



Módulo de Ensino

**Automatização em
miniatura**

Todos os materiais e detalhes de contato podem ser encontrados nos sites do projeto, bem como no perfil do projeto Erasmus+.

<https://sites.google.com/campus.ul.pt/hands-on-remote-language/home>
<https://erasmus-plus.ec.europa.eu/projects/search/details/2020-1-DE02-KA226-VET-008295>

Equipa de desenvolvimento

- Marion Pellowski and Lorenz Kampschulte, Deutsches Museum, Munich, Germany
- Pedro Reis, Mónica Baptista, Luís Alexandre da Fonseca Tinoca, Lisbon University, Institute of Education, Lisbon, Portugal
- Wojciech Karcz, Adam Zahler, Anna Strzeszewska-Potyrała, Karolina Klimaszewska, Copernicus Science Center, Warsaw, Poland
- Miriam Voss, Mike Kramler, Marion Pellowski, Technical University Munich, Munich, Germany

Declaração

O apoio da Comissão Europeia à produção desta publicação não constitui um endosso do conteúdo, o qual reflete apenas as opiniões dos autores, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito das informações nela contidas.



Co-funded by the
Erasmus+ Programme
of the European Union

Imprint

Autores: Marion Pellowski, Deutsches Museum and Technical University of Munich, Mike Kramler, Miriam Voss, Technical University of Munich, Munich, Germany

A versão portuguesa foi traduzida e adaptada por Pedro Reis, Mónica Baptista, Luís Alexandre da Fonseca Tinoca, Universidade de Lisboa, Instituto de Educação, Lisboa, Portugal

Deutsches Museum, Munich, Germany
Layout & Design: Michał Romański
Print: Fevereiro 2023



Este trabalho está licenciado sob uma Licença Internacional Creative Commons Attribution 4.0. Elementos marcados como citações podem estar sujeitos a outras licenças.

Index

Agradecimentos.....	3
1 Introdução	4
Breve descrição do módulo.....	4
1.1 Objetivos gerais	5
1.2 Racional didático.....	6
1.3 Comunidade e colaboração digital	7
1.4 Breve resumo.....	9
2 Integração Curricular	12
3 Overview do projeto e sequência do módulo "Automatização em miniatura"	13
4 Sequência do projeto para o módulo de ensino "Automatização em miniatura" em detalhes	0
4.1 Unidade 1.a introdução ao tópico.....	0
4.2 Unidade 1.b Introdução aos circuitos elétricos e à programação com o Arduino	2
4.3 Unidade 2.a Online-Simulation	5
4.4 Unidade 2.b Servomotor	6
4.5 Unidade 3.a plataforma giratória.....	8
4.6 Unidade 3.b Barreira de Luz.....	10
4.7 Unidade 4. Extensão para o processo de automatização.....	12
4.8 Unidade 5. Colaboração entre alunos	14
4.9 Unidade 6. Colaboração digital.....	15
4.10 Unidade 7. Automatização no seu contexto social.....	16
4.11 Unit 8. Outras aplicações das plataformas giratórias	19
4.12 Notas gerais sobre videoconferência e trabalho prático em situações remotas.....	21
5 Possibilidades de uso do módulo – diferentes percursos	23
6 Apêndice: Perspectivas da página de instrução multimédia interactiva	25

Agradecimentos

Agradecemos cordialmente aos professores e alunos da Escola Profissional Municipal de Tecnologia da Informação e Integração de Sistemas de Munique pela sua cooperação ao longo do projeto e pelo seu constante apoio no desenvolvimento deste Módulo de Ensino. Agradecemos também a todos os professores que participaram na nossa avaliação de necessidades pelos seus valiosos contributos e comentários para este projeto.

Breve descrição do módulo

Módulo de ensino

**Automação
em miniatura**

Os estudantes programam uma pequena fábrica de produção e põem-na em marcha. O sistema de enchimento automático é feito de materiais simples e pode ser levado para casa pelos estudantes. Os estudantes podem trabalhar em conjunto como uma equipa, mesmo em situações remotas.

Módulo principal de ensino			Unidades adicionais / opcionais	
Unidade 1a Breve introdução ao tema Objetivo: Estabelecer referências à vida quotidiana e aos ambientes profissionais Tarefa: Exibição de vídeos de várias aplicações de gira-discos em automatização	Unidade 2a	Unidade 3a	Unidade 4 Extensões ao processo de automatização Objetivo: Transferência de conhecimentos previamente adquiridos para novos elementos, âmbito dos projetos dos próprios estudantes Tarefa: Livre escolha para a instalação de outras peças funcionais como <ul style="list-style-type: none"> • Indicador de estado do LED • Deslizamento de entrada • Ejetor 	Unidade 5 / 6 Colaboração / Colaboração Digital Objetivo: Promover a colaboração entre estudantes, incluindo o trabalho de equipa e um sentido de comunidade, mesmo no ensino à distância Tarefa: Acoplamento de mesas giratórias de diferentes equipas; Utilização de transmissão de luz para induzir efeitos em situações remotas
Unidade 1b Introdução ao Arduino Objetivo: Introdução à programação com Arduino IDE, utilizando um primeiro desenho de circuito Tarefa: Programação de LEDs de cores diferentes, aprendizagem de comandos básicos e erros comuns de programação	Unidade 2b Servomotor Objetivo: Aprender sobre as características e funções dos servomotores Tarefa: Utilização de servomotores para mover partes funcionais da instalação de produção	Unidade 3b Barreira de luz Objetivo: Utilização de sensores para produção automatizada, discussão das medições com sensores Tarefa: Utilização da resistência dependente da luz (LDR) como sensor numa barreira de luz reflexa para controlar o processo de produção		Unidade 8 Outras utilizações do prato giratório Objetivo: Encorajar um maior envolvimento com a mesa giratória Tarefa: Utilização da mesa rotativa como máquina de classificação detetando diferenças de luminosidade ou cor; Utilização da plataforma rotativa como leitor de luz induzida pelo som, medindo a luminosidade



Os professores podem alterar a sequência das unidades e começar com 3a/3b e seguir com 2a/2b.



O módulo "Automação em Miniatura" permite aos alunos programar uma mini-fábrica como um sistema mecânico real, considerando e resolvendo problemas que não ocorreriam num ambiente puramente virtual. Para criar um processo produtivo funcional, a interação entre a programação e as partes mecânicas funcionais da mini-fábrica é fundamental. Quanto ao ensino à distância, a ideia geral deste módulo é substituir um produto de sala de aula de alta qualidade e caro por uma versão mais simples que os professores possam fornecer aos seus alunos para levarem para casa.

Usando o exemplo da mini-fábrica, os alunos aprendem sobre a estrutura e o funcionamento de um sistema técnico complexo. Ao controlar servomotores como atuadores e utilizar um sensor de luz para controlar o processo de enchimento, programam e controlam a pequena fábrica de produção. Programam baseados em texto com um microcontrolador, o Arduino, que também podem usar para gravar valores de medição. Os alunos testam seus programas no exemplo real da fábrica de produção e aplicam a resolução de problemas durante os testes. Ao fazê-lo, também devem ter em conta as influências ambientais (por exemplo, a influência da luz ambiente nos valores de medição do sensor de luz).

Os professores podem adaptar o módulo a diferentes níveis de habilidades de programação. Essencialmente, os alunos precisam apenas de algumas linhas de código para a programação básica dos processos de automação. Devido ao seu conceito, o módulo também é adequado para uma introdução à programação orientada a objetos (definição de objetos, estrutura de objetos, uso de classes e objetos baseados na realidade). Os professores também podem usá-lo para transferir conhecimentos existentes de programação orientada a objetos para um exemplo de aplicação. No entanto, é possível programar todo o processo de automação mesmo sem compreender a programação orientada a objetos.

O tema da automação e suas consequências sociais afetam muitos alunos de formação profissional diretamente, pois irão trabalhar com esses sistemas produtivos de alguma forma. Na última unidade, os alunos são incentivados a discutir e refletir sobre questões sociais relacionadas à automação para considerar suas próprias carreiras futuras num contexto mais amplo.

Tendo em vista a alta relevância social do tema, os alunos lidam com várias vantagens e desvantagens da automação para nossa sociedade, incluindo aspectos ecológicos e relacionados ao trabalho.

1.2 Racional didático



O projeto utiliza ferramentas online fáceis de usar, como um ambiente de simulação online e um quadro branco online para facilitar o ensino a distância. Os professores podem introduzir essas ferramentas, bem como os experimentos práticos, tanto em ambientes presenciais quanto de ensino à distância. Isso cria uma base comum para o ensino, independentemente da situação de ensino.

A videoconferência para uma introdução ao tópico e o trabalho com o minissistema de produção são fundamentais no ensino a distância, especialmente no início, e continua importante para manter contato com os alunos e apoiá-los em seu trabalho. Diferentes formas são possíveis: videoconferência com toda a turma, videoconferência com subgrupos em salas virtuais e estar disponível como suporte em horários fixos para solicitações dos alunos. O módulo também usa videoconferência para promover a colaboração entre os alunos.

Após uma introdução aos conceitos básicos, os alunos podem continuar trabalhando de forma independente usando uma página de instruções multimídia. Eles podem decidir por si próprios como proceder. O professor tem o papel de ser o contato para qualquer problema que possa surgir. O uso da página de instruções multimídia permite que os alunos trabalhem independentemente com os materiais e facilita o trabalho tanto em situações presenciais quanto remotas. A combinação de ferramentas online, diferentes formas de videoconferência e a página de instruções multimídia permite o uso do módulo tanto para o ensino à distância quanto para a instrução híbrida. Os

professores atuam como facilitadores da aprendizagem nesse processo, seja remotamente ou no local de ensino.

A automação é uma parte importante das sociedades modernas e determina uma grande parte de nossas vidas. Portanto, é fácil fazer referência a processos de automação reais e à vida cotidiana dos alunos. Com o processo de automação, referências a vários campos ocupacionais são possíveis. Em comparação com os sistemas de produção fabricados industrialmente para as escolas, a mini planta de produção consiste em materiais simples e as peças são relativamente fáceis de substituir. Isso é crucial para disponibilizar os kits de materiais aos alunos no ensino à distância. Além disso, pareceu importante para nós tornar o conjunto tão atraente que os alunos possam ter prazer em continuar trabalhando com ele em seu tempo livre.

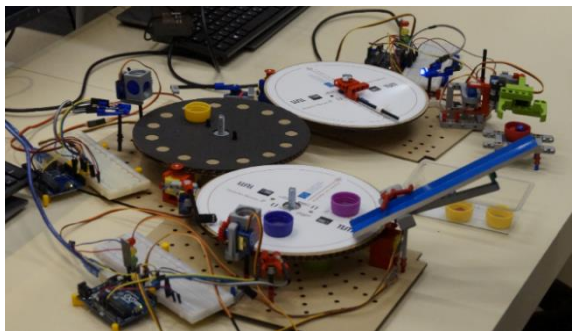
Para uso em um contexto europeu muito heterogêneo, o grau de complexidade pode ser adaptado ao grupo alvo respectivo por meio do uso flexível e redesenho da mini planta de produção para outras aplicações.

