**Soluție conducere quadcopter cu Arduino**

**Capitolul 2. Echipamente utilizate**

**2.1 Echipamente hardware**

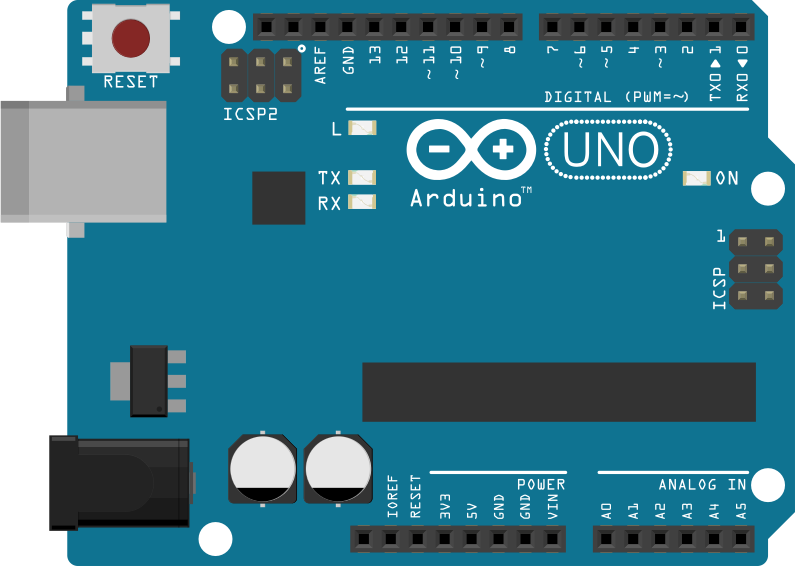
**2.1.1 Placa de dezvoltare Arduino Uno**

Arduino este o companie și comunitate ce produce hardware și software open-source folosite în special pentru construirea obiectelor interactive. Proiectul este bazat pe designul de plăci cu microcontroller ce ofera seturi de pini digitali și analog ce pot interfața anumite plăci de extensie, numite shields sau alte circuite. Plăcile includ interfețe de comunicare serială pentru scrierea de programe cu ajutorul computerelor.

Arduino Uno este o placa ce folosește un microcontroller ATmega328P. Este alcătuita din:

* 14 pini de intrare ieșire digitali din care 6 pot fi folosiți ca și iesiri PWM;
* 6 pini de intrare analog;
* un cristal de quart de 16 MHz;
* o mufă USB;
* o mufă de alimentare jack;
* un buton de reset;
* un header ICSP;

Placa de dezvoltare Arduino Uno conține tot ce e nevoie pentru a începe designul de obiecte interactive, doar trebuie conectată la un computer sau la un acumulator.



|  |  |
| --- | --- |
| Microcontroller | ATmega328P |
| Voltaj de operare | 5 V |
| Voltaj de intrare (recomandat) | 7-12 V |
| Voltaj de intrare (limita) | 6-20 V |
| Pini digitali intrare / ieșire | 14 |
| Pini PWM | 6 |
| Pini intrare analog | 6 |
| Curent continuu pini intrare / ieșire | 20 mA |
| Curent continuu pentru pin de 3,3 V | 50 mA |
| Memorie flash | 32 KB |
| SRAM | 2 KB |
| EEPROM | 1 KB |
| Frecvența de tact | 16 MHz |
| Lungime | 68,6 mm |
| Lațime | 53,4 mm |
| Greutate | 25 g |

Specificații tehnice:

In contexul pinilor de intrare / ieșire , fiecare din cei 14 pini digitali poate fi folosit ca pin de intrare sau ieșire folosind functiile pinMode(), digitalWrite() și digitalRead(). Opereaza la 5 volți.

In plus unii pini au funcții specializate:

* Serial: 0 (RX) și 1 (TX). Folosiți pentru a primii și a transmite date cu ajutorul comunicarii seriale.
* Intreruperi externe: 2 și 3. Acești pini pot fi configurați sa declanșeze intreruperi de nivel jos.
* PWM: 3, 5, 6, 9, 10 și 11. Oferă ieșire de PWM 8 biți.
* SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Acești pini suportă comunicare SPI folosind libraria SPI.
* LED: 13. Este un LED ce poate fi pornit setându-l pe HIGH.
* TWI: A4 sau SDA pin și A5 sau SCL pin. Suporta comunicare TWI folosind libraria Wire.

Are în plus 6 intrari analog, etichetate de la A0 la A5, fiecare oferind o rezolutie de 10 biți.

