri (**hum**(çift),

Bu yapı, verilerin kronolojik olarak alınmasına ve basit istatistiksel işlemlerin yapılmasına olanak tanır.

Python uygulamanızı veritabanı yönetimiyle tamamlayın

Mevcut Flask tabanlı uygulamanın genişletilmesi sırasında, veritabanı yönetimi ayrı bir mantıksal birim olarak entegre edildi. Sensör verileri hala Modbus TCP iletişimi üzerinden döngüsel olarak okunuyor, ancak güncellenen değerler de veritabanına giriliyor.

Mevcut uygulamada veriler sürekli olarak veritabanına gönderilse de, öğrencilerin dikkatini karmaşık bir sistemde farklı fonksiyonların farklı zaman önem dereceleriyle çalıştığı gerçeğine çekti. Buna dayanarak, uygulamanın geliştirilmesindeki bir sonraki adım, veri kaydının sıklığını yalnızca birkaç saniyede bir olacak şekilde azaltmak olabilir.

Python programını genişletmek için öncelikle mysql.connector modülünü içe aktarmak gerekiyordu.

```
9 # ===== MODULOK IMPORTÁLÁSA =====
10 from flask import Flask, render_template, jsonify
11 from pymodbus.client import ModbusTcpClient
12 import threading, time
13 import mysql.connector #Az SQL kapcsolatot kezelő modul importálása.
```

Bir sonraki adımda, veritabanı bağlantısını yönetmek ve veri göndermek için gerekli nesnelere oluşturduk.

```
26 # ===== ADATBÁZIS KAPCSOLAT KONFIGURÁLÁSA =====
27 # Az adatbázis kapcsolatot kezelő objektum létrehozása.
28 db = mysql.connector.connect(
29     host="localhost",
30     port="3306",
31     user="root",
32     password="",
33     database="sensordatabase"
34 )
35
36 #Az adstbázist kezelő cursor objektum létrehozása.
37 dbcursor = db.cursor()
```

Son olarak, analog sensör verilerini sorgularken, verileri veritabanı sunucusuna göndererek kodu genişlettik:

```
85 # ===== ANALÓG ADATOK OLVASÁSA =====
86 # Holding Register 0 – hőmérséklet
87 # Holding Register 1 – páratartalom
88 # Az ESP tizedes pontossággal küldi az adatot.
89
90 rr_temp = esp.read_holding_registers(address=0, count=2)
91
92 if not rr_temp.isError():
93     temperature = rr_temp.registers[0] / 10.0
94     humidity = rr_temp.registers[1] / 10.0
95     dbcursor.execute("INSERT INTO sensordata (temp,hum) VALUES (" + temperature + "," + humidity + ")")
96
97
98 # ===== PLC CIKLUS IDŐZÍTÉS =====
99 # 200 ms ciklusidő (ipari PLC mintára)
100 time.sleep(0.2)
```

Eğitimsel öneme ve öğrenci deneyimleri

SQL veritabanı entegrasyonunun getirilmesi, projenin öğrenme değerini önemli ölçüde artırdı. Öğrenciler yalnızca gerçek zamanlı veri yönetimiyle değil, aynı zamanda yapılandırılmış veri depolama ve daha sonra kullanılabilirliğiyle de tanıştılar.

Geliştirme sürecinde aşağıdakiler ortaya çıktı:

- Veri yönetimi ve veri modellemesi konusunda temel bilgi,
- sistem düşüncesi,
- ve veri güvenliğine ve veri tutarlılığına duyarlılık.

Sensör verilerinin bir veritabanına kaydedilmesi, projenin bütününe organik olarak uyum sağladı ve modern BT ve endüstriyel sistemlerin temelini verilerin bilinçli ve yapılandırılmış yönetimi olduğu görüşünü daha da güçlendirdi.