Para mais detalhes sobre o desenvolvimento e implementação, consulte o Guia. De acordo com a categorização feita no Guia (capítulo 2), este módulo combina as abordagens de experimentos locais e reais, que requerem o envio ou coleta de materiais físicos, e experimentos remotos e virtuais, que podem fornecer uma base melhor para a compreensão conceitual.

1.3 Comunidade e colaboração digital

O trabalho em equipa com experiências práticas tem um grande potencial para promover o sentimento de comunidade entre os alunos. Algumas tarefas neste módulo são projetadas para promover a aprendizagem colaborativa, especialmente quando os alunos são solicitados a fazer conexões entre suas diferentes plantas de produção e fazê-las trabalhar juntas.





Os alunos podem ligar as diferentes plataformas giratórias das equipas, utilizando o ejetor para transferir copos de uma plataforma para outra.

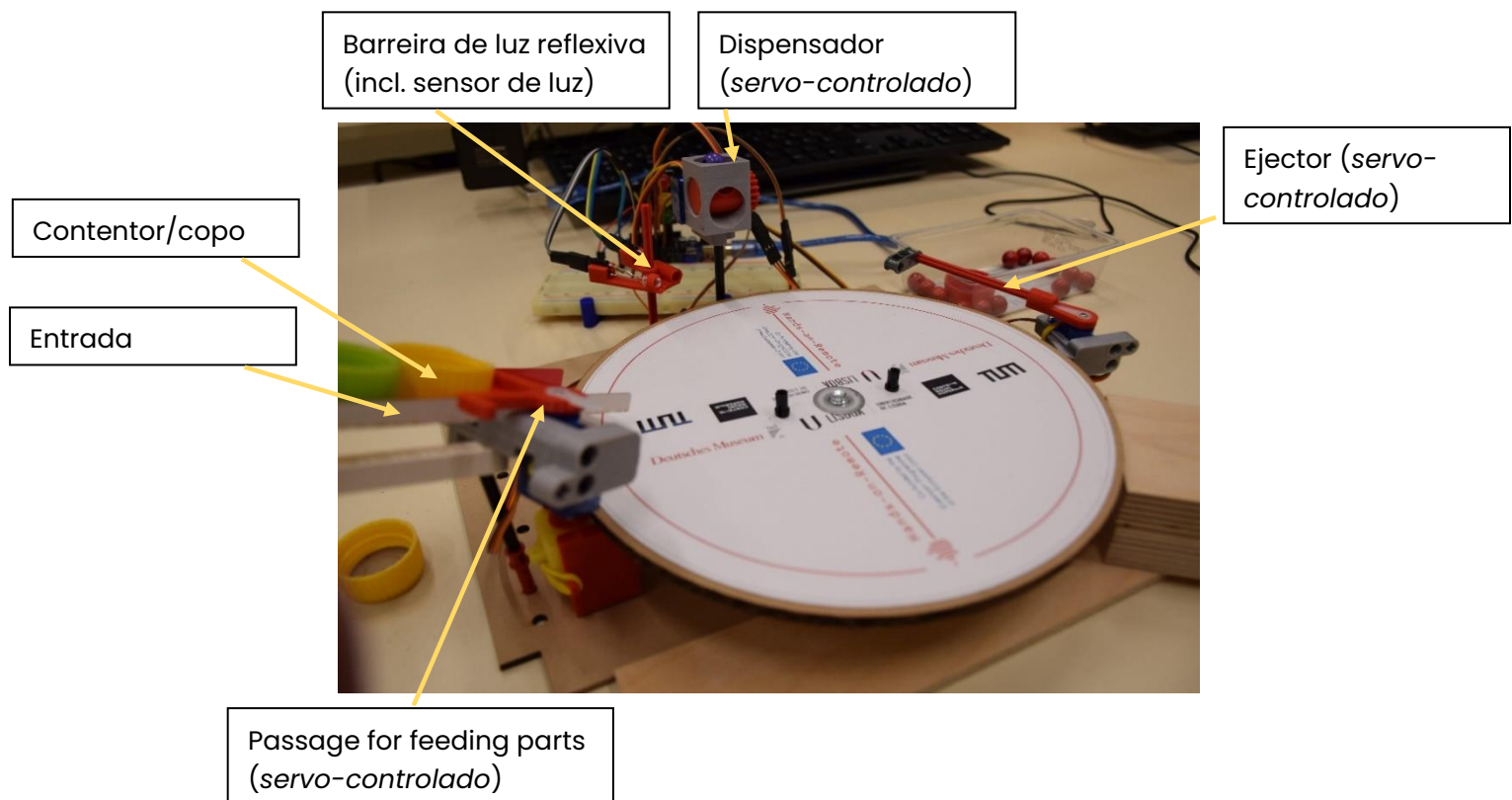
A acoplagem das plataformas giratórias é possível, mesmo que nenhuma das equipas tenha construído completamente o processo de automação. Nesse caso, os alunos usam apenas uma parte funcional diferente em cada plataforma giratória e estabelecem um mecanismo de transferência comum entre as suas plataformas giratórias.

O ambiente de simulação online permite que os alunos partilhem o seu código e trabalhem juntos nas tarefas de programação. Desta forma, uma cooperação próxima continua a ser possível mesmo no ensino a distância.

Para aproveitar os efeitos motivacionais da colaboração e combater o isolamento, este módulo envolve um tipo especial de colaboração: os alunos podem interagir uns com os outros através de transmissão de luz em videoconferências, movendo partes da mesa giratória de outra equipa num local diferente. No laptop do primeiro aluno, um LED brilha na webcam para a videoconferência. Na frente do segundo laptop do aluno, o sensor de luz é montado de forma a poder detetar o sinal. Em resposta ao sinal de luz, um servo é acionado no local do segundo aluno e pode mover uma parte da sua mesa giratória. O conceito é usar esta abordagem para criar uma sensação de inclusão apesar da distância física – e simplesmente para desencadear alegria. É uma abordagem muito simples, mas tem a grande vantagem de ser fácil de entender, de baixo custo e muito rapidamente implementável. A abordagem baseia-se no conhecimento prévio dos professores, o que a torna muito adaptável mesmo para professores com pouca experiência em meios técnicos no ensino remoto.

1.4 Breve resumo

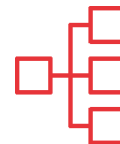
O módulo de ensino "Automação em Miniatura" permite que os alunos programem uma pequena fábrica de produção e a ponham em funcionamento. Um módulo de semáforo como indicador de estado, uma barreira de luz e uma plataforma giratória servem como elementos práticos, que juntamente com várias partes funcionais em movimento formam uma mini fábrica de produção.



Os alunos podem controlar a plataforma giratória como um sistema de enchimento – no qual a barreira de luz deteta um recipiente na plataforma giratória, o indicador de estado indica uma mudança de estado mudando de vermelho para verde e um dispensador é usado para preencher o recipiente correspondente.

O processo de produção depende do sensor de tal forma que uma barreira de luz reflexa deteta um copo a passar e, em seguida, aciona o processo de enchimento.

Estrutura das unidades



Este módulo é constituído por uma sequência de unidades de ensino que têm uma determinada sequência, mas é possível ter percursos diferenciados que permitem uma utilização mais avançada ou mais principiante. Para motivar os alunos para o desenvolvimento deste módulo o professor pode fazer a ligação com uma aplicação industrial, i.e., a plataforma giratória representa uma fábrica fictícia. Para a sua minifábrica, os alunos em grupo podem escolher o que “produzem” desenvolvendo o seu próprio cenário fictício. Como introdução à programação com o Arduino, os alunos primeiro ligam e desligam os LEDs do display. Para isso, trabalham com um módulo de semáforo real e virtual. Um ambiente de simulação baseado na Internet facilita a colaboração e a troca da programação que desenvolveram, de modo a que uma cooperação estreita permanece possível mesmo no ensino à distância. Em seguida, a plataforma giratória é colocada em movimento com um motor redutor. Outros elementos funcionais do sistema são movidos para frente e para trás entre duas posições por motores. Por exemplo, os alunos programam o motor do dispensador, giram duas engrenagens para abrir uma passagem e colocam o produto no recipiente. Os alunos também podem programar outros elementos com motores, como a passagem para alimentação de peças e o braço giratório para o ejetor.

O trabalho em grupo tem um grande potencial para promover um sentido de comunidade entre os alunos. Os alunos podem trabalhar em conjunto para acoplar as plataformas giratórias de diferentes grupos e estabelecer um funcionamento comum de diferentes plataformas giratórias. A colaboração também é possível em uma situação remota: por videoconferência, os alunos podem usar a transmissão de luz para acionar movimentos de motores em locais remotos.

Programação

O nível de dificuldade também pode ser definido no nível de programação: se o professor pretende usar o módulo para um nível inicial, os alunos inserem apenas alguns comandos de programação numa programação amplamente predefinida. Se o professor quiser usar o módulo para programação avançada, os alunos terão tarefas adicionais.

Materiais para o módulo de ensino

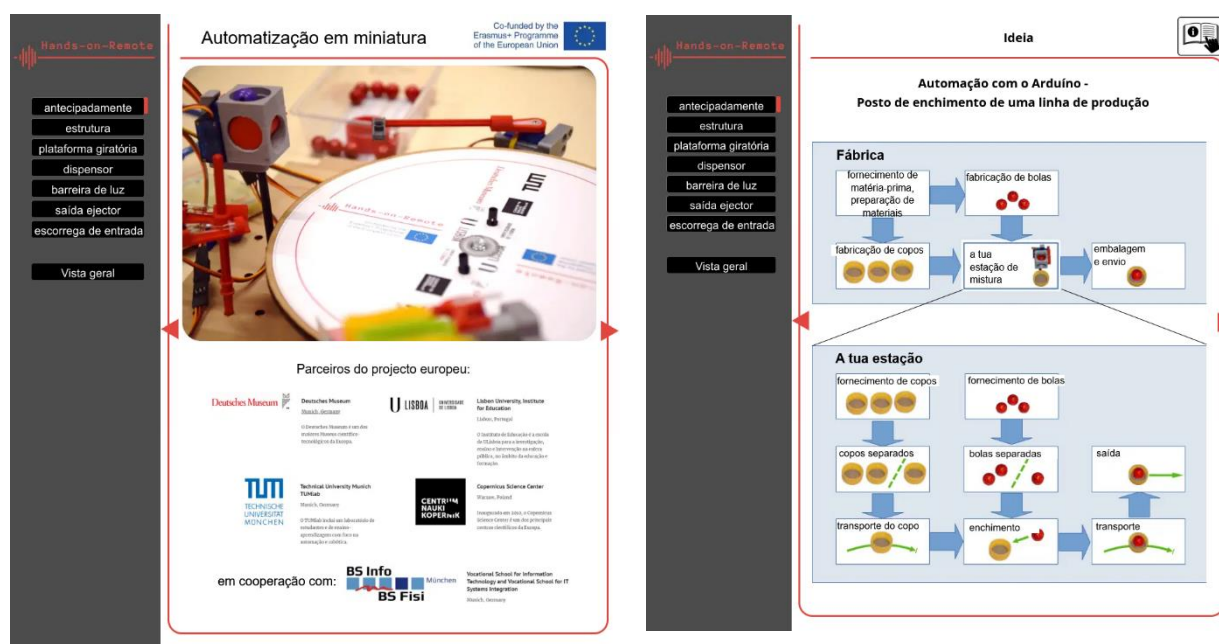
Uma parte do módulo de ensino "Automação em Miniatura" são vários ficheiros e materiais que podem ser descarregados das seguintes páginas:

