



**Escola Politècnica Superior
d'Enginyeria de Vilanova i la Geltrú**

UNIVERSITAT POLITÈCNICA DE CATALUNYA

TREBALL FINAL DE GRAU

**TÍTOL: DISSENY I CONTROL EN LENGUATGE C D'UN CUB DE RUBIK
ELECTRÒNIC BASAT EN LED**

AUTORS: SABIO MARTÍNEZ DE HUETE, ERIC

DATA DE PRESENTACIÓ: 30 de Maig, 2016

RESUM

El projecte tracta sobre el disseny i la creació de un cub electrònic que emula el famós joc clàssic del cub de Rubik. El usuari, usant un ordinador qualsevol, 27 sortides digitals d'una plataforma de hardware i un software controlat per un llenguatge de programació determinat, tot connectat entre sí i amb el cub, és capaç de manipular la il·luminació dels 54 LED (light emitter diode) que formen aquesta figura electrònica, a través de un teclat. D'aquesta manera, se li pregunta al usuari per la pantalla del ordinador i a través del programa en qüestió quin moviment del joc permès vol realitzar en cada ocasió i aquest, al executar-lo, provoca que els LED canviïn de color emulant els moviments mecànics del joc tradicional.

ABSTRACT

The project consist about the design and the creation of a electronic cube it emulate the classic famous game Rubik's cube. The user use a anyone computer, 27 digital output of a hardware platform and a software control with a programming language, and everything is interconnected with the cube, he is capable to manipulated the illumination of the 54 LED (light emitting diode) who form this electronic figure, through a keyboard. In this way, it question to the user through the computer screen and through this software what movement want to do and it's allow, and when the program run, this movement cause the change of the colours to the LED, emulating the mechanicals movements of the traditional game.

SUMARI

1. MOTIVACIONS	8
2. INTRODUCCIÓ	9
3. OBJECTIUS DEL PROJECTE.....	10
4. VIABILITAT DEL PROJECTE.....	11
4.1. COMUNICACIÓ ENTRE USUARI I LED'S	11
4.2. IL·LUMINACIÓ LED	11
5. FASE INICIAL: ESTUDI/SELECCIÓ MATERIAL BASE	12
5.1. HARDWARE.....	12
5.1.1. PLACA ARDUINO: MEGA 2560	12
5.1.2. LED RGB.....	13
5.2. SOFTWARE	14
5.2.1. PROGRAMA ARDUINO	14
5.2.2. LENGUATGE C.....	14
5.2.3. PROGRAMA KiCad.....	15
6. DISSENY DE L'ESQUEMA ELECTRÒNIC.....	16
6.1. PRIMER MODEL D'ESQUEMA ELECTRÒNIC	17
6.1.1. REGISTRE DE DESPLAÇAMENT 74HC595	17
6.1.2. EL TRANSISTOR COM A ELEMENT ACTIU	18
6.1.3. PROBLEMÀTICA DEL PRIMER MODEL.....	18
6.2. MODEL DEFINITIU.....	19
7. ELECCIÓ COLOR LED.....	20
8. CODI.....	21
8.1. PSEUDOCODI.....	22
8.2. IDE ARDUINO	22
9. NOMENCLATURA CUB DE RUBIK	23
10. MILLORES PROJECTE.....	25
CONCLUSIONS.....	26
AVALUACIO ECONÓMICA PROJECTE.....	27

BIBLIOGRAFIA	29
ANNEX.....	30

SUMARI DE FIGURES

FIGURA 1. PLACA ARDUINO MEGA 2560	12
FIGURA 2. ENCAPSULAT LED RGB	13
FIGURA 3. EXEMPLE TIPUS ENTORN SOFTWARE KICAD	15
FIGURA 4. BLOCS ESQUEMA ELECTRÒNIC	16
FIGURA 5. PLACA ARDUINO UNO	17
FIGURA 6. ENCAPSULAT REGISTRE DE DESPLAÇAMENT 74HC595	17
FIGURA 7. ENCAPSULAT I ESQUEMA TRANSISTOR 2N3904	18
FIGURA 8. DIAGRAMA DE FLUX CODI PROGRAMAT	21
FIGURA 9. EXEMPLE TIPUS ENTORN IDE ARDUINO	22
FIGURA 10. MOVIMENTS I NOMENCLATURES CUB DE RUBIK ORIGINAL	23
FIGURA 11. FRAGMENT ESQUEMA	30

SUMARI DE TAULES

TAULA 1. DESPESES DE MATERIAL DEL PROJECTE PER A CADA CUB CONSTRUÏT	27
TAULA 2. DESPESES DE MATERIAL DEL PROJECTE AL LLARG DEL ANY	27
TAULA 3. DESPESES DURANT EL PRIMER ANY TREBALLANT EN RÈGIM DE TREBALLADOR AUTÒNOM	27
TAULA 4. INGRESSOS ANUALS A REBRE PER A MANTENIR EL RÈGIM DE TREBALLADOR AUTÒNOM	28
TAULA 5. MÍNIMS HONORARIS A REBRE PER MANTENIR EL RÈGIM DE TREBALL I GENERAR INGRESSOS	28

1. MOTIVACIONS

Principalment, s'ha volgut realitzar un projecte basat en la creació d'un producte que estigui vinculat al món de l'oci i que pugui ser innovador en el mercat d'avui en dia. També el ús de elements com són la tecnologia LED i els altres components electrònics que formen part del disseny, així com el ús del llenguatge de programació informàtic C, han estat importants per a la elecció d'aquest projecte. El fet de poder posar en pràctica tota la teoria relacionada amb aquests components perquè funcionin de una forma correcta i sincronitzada, permeten adoptar solucions als problemes trobats al llarg d'aquest recorregut per finalment aconseguir la creació d'aquest producte, ha suposat tot un repte i un estímul a nivell personal.

2. INTRODUCCIÓ

El cub de Rubik és un trencaclosques tridimensional de dimensions 3x3x3 (alçada, llargada i profunditat) inventat al 1974 per el escultor, arquitecta i dissenyador hongarès Erno Rubik.

El joc consisteix en que, inicialment, cada cara del cub conté un color determinat, i aquest es format per un mecanisme de eixos que permet a cadascuna de les seves cares moure's de forma independent, provocant així la barreja dels diferents colors que el formen. L'objectiu del joc és desfer aquesta barreja i aconseguir que cada cara torni a estar formada per un únic color.

És considerada la joguina més venuda arreu del món, superant la xifra dels 350 milions d'exemplars. No és d'estranyar llavors que, en els últims anys, s'hagi volgut explotar aquest fenomen i crear diferents variants del cub més complexes i originals per estímul del consumidor.

Actualment, la tipologia és molt variable i aquests són alguns exemples: Diferents dimensions (2x2x2, 5x5x5, 9x9x9), diferents formats geomètrics (esfèrics, piramidals) o inclús formats electrònics.

Tenint en compte aquest últim format nombrat, s'ha partit de la idea ja creada d'un cub de Rubik electrònic que és il·luminat a partir de LED, per tal de dur-lo a un altre nivell i poder aconseguir diferències respecte els altres cubs electrònics ja existents.