Pentru comunicarea serială se folosește UART. Universal asynchronous receiver/transmitter este un dispozitiv hardware ce traduce date din paralel in serial. UART-ul este folosit de obicei împreuna cu standarde de comunicare cum ar fi TIA, RS-232, RS-422 , RS-485. Designul universal arată că formatul de date transmis și viteza de transmitere sunt configurabile. Nivelele de semnal electric și metodele (ca diferențele de semnal) sunt gestionate de un circuit driver exterior UART-ului.

Protocolul de UART este de obicei parte dintr-un circuit integrat folosit pentru comunicare serială cu un computer sau un periferic. Cel mai comun sunt incluse in microcontrollere.

**Transmiterea și primirea datelor**

UART-ul primește byți de date și transmite biți individuali in mod secvențial. La destinație un al doilea UART reasambleaza biții in bytes compleți. Fiecare UART conține un registru de shiftare, care este metoda fundamentală de conversie intre formele serial și paralel. Transmiterea serială a informațiilor digitale printr-un singru cablu este mai ieftină decât transmiterea paralela prin mai multe cabluri.

UART-ul de obicei nu genereaza sau primește direct semnalele externe folosite între diferite echipamente. Interfețe separate ale dispozitivelor sunt folosite pentru a transforma semnalele UART din și in semnale externe. Semnalele externe pot fi de diferite forme , standardele sunt RS-232, RS-422 , RS-485. Unele scheme de semnal nu folosesc cabluri electrice, exemple sunt, fibra optica, infraroșu, bluetooth.

**Data Framing**

Starea de așteptare este de voltaj ridicat sau pornit din motive istorice din telegrafie, cand linia era ținută pe voltaj ridicat cand nu se trimiteau informatii, pentru a arata ca linia și transmițătorul nu sunt defecte. Fiecare caracter este transmis ca un bit de start logic scazut, un număr configurabil de biți de date și opțional un bit de paritate.

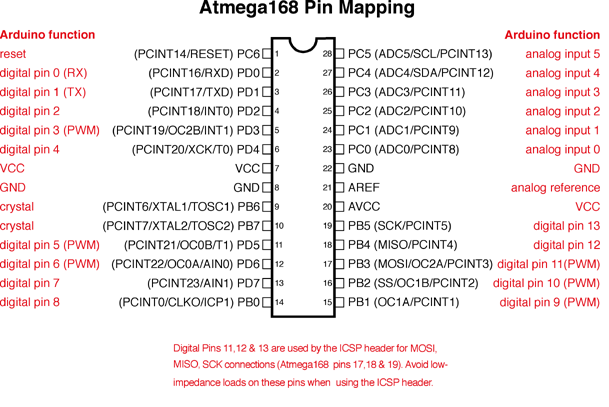
Bitul de start semnalează receptorului că un nou caracter urmează să fie trimis. Următorii nouă biți, reprezintă datele. Dacă se folosește un bit de paritate, va fi așezat dupa terminarea transmiterii biților de date. Următorii unu sau doi biți sunt folosiți ca biți de stop, arată receptorului că transmiterea datelor este completă.

**Receiver**

Toate operațiile efectuate de UART sunt controlate de un semnal de tact care rulează la multiplul ratei de transmitere a datelor, de obicei de 8 ori rata de transmitere. Receptorul testează starea semnalului primit la fiecare puls de tact, căutand începutul bitului de start. Daca bitul de start rezistă cel puțin jumătate din durata unui timp de bit, este valid și semnalează inceperea transmiterii datelor. Dacă nu, este considerat ca puls fals și este ignorat. Dupa așteptarea timpului specificat, starea liniei este eșantionată din nou și rezultatul este trecut in registrul de shiftare. După ce au venit toți biții de date conținutul registrului de shiftare este pus la dispoziție receptorului. UART-ul va seta un flag ce va indica valabilitatea unor date noi și va genera o intrerupere pentru a cere procesorului transferul datelor primite.

Uart-urile de comunicare nu au de obice un sistem de timing care să îl împartă. De obicei au două generatoare de tact diferite ce se sincronizeaza la fiecare transmitere de date ce nu este falsă. Obținerea informațiilor de timing în aceasta maniera folosește o linie mai puțin de semnal.