6.11. Web yönetim katmanı – sistemdeki rolü

Python uygulaması HTTP uç noktaları aracılığıyla verilere erişim sağlarken, tarayıcıda çalışan HTML + JavaScript arayüzü ise görüntüleme ve kullanıcı etkileşimini üstlenir.

Bu katman:

- Fiziksel giriş/çıkışı doğrudan kontrol etmez.
- ESP durumunun dijital bir gölgesini gösterir.
- Kullanıcı komutlarını Python uygulamasına iletir.

Veri yolu:



LED durum bloğu

Bu düğme doğrudan ESP'yi değil, Python yönetim sistemini kontrol eder.

```
11 <!-- ===== LED KÁRTYA ===== -->
12 <!-- Ez a blokk a LED állapotát mutatja és innen lehet vezérelni -->
13 <div class="card">
14   <h2>LED állapot</h2>
15
16   <!-- A LED vizuális visszajelzője
17       A színét a JavaScript módosítja:
18       zöld = világít
19       piros = nem világít -->
20   <div id="led" class="led"></div>
21
22   <!-- Virtuális kapcsoló gomb
23       onclick esemény - toggleLED() JS függvény fut le
24       Ez HTTP POST kérést küld a Flask szervernek (/toggle_led)
25       Ez a gomb nem közvetlenül az ESP-t vezérli, hanem a Python felügyeleti rendszert.-->
26   <button onclick="toggleLED()">LED kapcsolás</button>
27 </div>
```

Düğme durumu bloğu

```
30 <!-- ===== FIZIKAI GOMB ÁLLAPOT ===== -->
31 <!-- Ez CSAK kijelzés, nem vezérlő szerv -->
32 <div class="card">
33   <h2>Gomb állapot</h2>
34
35   <!-- A fizikai gomb állapotának szöveges kijelzése.
36       A Flask szerver PLC ciklusa frissíti.
37       "Nyomva" / "Nincs nyomva" szöveg jelenik meg -->
38   <div id="btn">Nincs nyomva</div>
39 </div>
40
```

Fiziksel düğmenin durumunu gösteren metin.

Senzör bloğu

```
42 <!-- ===== KÖRNYEZETI ADATOK ===== -->
43 <!-- DHT11 szenzor által mért adatok -->
44 <div class="card">
45   <h2>Környezeti adatok</h2>
46
47   <!-- Hőmérséklet kijelzés
48       Ezek a DHT szenzor értékei számára fenntartott mezők.
49       A program ide írja be a /status API-ből érkező értéket. -->
50   <div>🌡 Hőmérséklet: <span id="temp">--</span> °C</div>
51
52   <!-- Páratartalom kijelzés -->
53   <div>💧 Páratartalom: <span id="hum">--</span> %</div>
54 </div>
```

Bunlar DHT szenzor değerleri için ayrılmış alanlardır.

JavaScript - canlı güncelleme

```
1 // ===== ÁLLAPOTFRISSÍTŐ FÜGGVÉNY =====
2 // Ez a függvény 200 ms-onként lefut.
3 // Feladata: lekérdezni a Flask szervertől az aktuális rendszerállapotot és frissíteni a weboldalon
  látható adatokat.
4
5 function update(){
6
7 // HTTP GET kérés a szerver /status végpontjára
8 // A szerver JSON formátumban küldi vissza az adatokat:
9 // { led: bool, btn: bool, temp: float, hum: float }
10 fetch("/status")
11
12 // A válasz átalakítása JSON objektummá
13 .then(r => r.json())
14
15 // A kapott adatok feldolgozása
16 .then(data => {
17
18 // ===== LED VIZUÁLIS ÁLLAPOT =====
19 // A LED kör színének beállítása:
20 // zöld ha világít (true), sötétpiros ha nem (false)
21 document.getElementById("led").style.background =
22 data.led ? "limegreen" : "darkred";
23
24 // ===== FIZIKAI GOMB ÁLLAPOT KIÍRÁS =====
25 // A Flask PLC ciklus által olvasott bemenet jelenik meg
26 document.getElementById("btn").innerText =
27 data.btn ? "Nyomva" : "Nincs nyomva";
28
29 // ===== HŐMÉRSÉKLET KIÍRÁS =====
30 // DHT11 szenzor adata az ESP - Modbus - Flask útvonalon
31 document.getElementById("temp").innerText = data.temp;
32
33 // ===== PÁRATARTALOM KIÍRÁS =====
34 document.getElementById("hum").innerText = data.hum;
35
36 });
37
38
39 }
```

JavaScript bu deĝişkenleri alır ve günceller.