Página de download do projecto:

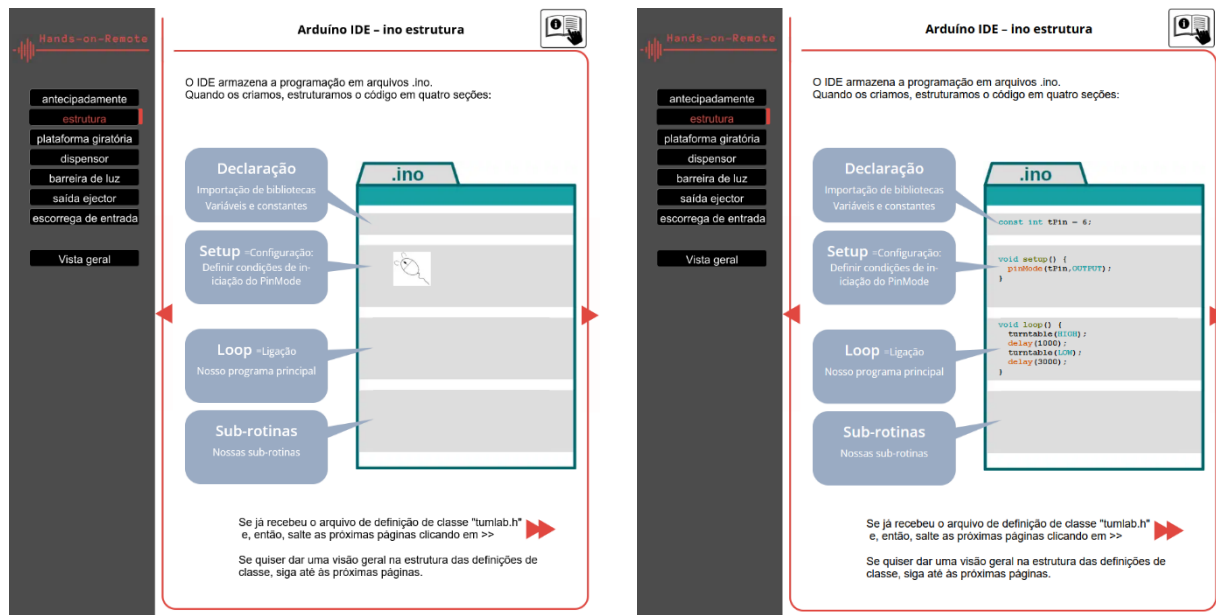
<https://sites.google.com/campus.ul.pt/hands-on-remote-de/downloads>

Página do projecto do Museu Alemão: <https://www.deutsches-museum.de/forschung/forschungsinstitut/projekte/detailseite/erasmus-hands-on-remote>

- Ficheiro de apresentação para o professor (*pptx*), bem como um vídeo sobre o circuito eléctrico (*mp4*)
- Página de instruções multimédia interactiva para os alunos, instruções passo a passo para trabalhar autonomamente com a unidade de mini-produção, incluindo vídeos e animações, *tutorial interactivo para estudantes (html)*



- Fichas de trabalho para a unidade "Automação no contexto social" e ligação ao quadro branco em linha, *Fichas de trabalho Automação no contexto social (pdf)*.
- Ficheiros de impressão 3D e de cortador a laser para o conjunto de materiais, Conjunto de materiais (*lista, ficheiros stl e de corte a laser*).
- Um ficheiro com erros e dificuldades comuns, *Troubeshooting_Automation_in_Miniature (pdf)*



Os passos individuais para a construção e programação de todo o sistema ultrapassam o âmbito desta descrição do módulo de ensino e podem ser encontrados nos ficheiros de acompanhamento acima mencionados. Para a mini-unidade de produção, a estrutura e a programação são mostradas em particular na página de instruções multimédia. As capturas de ecrã em anexo dão uma primeira ideia da estrutura e da concepção da página de instruções (ver capítulo 6, Apêndice: Perspectivas da página de instrução multimédia interactiva). Isto dá uma ideia melhor de como trabalhar remotamente com o Arduino e o sistema.

2 Integração Curricular

O módulo "Automação em Miniatura" está relacionado com o currículo de várias profissões de formação e com as aulas da escola secundária técnica. Algumas profissões para as quais diferentes partes do módulo podem ter um papel importante são, por exemplo, ciência da computação, técnico eletrónico, técnico em automação, eletricista industrial e desenvolvedor de software.

A unidade de aprendizagem sobre os aspetos sociais da automação tem ligações com a ética aplicada (ética da tecnologia). Palavras-chave para os tópicos incluem sistemas e processos, sistemas técnicos complexos, tecnologia de controlo e programação orientada a objetos. Geralmente, as conexões com o currículo existem para as séries 10-13.

3 Overview do projeto e sequência do módulo "Automatização em miniatura"

Unidade 1.a Introdução ao uso da plataforma giratória na automati-zação e o objetivo de transformar em miniatura um pro-cesso de automatização. (15 min)		Unidade 1.b I Introdução ao trabalho com o Arduíno usando um primeiro projeto de circuito e exemplo de programação com diferentes cores LEDs. (30-45 min)		
Unidade 2.a Introdução a um ambiente de simulação online para programação do Arduíno, permitindo trabalhar em programação em situações de ensino a distância. Projeto de um circuito simulado para um motor e programação do motor. (15 min)		Unidade 2.b Introdução aos motores, incluindo circuitos e programação de um motor real (30 min).		
Unidade 3.a Configuração da plataforma giratória, circuitos e programação do motor. (20 min)		Unidade 3.b Funcionamento da barreira luminosa, incluindo circuitos, programação e apresentação de valores no plotter. (25 min)		
Unidade 4. Os alunos trabalham de forma independente e podem escolher quais componentes para a plataforma giratória (vd. tabela abaixo). (Pelo menos 45 min)				
Versão curta	Variação	Variação	Variação	Versão longa
Plataforma giratória Barreira de luz Distribuidor	Plataforma giratória Barreira de luz Distribuidor LED para o dispen-sador	Plataforma giratória Barreira de luz Distribuidor Ejector	Plataforma giratória Barreira de luz Distribuidor Ejector Entrada	Plataforma giratória Barreira de luz Distribuidor Ejector Entrada LED para o dispensador
Unidade 5. Colaboração entre alunos: Acoplamento das plataformas giratórias de diferentes grupos Colaboração digital com transmissão por videoconferência.				
Unidade 6. Colaboração digital: Usando transmissão de luz para induzir efeitos físicos em situações remotas (duração flexível)				
Unidade 7. Automatização em contexto social (pelo menos 45 min)				
Unidade 8. Possíveis outros usos da plataforma giratória (vd. Tabela abaixo) (pelo menos 45 min)				
Máquina de classificação		Reprodutor de som induzido por luz		
Plataforma giratória Barreira de luz LED indicador de status Brilho/cor Uso do ejtor para classifica-ção		Plataforma giratória com sobreposições de papel num padrão preto e branco Sensor de luz Altifalante (não incluído no kit de material)		

4 Sequência do projeto para o módulo de ensino "Automatização em miniatura" em detalhes



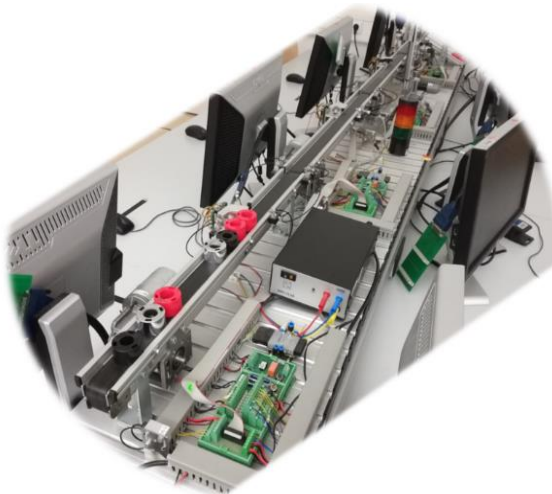
4.1 Unidade 1.a introdução ao tópico

Tópico: Diferentes exemplos reais de processos de automatização

Objetivo: Breve introdução dos alunos ao assunto, centrando-se na ideia ligada com o processo de automatização em miniatura

Duração: aprox. 15 min

Background/Introdução à ideia:



Muitos alunos dos cursos profissionais entrarão em contato com processos de automatização nas suas vidas profissionais – seja porque os próprios poderão programar linhas de produção no futuro ou porque trabalharão com tais sistemas – por exemplo, como

técnicos de alimentos. No geral, a automatização oferece muitas referências relaciona-as com o mundo real.

A ideia de um processo de mini-automatização, remonta a uma linha de produção maior no TUMlab. Neste laboratório, o "processo de produção" correspondente é programado e automatizado pelos alunos. Os professores têm manifestado reiteradamente o desejo de que se desenvolva um protótipo em formato menor que represente uma linha de produção que os alunos possam programar na escola – ou também em casa. Essa foi a origem da ideia de usar o Arduino para possibilitar um processo de automatização em pequena escala.

Forma de trabalho: online via videoconferência ou em sala de aula

Materiais necessários para o professor: apresentação em PowerPoint, incluindo links para vídeos do Youtube

Materiais necessários para os alunos: Nenhum

Conteúdo:

Atividades do professor:

Os professores fornecem informações sobre exemplos reais de plantas industriais automatizadas a partir de vários vídeos do Youtube. Os vídeos são sobre plataformas giratórias que desempenham um papel importante nos processos de automação. Um desses vídeos mostra alguém a colocar a cobertura numa pizza, usando uma plataforma giratória e um processo automatizado. Essa ideia pode relacionar-se com a própria vida quotidiana dos alunos.

Na introdução do projeto é importante transmitir aos alunos a ideia de que vão usar uma plataforma giratória, em vez de uma linha reta para o processo de automação miniaturizado. O professor pode usar as plataformas giratórias de várias formas:

- seja como disco transportador de um sistema de automação com várias etapas de trabalho
- ou como um único gira-discos, por ex., como uma única estação de processamento de uma linha de produção completa, como uma estação de enchimento.

Atividades dos alunos:

Os alunos visualizam vídeos.

4.2 Unidade 1.b Introdução aos circuitos elétricos e à programação com o Arduino

Tópicos: Projeto de circuitos, comandos básicos de programação (Arduino IDE), erros comuns de programação

Objetivo: Os alunos têm contacto com o microcontrolador pela primeira vez. Aprendem métodos básicos de trabalho para projeto de circuitos e programação com um microcontrolador. Para iniciar, os alunos trabalham com LEDs ou o módulo de semáforo, que posteriormente pode servir como *display* de status para processos de automatização.

Background /Introdução à ideia:

Os microcontroladores são incorporados em muitos dispositivos tecnológicos quotidianos, embora muitas vezes sejam invisíveis. Um microcontrolador requer circuitos e programação para as suas funções, dando aos professores a oportunidade de apresentar ambos os tópicos. Para os alunos, é motivador que possam produzir um resultado final do seu trabalho. Se os alunos não atingirem o resultado esperado, eles ou seus professores podem identificar e corrigir rapidamente os erros.

