

Arduino assignment

(9 ქულა) ამოცანის მიზანია შევექმნათ ისრებიანი საათი 2 სახის - stepper და servo ძრავების დახმარებით.

დავავლების სრულფასოვნად შესასრულებლად საჭიროა რამდენიმე კომპონენტის: servo და stepper ძრავების, buzzer-ის და ღილაკების გამოყენება. ამ კომპონენტებით შექმნილ მოწყობილობას უნდა ჰქონდეს ბიბლიოთეკა, რომელსაც მომხმარებელი გამოიყენებს.

Motorclock.h:

- `Motorclock(int hour, int minutes, uint8_t servoP, uint8_t stepperN1P, uint8_t stepperN2P, uint8_t stepperN3P, uint8_t stepperN4P, uint8_t buzzerP, uint8_t button1P, uint8_t button2P, uint8_t rgbRP, uint8_t rgbGP, uint8_t rgbBP);` - აინიციალიზებს საათს, რომლის ჩართვის დრო გადაცემული საათისა და წუთის შესაბამისია. აღსანიშნავია, რომ საათი მუდმივ რეჟიმში არ ტრიალებს, მხოლოდ მაშინ, როცა ამას მომხმარებელი მოითხოვს.
- `String showTime();` - მოცემულმა ფუნქციამ უნდა დაატრიალოს საათის და წუთების ისრები და დააყენოს მიმდინარე დროის შესაბამისად. ასევე უნდა დააბრუნოს მნიშვნელობა `String` ფორმატში - „hour:minutes“.
- `void setTimer(double minutes);` - გადაეცემა `minutes`, პროგრამა იცდის აღნიშნული დროის განმავლობაში და დროის გასვლას მომხმარებელს buzzer-ის ხმის გამოყენებით ამცნობს. Buzzer-ის ხმა უნდა ისმოდეს მანამ, სანამ მომხმარებელი II ღილაკს არ დააწვება.
- `void setTimerByHand();` - მომხმარებელს საშუალება უნდა ჰქონდეს მისთვის სასურველი ტაიმერის დრო ხელით 2 ღილაკის დახმარებით დააყენოს. I ღილაკი დროის დიფერენცირებისთვის გამოიყენება - ღილაკზე 1 დანოლა = 1 წუთს. II ღილაკი აფიქსირებს შეყვანილ დროს. დროის ამოწურვის შემდეგ ამ შემთხვევაშიც buzzer გამოსცემს ხმას მანამ, სანამ მომხმარებელი II ღილაკს არ დააწვება. მაგალითად: თუკი მომხმარებელს უნდა 7 წუთიანი ტაიმერის დაყენება - აღნიშნული ბრძანების შემდეგ ის 7-ჯერ აწვება I ღილაკს, შემდეგ აწვება II ღილაკს, რომ დააფიქსიროს წუთების ოდენობა. II ღილაკზე დანოლიდან 7 წუთის შემდეგ buzzer გამოსცემს ხმას, მანამ სანამ კვლავ II ღილაკზე არ დააჭერს მომხმარებელი, რის შემდეგაც ბრძანება სრულდება.