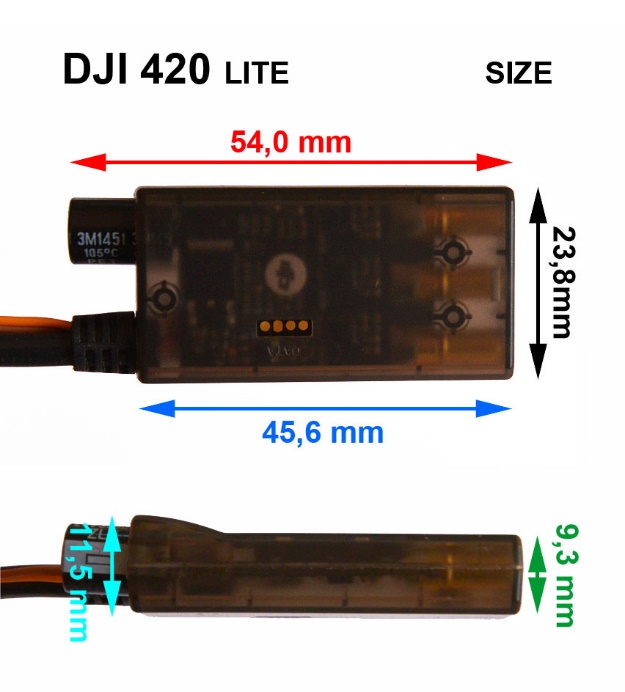
Este un feature standard al UART-ului să stocheze cel mai recent caracter primit până îl primește pe următorul. Acest dublu buffer oferă computerului receptor durata transmisiei unui caracter intreg pentru a lua caracterul primit. Multe UART-uri au o mica memorie de First In First Out între receptor și host, asta permite procesorului hot mai mult timp să manipuleze întreruperea venită de UART și previne pierderea de date.



**2.1.2 Regulatorul electronic de viteză E420 lite**

Un regulator de viteză electronic este un circuit electronic ce are scopul de a varia viteza unui motor electric, de a îi controla direcția și de a îl frâna dinamic. Sunt des folosite pentru drone sau alte dispozitive controlate electronic, cel mai des folosite pentru motoare brushless, esențial punând la dispoziție un semnal electric tri-fazic de voltaj scăzut .

Un ESC poate fi conectat la un receptor radio pentru controlul accelerației sau chiar încorporat în receptor .



**Functionalitate**

Indiferent de tipul folosit un ESC interpreteaza informația de control, și o transforma intr-un mod care variază starea tranzistorilor care controlează un câmp electric. Variația rapidă a stării tranzistorilor este ceea ce cauzează motorului să scoată un piuit ce poate fi foarte bine observat la viteze scăzute. Permite ajustarea mult mai fina și precisă a vitezei motorului intr-o manieră mult mai eficienta decât cea mecanica ce folosește bobina rezistivă.

Cele mai multe ESC-uri moderne incorporeaza un circuit eliminator cu baterie ce reguleaza voltajul pentru receptor.

ESC-urile de obicei acceptă un semnal PWM de 50HZ al cărui puls variază de la 1 la 2 ms. Când primește puls de 1 milisecundă, ESC-ul răspunde prin oprirea motorului, pe când un puls de 2 ms invârte motorul la viteză maxima.

**Clasificare**

De obicei ESC-urile sunt clasificate în concordanță cu maximul de amperi, de exemplu 25A. Cu cât este mai mare numărul de amperi, cu atât mai mare și mai greu este dispozitivul. Multe ESC-uri moderne conțin o baterie cu o rază de intrare și oprire de voltaj. Tipul bateriei și numărul de celule este un considerent important în alegerea BEC-ului (battery eliminator circuit). Un număr mai mare de celule conectate va rezulta o putere redusă.

**Folosirea în vehicule**

**Masini electrice**

Prețul în creștere a petrolului și insuficienta resurselor de combustibili fosili împing producatorii de masini sa investeasca în cercetarea propulsiei electrice. Motoarele electrice brushless sunt prognozate sa fie folosite în majoritatea autovehiculelor în următorii 20 ani.

**Biciclete electrice**

Un motor folosit pentru o bicicleta electrică trebuie să ofere o valoare mare de cuplu așa că folosește un senzor Hall. Controllerele ESC folosite pentru biciclete electrice folosesc de obice senzori de frânare, de poziție a pedalei, și ofera ajustare rapidă a vitezei și a cuplului

**Quadcoptere**

ESC-urile sunt o parte esențiala a quadcopterelor ce ofera putere mare și frecvența ridicată motoarelor electrice intr-un pachet compact. Aceste modele depind în totaliate de variatia vitezei motoarelor ce învart elicele. Această variație mare și controlul fin al rotației oferă controlul necesar pentru un quadcopter.

Inalțimea este determinată de puterea trimisă motoarelor. Mișcarea în fața este obținută prin invărtirea mai rapidă a motoarelor din spate. Mișcarea laterală este obținută prin accelerearea motoarelor de pe o parte mai puternic decăt cealaltă.