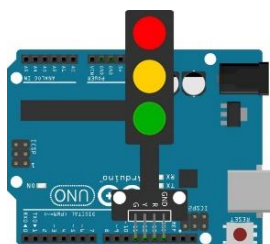
Duração: aprox.30 min,
com uso do vídeo em circuitos elétricos e experimentando o DC Virtual Lab, aprox. 45 minutos

Materiais necessários para o professor: apresentação PowerPoint



Materiais necessários para os alunos:

Arduino, LEDs ou módulo de semáforo, placa, Arduino IDE, se os professores quiserem usar:
Vídeo sobre circuitos, arquivos portáteis numa pendrive



Importante notar: O Arduino está ligado ao computador através de um hub USB. Cada vez que um circuito é construído, o Arduino é desligado do concentrador USB. Este procedimento serve para proteger o computador no caso de um possível curto-circuito.

Forma de trabalho: online via videoconferência ou num laboratório de informática da escola

Conteúdo:

Atividades do professor:

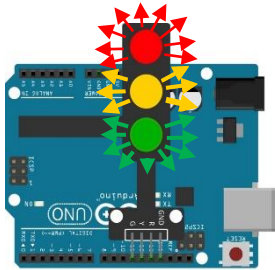
- Os professores familiarizam os alunos com o Arduino e os passos básicos dos projetos Arduino.
- Os professores apresentam o ambiente de programação baseado em linguagem do Arduino IDE.
- Dependendo do nível de aprendizagem dos alunos:
 - Os professores relembrar os alunos sobre os circuitos elétricos simples que permitem fazer um LED acender
 - Os professores orientam os alunos na construção do circuito no Arduino com um LED ou o módulo de semáforo.
 - Os professores perguntam aos alunos sobre seu progresso e pedem para ver os LEDs a piscar.
 - Os professores apoiam os alunos com problemas técnicos, usando uma lista de erros e dificuldades comuns.
 - No ensino a distância, os professores permitem que os alunos compartilhem o ecrã na videoconferência para ajudar em problemas de programação.
 - Os professores interligam a programação dos semáforos à automatização, ensinando os alunos sobre o uso de LEDs/semáforos como indicadores de *status* em sistemas de automatização.



Atividades dos alunos:

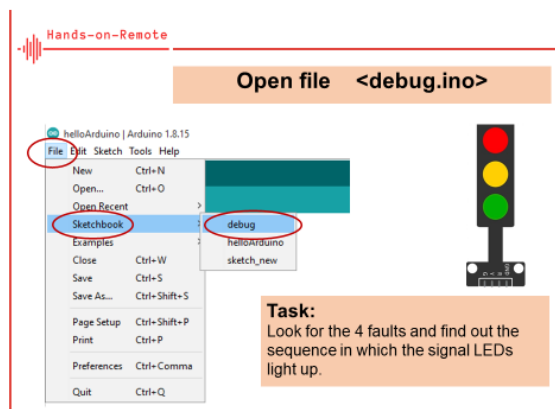
- Os alunos assistem a um vídeo sobre circuitos e constroem o seu próprio circuito online com o Kit de Construção de Circuitos: DC Virtual Lab (https://phet.colorado.edu/sims/html/circuit-construction-kit-dc-virtual-lab/latest/circuit-construction-kit-dc-virtual-lab_en.html)
- Os alunos constroem o seu primeiro circuito Arduino com um LED ou o módulo de semáforo
- Os alunos ligam o Arduino e carregam o primeiro código para deixar um LED piscar
- A primeira tarefa de programação para os alunos é alterar o tempo de piscar de um LED

- Os alunos superam problemas técnicos, usando uma lista de erros e dificuldades comuns
- Se necessário, os alunos discutem sua programação com seus professores



- A tarefa dos alunos é deixar os 3 LEDs a piscar ao mesmo tempo.

Os professores podem mudar para a programação orientada a objetos neste ponto: 3 LEDs = vários objetos do mesmo tipo. Na programação orientada a objetos, o professor pode combiná-los em turma. Os alunos que já possuem um conhecimento aprofundado de programação orientada a objetos podem definir seus próprios códigos. Todos os outros podem usar códigos predefinidos. O uso de definições de códigos predefinidas simplifica a programação para o processo posterior.



- Os alunos aprofundam os seus conhecimentos da linguagem de programação, procurando por erros típicos de semântica/sintaxe num arquivo sobre o melhoramento do pré-projeto.

4.3 Unidade 2.a Online-Simulation

Os professores podem continuar esta sequência ou passar para a plataforma giratória (3.1) e a barreira de luz (3.2) primeiro.

Tópicos: Uso de um ambiente de simulação online para detecção de erros comuns de programação e compartilhamento de código, primeira programação de um motor

Objetivos: Os alunos conhecem o ambiente de simulação online “Wokwi” e aprendem a utilizá-lo. Isso pode tornar-se particularmente importante quando pretendem compartilhar código dentro do grupo, numa situação de aprendizagem à distância. Transferem o código entre o Wokwi e o IDE do Arduino.

Background/Introdução à ideia:

Os professores podem usar a simulação online do Wokwi tanto no ensino presencial, como também no ensino à distância. Os alunos podem usá-lo para consolidar os seus conhecimentos adquiridos para encontrar soluções para os problemas no ambiente de programação.

Duração: aprox. 15 min

Materiais necessários para o professor: Apresentação de PowerPoint



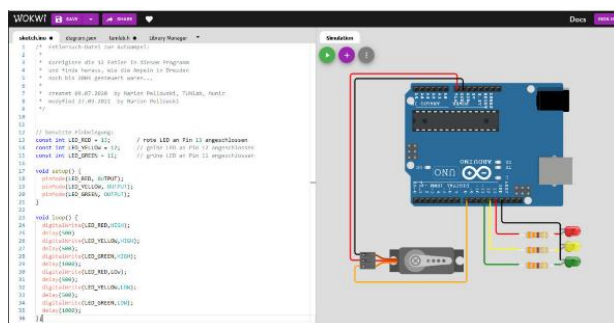
Materiais necessário para os alunos: Arquivo Wokwi (<https://wokwi.com/projects/328915985532715602>), Conexão de internet

Forma de trabalho: Os alunos trabalham de forma autónoma com o arquivo Wokwi, quer online e com suporte do professor por videoconferência, quer num laboratório de informática na escola

Conteúdo:

Atividades do professor:

- Distribuir o arquivo Wokwi e atribuir a tarefa



Atividades do aluno:

- Os alunos encontram erros no arquivo do Wokwi e podem observar os efeitos das suas alterações nos LEDs simulados.

4.4 Unidade 2.b Servomotor

Tópicos: Conhecer o motor e sua função, programando as posições angulares de um motor.

Objetivos:

Os alunos aprendem sobre as propriedades e funções dos motores. Na planta de produção miniaturizada, os grupos de alunos usarão motores como atuadores para mover engrenagens ou barreiras, por exemplo. Em geral, o uso de motores como atuadores é bastante comum em processos de automatização em miniatura ou na confecção de modelos.

Background/Introdução à ideia:

Três motores movimentam componentes essenciais da mini linha de produção: um motor permite movimentar a passadeira de entrada, um motor permite a passagem das bolas no dispensador e um motor movimenta o ejetor. Os alunos podem, assim, transferir os seus conhecimentos de programação do motor para diferentes aplicações. Os alunos podem aprender que diferentes objetos reais do mesmo tipo são, em termos de programação, objetos diferentes da mesma classe - um *insight* importante na programação orientada a objetos.

Duração: aprox. 30 min



Materiais necessários para o professor: Apresentação de PowerPoint

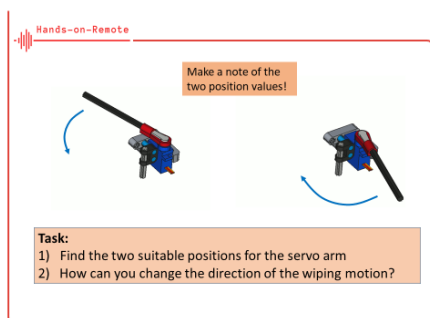
Materiais necessários para os alunos: Manual interativo, Arduino IDE, simulação Wokwi, Arduino, servomotor, breadboard, fios

Forma de trabalho: Online ou no computador da escola

Conteúdo:

Atividades do professor:

- Introdução: O que é que um motor pode fazer? Para que queremos usá-lo?



Atividades dos estudantes:

- Os alunos podem optar por trabalhar primeiro com o dispensador, barreira ou braço. O motor só muda entre duas posições; ele varia entre a posição de repouso e a posição ativa.
- Os alunos podem copiar o novo código necessário primeiro na simulação e depois no IDE do Arduino.
- Os alunos constroem um circuito com um motor real e o Arduino e tentam encontrar posições de repouso ou ativas adequadas.

4.5 Unidade 3.a plataforma giratória

Tópicos: aplicar competências práticas para desenvolver uma parte mecânica, construir outro circuito simples e inserir um novo código para criar um novo objeto.

Objetivos: Os alunos praticam enquanto configuram a plataforma giratória e ajustam o motor de acionamento. Os alunos podem ter que resolver pequenos problemas mecânicos durante a montagem, por exemplo, para encontrar a posição correta para o motor de acionamento. Os alunos constroem outro circuito com o Arduino e adicionam um novo código para mover um novo elemento.

Background/Introdução à ideia: A mesa giratória é central para o processo de automatização miniaturizado, pois possibilita o transporte de objetos entre as diversas estações do processo de automatização.

Formulário: Os alunos trabalham de forma independente sob a orientação do manual interativo, quer on-line e com apoio do professor por videoconferência, quer num laboratório de informática na escola.

Duração: aprox. 20 min

Materiais necessários para o professor: apresentação em PowerPoint, incluindo links para vídeos do Youtube

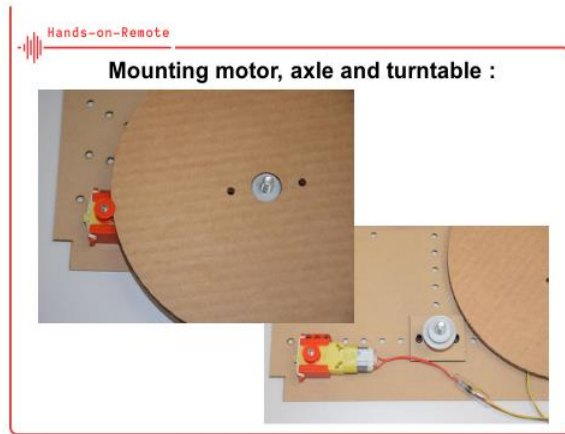


Materiais necessários para os alunos: Manual interativo, Arduino IDE, placa base, plataformas giratórias, motor de acionamento com roda de fricção, suporte como suporte do motor, anel de borracha, eixo (parafuso), porcas, conectores pretos, Arduino, protoboard, fios

Conteúdo:

Atividades do professor:

- Breve introdução à montagem das plataformas giratórias, usando a apresentação em PowerPoint



Atividades dos estudantes:

- Os alunos montam a plataforma giratória e o motor que se encontra na base
- Os alunos expandem o circuito do Arduino e conectam o motor de acionamento

```
//Create a motor object
Motor xxx(6);
...
xxx.go(HIGH);
...
xxx.go(LOW);
```

- Os alunos adicionam apenas algumas linhas de novo código para o motor de acionamento. Os alunos criam um objeto motor e ligam ou desligam o motor. O código necessário para ligar ou desligar o motor é muito semelhante ao código necessário para deixar um LED piscar, para que os alunos possam transferir os conhecimentos adquiridos anteriormente. A tarefa do professor é deixar as plataformas giratórias se moverem um quarto.