- `void setAlarm(int hour, int minutes);` - ამ ბრძანების დახმარებით მომხმარებელი აყენებს მაღვიძარას, გადაეცემა შესაბამისი საათისა და წუთების მნიშვნელობა.
- `void setAlarmByHand();` - ეს ბრძანება `setTimerByHand();`-ის მსგავსია, თუმცა ამ შემთხვევაში საჭიროა საათების და წუთების დაფიქსირება. თავდაპირველად მომხმარებელს I ლილაკის დახმარებით შეჰყავს სასურველი საათი - ლილაკზე 1 დაწოლა = 1 საათს.(თუკი მომხმარებელი 12-ზე მეტჯერ დააწვება I ლილაკს საათმა თავიდან უნდა დაიწყოს თვლა.) შემდეგ აწვება II ლილაკს - აფიქსირებს საათს. საათის დაფიქსირების შემდეგ კვლავ I ლილაკის დახმარებით მომხმარებელს შეჰყავს სასურველი წუთების რაოდენობა - ლილაკზე 1 დაწოლა = 10 წუთს(თუკი მომხმარებელი 6-ზე მეტჯერ აწვება I ლილაკს საათმა თავიდან უნდა დაიწყოს თვლა).
- `void checkAlarm();` - თუკი მაღვიძარა არის დაყენებული და ამ ბრძანების შემოწმებისას მიმდინარე დრო მეტია ვიდრე მაღვიძარაზე დაყენებული დრო, buzzer-მა უნდა ამცნოს მომხმარებელს. ხმა უნდა გაითიშოს მაშინ, როცა მომხმარებელი II ლილაკს დააწვება. ამ ბრძანების გამოძახების შემდეგ, თუკი მომხმარებელმა ხელმეორედ არ დააყენა მაღვიძარა buzzer-მა ხმა აღარ უნდა გამოსცეს.
- `double setStopWatch();` - აყენებს წამზომს, რომელიც ითვლის დროს მანამ, სანამ მომხმარებელი II ლილაკზე არ დააჭერს ხელს. აბრუნებს შესაბამისი წუთების რაოდენობას. გარდა ამისა წუთების ისარი დაბრუნებული მნიშვნელობის შესაბამისად უნდა დატრიალდეს მარჯვნივ. მაგალითად თუ წამზომი 1 საათის განმავლობაში იყო დაყენებული, წუთების ისარმა 1 სრული ბრუნი უნდა გააკეთოს. აღსანიშნავია, რომ წამზომის ბრძანების დასრულების შემდეგ წუთების ისარი 1 წმ-ის განმავლობაში უნდა იყოს გაჩერებული, შემდეგ კი იმავე ნიშნულზე უნდა დადგეს, რომელზეც ბრძანების დაწყებამდე იდგა.
- `double setStopWatchByHand();` - წინა ბრძანების მსგავსია, იმ განსხვავებით, რომ წამზომის დაყენება I ლილაკით ხდება - პროგრამა იცდის, მანამ სანამ მომხმარებელი არ დააწვება I ლილაკს, შემდეგ კი იწყებს დროის ათვლას. II ლილაკზე დაჭერის შემდეგ აბრუნებს შესაბამის მნიშვნელობას და ატრიალებს წუთების ისარს.

Servo ძრავი:

- ამ ძრავმა უნდა აკონტროლოს საათების ისარი.
- საათების რაოდენობა 12-ია.
- სერვოს 0 გრადუსი ნიშნავს 0 საათს, 180 გრადუსი - 12 საათს.
- საათების ისრად შეგიძლიათ სერვოს მზა თავი გამოიყენოთ.

Stepper ძრავი:

- Stepper ძრავმა უნდა აკონტროლოს წუთების ისარი, რომელიც 0-60 დიაპაზონში მერყეობს.
- ჩართვისას ითვლება, რომ stepper ძრავის ისარი 0 წუთზე მოუთითებს.
- წუთების ისრად შეგიძლიათ გამოიყენოთ ნებისმიერი რამ.

Buzzer:

- გამოიყენება მაღვიძარას, ტაიმერის და წამზომის ფუნქციონალისთვის.

ლილაკები:

- ლილაკების რაოდენობა – 2.
- I ლილაკი გამოიყენება ხელით რეჟიმში მართვისას დროის დასაყენებლად.
- II ლილაკი გამოიყენება ხელით რეჟიმში მართვისას დაყენებული დროის დასაფიქსირებლად, ასევე buzzer-ის გამოსართავად.

RGB led:

- შექდიოდის ფუნქციაა მომხმარებელს ამცნოს ...ByHand() ტიპის ბრძანებებში შესვლის შესახებ.
- იმ შემთხვევაში თუ სრულდება setTimerByHand(); ბრძანება, უნდა აინთოს წითელი ფერი.
- იმ შემთხვევაში თუ სრულდება setAlarmBbyHand(); ბრძანება, უნდა აინთოს მწვანე ფერი.
- იმ შემთხვევაში თუ სრულდება setStopWatchByHand(); ბრძანება, უნდა აინთოს ლურჯი ფერი.