3. OBJECTIUS DEL PROJECTE

S'han establert els objectius següents per a aquest projecte:

- Estudi de les variables i elements que intervenen en el projecte per tal de analitzar si aquest és realitzable o no.
- Selecció dels dispositius adients necessaris per tal de dissenyar, executar i controlar el cub.
- Realització de l'esquema electrònic capaç de dur la teoria cap a la pràctica.
- Creació del codi que permet el control de l'esquema.
- Construcció física del projecte.

4. VIABILITAT DEL PROJECTE

De bon inici, s'ha estudiat si aquest projecte és capaç de dur-se a terme. Aquest té que ser capaç de cobrir els aspectes fonamentals com són el fet de que l'usuari es pugui comunicar amb la màquina (PC), aquesta amb el cub (LED) i que aquests siguin capaços d'il·luminar de la manera adequada.

4.1. COMUNICACIÓ ENTRE USUARI I LED'S

En aquest punt, es sap que l'usuari es comunicarà amb l'ordinador a través d'un teclat principalment, però es necessita que les comandes que entri el jugador a través d'aquest puguin ser interpretades per el ordinador. No només això, sinó també fa falta que aquest, un cop feta la interpretació, es pugui comunicar immediatament amb els LED.

Per tal de poder fer aquesta arquitectura comunicativa, farà falta d'un programa (software) que entengui el que l'usuari escriu i envia, i d'un microcontrolador (hardware) compatible amb el programa que sigui capaç de poder gestionar aquestes ordres i actuar en conseqüència.

Afortunadament, avui en dia existeixen moltes plataformes electròniques en el mercat que ens permeten realitzar aquest procés d'una forma competent, tant a nivell tècnic com econòmic.

4.2. IL·LUMINACIÓ LED

Aquest component electrònic, els LED, és l'altre aspecte imprescindible a tenir present, ja que és el cor d'aquest projecte.

Es vol fer un cub que s'il·lumini, i que aquesta il·luminació estigui controlada, és a dir, que respongui a les peticions que l'usuari entri mitjançant les comandes pertinents. Com aquestes accions es fan a través d'un dispositiu electrònic (com el ordinador) el control de la llum té que ser de forma electrònica. Arribat a aquest punt, es sap que a dia d'avui els LED són un tipus de díode o, dit d'una altra manera, un tipus de component electrònic que respon als estímuls elèctrics per ser controlat, i és molt eficient, estètic, de baix consum i de llarga vida.

Sent per a tots aquests motius l'elecció per excel·lència en el disseny del projecte en qüestió, cal tenir en compte un altre aspecte.

El cub de Rubik a dissenyar, igual que el cub original, té que ser capaç de tenir diferents colors (sis concretament) per poder ser emulat i així poder jugar amb ell. Aquest aspecte obliga a disposar de LED que siguin capaços de produir aquesta diversitat de colors, cadascun d'ells.

També es pot cobrir aquest aspecte clau en el projecte, ja que a dia d'avui existeix una àmplia gamma sobre aquest producte que satisfà les necessitats del projecte.

5. FASE INICIAL: ESTUDI/SELECCIÓ MATERIAL BASE

Tot seguit, ja es saben els tipus d'elements principals que es necessiten per poder crear el cub de Rubik electrònic. Arribat a aquest punt, es tenen que seleccionar concretament quins i perquè s'usaran en el projecte .

5.1. HARDWARE

En quant a la part física d'aquest disseny, fent referència a la comunicació entre l'usuari i els LED, es necessita un microcontrolador acompanyat de la seva arquitectura pertinent, és a dir, una placa electrònica que es pugui adaptar i manipular des del PC. Juntament amb l'elecció dels LED oportuns, es té aquesta fase coberta.

5.1.1. PLACA ARDUINO: MEGA 2560

Tal com s'ha dit en apartats anteriors, existeixen multitud de plaques electròniques controladores per un microcontrolador en el mercat.

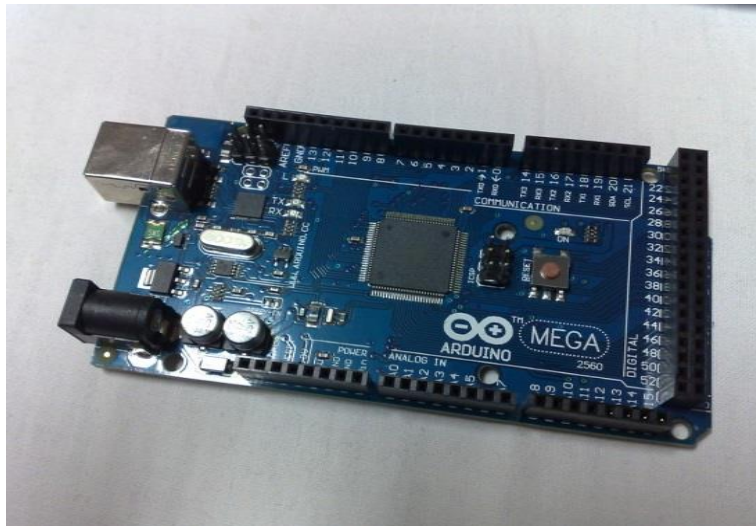


Figura 1. Placa Arduino Mega 2560

Per aquest projecte, s'ha seleccionat la placa Arduino, concretament el model MEGA 2560. En primera instància, s'havia escollit el model UNO, però finalment es va descartar per motius d'incompatibilitat amb el projecte, els quals s'explicaran més endavant en aquesta memòria en el seu apartat corresponent.

La selecció realitzada s'ha basat en tres aspectes fonamentals:

- a) Les plaques de la gamma Arduino són molt completes, tenint tots els components essencials per reproduir multitud de projectes diferents en l'àmbit de l'electrònica. Disposen de connexió USB que els hi proporciona l'alimentació i que permet que aquestes siguin manipulades a nivell de programació mitjançant qualsevol dispositiu electrònic que tingui aquest port d'entrada. També té multitud d'entrades i de sortides per els diferents perifèrics a connectar o connexions a realitzar i el microcontrolador que gestiona les dades de l'usuari.

- b) Aquesta placa disposa de 54 ports d'entrada/sortida, poden usar-se 27 d'aquestes connexions per els LED del cub.
- c) He treballat amb aquests tipus de plaques al llarg de la carrera, el que fa que hi estigui familiaritzat i pugui encarar amb més garanties aquest treball.

5.1.2. LED RGB

Es troba també en el terreny dels semiconductors, els LED, un ampli ventall de possibilitats. En aquest cas la selecció es basa, en primer lloc, en trobar un tipus que proporcioni, almenys, els sis colors diferents propis del cub de Rubik. Posteriorment, s'haurà de veure la demanda de corrent i tensions que aquests requereixen, per tal de ser compatibles amb la placa Arduino, és a dir, que aquesta pugui proporcionar l'essencial sense posar en risc la seva integritat.



Figura 2. Encapsulat LED RGB

Amb tot això, l'elecció ideal és el LED RGB. Es tracta d'un tipus concret que té encapsulat dins seu tres LED: el vermell (red), el verd (green) i el blau (blue), d'aquí el seu nom. Com que es tracta dels tres colors primaris, i aquests poden ser controlats electrònicament mitjançant el dispositiu en qüestió, es pot obrir i tancar el que sigui necessari en cada cas i així, obtenir qualsevol combinació possible de colors.