4.6 Unidade 3.b Barreira de Luz

Tópicos: Circuitos relacionados com a barreira de luz, Resistor dependente de luz (LDR) como sensor de brilho, relação entre valores de resistência e intensidade, controlo de processos dependente de sensor.

Objetivo: Os alunos identificam a necessidade de sensores para o controlo automatizado de uma planta de produção. Os alunos aprendem como funciona uma barreira de luz e aplicam a barreira como um sensor para controlar o seu “processo de produção”.

Duração: aprox. 25 minutos

Materiais necessários para o professor: apresentação em PowerPoint



Materiais necessários para os alunos: Manual interativo, Arduino IDE, plataformas giratórias já montadas, barreira de luz LDR/reflex, suporte para a barreira de luz, Arduino, placa de ensaio, fios

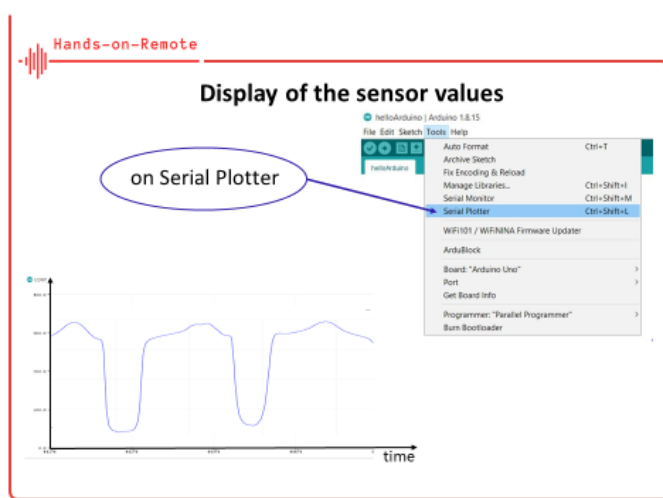
Background/Introdução à ideia: As barreiras de luz são elementos frequentemente usados na vida quotidiana, por exemplo, ao fechar as portas dos comboios suburbanos. Os alunos aprendem sobre uma possível aplicação de processos de automatização com a minifábrica. Os alunos obterão informações sobre como o resistor dependente da luz funciona e como os valores medidos podem ser influenciados mesmo por pequenos desvios no brilho. Os professores podem usar essas medições para lembrar os alunos das dificuldades de obter medições precisas e significativas e discutir com os grupos de alunos o que isso significa.

Forma de trabalho: Os alunos trabalham de forma independente sob a orientação do manual interativo, quer online e com apoio do professor por videoconferência quer num laboratório de informática na escola. Num ambiente online, os alunos podem trabalhar juntos em salas temáticas.

Conteúdo:

Atividades do professor:

- Os professores introduzem o resistor dependente da luz (LDR) como um sensor de brilho
- Os professores auxiliam os alunos no uso da barreira de luz e na geração dos gráficos de valores medidos.
- Os professores estão disponíveis para tirar dúvidas aos alunos, por exemplo, por videoconferência



Atividades dos alunos:

- Os alunos aprendem sobre as propriedades de uma barreira de luz reflexa usando as instruções interativas
- Os alunos montam a barreira de luz reflexa na placa de base
- Os alunos expandem o circuito do Arduino e ligam a barreira de luz LDR/reflex ao Arduino
- Os alunos traçam os valores dos sensores que obtêm do LDR e interpretam os gráficos. Também podem rastrear mudanças sob diferentes condições de iluminação.
- Os alunos usam a barreira de luz reflexa para detetar copos que se encontram à sua frente e parar ou iniciar a plataforma giratória.
- Os alunos adicionam linhas de novo código para a barreira de luz, especialmente o comando de espera 'waitForCan' – somente quando o sensor deteta um copo, o programa continua.

4.7 Unidade 4. Extensão para o processo de automatização

Tópicos: Transferência de conhecimentos previamente adquiridos para novos elementos

Objetivo: Os alunos trabalham de forma independente e podem escolher quais componentes para as plataformas giratórias que podem adicionar ao processo de automatização. Ao fazer isso, os alunos podem transferir conhecimentos previamente adquiridos sobre programação, circuitos ou processos de automatização e aplicá-los nas suas novas tarefas. Também dá aos alunos oportunidade para desenvolverem os seus próprios projetos.

Duração: flexível, pelo menos 45 min



Materiais necessários para o professor: Se necessário, lista de erros e dificuldades comuns

Materiais necessários para os alunos: Manual interativo, Arduíno IDE, plataformas giratórias já montadas, LDR/reflex light bar-rier, suporte para barreira de luz, Arduíno, breadboard, fios, como material para escolha dos alunos: dispensador, ejetor, slide de entrada, LEDs ou módulo de semáforo

Antecedentes/Introdução à ideia: Os alunos podem obter um processo de automatização com vários elementos funcionais.



Forma de trabalho: Os alunos trabalham de forma independente sob a orientação do manual interativo, quer on-line e com apoio do professor por videoconferência quer num laboratório de informática na escola. N um ambiente online, os alunos podem trabalhar juntos em salas temáticas.

Conteúdo:

Atividades do professor:

Os professores estão disponíveis para tirar dúvidas aos alunos, por exemplo, por videoconferência

Atividades dos alunos:



Os alunos trabalham com o manual interativo para adicionar mais componentes ao processo de automatização. Na página do índice do manual interativo, os alunos podem clicar para aceder aos diferentes componentes que lhes interessam. Os alunos podem escolher quais componentes que pretendem adicionar.

<i>Versão curta</i>	<i>Variação</i>	<i>Variação</i>	<i>Variação</i>	<i>Versão longa</i>
Plataforma giratória Barreira de Luz Dispensor	Plataforma giratória Barreira de Luz Dispensor LED para o indicador	Plataforma giratória Barreira de Luz Dispensor Ejetor	Plataforma giratória Barreira de Luz Dispensor Ejetor Slide de entrada	Plataforma giratória Barreira de Luz Dispensor Ejetor Slide de entrada LED para o atuador

Esta unidade pode ser realizada de forma flexível tendo em conta a possibilidade de escolher diferentes percursos (vd. tabela acima), dependendo do tempo que o professor tenha disponível. Com os diferentes percursos, os professores também podem levar em conta diferentes capacidades de programação e a experiência anterior dos alunos.

4.8 Unidade 5. Colaboração entre alunos



Tópicos: Expandir as competências de programação, trabalho de equipa e colaboração

Objectivo: Trabalho colaborativo dos alunos

Duração: min. 45 min

Materiais para o professor: Nenhum



Materiais para os alunos: Mini-usinas de produção já construídas pelas equipas

Conhecimentos prévios necessários: As equipas de alunos devem ter programado pelo menos uma das partes funcionais do processo de automatização. O ideal é que os alunos de equipas diferentes tenham programado partes funcionais diferentes.

Forma: Num laboratório de informática na escola

Conteúdo:

Atividades do professor:

O professor inicia e apoia a colaboração e o trabalho de equipa dos alunos.



Atividades dos alunos:

A tarefa dos jovens é ligar os seus gira-discos entre equipas. Os alunos decidem como querem emparelhar os gira-discos e programam em conjunto os passos de programação necessários.

4.9 Unidade 6. Colaboração digital



Tópicos: Medição com sensores, desencadeamento de diferentes atuadores do processo de automação dependentes do sensor, mesmo em locais remotos

Objetivo: fomentar um sentido de comunidade entre os alunos

Duração: flexível

Materiais necessários para o professor: Nenhum



Conhecimentos necessários: Os alunos precisam ter programado processos dependentes de sensores com atuadores anteriormente (Unidade 3.2)

Forma: Online ou em um laboratório de informática na escola

Conteúdo:

Atividades do professor:

Os professores introduzem a ideia de um processo de comunicação via transmissão de luz e auxiliam os alunos a ajustar seus sensores.



Atividades dos alunos:

Os alunos programam um dos atuadores do processo de automação e o colocam em movimento se o sensor de luz detectar um aumento de brilho. O primeiro aluno posiciona a barreira de luz, para que o LED brilhe na câmara do laptop quando ele for ligado. O segundo aluno posiciona a barreira de luz, para que o sensor de luz possa detectar a luz transmitida pelo primeiro aluno em um ambiente de videoconferência. Ao ligar a luz, o primeiro aluno coloca o atuador do segundo aluno em movimento. O segundo aluno pode então enviar um sinal de luz para o terceiro aluno e assim por diante.

4.10 Unidade 7. Automação no seu contexto social

Os professores podem optar por continuar com esta sequência ou passar para outras aplicações das plataformas giratórias primeiro (unidade 8.).

Tópicos: Comportamento do consumidor, dependência de bens manufaturados, efeitos colaterais do consumo (demanda de energia, desperdício, transporte, etc.), vantagens e desvantagens dos processos de automação

Objetivo: Os alunos exploram vários impactos dos processos de automação na sociedade. Os alunos refletem sobre seu próprio comportamento de consumo.

Duração: flexível, pelo menos 45 min

Materiais necessários para o professor: Acesso ao quadro branco online que o professor pode distribuir aos alunos, cartões com questões que o professor pode distribuir aos alunos

Materiais necessários para os alunos: Cartões de questões, quadro branco on-line e conexão à internet, se os professores quiserem usá-los: plataformas giratórias já montadas

Formulário: Online ou num laboratório de informática da escola

Conteúdo:

Atividades do professor:

O professor entrega os cartões aos alunos e/ou conduz uma discussão em grupo. Introdução e transição para o novo tópico, dando os parabéns pela conquista e perguntando o que os alunos gostariam de fazer depois de terminar as suas tarefas. Se os professores puderem recolher diretamente as respostas dos alunos, sugere-se que incorporem as suas nas aulas.

O professor discute as respostas com os alunos e distingue se o consumo de produtos manufaturados faz parte ou não das guloseimas propostas.





Parabéns pela sua primeira produção automatizada!

Agora que conseguiu realmente algo, pode mimar-se com algo agradável.....

Com o que é que gostaria de se mimar agora?

Não hesite em discutir com a equipa o que lhe ocorrer e, se quiser, escreva-o aqui....



Nas suas ideias sobre o que quer fazer, há alguma coisa que não tenha nada a ver com consumo ou produtos de consumo?

Consegue imaginar-se a mimar-se sem precisar ou consumir nada? O que é que isso seria?

Se todas as respostas forem baseadas no consumo, o professor entrega o segundo cartão, que pergunta se um mimo sem consumo também seria possível. Discute com os alunos os efeitos colaterais do alto consumo (custos de energia, desperdício, transporte, etc.).