LED anahtar düğmesinin çalışması

```
43 // ===== VIRTUÁLIS LED KAPCSOLÓ =====
44 // Ez a függvény akkor fut le, amikor a felhasználó megnyomja a weboldalon a "LED kapcsolás" gombot.
45
46 function toggleLED(){
47
48 // HTTP POST kérés küldése a szervernek
49 // A Flask oldalon ez a /toggle_led útvonalat hívja meg, ami megfordítja a LED állapotát és kiírja
  Modbus-on az ESP-re.
50 fetch("/toggle_led", { method: "POST" });
51
52 // Fontos: itt nincs válaszfeldolgozás, a következő update() ciklus már lekéri az új állapotot.
53 }
```

İşlem:



Güncelleme döngüsü

```
57 // ===== IDŐZİTETT FRISSİTÉS =====
58 // A böngésző 200 ms-onként automatikusan meghívja az update() függvényt.
59 // Ez kvázi PLC ciklus a weboldalon (HMI polling).
60 // Ez megfelel a Python programban lévő ciklus ütemének.
61
62 setInterval(update, 200);
```

Bu, Python'ın döngü hızıyla eşleştiği için sistem senkronize kalır.

Stilin sistemdeki rolü

CSS, HMI web arayüzünün şu özelliklere sahip olmasını sağlar:

- düşük ışık koşullarında bile okunabilir
- Durumlar hakkında net görsel geri bildirim sağlayın.
- ayrı bilgi blokları (HMI panel mantığı)

Bu yapı, SCADA / endüstriyel HMI (İnsan-Makine Arayüzü) sistemlerinin minimalist yaklaşımını takip etmektedir.

Tam CSS – detaylı açıklamalarla birlikte.

```
1  /* ===== OLDAL ALAPSTÍLUS ===== */
2  body {
3      font-family: Arial;
4      background: linear-gradient(135deg,#0f2027,#203a43,#2c5364);
5      color: white;
6      text-align: center;
7      margin: 0;
8  }
9
10 /* ===== KÁRTYA (PANEL) STÍLUS =====
11 Minden információs blokk ezt használja:
12 LED, Gomb, Szenzor adatok*/
13 .card {
14     background: #222;
15     margin: 20px auto;
16     padding: 20px;
17     width: 320px;
18     border-radius: 12px;
19     box-shadow: 0 0 10px #000;
20 }
21
22 /* ===== LED VISSZAJELZŐ KÖR =====
23 Ez a vizuális LED indikátor*/
24 .led {
25     width: 40px;
26     height: 40px;
27     margin: 15px auto;
28     border-radius: 50%;
29     background: darkred; /* Alapértelmezett állapot: LED kikapcsolva. A program dinamikusan felülírja:
30     limegreen = bekapcsolva */
31 }
32
33 /* ===== GOMB STÍLUS =====
34 Virtuális LED kapcsoló*/
35 button {
36     padding: 10px 15px;
37     border-radius: 8px;
38     border: none;
39     background: #00aaff;
40     color: white;
41     cursor: pointer;
42 }
```

Bu, gerçek PLC HMI, SCADA terminali ve endüstriyel kontrol panellerinin görüntüleme prensiplerine karşılık gelir.

Bu, eğitimde neden önemlidir?