Com que la il·luminació que fa falta d'aquests LED en aquest projecte no requereix que sigui elevada, un cop triat aquest tipus de LED, només ha fet falta buscar un de baixa potència, és a dir, on la demanda de corrent i tensió sigui baixa, i així assegurar el bon funcionament del projecte.

5.2. SOFTWARE

La part no física del projecte es centra en un programa el qual controli la placa Arduino i, en conseqüència, el cub de Rubik. Per això, farà falta que aquesta aplicació sigui compatible amb la placa per tal de comunicar-se adequadament amb ella. Això també implicarà l'ús d'un llenguatge de programació propi d'aquesta arquitectura per poder aconseguir-ho.

Per altre banda, també s'ha fet ús de un altre programa, però aquest orientat al disseny del esquema electrònic que fa funcionar aquest projecte.

5.2.1. PROGRAMA ARDUINO

Referent a aquest apartat, cal dir que no ha calgut fer cap tria respecte al programa a usar, ja que el fet d'usar la placa electrònica Arduino implica de forma obligatòria usar una aplicació de programació de la mateixa gamma d'aquest producte.

De fet, aquesta ha estat una de les motivacions per tal d'emprendre aquest tipus de projecte, poder fer ús d'un entorn de programació que permetés utilitzar la programació apresada al llarg dels estudis, com és l'IDE (Entorn de Desenvolupament Integrat) Arduino. Aquest és concretament el programari emprat en el projecte.

5.2.2. LLENGUATGE C

Amb l'IDE d'Arduino s'aconsegueix el pas definitiu per poder comunicar l'usuari amb el cub on hi ha els LED col·locats, mitjançant la realització del codi adequat a través de l'ordinador, visualitzant-ho en la seva pantalla. Aquí és on es dissenya el "cervell" del projecte, totes les instruccions que tindrà que interpretar el microcontrolador, per tal de simular el joc del cub de Rubik en funció de les peticions entrades per teclat per l'usuari.

Per poder dissenyar aquest codi de control es necessita usar un llenguatge que l'Arduino entén, i aquest és el llenguatge C. Es tracta d'un llenguatge de baix nivell on un dels seus usos és la programació de microcontroladors. Per aquest motiu i per la compatibilitat ja esmentada, és l'escollit per a dur a terme l'objectiu marcat.

5.2.3. PROGRAMA KICAD

Per tal de plasmar d'una forma correcta les connexions dels diferents components que intervenen en el cub i que provoquen el seu funcionament, és necessari dissenyar un esquema electrònic el qual mostri el circuit en qüestió. A partir d'aquí, existeixen molts tipus de software que ens faciliten aquesta labor de creació. Un d'ells és el programa KiCad, el qual ha estat escollit per a realitzar aquesta labor.

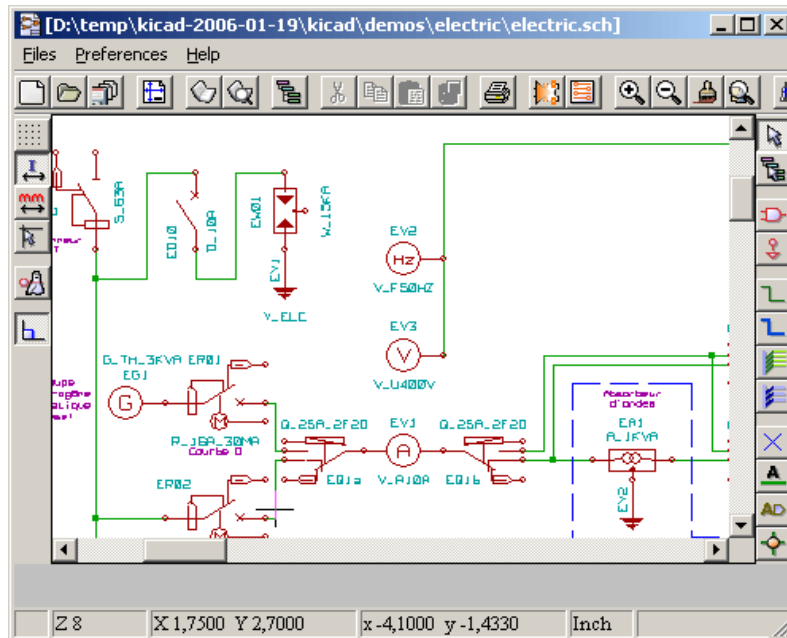


Figura 3. Exemple tipus entorn software KiCad

S'ha optat per aquesta opció per dos motius principals:

- Gratuïtat: és d'ús gratuït (no és necessari pagar cap llicència).
- Senzillesa d'ús: amb una corba d'aprenentatge ràpida que permet crear diversitat de circuits electrònics sense necessitat d'invertir molt temps.

6. DISSENY DE L'ESQUEMA ELECTRÒNIC

Per tal de poder controlar tots els LED del disseny, com que cadascun d'ells segueix un patró en particular, independentment del que realitzin els altres, farà falta un control individual. Això és degut a que cada LED pot tenir qualsevol dels sis colors dels quals disposa el cub, en qualsevol moment, en funció de les comandes entrades per l'usuari.

Totes aquestes raons han conduït a utilitzar un mètode eficaç per aquests casos, el qual estalvia moltes entrades digitals i el fa viable per ser usat per una placa amb microcontrolador. Aquest sistema consisteix en una matriu de LED.

Naturalment, això ha conduït a tenir que usar uns certs components electrònics com són les resistències o els transistors, per aconseguir fer viable el sistema esmentat.

Les resistències han complert la funció bàsicament de controlar les tensions i corrents del disseny, per adaptar-les a les necessitats dels LED, de la placa Arduino i també dels transistors.

Els transistors han desenvolupat un paper més fonamental i primordial que serà esmentat dins el següent apartat.

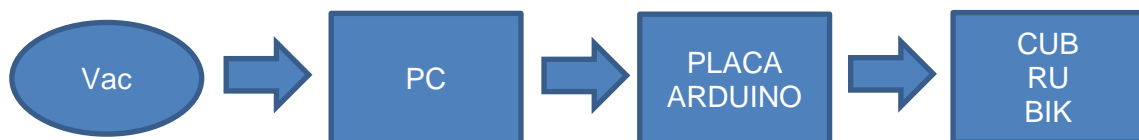


Figura 4. Blocs esquema electrònic

En definitiva, el projecte queda constituït per un conjunt de blocs que es podrien definir com l'entrada o alimentació que se li proporciona al PC, el qual s'uneix mitjançant el port USB a la placa del microcontrolador, la qual finalment s'uneix a tota l'electrònica del cub que habilita el funcionament dels LED.

6.1. PRIMER MODEL D'ESQUEMA ELECTRÒNIC



Figura 5. Placa Arduino UNO

En primera instància el disseny s'ha basat en un model el qual té com a protagonista la placa del microcontrolador Arduino UNO. La raó inicialment ha estat buscar la placa més barata possible dins d'aquesta gamma, i degut a que aquesta disposava de poques entrades digitals per al control dels 54 LED del disseny, ha estat necessari usar un circuit integrat per tal de que els LED i la placa s'hagin pogut acoblar entre ells.