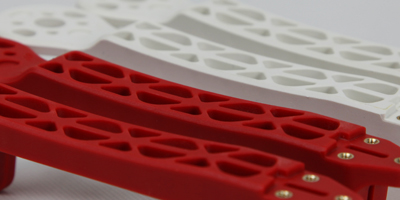
Quadcopterele sunt un hobby ce crește rapid, dar oferă și capacitatea filmării activităților sportive, pentru cercetarea în agricultură, explorare istorică.

**2.1.3 Structura quad. DJI Flame Wheel F450**

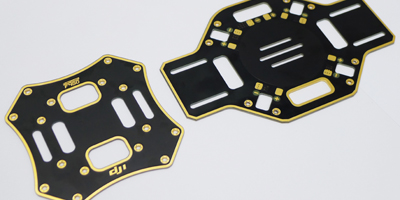
Este o platforma quatrocopter proiectată pentru zbor de divertisment sau pentru fotografie aeriana de tip hobby/amator. Această platforma poate fi echipată cu orice modul de stabilizare. Odată echipată cu unul dintre aceste stabilizatoare platforma este capabilă de performanțe superioare în hovering, zbor de croazieră sau zbor acrobatic în forța.   
Construcția sa este integral modulată și partile sale componente se pot inlocui ușor și foarte rapid.

Caracteristici principale:  
- Material ultra-rezistent  
- PCB integrat, fara cabluri vizibile  
- Spațiu interior generos  
- Brațe rezistente cu aspect plăcuthttp://www.sierra.ro/images/spacer.gif

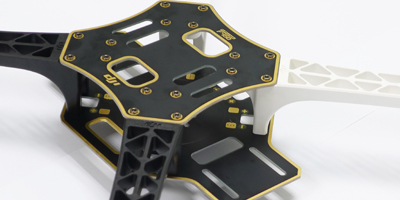
**Materiale ultra-rezistente:**  
Bratele structurii PA66+30GF sunt produse din materiale ultra rezistente și ofera o protecție mai buna la accidente.



**Cabluri integrate în PCB**



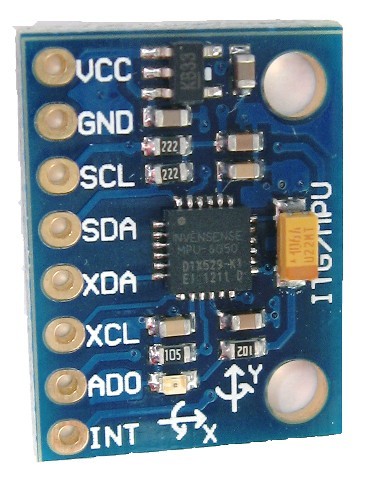
**Spatiu de asamblare mare**  
Structura este construita pentru a oferii un spatiu suficent pentru sistemele de control.



**2.1.4 Giroscop și accelerator MPU 6050**

**Introducere**

Senzorul MPU 6050 conține un accelerometru MEMS și un giroscop MEMS intr-un singur chip. Este foarte precis, conținănd hardware de conversie din analog în digital cu o rezolutie de 16 biți. Rezultatul este permiterea captarii canalelor x,y,z în același timp. Senzorul folosește I2C-bus pentru interfațarea cu arduino.



Giroscopul și accelerometrul conțin fiecare câte trei convertoarea ADC pentru digitizarea datelor de ieșire. Pentru precizia mișcărilor rapide și încete senzorul poate fi configurat pentru greutatea aparatului căruia îi măsoară accelerația și inclinația.

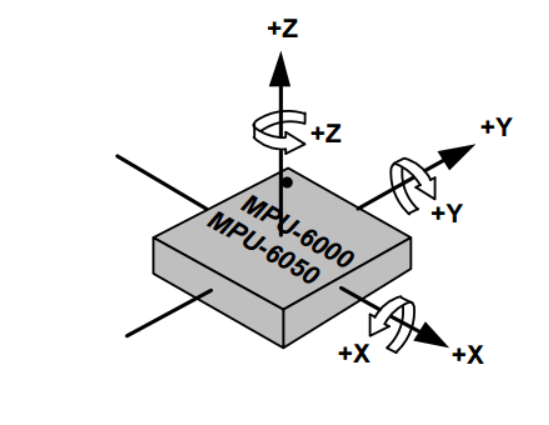


FIG Orientarea axelor și polaritatea rotațiilor

Un buffer FIFO este folosit pentru a reduce consumul de curent, permițănd procesorului să citească informația de la senzor toată odată după care să intre în low power mode.