Öğrenciler yalnızca verileri görüntülemekle kalmaz, aynı zamanda bir HMI'nin (İnsan-Makine Arayüzü) nasıl çalışabileceğini de öğrenirler:

- işlevsel görsel sistem
- renk, boyut ve düzenlemenin anlam taşıdığı yer

Web katmanı artık tamamlandı:



6.12. Öğrencilerin ekip çalışmasıyla web tabanlı HMI arayüzü geliştirmesi

İzleme sisteminin temel fonksiyonlarının uygulanmasının ardından, projenin bir sonraki aşaması web tabanlı İnsan-Makine Arayüzünün (HMI) daha da geliştirilmesiydi. Bu aşamada öğrenciler artık önceden tanımlanmış çözümleri uygulamak yerine, mevcut HTML ve CSS tabanlı arayüzü kendi fikirlerine dayanarak ve ekip çalışmasıyla geliştirdiler.

Bu görevin amacı, öğrencilerin işlevsel istikrarı korunurken, çalışan ancak basit görünümlü bir arayüzün nasıl daha kullanıcı dostu, daha net ve görsel olarak daha bilgilendirici hale getirilebileceğini deneyimlemelerini sağlamaktır.

Takım çalışması ve görev paylaşımı

Öğrenciler, görevlerin bilinçli bir şekilde bölüştüğü küçük çalışma grupları halinde çalıştılar. Bazı öğrenciler öncelikle yapıyı dönüştürmeye (HTML öğelerini düzenleme, mantıksal gruplandırma) odaklanırken, diğerleri görünümü iyileştirmeye (renkler, düzen, görsel geri bildirim) odaklandı.

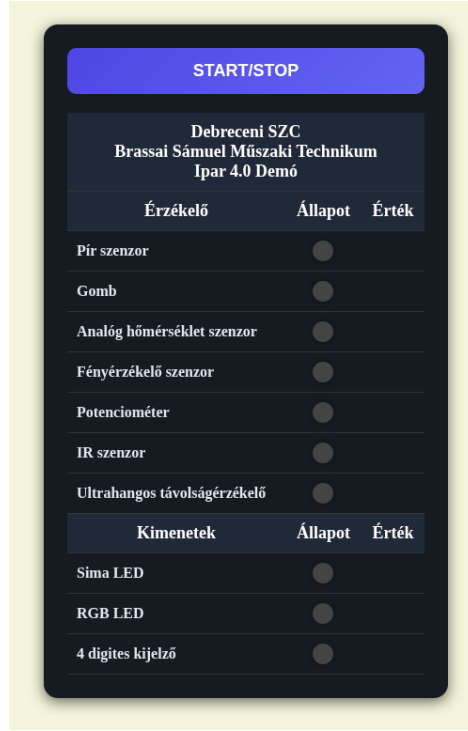
Bu çalışma organizasyon biçimi, öğrencilerin ortak geliştirmenin faydalarını ve zorluklarını deneyimlemelerine, ayrıca birbirlerinin çalışmalarına dayalı koordinasyon ve geliştirme pratiği yapmalarına olanak sağladı.

Arayüzün işlevsel ve görsel olarak iyileştirilmesi

Başlangıç noktası olarak kullanılan web arayüzü, sistemin temel durumlarını ve temel kontrol seçeneklerini zaten içeriyordu. Öğrenciler bu arayüzü daha da geliştirdiler:

- Bilgilerin görüntülenme şekli yeniden düzenlendi.
- Daha sonraki bir geliştirme için hazırlık amacıyla görüntülenen sensör verilerinin aralığını genişlettiler.
- Daha net görsel geri bildirim geliştirildi.
- ve görünüm tarzı, ekiplerinin ihtiyaçlarına göre değiştirildi.

Değişiklikler sırasında, özellikle arayüzün gerçek dünya izleme ortamında kolayca yorumlanabilmesini sağlamak amacıyla, açıklık ve kullanılabilirliğe özel önem verildi.