6.1.1. REGISTRE DE DESPLAÇAMENT 74HC595

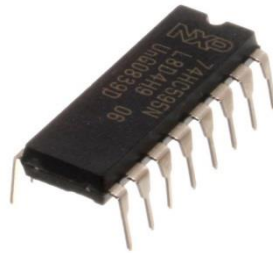


Figura 6. Encapsulat registre de desplaçament 74HC595

El circuit integrat 74HC595 és un tipus de registre de desplaçament que bàsicament serveix per augmentar el nombre de sortides de les plaques amb microcontroladors, en aquest cas l'Arduino. Mitjançant tres pins d'aquest, es poden tenir vuit connexions digitals de sortida per els LED. Es poden anar afegint més registres de desplaçament d'aquests en paral·lel, i així guanyar vuit sortides més per a cada nou xip que es posi.

6.1.2. EL TRANSISTOR COM A ELEMENT ACTIU

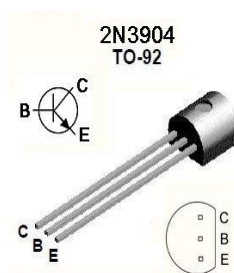


Figura 7. Encapsulat i esquema transistor 2N3904

Aquest dispositiu electrònic és un element semiconductor que ha actuat com un interruptor actiu per al control dels LED. Això significa que aquest deixa o no passar la corrent elèctrica a través seu en funció de les tensions a les quals està sotmès, a diferència d'un element passiu, que hauríem de pressionar directament sobre ell per aconseguir aquest efecte.

Aquest canvi es clau per al disseny, ja que vol dir que es pot sotmetre els transistors a les tensions que proporcioni la placa del microcontrolador, i com que aquesta, al mateix temps, està manipulada per les comandes que l'usuari introdueixi en el programa, significa que els transistors són manejats mitjançant la programació o la informàtica. En conseqüència, connectant aquests amb els LED, s'acaba controlant la il·luminació del cub de Rubik.

6.1.3. PROBLEMÀTICA DEL PRIMER MODEL

A l'hora de fer i provar aquest muntatge, han sorgit problemes importants que han fet que aquest model no hagi estat viable per a dur a terme el projecte amb bon funcionament.

El principal problema que té aquest model i que el fa inviable es que no hi ha un control correcte dels LED del cub. Segons les ordres donades per l'usuari, hi ha LED que s'obren quant això no és necessari, provocant una barreja de colors incorrecte que fa que sigui impossible poder jugar a aquest cub de Rubik, és a dir, poder resoldre'l. Posteriorment s'ha vist que l'error ha tingut que veure amb la col·locació d'alguns transistors en l'esquema, fet que s'ha arreglat més endavant amb la millora del model feta.

L'altre problemàtica que s'ha tingut que resoldre ha estat referent als registres de desplaçament i com aquests intervenen en aconseguir els colors desitjats dels LED. Per poder donar la sensació que tots els LED són encesos al mateix temps es necessita que la placa del microcontrolador proporcioni corrent un per un a molta velocitat i de forma regular. D'aquesta manera, l'ull humà és enganyat i té la sensació de veure tots els LED del cub encesos contínuament, quant en realitat la placa Arduino els està encenent un darrere l'altre de forma ordenada. Aquest fenomen s'anomena refresc. Això s'aconsegueix gràcies a un element de la placa que es un cristall de quars, el qual té la propietat de controlar intervals de temps regulars quant se li aplica una tensió entre els seus extrems, i així poder refrescar els LED quant és necessari. Posteriorment s'ha observat que el registre de desplaçament provoca errors en els colors obtinguts en el cub, el que es creu que és degut a que aquest té un temps de resposta força més lent que el temps de refresc que proporciona l'Arduino, ocasionant que aquests estiguin desfasats i obtinguem colors no desitjats. Aquest fet ha provocat que s'hagi descartat el 74HC595 del projecte i s'hagi buscat un nou camí per a fer un bon disseny del model.

6.2. MODEL DEFINITIU

En aquest nou esquema, cal ressenyar que els transistors connectats a les files de LED són controlats de forma individual, enlloc d'estar en paral·lel com en el model anterior. Això suposo un augment dels pins de connexió al microcontrolador, però és necessari per tal de corregir el problema del control dels LED. Amb la nova disposició, en el nou model s'han estalviat sis transistors en el muntatge, però en contrapartida han fet falta nou resistències més.

Com que els registres de desplaçament han estat descartats definitivament degut al problema ja explicat, la placa Arduino UNO no és capaç d'alimentar a tots els LED degut a que no té suficients ports d'entrada. Bàsicament, aquest ha estat el motiu pel qual s'ha tingut que optar per una placa amb un major nombre d'entrades i, ja de pas, també ha vingut acompanyat d'un microcontrolador una mica més potent. La placa electrònica ja comentada, l'Arduino Mega 2560, ha estat l'escollida per aquestes circumstàncies per tal de poder fer funcionar el definitiu model del projecte del cub de Rubik electrònic.

Tot i l'elecció d'un microcontrolador una mica més potent i amb moltes més entrades o ports electrònics, no ha estat possible treure el màxim benefici lumínic als LED. La hipòtesis és que, per tal de poder veure aquests permanentment encesos, és necessari una velocitat de refresc elevada, i això provoca que aquests no tinguin temps de rebre la intensitat elèctrica necessària per a lluir al potencial requerit. Això ha conduït a alguna problemàtica a l'hora de seleccionar els colors que formarien part del joc.

7. ELECCIÓ COLOR LED

Per tal que aquest projecte pugui simular de la millor forma possible el joc clàssic del cub de Rubik, ha estat necessari escollir el color de la il·luminació dels LED en funció de dos criteris principals: la disponibilitat i facilitat electrònica per poder crear-lo, i la distinció d'aquest respecte els altres.

Els tres colors primaris o principals (el vermell, el verd i el blau) que formen part de tots o quasi tots els cubs de Rubik del mercat són fàcils de reproduir. A nivell electrònic es regulen bé, ja que només és necessari tenir en compte un camí de corrent elèctrica el qual es controla amb facilitat amb una resistència i el transistor en qüestió, actuant com a element actiu. Aquests colors primaris queden perfectament definits i nítids a la vista, distingint-se clarament a nivell individual. L'elecció d'aquests ha estat fàcil per el projecte.

Els altres tres colors propis d'un cub de Rubik clàssic són el taronja, el groc i el blanc. El groc s'ha pogut reproduir bé a partir de la barreja del vermell i el verd, és a dir, utilitzant dues vies de corrent elèctrica o dos connexions diferents. El repartiment d'intensitat no ha estat equitatiu, ja que cal tenir en compte que per a fer barreges de colors partint dels tres colors elementals i que cadascun d'ells té una naturalesa lumínica diferent, ha estat necessari canalitzar més corrent elèctrica per al LED vermell que per el verd, ja que aquest últim és molt més intens i, per tant, en fa falta menys per aconseguir l'objectiu. Amb tot això, s'ha aconseguit un color groc amb tonalitat ataronjada ben definit i que es distingeix clarament de la resta. Aquest també ha estat escollit per al projecte.