Dependendo do curso da discussão e do foco escolhido, o professor pode fazer mais perguntas. O professor familiariza os alunos com o quadro branco on-line usado e explica a sua tarefa aos alunos. O professor pode deixar que os alunos escolham um tópico que desejam explorar com mais profundidade. Se os professores quiserem, podem usar uma sobreposição de papel para as plataformas giratórias e deixar que os alunos escolham seu tópico dessa maneira.



Atividades dos alunos:

- Espera-se que os alunos gostem de construir e programar o sistema de mini-fabricação. Ao fazê-lo, também pensam em alternativas ao consumo de produtos manufaturados.
- Os alunos podem olhar para as plataformas giratórias que acabaram de construir e reconhecer que também é um produto "manufaturado": Que peças podem estar relacionadas com o processo de automatização?
- Os alunos podem começar por discutir quais as vantagens e desvantagens que os processos de automatização. Os alunos recolhem as suas ideias associadas à automatização e suas consequências sociais num quadro branco online. Os alunos podem adicionar diferentes materiais, como imagens, vídeos, depoimentos e assim por diante.
- Os alunos podem escolher um tópico que gostariam de explorar com mais profundidade e discuti-lo em grupo.

4.11 Unit 8. Outras aplicações das plataformas giratórias

Tópicos: Valores de brilho para distinguir objetos ou reproduzir sons

Objetivo: Explorar outras aplicações para os mesmos materiais de forma a continuar a trabalhar com o kit de materiais e experimentar algo novo. Alguns dos alunos também estão dispostos a explorar ainda mais com o kit de material no seu tempo livre. Os alunos podem continuar a trabalhar de forma muito independente.

Ao mesmo tempo, essas outras aplicações podem representar alternativas, caso professores e alunos não pretendam se envolver demais na programação.

Duração: flexível, pelo menos 45 min



Materiais necessários para o professor: Se necessário: Impressora para imprimir escalas de cinza feita pelos alunos

Materiais necessários para os alunos: Arduíno, breadboard, fios, Arduíno IDE, plataformas giratórias já montadas, barreira de luz. Adicionalmente para a máquina de triagem: ejetor e indicador de status LED.

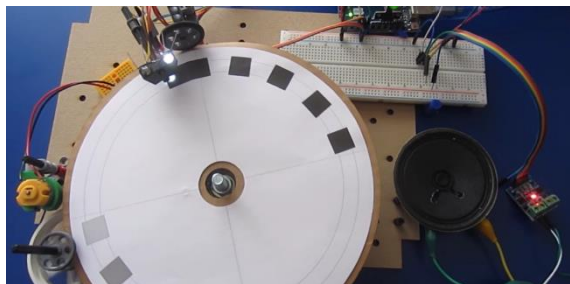
Além disso, para o reproduzidor de som induzido pela luz: papelão em tamanho de disco para colocar firmemente nas plataformas giratórias, uma impressora para imprimir o padrão preto e branco ou a escala de cinza no disco de papel, uma campainha, um altifalante

Sorting machine	Light-induced sound player	...
Plataforma giratória Barreira de luz Indicador do status do LED brilho/cor Uso do ejetor para classificação	Plataforma giratória com sobreposições de papel em um padrão preto e branco Sensor de luz Altifalante (não incluído no kit)	

Modo de trabalho: Trabalho autónomo dos alunos

Conteúdo:

- Máquina de classificação: use o ejetor para classificar diferentes tipos de objetos que diferem, por exemplo, em brilho ou cor



- Reprodutor de som induzido por luz: Se colocar um papel com um padrão preto e branco no plataformas giratórias, poderá usar o sensor de luz para converter essas informações de brilho em tons e ritmos simples para um dispositivo de saída de som simples. Também pode usar uma escala de cinza impressa para o mesmo efeito.



Atividades dos alunos:

- Os jovens programam a mini linha de produção como uma máquina de selecção e distinguem os copos de acordo com a cor ou o brilho.
- Os alunos desenharam padrões em escala de cinzentos para o gira-discos. O Arduino reconhece os valores de brilho e reproduz os sons correspondentes. Os jovens também podem atribuir sons a valores de brilho na programação.

4.12 Notas gerais sobre videoconferência e trabalho prático em situações remotas

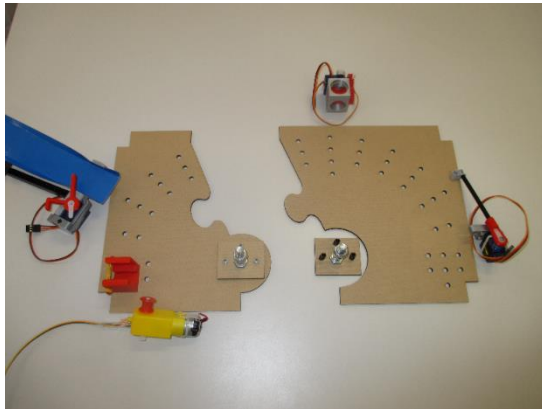
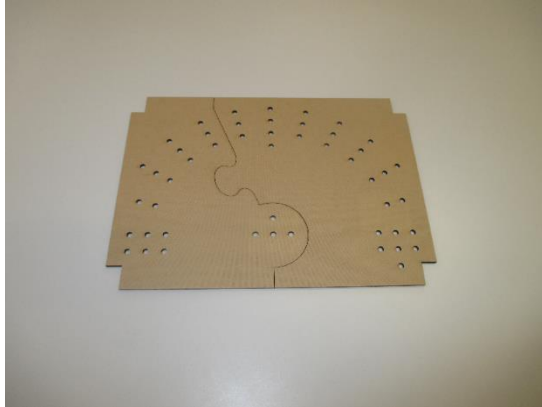
Proteção de dados

A escola deve esclarecer dúvidas sobre o processamento de dados necessário durante a videoconferência – por exemplo, o uso de compartilhamento de tela ou o uso da câmara – de acordo com o CNPD. Em estreita consulta com os alunos, os professores podem conduzir aulas sem a câmara ligada; no entanto, se os alunos ligarem a câmara, é muito mais fácil ver como os alunos estão a progredir e pode melhorar significativamente o sentimento do grupo. Uma opção que os professores podem oferecer é fazer com que os alunos apontem a câmara para os seus projetos sem estarem na imagem – mas essa opção não estará disponível para todos os alunos.

Gerir os materiais

A primeira parte da sequência do projeto, nomeadamente as secções 1.1 Introdução ao tema a 2.2. motor, pode ser feito facilmente num ambiente de aprendizagem online. Para isso, os professores precisam de um sistema de videoconferência adequado que lhes permita fazer uma apresentação online e, se necessário, criar salas de apoio onde os alunos possam trabalhar de forma autónoma. A proporção de peças que os alunos usam para o seu trabalho em casa consiste num Arduíno, uma placa, cabos, um módulo de semáforo ou LEDs e um motor, ou seja, um conjunto de materiais que se consegue gerir. No entanto, num ambiente totalmente remoto, isso exige que todos os alunos de uma turma recebam esse conjunto. Caso não seja possível, o professor pode aprofundar o trabalho com o ambiente de simulação online Wokwi. No entanto, isso será à custa das competências para aprender a manusear um microcontrolador, capacidades que os alunos ainda precisarão para as unidades posteriores.

Quando se trata das plataformas giratórias, a implementação torna-se mais complicada. O ideal é que cada aluno receba um kit de material completo para trabalhar em casa. Se isso não for possível (seja pelo custo, pelo esforço necessário para a produção, pela dificuldade de empréstimo ou envio dos materiais), há necessidade de uma alternativa.



Uma alternativa poderia ser, por exemplo, dividir a placa de base para o prato giratório e dar aos alunos diferentes partes para que possam trabalhar em equipa para conceber o processo de automatização.

5 Possibilidades de uso do módulo – diferentes percursos

Se os pré-requisitos forem diferentes, pode ser útil usar o módulo de forma diferente. Dois percursos alternativos são sugeridos abaixo, provavelmente há mais possibilidades.

A Percurso 1 continua a depender do processo de automatização. No entanto, o processo de automatização é reduzido ao transporte de copos com a plataforma giratória, o uso do motor do dispensador e um controlo dependente do sensor do processo de produção.

O Percurso 2 permite que use as plataformas giratórias de uma maneira completamente diferente, ou seja, apenas como reprodutor de som induzido pela luz. Portanto, apenas os passos básicos são necessários para trabalhar com o Arduino, colocar as plataformas giratórias em movimento e trabalhar com a barreira de luz.

Percurso 1

1.1 Introdução ao uso de plataformas giratórias na automatização e o objetivo de miniaturizar um processo de automatização.
1.2 Introdução ao trabalho com o Arduino usando um primeiro projeto de circuito e exemplo de programação com LEDs de cores diferentes.
2.1 Introdução aos motores, incluindo circuitos e programação de um motor real. Usando o motor do dispensador.
2.2 Configuração da plataforma giratória, circuitos e programação do motor de acionamento.
2.3 Funcionamento da barreira luminosa, incluindo circuitos, programação e visualização de valores no plotter. Usando o sensor de luz para controlar o “processo de produção”, ou seja, parar a plataforma giratória, se houver um copo na frente do dispensador.
3. Automatização no seu contexto social

Percurso 2

1.1 Introdução a um tema da sua escolha
1.2 Introdução ao trabalho com o Arduino, usando um primeiro projeto de circuito e exemplo de programação com LEDs de cores diferentes.
2.1 Configuração da plataforma giratória, circuitos e programação do motor de acionamento.
2.2 Funcionamento da barreira luminosa, incluindo circuitos, programação e visualização de valores no plotter.
3. Uso da plataforma como reproduutor de som induzido pela luz

6 Apêndice: Perspectivas da página de instrução multimédia interactiva

As capturas de ecrã mostram um exemplo das instruções passo a passo com imagens das animações, vídeos e efeitos de passagem do rato para a instalação, cablagem e programação do escorrega de entrada e da barreira luminosa de reflexo. Para obter uma melhor impressão, pode ser aconselhável aumentar o zoom para uma vista maior.

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada
- Vista geral

Automatização em miniatura

Co-funded by the Erasmus+ Programme of the European Union

Parceiros do projecto europeu:

Deutsches Museum, LISBOA, TUM, CENTRALE NORD ROBERT, BS Info, BS Fisi

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada
- Vista geral

Visão geral

Automação com o Arduino

- Posto de enchimento de uma linha de produção

Mova o rato sobre as caixas

```

graph LR
    A[Copos separados] --> B[transportar o copo]
    B --> C[encher]
    C --> D[transportar]
    D --> E[sair]
    E --> A
    
```

Tarefa: Clique numa caixa e salte para o capítulo correspondente.
Terminou com tudo? Então clique aqui: [finalizado](#)

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada
- Vista geral

Visão geral

Automação com o Arduino

- Posto de enchimento de uma linha de produção

Mova o rato sobre as caixas

```

graph LR
    A[Copos separados] --> B[transportar o copo]
    B --> C[encher]
    C --> D[transportar]
    D --> E[sair]
    E --> A
    
```

Tarefa: Clique numa caixa e salte para o capítulo correspondente.
Terminou com tudo? Então clique aqui: [finalizado](#)

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada
- Vista geral

Hardware - Escorrega de entrada

O escorrega de entrada faz com que os copos deslizem um a um sobre a plataforma giratória. Para este propósito, o servo motor gira a barreira para frente e para trás entre duas posições. Isso significa que o copo não precisa ser colocado na plataforma à mão.