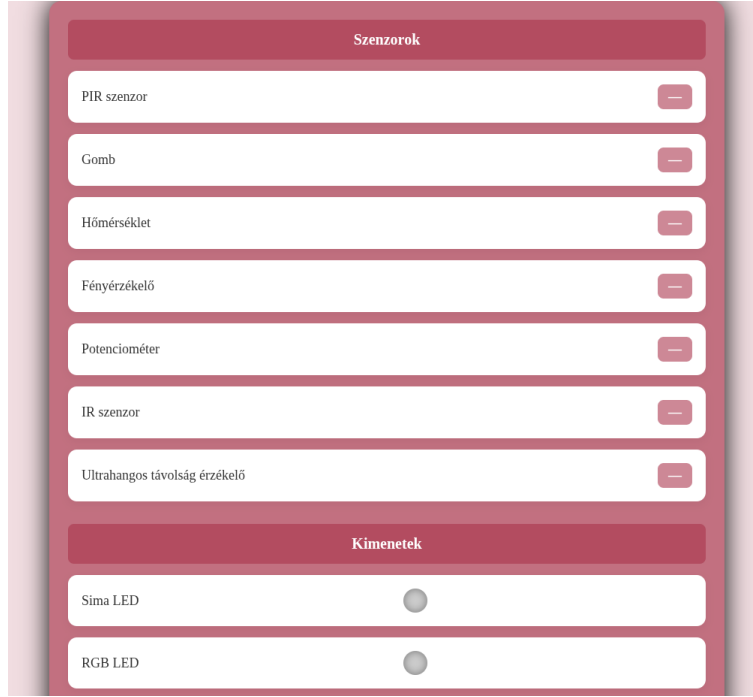
Şekil 69: Öğrencilerin geliştirilmiş çalışmaları 1. (Kaynak: Kendi düzenlemem)

Bireysel ihtiyaçların ortaya çıkışı ve yaratıcı çözümler

Geliştirme sürecinde öğrenciler, arayüzün görünümünü ve işlevselliğini kendi fikirlerine göre şekillendirme fırsatı buldular. Sonuç olarak, örneğin çeşitli çözümler ortaya çıktı:

- Durum göstergeleri için farklı renk kodlaması,
- Sensör verilerini görüntülemek için alternatif düzenler,
- Müdahaleler sırasında daha belirgin geri bildirim.

Bu özgürlük, yaratıcılığın gelişmesi için bir fırsat sağlarken, öğrenciler bireysel fikirlerin sistemin teknik ve işlevsel sınırlamalarıyla dengelenmesi gerektiğini öğrendiler.



Şekil 70: Öğrencilerin geliştirilmiş çalışmaları 2. (Kaynak: Kendi düzenlemem)

Eğitimsel önemi ve gelişim deneyimleri

Web tabanlı HMI arayüzünün daha da geliştirilmesi sırasında öğrenciler, daha önce edindikleri bilgileri karmaşık bir şekilde uyguladılar. Bu görev eş zamanlı olarak şunları geliştirdi:

- dijital ve bilişim teknolojileri yetkinlikleri,
- işbirliği ve iletişim becerileri,
- ve kullanıcı merkezli düşünme.

Süreç, proje hedefleriyle son derece uyumluydu; çünkü öğrenciler sadece teknik çözümler üretmekle kalmayıp, arayüzü genel gözetim mimarisindeki rolünü de göz önünde bulundurarak çalışan bir sistem olarak düşündüler.

6.13. Özet - sistem entegrasyonu ve öğrenme sonuçları

Bu bölümde sunulan faaliyetler, projenin en karmaşık ve pedagojik açıdan en değerli aşamalarından birini oluşturmuştur. Bu aşamada öğrenciler, donanım unsurları, ağ iletişimi, yazılım kontrol mantığı ve web ekranının birleşik bir sistem olarak çalıştığı eksiksiz ve işlevsel bir gözetim sistemi uygulamışlardır.