El següent color seleccionat ha estat el magenta, que és la barreja del vermell i el blau. Un altre color fàcil de crear del estil del groc, és a dir, a partir de la barreja de dos colors primaris i per tant fàcils de fer a nivell electrònic que, a més a més, es distingeix clarament de la resta i també queda nítid i ben definit. Aquest color ha estat una elecció més atípica en substitució del taronja, el qual sí es més propi de trobar. L'elecció ha estat personal i és degut a que el taronja es molt més difícil de reproduir (no s'aconsegueix a partir de la barreja de dos colors primaris) en la forma electrònica, mentre que el magenta sí.

L'últim d'ells, el blanc, ha estat escollit en primera instància, ja que també és fàcil de fer (a partir de tres corrents diferents ben regulades). Per desgràcia, i en funció del que hem comentat al final del apartat superior, en que la velocitat de refresc de la placa obliga a no poder treure el màxim partit als LED, provoca que no es compleixi l'altre requisit esmentat necessari, que és aconseguir un color ben distingit de la resta. El color blanc brilla massa poc i, en conseqüència, acaba predominant el color vermell sobre la resta degut a la configuració feta per a l'obtenció dels altres cinc colors escollits i això, per desgràcia, crea un color visualment deficient que a més té certa semblança amb el magenta.

Per tots aquest inconvenients, s'ha acabat optant per el cian, que és la barreja dels colors primaris blau i verd. Si bé és veritat que tampoc és la millor elecció a fer si ho mirem amb un criteri plenament objectiu, ja que té semblança amb el blau. Tot i així, l'efecte visual o reproducció d'aquest és de més qualitat i s'acaba distingint millor de la resta que no pas el blanc respecte els altres.

8. CODI

Tota aquesta part representa el control del projecte, concretament el control del microcontrolador, el qual manipula el cub de Rubik creat a partir de les directrius que se li marquen mitjançant codificació en llenguatge C.

Per establir aquesta part ha estat necessari saber amb total certesa l'objectiu del projecte, és a dir, què té que fer. D'entrada això sembla evident, però s'ha tingut que analitzar cadascun dels passos a desenvolupar per el cub, per tal de poder estructurar de la millor forma possible tot aquest codi i que aquest realitzi totes les accions en l'ordre indicat i de la forma correcta. Aquest s'ha distribuït en funcions o, el que és el mateix, accions que té que desenvolupar el cub (canvi de color i refresc dels LED, moviment aleatori inicial, etc). D'aquesta manera, s'ha aconseguit que el codi quedi més ordenat i visible i, en conseqüència, amb més facilitat per la correcció d'errors que han anat sorgint durant el procés de creació i proves d'aquest. Això també ajuda als altres usuaris que no han intervingut en el projecte i que vulguin realitzar qualsevol modificació o que simplement vulguin entendre el codi. Per aplicar aquest últim fi, s'hi han afegit comentaris en quasi totes les línies de codi, fent aquest més entenedor.

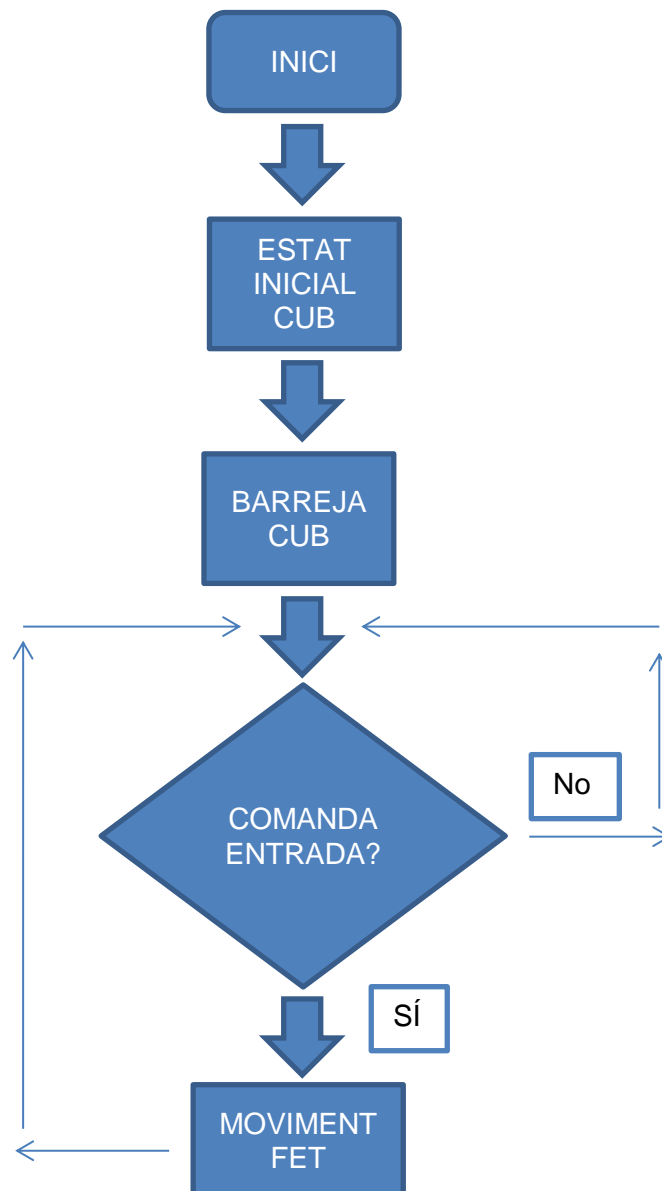


Figura 8. Diagrama de flux codi programat

En el apartat del annex (al final del document) s'hi troba el enllaç corresponent al codi programat.

8.1. PSEUDOCODI

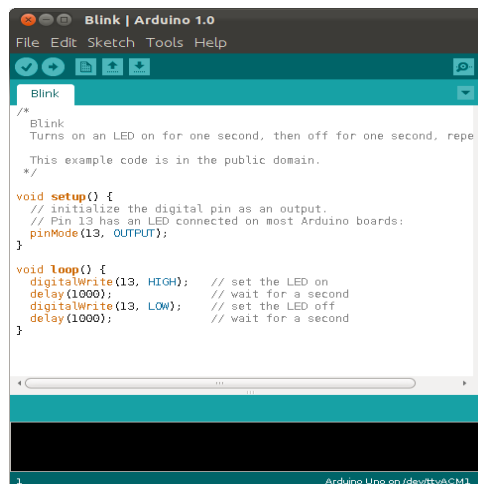
És important anomenar aquest punt, ja que ha estat el pas previ a la realització del codi definitiu.

El pseudocodi és un tipus de descripció del programa en un llenguatge que sigui entenedor per les persones, consistint en explicar pas a pas el codi de programació a realitzar. Amb això s'aconsegueix que l'usuari que vol crear un programa, per tal de poder fer aquest de la forma adient, pugui plasmar les idees d'aquest mitjançant aquest tipus de codificació. Un cop aquest és definitiu, el qual té que poder ser entès per a qualsevol persona sense coneixements de programació, el pas següent es traduir-lo a codi màquina (en aquest cas llenguatge C) per tal que el ordinador compregui què vol fer l'usuari i així ho pugui executar.