Mova o rato sobre a imagem...

A seguir, a lâmina de entrada deve primeiro ser montada, então o servo motor deve ser conectado ao Arduino e, finalmente, o Arduino deve ser programado.

Hands-on-Remote

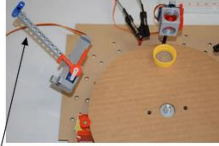
- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

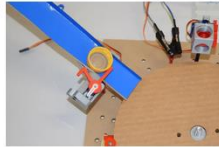
Hardware - monte o slide de entrada

Monte o escorrega de entrada com a porta:

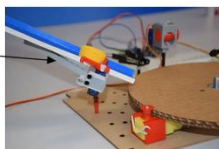
1. Prenda o escorrega de entrada com o suporte na placa de base.



2. Empurre o slide para dentro da guia.



3. O braço Lego (feixe) serve como suporte para o slide.

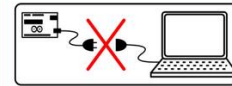


Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Fiação - Slide de entrada



Por favor, desconecte o Arduino do PC antes de mudar o fio!

Trabalhe no Arduino apenas no estado 'desenergizado'!

Agora o servo da porta do escorrega de entrada tem que ser conectado ao Arduino:

Use o autocarro de abastecimento na breadboard (vermelho e azul):

No Arduino: fio vermelho: V in - Pin -> autocarro vermelho

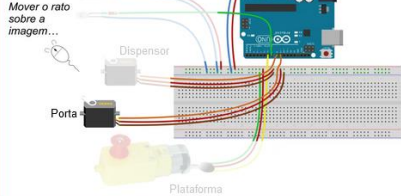
No breadboard: fio azul: GND - Pin -> autocarro azul

Conecte o servomotor:

Laranja -> Pin 9

Vermelho -> V in - autocarro

castanho -> GND - autocarro



Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Fiação - Slide de entrada



Por favor, desconecte o Arduino do PC antes de mudar o fio!

Trabalhe no Arduino apenas no estado 'desenergizado'!

Agora o servo da porta do escorrega de entrada tem que ser conectado ao Arduino:

Use o autocarro de abastecimento na breadboard (vermelho e azul):

No Arduino: fio vermelho: V in - Pin -> autocarro vermelho

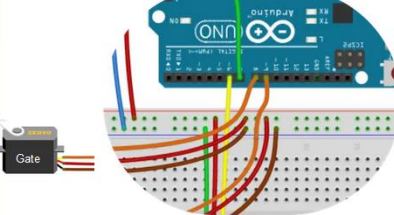
No breadboard: fio azul: GND - Pin -> autocarro azul

Conecte o servomotor:

Laranja -> Pin 9

Vermelho -> V in - autocarro

castanho -> GND - autocarro



Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

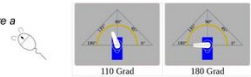
Vista geral

Hardware - Porta-Servo

A barreira do escorrega de entrada (= porta) é movida por um servo.

Servos são pequenos motores que geralmente podem ...

Mova o rato sobre a imagem...



Para a barreira, precisamos de duas posições:

-> barreira fechada (= posição de descanso, retém todos os copos

-> barreira aberta (= posição de passagem um copo desliza para a plataforma)

O ponteiro angulado...

Clique na imagem para iniciar a animação...

Os graus de ambas as posições devem ser determinados por tentativa e erro.

A propósito: O valor do ângulo da posição de repouso é maior do que o valor do ângulo da posição de passagem.

00:00 00:05

Hands-on-Remote

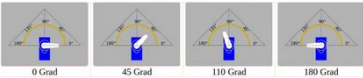
- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Hardware - Porta-Servo

A barreira do escorrega de entrada (= porta) é movida por um servo.

Servos são pequenos motores que geralmente podem mover apenas meia volta, entre 0° e 180°. Envie ao servo o número desejado de graus e o servo se move para a posição.



Para a barreira, precisamos de duas posições:

-> barreira fechada (= posição de descanso, retém todos os copos

-> barreira aberta (= posição de passagem um copo desliza para a plataforma)

O ponteiro angulado...

Clique na imagem para iniciar a animação...

Os graus de ambas as posições devem ser determinados por tentativa e erro.

A propósito: O valor do ângulo da posição de repouso é maior do que o valor do ângulo da posição de passagem.

00:00 00:05

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

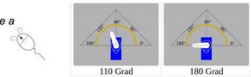
Vista geral

Hardware - Porta-Servo

A barreira do escorrega de entrada (= porta) é movida por um servo.

Servos são pequenos motores que geralmente podem ...

Mova o rato sobre a imagem...



Para a barreira, precisamos de duas posições:

-> barreira fechada (= posição de descanso, retém todos os copos

-> barreira aberta (= posição de passagem um copo desliza para a plataforma)

O ponteiro angulado do portão impede que outros copos escoreguem ao abrir a barreira.

Clique na imagem para iniciar a animação...

Os graus de ambas as posições devem ser determinados por tentativa e erro.

A propósito: O valor do ângulo da posição de repouso é maior do que o valor do ângulo da posição de passagem.

00:00 00:05

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Programação: Porta de controle

Trabalhe na guia ino.

Tarefa: Programe o servo da porta para que ele possa produzir um copo. Encontre os valores de ângulo apropriados para as posições de descanso e passagem.

- Inclua o arquivo ".h" que você está usando com a instrução #include. ...
- Crie um objeto servo, nomeie-o e ...
- Em setup() ...
- Gire a porta no loop() ...
- Comente o programa...

```

// sketch-00P $ tumbler.h
/*
 *
 */
#include "tumbler.h"

ServoMotor name(pin,pos0,pos1);

void setup() {
  - go (LOW);
}

void loop() {
  - goSlow (HIGH);
}

```

Tarefa adicional: um LED deve indicar a atividade da porta (processo total: virar para a posição de passagem e de volta para a posição de repouso).

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Programação: Porta de controle

Trabalhe na guia ino.

Tarefa: Programe o servo da porta para que ele possa produzir um copo. Encontre os valores de ângulo apropriados para as posições de descanso e passagem.

- Inclua o arquivo ".h" que você está usando com a instrução #include. Isso torna as definições de classe disponíveis. O nome do arquivo inteiro deve estar em ". ". Esta linha não precisa de um ponto-e-vírgula;
- Crie um objeto servo, nomeie-o e ...
- Em setup() ...
- Gire a porta no loop() ...
- Comente o programa...

```

// sketch-00P $ tumbler.h
/*
 *
 */
#include "tumbler.h"

ServoMotor name(pin,pos0,pos1);

void setup() {
  - go (LOW);
}

void loop() {
  - goSlow (HIGH);
}

```

Tarefa adicional: um LED deve indicar a atividade da porta (processo total: virar para a posição de passagem e de volta para a posição de repouso).

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Programação: Porta de controle

Trabalhe na guia ino.

Tarefa: Programe o servo da porta para que ele possa produzir um copo. Encontre os valores de ângulo apropriados para as posições de descanso e passagem.

- Inclua o arquivo ".h" que você está usando com a instrução #include. ...
- Crie um objeto servo, nomeie-o e passe o número do pin onde conectou o servo e as duas posições angulares entre chaves ().
- Em setup() ...
- Gire a porta no loop() ...
- Comente o programa...

```

// sketch-00P $ tumbler.h
/*
 *
 */
#include "tumbler.h"

ServoMotor gate(9,100,10);

void setup() {
  - go (LOW);
}

void loop() {
  - goSlow (HIGH);
}

```

Tarefa adicional: um LED deve indicar a atividade da porta (processo total: virar para a posição de passagem e de volta para a posição de repouso).

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Programação: Porta de controle

Trabalhe na guia ino.

Tarefa: Programe o servo da porta para que ele possa produzir um copo. Encontre os valores de ângulo apropriados para as posições de descanso e passagem.

- Inclua o arquivo ".h" que você está usando com a instrução #include. ...
- Crie um objeto servo, nomeie-o e ...
- Em setup() pode trazer o servo para sua posição de repouso com o método go().
- Gire a porta no loop() ...
- Comente o programa...

```

// sketch-00P $ tumbler.h
/*
 *
 */
#include "tumbler.h"

ServoMotor gate(9,100,10);

void setup() {
  gate.go (LOW);
}

void loop() {
  - goSlow (HIGH);
}

```

Tarefa adicional: um LED deve indicar a atividade da porta (processo total: virar para a posição de passagem e de volta para a posição de repouso).

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Programação: Porta de controle

Trabalhe na guia ino.

Tarefa: Programe o servo da porta para que ele possa produzir um copo. Encontre os valores de ângulo apropriados para as posições de descanso e passagem.

- Inclua o arquivo ".h" que você está usando com a instrução #include. ...
- Crie um objeto servo, nomeie-o e ...
- Em setup() ...
- Gire a porta no loop() com o método go() na posição de passagem e de volta novamente passando os valores "HIGH" e "LOW", respectivamente.
- Comente o programa...

```

// sketch-00P $ tumbler.h
/*
 *
 */
#include "tumbler.h"

ServoMotor gate(9,100,10);

void setup() {
  gate.go (LOW);
}

void loop() {
  gate.goSlow (HIGH);
  delay(100);
  gate.goSlow (LOW);
  delay(2000);
}

```

Tarefa adicional: um LED deve indicar a atividade da porta (processo total: virar para a posição de passagem e de volta para a posição de repouso).

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Programação: Porta de controle

Trabalhe na guia ino.

Tarefa: Programe o servo da porta para que ele possa produzir um copo. Encontre os valores de ângulo apropriados para as posições de descanso e passagem.

- Inclua o arquivo ".h" que você está usando com a instrução #include. ...
- Crie um objeto servo, nomeie-o e ...
- Em setup() ...
- Gire a porta no loop() ...
- Comente o programa: comentários de várias linhas começam com /* e terminam com */ // introduz comentário de linha única.

```

// sketch-00P $ tumbler.h
/*
 * a cup slips through
 * 29.01.2022 Mike
 */
#include "tumbler.h"

ServoMotor gate(9,100,10);

void setup() {
  // gate in rest position:
  gate.go (LOW);
}

void loop() {
  // gate in action:
  gate.goSlow (HIGH);
  delay(100);
  gate.goSlow (LOW);
  delay(2000);
}

```

Tarefa adicional: um LED deve indicar a atividade da porta (processo total: virar para a posição de passagem e de volta para a posição de repouso).

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Fiação - barreira de luz

Por favor, desconecte o Arduino do PC antes de mudar o fio!

Trabalhe no Arduino apenas no estado 'desenergizado'!

Agora conecte o LDR (sensor de luz) e o LED (fonte de luz) ao Arduino.

Use o autocarro de alimentação (vermelho e azul) na breadboard novamente:

	No Arduino	Na breadboard
fio vermelho:	V in - Pin	autocarro vermelho
fio azul:	GND - Pin	autocarro azul

- Ligar LDR:**
 - Amarelo -> Pin A0
 - Verde -> power bus
 - Azul -> GND bus

Mover o rato sobre a imagem...
- Ligar LED:**
 - Cinza -> Pin 7
 - Preto -> GND

(não precisa do cabo branco para o LED).