Raspberry Pi tabanlı çalışma ortamını hazırlamak ve Flask tabanlı uygulamayı dağıtmak, öğrencilere programlama ve sistem işletimi becerilerini gerçek BT ve endüstriyel ortamlarda uygulama fırsatı verdi. Özellikle Modbus TCP iletişimini kullanmak önemli bir deneyimdi, çünkü öğrenciler şunları yapabildiler:

Sektör genelinde kullanılan bir protokolle çalışabildiler ve bu protokolün gözetim mimarisindeki rolünü anladılar.

Web HMI arayüzünün daha da geliştirilmesi sırasında öğrenciler sadece teknik çözümler üretmekle kalmadılar, aynı zamanda kullanıcı merkezli bir yaklaşımı da benimsediler. Takım çalışmasıyla gerçekleştirilen geliştirmeler sırasında bireysel yaratıcılık, bağımsız karar verme ve ortak profesyonel danışmanlık ortaya çıktı. Arayüzlerin özelleştirilmesi, öğrencilerin sistemi kendilerine ait hissetmelerine ve işletiminden sorumlu olmalarına katkıda bulundu.

Pedagojik açıdan bakıldığında, öğrencilerin tüm geliştirme süreci boyunca kararlarının sonuçları hakkında sürekli geri bildirim aldıklarını vurgulamak önemlidir. Sistemin çalışmasını test etmek, hataları belirlemek ve düzeltmek ve çözümleri birlikte değerlendirmek, öz yansıtma ve problem çözme düşünme becerilerini güçlendirdi. Hatalarla başa çıkmak bir başarısızlık olarak değil, öğrenme sürecinin doğal bir parçası olarak görüldü.

6. Bölümde uygulanan faaliyetler, özellikle aşağıdaki temel yetkinliklerin geliştirilmesine katkıda bulunmuştur:

- dijital yetkinlik,
- işbirliği ve iletişim becerileri,
- algoritmik ve sistem düşüncesi,
- ve bağımsız öğrenme ve sorumluluk alanlarında.

Genel olarak, bu bölüm, proje tabanlı, uygulama odaklı bir yaklaşımın teorik bilgiyi gerçek dünya uygulama ortamıyla nasıl birleştirebileceğinin iyi bir örneğidir. Öğrenciler sadece çalışan bir teknik sistem oluşturmakla kalmadılar, aynı zamanda uzun vadede mesleki gelişimleri ve kariyer yönelimleri için temel oluşturacak deneyimler de kazandılar.

7. Özet

Bu belge, akıllı ev/akıllı çiftlik modelinin geliştirilmesine odaklanan, Endüstri 4.0 ve Nesnelerin İnterneti (IoT) tabanlı karmaşık bir eğitim projesini sunmaktadır. Proje felsefesine göre, öğrenme pasif bir alımlama değil, öğrencilerin gerçek sorunlar üzerinden çözümlere ulaştığı aktif ve yaratıcı bir süreçtir. Vurgu, hazır cevaplar üzerinde değil, düşünme süreci, kararlar ve yinelemeler üzerindedir. Öğrenciler sadece teknolojiyi kullanmakla kalmaz, aynı zamanda bir sistemin çalışmasının her zaman uzlaşmaların ve bilinçli seçimlerin sonucu olduğunu deneyimleyerek onu tasarlar, analiz eder ve geliştirirler.

Bu doğrultuda eğitim hedefleri, klasik teknik bilginin ötesine geçmektedir. Proje, aynı anda programlama ve elektronik bilgisi, sistem düşünme ve iş birliği becerilerini geliştirmektedir. Öğrenciler, fiziksel dünyanın olaylarını dijital sistemlerde yorumlamayı, neden-sonuç ilişkilerini tanımayı ve karmaşık problemleri birkaç adımda çözmeyi öğrenirler. Özellikle önemli olan, projenin farklı geçmişlere sahip öğrencileri dahil etme fırsatı sunmasıdır; bu da açıklayıcı düşünmeyi ve ortak problem çözmeyi güçlendirir.