Aquest ha estat el procés seguit per poder elaborar el codi d'aquest cub de Rubik.

8.2. IDE ARDUINO

Tal com s'ha anomenat en apartats anteriors, el codi del programa que és capaç de crear els algorismes adequats per fer funcionar el projecte elaborat s'ha pogut executar gràcies a l'entorn de desenvolupament IDE Arduino.

The image shows a screenshot of the Arduino IDE interface. The title bar reads "Blink | Arduino 1.0". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for opening, saving, and running. The main text area contains the following code:

```
/*  
 * Blink  
 * Turns on an LED on for one second, then off for one second, repe  
 * This example code is in the public domain.  
 */  
  
void setup() {  
  // initialize the digital pin as an output.  
  // Pin 13 has an LED connected on most Arduino boards:  
  pinMode(13, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(13, HIGH); // set the LED on  
  delay(1000);           // wait for a second  
  digitalWrite(13, LOW); // set the LED off  
  delay(1000);           // wait for a second  
}
```

The status bar at the bottom indicates "Arduino Uno on /dev/ttyACM1".

Figura 9. Exemple tipus entorn IDE Arduino

Aquest tipus de software és fonamental per a l'elaboració de codi de programació. Té com a objectiu poder compilar (processar el codi) del programa que s'elabora i verificar si aquest té algun error de codificació. A continuació, quant aquest és correcte, permet passar les dades del programa a la placa del microcontrolador Arduino per tal de que aquest executi el codi programat.

9. NOMENCLATURA CUB DE RUBIK

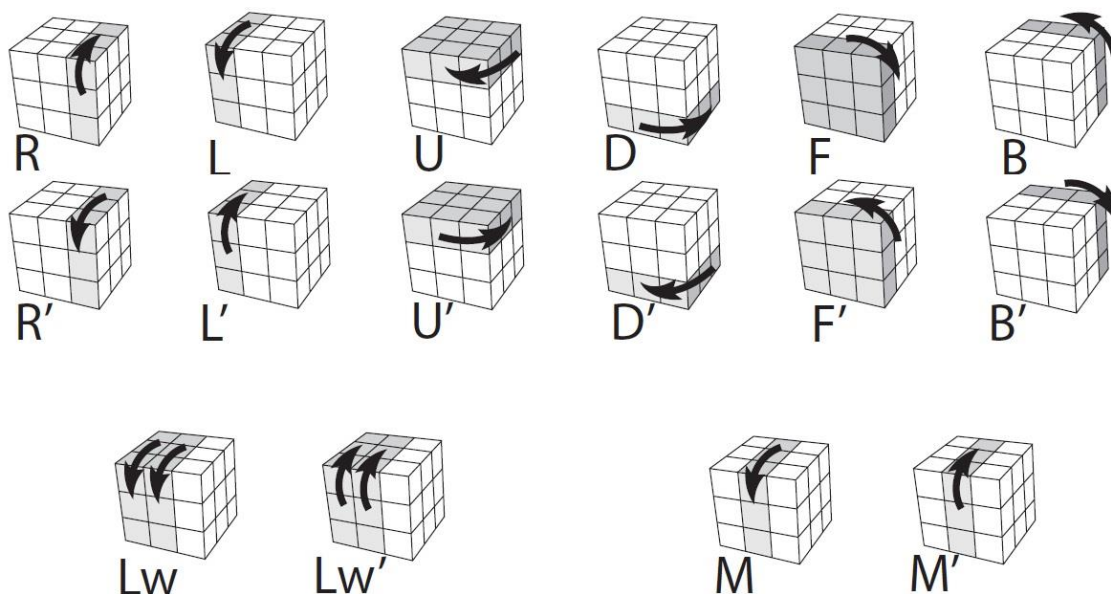


Figura 10. Moviments i nomenclatures cub de Rubik original

Els moviments del cub creat en aquest projecte són possibles gràcies a que l'usuari es comunica amb aquest a través de l'entorn Arduino i, més concretament, a través del port sèrie del que aquest disposa que s'uneix al PC mitjançant una connexió USB. Tenint en compte això i fent ús d'aquest entorn que genera el port sèrie, per facilitar el joc i que resulti familiar a usuaris habituats a experimentar amb aquest, la majoria de moviments es basen en la nomenclatura original (tal com es veu a la figura 9).

S'ha cregut oportú no incloure moviments dobles com són el Lw i Lw', i en canvi incloure dos nous moviments com són el MR i el ML. Ha estat una elecció personal, creient que d'aquesta manera el usuari té més facilitat a la hora de familiaritzar-se i fer ús de les diferents nomenclatures, ja que d'aquesta manera totes elles tan sols constitueixen un únic moviment al mateix temps. Les eliminades representant dos moviments bàsics de forma simultània, creient que això pot representar més confusió al usuari a la hora de resoldre el cub, principalment al novell.

La nomenclatura original fa ús de l'apòstrof (') per designar aquells moviments que van en contra les agulles del rellotge. En el cas del projecte realitzat, s'ha canviat aquesta part per un signe negatiu (-) i els moviments originals que van en el sentit de les agulles del rellotge se'ls hi ha afegit el signe positiu (+).

El criteri per a la realització d'aquests canvis s'ha basat en que el signe de l'apòstrof resulta confús per a l'entorn IDE Arduino quan programem en llenguatge C, ja que és un element reservat en aquest llenguatge i no es pot usar així com així en qualsevol circumstància. Per aquest motiu s'ha fet el canvi esmentat en els moviments negatius, i en conseqüència, també ha estat oportú fer el canvi en els moviments positius per tenir més facilitat a l'hora de programar.

Cal esmentar el fet d'haver introduït també comandes que tenen la capacitat de canviar l'orientació del cub de Rubik, a partir dels seus eixos de coordenades. S'ha cregut que era oportú afegir aquests moviments, tenint en compte que aquest cub no està pensat per a ser manipulat amb les mans.

Aquests són els significats de tots els moviments inclosos en aquest projecte:

- | | |
|---|--|
| R+ → Right (dreta) sentit positiu | R- → Right (dreta) sentit negatiu |
| L+ → Left (esquerra) sentit positiu | L- → Left (esquerra) sentit negatiu |
| U+ → Up (amunt) sentit positiu | U- → Up (amunt) sentit negatiu |
| D+ → Down (abaix) sentit positiu | D- → Down (abaix) sentit negatiu |
| F+ → Front (frontal) sentit positiu | F- → Front (frontal) sentit negatiu |
| B+ → Back (esquena) sentit positiu | B- → Back (esquena) sentit negatiu |
| M+ → Middle (mig) sentit positiu | M- → Middle (mig) sentit negatiu |
| ML → Middle Left (mig esquerra) | MR → Middle Right (mig dreta) |
| X+ → Moviment respecte coordenada "x" sentit positiu | |
| X- → Moviment respecte coordenada "x" sentit negatiu | |
| Y+ → Moviment respecte coordenada "y" sentit positiu | |
| Y- → Moviment respecte coordenada "y" sentit negatiu | |
| Z+ → Moviment respecte coordenada "z" sentit positiu | |
| Z- → Moviment respecte coordenada "z" sentit negatiu | |

10. MILLORES PROJECTE

Cal tenir present que, durant tot el procés de disseny i muntatge d'aquest projecte, s'han anat observant aspectes que serien millorables, els quals caldria tenir presents en cas de fer un segon prototip que superés a l'anterior, és a dir, al realitzat.