Plataforma giratória

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Fiação - barreira de luz

Por favor, desconecte o Arduino do PC antes de mudar o fio!

Trabalhe no Arduino apenas no estado 'desenergizado'!

Agora conecte o LDR (sensor de luz) e o LED (fonte de luz) ao Arduino.

Use o autocarro de alimentação (vermelho e azul) na breadboard novamente:

	No Arduino	Na breadboard
fio vermelho:	V in - Pin	autocarro vermelho
fio azul:	GND - Pin	autocarro azul

- Ligar LDR:**
 - Amarelo -> Pin A0
 - Verde -> power bus
 - Azul -> GND bus
- Ligar LED:**
 - Cinza -> Pin 7
 - Preto -> GND

(não precisa do cabo branco para o LED).

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Valores medidos - luz refletida e ambiente

Se o feixe de luz do LED não for a lugar nenhum (nenhum copo na frente da barreira de luz), a luz não é refletida, o sensor de luz só vê a luz ambiente.

Se um copo estiver no feixe de luz, a luz do LED refletida pelo copo é adicionada à luz ambiente, o sensor de luz mede mais brilho.

Atenção: quanto mais brilho, menor o valor medido. Isto é devido à forma como o resistor em série e o sensor de luz estão conectados.

copo está passando a barreira de luz

Linha azul: Valores medidos com LED ligado
Linha vermelha: Valores medidos com LED desligado
Linha verde: Diferença entre os dois valores medidos

Somente se um copo passar na frente da barreira de luz e refletir a luz, os dois valores diferem significativamente. A linha verde mostra a diferença dos valores medidos. Avaliamos essa diferença.

Se a diferença exceder o valor definido por si, considera-se detetado um copo.

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Programação: A classe "ReflexSensor"

ReflexSensor

- name: String
- pinLEDnumber: int
- pinLDRnumber: int
- trigger: int

- ReflexSensor(pinLED, pinLDR, trigger)
- ReflexSensor(pinLDR)
- showLDR(void): void
- showValues(void): void
- waitForCan(void): void
- detectCan(void): bool

Ao criar o objeto, a classe "ReflexSensor" espera ...

```
// create light sensor objects:
ReflexSensor xxx (A0);
ReflexSensor yyy (7, A5, B0);

void setup() {
  xxx.showLDR(); //
  yyy.showValues();
}

void loop() {
  xxx.waitForCan();
  if (xxx.detectCan() == HIGH) {
    //
  }
}
```

Também é possível usar o LDR sozinho. Nesse caso ...

Os métodos 'showLDR()' e 'showValues()' podem ser usados para ...

A classe "ReflexSensor" fornece o método "waitForCan()", que ...

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Programação: A classe "ReflexSensor"

ReflexSensor

- name: String
- pinLEDnumber: int
- pinLDRnumber: int
- trigger: int

- ReflexSensor(pinLED, pinLDR, trigger)
- ReflexSensor(pinLDR)
- showLDR(void): void
- showValues(void): void
- waitForCan(void): void
- detectCan(void): bool

Ao criar o objeto, a classe "ReflexSensor" espera, além do nome, os números de pin aos quais o LED e o LDR estão conectados e o valor limite a partir do qual um copo é considerado detetado.

Também é possível usar o LDR sozinho. Nesse caso ...

Os métodos 'showLDR()' e 'showValues()' podem ser usados para ...

A classe "ReflexSensor" fornece o método "waitForCan()", que ...

```
// create light sensor objects:
ReflexSensor xxx (A0);
ReflexSensor yyy (7, A5, B0);

void setup() {
  xxx.showLDR(); //
  yyy.showValues();
}

void loop() {
  xxx.waitForCan();
  if (xxx.detectCan() == HIGH) {
    //
  }
}
```

Hands-on-Remote

- antecipadamente
- estrutura
- plataforma giratória
- dispensor
- barreira de luz
- saída ejector
- escorrega de entrada

Vista geral

Programação: A classe "ReflexSensor"

ReflexSensor

- name: String
- pinLEDnumber: int
- pinLDRnumber: int
- trigger: int

- ReflexSensor(pinLED, pinLDR, trigger)
- ReflexSensor(pinLDR)
- showLDR(void): void
- showValues(void): void
- waitForCan(void): void
- detectCan(void): bool

Ao criar o objeto, a classe "ReflexSensor" espera ...

```
// create light sensor objects:
ReflexSensor xxx (A0);
ReflexSensor yyy (7, A5, B0);

void setup() {
  xxx.showLDR(); //
  yyy.showValues();
}

void loop() {
  xxx.waitForCan();
  if (xxx.detectCan() == HIGH) {
    //
  }
}
```

Também é possível usar o LDR sozinho. Nesse caso, apenas o número PIN do LDR é necessário além do nome.

Os métodos 'showLDR()' e 'showValues()' podem ser usados para ...

A classe "ReflexSensor" fornece o método "waitForCan()", que ...

Programação: A classe "ReflexSensor"

antecipadamente
estrutura
plataforma giratória
dispensor
barreira de luz
saída ejector
escorrega de entrada
Vista geral

ReflexSensor

- name: String
- pinLEDnumber: int
- pinLDRnumber: int
- trigger: int
- ReflexSensor(pinLED, pinLDR, trigger)
- ReflexSensor(pinLDR)
- showLDR(void): void
- showValues(void): void
- waitForCan(void): void
- detectCan(void): bool

Mover o rato sobre as linhas de texto...

Ao criar o objeto, a classe "ReflexSensor" espera ...

Também é possível usar o LDR sozinho. Nesse caso ...

Os métodos 'showLDR()' e 'showValues()' podem ser usados para exibir graficamente os valores medidos retornados pelo LDR.

A classe "ReflexSensor" fornece o método "waitForCan()", que ...

```

// create light sensor object:
ReflexSensor xxx (A0);
ReflexSensor yyy (7, A0, 00);

void setup() {
  xxx.showLDR(); //
  yyy.showValues();
}

void loop() {
  xxx.waitForCan();
  if (xxx.detectCan() == HIGH) {
    ...
  }
}

```

Programação: A classe "ReflexSensor"

antecipadamente
estrutura
plataforma giratória
dispensor
barreira de luz
saída ejector
escorrega de entrada
Vista geral

ReflexSensor

- name: String
- pinLEDnumber: int
- pinLDRnumber: int
- trigger: int
- ReflexSensor(pinLED, pinLDR, trigger)
- ReflexSensor(pinLDR)
- showLDR(void): void
- showValues(void): void
- waitForCan(void): void
- detectCan(void): bool

Mover o rato sobre as linhas de texto...

Ao criar o objeto, a classe "ReflexSensor" espera ...

Também é possível usar o LDR sozinho. Nesse caso ...

Os métodos 'showLDR()' e 'showValues()' podem ser usados para ...

A classe "ReflexSensor" fornece o método "waitForCan()", que interrompe o fluxo do programa até que um copo seja detetado.

```

// create light sensor object:
ReflexSensor xxx (A0);
ReflexSensor yyy (7, A0, 00);

void setup() {
  xxx.showLDR(); //
  yyy.showValues();
}

void loop() {
  xxx.waitForCan();
  if (xxx.detectCan() == HIGH) {
    ...
  }
}

```

Programação: Configurar barreira de luz

antecipadamente
estrutura
plataforma giratória
dispensor
barreira de luz
saída ejector
escorrega de entrada
Vista geral

A sensibilidade da barreira de luz é determinada pelo valor de trigger.

Para descobrir o valor do trigger, pode mostrar os valores medidos do sensor de luz no gráfico.

Trabalhar na janela ino

- 1) Criar um novo objeto...
- 2) Determine o valor de trigger ...
- 3) Abra a série de pontos...
- 4) E determinar...
- 5) Nota: showValues() é executado num loop infinito. Portanto,
- 6) Agora adicione ...

```

//
// Extension of the programme code
//
ReflexSensor lightBarrier(7,A0,30);

void setup() {
  lightBarrier.showValues();
  // after the measurement this line
  // comment out again!
}

void loop() {
  ...
}

```

Programação: Configurar barreira de luz

antecipadamente
estrutura
plataforma giratória
dispensor
barreira de luz
saída ejector
escorrega de entrada
Vista geral

A sensibilidade da barreira de luz é determinada pelo valor de trigger.

Para descobrir o valor do trigger, pode mostrar os valores medidos do sensor de luz no gráfico.

Trabalhar na janela ino

- 1) Criar um novo objeto... da classe ReflexSensor com nome, pin e valor de trigger. Use o valor 30 por enquanto.
- 2) Determine o valor de trigger ...
- 3) Abra a série de pontos...
- 4) E determinar...
- 5) Nota: showValues() é executado num loop infinito. Portanto,
- 6) Agora adicione ...

Insira o valor assumido 30 como o valor de trigger e ajuste o valor na etapa (4).

```

//
// Extension of the programme code
//
ReflexSensor lightBarrier(7,A0,30);

void setup() {
  ...
}

void loop() {
  ...
}

```

Programação: Configurar barreira de luz

antecipadamente
estrutura
plataforma giratória
dispensor
barreira de luz
saída ejector
escorrega de entrada
Vista geral

A sensibilidade da barreira de luz é determinada pelo valor de trigger.

Para descobrir o valor do trigger, pode mostrar os valores medidos do sensor de luz no gráfico.

Trabalhar na janela ino

- 1) Criar um novo objeto...
- 2) Determine o valor de trigger ...
- 3) Abra a série de pontos... Deixe os copos passarem pela barreira da luz.
- 4) E determinar...
- 5) Nota: showValues() é executado num loop infinito. Portanto,
- 6) Agora adicione ...

Abra a série de pontos... Deixe os copos passarem pela barreira da luz.

Valor de trigger

Copos estão passando barreira de luz

```

//
// Extension of the programme code
//
ReflexSensor lightBarrier(7,A0,30);

void setup() {
  ...
}

void loop() {
  ...
}

```

Programação: Configurar barreira de luz

antecipadamente
estrutura
plataforma giratória
dispensor
barreira de luz
saída ejector
escorrega de entrada
Vista geral

A sensibilidade da barreira de luz é determinada pelo valor de trigger.

Para descobrir o valor do trigger, pode mostrar os valores medidos do sensor de luz no gráfico.

Trabalhar na janela ino

- 1) Criar um novo objeto...
- 2) Determine o valor de trigger ...
- 3) Abra a série de pontos... E determinar... um valor de trigger de ajuste do gráfico e substitua o 30 por esse valor.
- 4) E determinar...
- 5) Nota: showValues() é executado num loop infinito. Portanto,
- 6) Agora adicione ...

Valor de trigger

Copos estão passando barreira de luz

```

//
// Extension of the programme code
//
ReflexSensor lightBarrier(7,A0,30);

void setup() {
  ...
}

void loop() {
  ...
}

```



**CENTRUM
NAUKI
KOPERNIK**



Technische Universität München

All materials are available at

<https://erasmus-plus.ec.europa.eu/projects/search/details/2020-1-DE02-KA226-VET-008295>

The Hand-on Remote project was funded in the Erasmus+ KA226 Partnerships for Digital Education Readiness program (2020-1-DE02-KA226-VET-008295)

Co-funded by the
Erasmus+ Programme
of the European Union