Teknik açıdan bakıldığında, proje sensörlerin, kontrolörlerin ve aktüatörlerin birleşik bir mantıksal sistem olarak birlikte çalıştığı IoT tabanlı bir sistem oluşturmayı amaçlamaktadır. Başlangıç noktası, fabrikada üretilmiş bir Akıllı Çiftlik Kiti'dir; ancak bu sadece bir temel görevi görür: öğrencilerin görevi onu yeniden üretmek değil, sistemi anlamak ve ardından kendi endüstri odaklı mimarilerine doğru geliştirmektir. Bu süreç, mevcut çözümler analiz edildikten sonra yeni, optimize edilmiş sistemlerin oluşturulduğu gerçek dünya mühendislik çalışmalarını iyi bir şekilde modellemektedir.

Uygulama, net bir şekilde yapılandırılmış ancak esnek bir süreçle ve açıkça tanımlanabilir önemli kilometre taşlarıyla gerçekleştirilir. Öğrenme yolculuğu doğrusal değildir, ancak tipik olarak aşağıdaki adımları izler:

- Fabrika sisteminin (Akıllı Çiftlik Kiti) öğrenilmesi ve montajı,
- Bireysel bileşenlerin (sensörler, aktüatörler, kontrol) çalışmasının analizi,
- Sistemin sınırlılıklarını ve eksikliklerini ortaya koyarak,
- kendi gelişim hedeflerini ve işlevlerini tanımlamak,
- Bireysel mekanik elemanların ve gövde parçalarının tasarımı (örneğin 3 boyutlu modelleme),
- Fiziksel unsurların üretimi (örneğin 3 boyutlu baskı),
- Elektronik sistemin yeniden tasarımı ve genişletilmesi,
- programlama (blok tabanlı → metin tabanlı),
- Alt sistemlerin tek bir sisteme entegrasyonu,
- Test etme, hata ayıklama ve yinelemeli geliştirme.

Bu süreç yalnızca teknik ilerlemeyle sonuçlanmakla kalmaz, aynı zamanda öğrencilere karmaşık bir sistemin asla "tamamlanmadığını", sürekli olarak geliştiğini ve iyileştirildiğini de öğretir.



Şekil 71: Akıllı ev (Kaynak: Kendi fotoğrafım, yapay zeka tarafından oluşturulmuş arka plan)

Proje sırasında karşılaşılan zorluklar – temel kavram eksikliği, analog ve dijital sinyaller arasındaki farkların anlaşılması ve sorun giderme – sürecin sonunda değerli öğrenme deneyimlerine dönüştü. Gürültülü ölçüm verileri veya uygun olmayan sınır değerleri nedeniyle kararsız çalışma gibi teknik sorunlar, öğrencilerin sistemin davranışının tasarım kararlarının doğrudan bir sonucu olduğunu anlamalarına yardımcı oldu.

Projenin sonunda öğrenciler sistemli düşünme, girdi işleme-çıktı mantığını yorumlama ve fiziksel ile dijital dünyalar arasındaki ilişkiyi anlama becerisi kazandılar. Oluşturulan çözümler zaten bilinçli olarak inşa edilmiş, işlevsel sistemlerdi.

En önemli ders, projenin sadece teknik bilgi sağlamakla kalmayıp, modern endüstriyel ve teknolojik ortamda doğrudan uygulanabilecek bir yaklaşım geliştirmesidir.

Avrupa Birliği tarafından finanse edilmiştir. Burada yer alan bilgiler ve açıklamalar, yazar(lar)ın görüşlerini temsil eder ve Avrupa Birliği veya Tempus Kamu Vakfı'nın resmi görüşünü yansıtmayabilir. Avrupa Birliği veya fon sağlayan kurum bunlardan sorumlu tutulamaz.