Un d'ells seria les dimensions del cub, clarament per sobre del cub clàssic. Tot i que aquest aspecte no és fonamental, ja que l'usuari no té que manipular-lo directament, això ens suposaria un avantatge tant en estètica com en espai. Una manera d'optimitzar l'espai seria fer el disseny de les plaques electròniques del cub a partir de circuits impresos. Suposaria un estalvi de material (dimensions placa i metall fos per a fer les soldadures) i com que el disseny es faria a partir d'un programari específic, si es domina, suposaria també un estalvi de temps.

Un altre aspecte és el fet d'haver soldat directament els LED sobre la placa. Aquest fet no interfereix pas en el funcionament d'aquests però, en cas que algun falli per la raó que sigui (error de fàbrica, curtcircuit provocat o involuntari, etc), es fa impossible dessoldar-lo, tenint que buscar alternatives complicades per el canvi del LED espatllat i poc estètiques en quant al disseny del cub es refereix. Una possible solució seria col·locar els LED en sòcols adaptats al seu encapsulat, sent els sòcols els quals anirien soldats a la placa. D'aquesta manera, en cas que un fallés, el podríem treure fàcilment i canviar-lo per un altre en perfectes condicions.

L'últim aspecte vist és el fet que els components han estat soldats per la cara visible del cub, els quals no suposen elements gaire estètics per l'usuari en el disseny del cub, amb cert risc de que puguin fallar abans de temps pel fet d'estar molt més exposats a possibles caigudes d'aquest i també a la intempèrie. La solució per millorar aquest fet seria soldar els components per la cara interna del cub mitjançant determinades tècniques de muntatge o habilitats en concret, o utilitzant tipus d'encapsulats diferents per els components que formen part del cub.

Introduir memòria al cub per recordar l'última posició realitzada durant una partida, és un altre aspecte important a millorar ja que si, per la raó que sigui, l'usuari que està jugant amb aquest cub de Rubik no pot resoldre el cub en aquest moment i vol continuar més tard, això obliga a deixar-lo encès i per tant a consumir electricitat per tal de no perdre els avenços fets.

També es pot fer alguna millora en la gestió que cal fer a l'hora de poder obrir i tancar el port sèrie del software Arduino per on s'introdueixen les comandes de moviment del cub, sense necessitat de provocar que aquest es reinici.

El port sèrie és un aspecte clau del projecte, ja que mitjançant el software, podem accedir a la part del programa (se'ns obre una finestra al seleccionar aquesta opció) on se li demana l'usuari que introdueixi les comandes de moviment del cub.

Per la manera en que la placa del microcontrolador està configurada, cada cop que s'obre la finestra del port sèrie el programa sencer es reinicia, o el que és el mateix, el cub de Rubik torna al punt de partida inicial. Això obliga a que un cop s'obre el port sèrie aquest estigui obert durant tota la partida, la qual cosa a priori no sembla un problema, però en cas de no poder completar-la o tancar la finestra del port per accident, es perden de nou tots els avenços fets i s'ha de tornar a començar.

CONCLUSIONS

Després de l'anàlisi dels resultats obtinguts es pot afirmar que:

- S'ha confirmat la viabilitat del projecte, amb el seu anàlisi previ, per tal de poder-lo dur a terme.
- S'han seleccionat de manera adequada els components (placa del microcontrolador, elements electrònics, etc) que ha fet possible la posterior construcció i funcionament del cub.
- S'ha dissenyat de manera òptima i simple un esquema electrònic basat en una matriu de LED capaços d'actuar en funció de les directrius adequades.
- S'ha realitzat un codi basat en el llenguatge C de programació a partir del software Arduino que permet el control i la manipulació del cub.
- S'ha construït un cub de Rubik electrònic amb LED que presenta variacions o modificacions en el seu aspecte respecte de l'original, però suficient per a complir amb l'objectiu marcat.

AVALUACIO ECONÓMICA PROJECTE

S'ha fet un estudi de l'impacte econòmic que ha suposat el projecte i el que suposaria treballar per compte propi (en règim de treballador autònom) realitzant aquest tipus de producte.

Primerament, s'han calculat les despeses de material que suposa el fet de construir un cub de Rubik d'aquestes característiques:

Despeses material (per cub)	
54 LED RGB	1,6€
27 transistors 2N3904	2,16€
42 resistències	0,84€
Placa Arduino MEGA 2560	47,19€
129 connectors femella	15,73€
6 metres cable flexible	5,91€
6 plaques baquelita (78x90mm)	19,38€
Rotllo aliatge estany sense plom	13,92€
TOTAL	106,73€

Taula 1. Despeses de material del projecte per a cada cub construït

El temps de realització per un projecte d'aquesta envergadura seria, com a molt, de dos mesos. Un cop creat el primer prototip, els temes del disseny del cub (esquema, programa) i les directrius a seguir són força clares. Això fa que per a dissenyar-lo de nou, els temps per a la creació d'aquest es basi únicament en el muntatge i les proves de funcionament, reduint-se de forma notable. En definitiva, això fa que s'arribi a la conclusió que en un any, tenint en compte una producció en paral·lel i sistemàtica, el nombre de cubs que es podrien realitzar seria aproximadament uns 200 (25 per mes). Aquests serien els costos anuals a tenir en compte:

Despeses material (anual)	
Despeses de material x 300cubs	32019€

Taula 2. Despeses de material del projecte al llarg del any

A més a més d'això, s'ha de tenir present que el fet de treballar com a autònom per a d'altres empreses suposa també una sèrie de despeses a realitzar. Així doncs, en total es gastaria al llarg del primer any:

Despeses primer any règim treballador autònom	
Quota autònoms base mínima cotització	600€
Gestoria	360€
Creació pàgina web	700€
IRPF	250,07€
Despeses de material del projecte (anual)	32019€
TOTAL	33929,07€

Taula 3. Despeses durant el primer any treballant en règim de treballador autònom

Disseny i control en llenguatge C d'un cub de Rubik electrònic basat en LED
SABIO MARTÍNEZ DE HUETE, ERIC

No només s'han de tenir en compte les despeses, i és que com a treballador autònom és necessari poder arribar a tenir un mínim determinat d'ingressos per poder arribar a cotitzar un mínim a la seguretat social i, així, mantenir aquest règim de treball. Tenint que vigilar aquests aspectes i al fet de les despeses calculades, aquests són els ingressos que es necessiten percebre al llarg del primer any:

Ingressos mantenir règim de treball (anual)	
Ingressos mínims per cotitzar (mensual)	893,10€
893,10€ x 12mesos	10717,20€
10717,20€ + Despeses primer any	44646,27€

Taula 4. Ingressos anuals a rebre per a mantenir el règim de treballador autònom

Definitivament, i tenint en compte els ingressos necessaris calculats, aquests serien els honoraris mínims a percebre com a enginyer per poder exercir en aquest règim de treball i generar beneficis, si ens basem en que exercim la professió un mínim de 160 hores al mes (temps mitjà d'un contracte laboral estàndard):

Mínims honoraris mantenir règim de treball (per hora)	
Ingressos a rebre/12mesos/4semanes/5dies/8hores	23,25€/h

Taula 5. Mínims honoraris a rebre per mantenir el règim de treball i generar ingressos

BIBLIOGRAFIA

Pàgines web consultades per a l'elaboració del projecte:

www.Rubikaz.com

www.geekytheory.com

www.LEDfacil.com.ar

www.electrocirc.blogspot.com.es

www.panamahitek.com

www.circuitoselectronicos.org

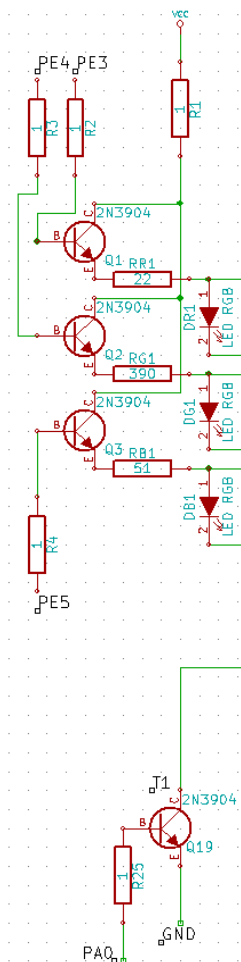
www.datasheetcatalog.com

www.alldatasheet.com

www.blog.bricogeek.com

ANNEX

CÀLCULS ESQUEMA



V_{CEsat} → Tensió saturació col·lector – emissor transistor

V_{BEsat} → Tensió saturació base – emissor transistor

I_C → Corrent col·lector transistor

I_B → Corrent base transistor

Tot seguit es mostren els mètodes i fórmules matemàtiques que s'han tingut en compte a nivell teòric per a la creació de l'esquema electrònic.

El càlcul es basa en un petit fragment de la matriu de LED, ja que un cop es sap aquest la resta segueix el mateix principi.

A partir de les fulles tècniques (*datasheets*) del transistor 2N3904 i el LED RGB, s'han agafat aquestes valors com a requisits per al bon funcionament del circuit:

$$V_{CEsat} = 0.1V \quad V_{BEsat} = 0.7V \quad I_B = 2mA$$

$$I_C = 20mA \quad D_{R1} = 1.85V \quad D_{G1} = D_{B1} = 3.5V$$

$$R_1 = \frac{V_{R1}}{I_C} = \frac{5-4.2}{20 \cdot 10^{-3}} = 40\Omega \quad R_2 = \frac{V_{R2}}{I_B} = \frac{5-4.8}{2 \cdot 10^{-3}} = 100\Omega$$

$$R_{R1} = \frac{V_{R1}}{I_C} = \frac{4.1-1.95}{20 \cdot 10^{-3}} = 107.5\Omega \quad R_{25} = \frac{V_{R25}}{I_B} = \frac{5-0.7}{2 \cdot 10^{-3}} = 2.15k\Omega$$

$$R_{G1} = R_{B1} = \frac{V_{R_{G1}}}{I_C} = \frac{4.1-3.6}{20 \cdot 10^{-3}} = 25\Omega$$

Figura 11. Fragment esquema

Degut a que en el mercat el valor de les resistències és limitat, a la pràctica s'han d'adaptar els valors teòrics als comercials:

$$R_1 = 39\Omega \quad R_2 = R_3 = R_4 = 100\Omega \quad R_{R1} = 100\Omega \quad R_{G1} = R_{B1} = 22\Omega \quad R_{25} = 2.2k\Omega$$

Tot i així, a l'hora de fer totes les connexions, tot el muntatge i fer les proves pertinents, s'ha vist que els LED brillaven amb molt poca intensitat amb aquests valors de resistències.

Per tant, s'ha partit d'aquests valors sobre el paper, però després d'anar experimentant, s'ha vist que aquests no eren correctes per tal d'aconseguir la llum i els colors correctes dels LED.

Finalment, aquests han estat seleccionats d'una forma empírica:

$$\boxed{R_1 = R_2 = R_3 = R_4 = R_{25} = 1\Omega} \quad \boxed{R_{R1} = 22\Omega} \quad \boxed{R_{G1} = 390\Omega} \quad \boxed{R_{B1} = 51\Omega}$$

També s'ha de tenir en compte el temps de refresc dels LED, el qual ha fet falta calcular per tal de que aquests estiguin il·luminats de la forma adequada. Per tal de poder fer aquest càlcul de forma teòrica, s'ha tingut en compte com a base la freqüència de temps dels fotogrames en les pel·lícules, el qual es de 24 imatges per segon, és a dir, 24Hz. També tenint en compte que per generar els colors vermell, verd i blau fa falta obrir 9 LED per cadascun d'ells siguin la cara que siguin, es a dir, 27 LED en total (ja que son colors simples), i per generar els colors groc, magenta i cian es necessari el doble de LED encesos (ja que són colors complexos formats a partir de la unió de 2 colors) i per tant ens fan falta 54 LED. En total tindrem 81 LED funcionant contínuament.

Tenint present la dada de la cadència i els 81 LED encesos, la freqüència de refresc total i teòrica és igual a:

$$f_{refresc} = 24Hz \cdot 81 \text{ led's} = 1944Hz$$

Llavors, el temps de refresc teòric en que haurem de encendre cada LED serà igual a :

$$T_{refresc} = \frac{1}{1944Hz} = 5.14 \cdot 10^{-4}s = 0.51 \cdot 10^{-3}s = 0.51ms$$

Finalment, però, després de realitzar diferents probes de forma empírica, s'ha vist que aquest temps era insuficient i s'ha tingut que anar disminuint, és a dir, augmentant la velocitat de commutació o de refresc, fins arribar a la xifra definitiva i correcta per el funcionament de:

$$T_{refresc} = 0.21ms$$

Això vol dir que cada LED és encès aproximadament cada quart de segon (una mica menys) per tal de donar l'efecte al ull humà de que aquest i tots els altres mai són apagats i romanen il·luminats durant tota l'estona en que el cub es manté en funcionament.

Amb aquesta dada, ja es pot configurar la part del codi del projecte referent al temps (en la plataforma Arduino, aquesta part de l'aplicació que controla el cristall de quars s'anomena timer). Es pot observar en detall en el annex on apareix tot el codi del projecte, anomenat Codi C del cub.

Disseny i control en llenguatge C d'un cub de Rubik electrònic basat en LED
SABIO MARTÍNEZ DE HUETE, ERIC

Aquí s'han afegit els següents enllaços als annexes corresponents a les dades tècniques dels components usats i el codi en llenguatge C del projecte:

- [Plaques Arduino](#)
- [Microcontrolador ATMEGA 2560](#)
- [Transistor 2N3904](#)
- [LED RGB](#)
- [Codi C del cub](#)

L'esquema del disseny electrònic ha estat realitzat mitjançant el software Kicad, tal com s'ha dit en apartats anteriors. Això genera un arxiu del circuit que indexat des d'aquesta pàgina no és capaç d'obrir el programa pertinent. Aquest ha estat el motiu pel qual no ha estat incorporat per aquí el seu enllaç.