**Caracteristicile giroscopului**

* Ieșiri digitale X,Y,Z.
* Pin extern de sincronizare
* ADC pe 16 biți integrat
* Calibrator automat sensibil la temperatură
* Filtru trage-jos programabil digital

**Caracteristicile accelerometrului**

* Ieșire digitala pe trei axe
* ADC pe 16 biți integrat permite eșantionarea accelerometrului fără multiplexor
* Detecție de orientare
* Întreruperi programabile

**Procesor digital de mișcare**

Procesorul digital de mișcare încapsulat este localizat în interiorul MPU-6050 și descarcă procesarea algoritmică de mișcare de la procesorul host. Primește date de la accelerometru, giroscop și alți senzorii cum ar fi magnetometrul și le procesează. Rezultatul poate fi citit din regiștrii sau poate fi pus in buffer. Scopul este ca procesorul să descarce cerințele de sincronizare și procesare

**Generarea de tact internă**

Senzorul MPU 6050 deține o schemă flexibilă de tact, ce permite o varietate internă sau externă de surse de tact ce pot fi folosite pentru sincronizarea circuitelor. Selecția surselor pentru generarea internă de tact depinde de disponibilitatea surselor externe și de cerintele de consum de energie și acuratețe de cronometrare. Acuratețea de cronometrare este importantă din cauză ca erorile de sincronizare afectează direct distanța și unghiul calculelor efectuate de procesorul digital de mișcare.

**Regiștrii senzorului**

Registrul senzorului conține ultimele măsuratori ale giroscopului, accelerometrului și senzorilor auxiliari. Pot fi doar citiți și se acceseaza cu interfața serială.

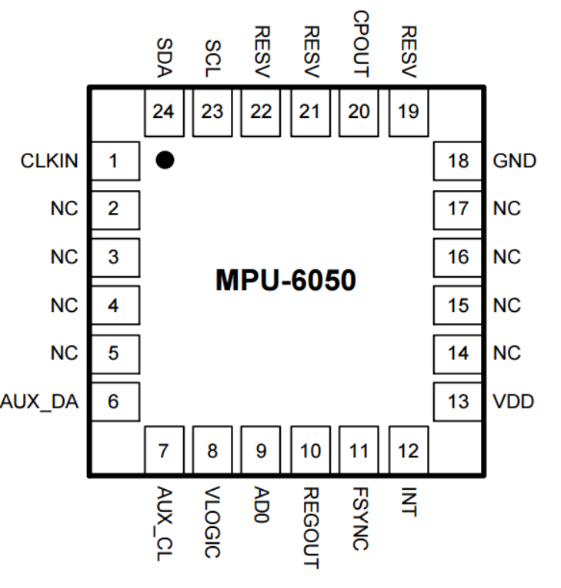


FIG QFN Package

**2.1.5 Senzor temperatura și umiditate DHT11**

**Introducere**

Senzorul de temperatură și umiditate DFRobot DHT11 conține un complex de senzori de temperatură și umiditate cu un semnal de ieșire digital calibrat. Folosind exclusiv tehnica de semnal digital oferă o fiabilitate ridicată și stabilitate mare în timp. Acest senzor include o componentă rezistivă de măsurare a umiditații, o componentă NTC de măsurare a temperaturii și se conectează la un microcontroller de performanță inaltă pe 8 biți oferind calitate excelentă, răspuns rapid, abilitați anti-interferente.

**Specificații tehnice**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parametrii** | **Conditii** | **Minimum** | **Tipic** | **Maximum** |
| **Umiditate** | | | | |
| **Rezolutie** |  | 1%RH | 1%RH | 1%RH |
|  | 8 Bit |  |
| **Repetabilitate** |  |  | ± 1%RH |  |
| **Acuratete** | 25 °C |  | ± 4%RH |  |
| 0-50 °C |  |  | ± 5%RH |
| **Interschimbabilitate** | Total interschimbabil | | | |
| **Plaja masuratorii** | 0 °C | 30%RH |  | 90%RH |
| 25 °C | 20%RH |  | 90%RH |
| 50 °C | 20%RH |  | 80%RH |
| **Timp de raspuns (Secunde)** | 1/e(63%)25 °C,1m/s Aer | 6 S | 10 S | 15 S |
| **Histerezis** |  |  | ± 1%RH |  |
| **Stabilitate de lunga durata** | Tipic |  | ± 1%RH/an |  |
| **Temperatura** |  |  |  |  |
| **Rezolutie** |  | 1 °C | 1 °C | 1 °C |
|  | 8 Bit | 8 Bit | 8 Bit |
| **Repetabilitate** |  |  | ± 1 °C |  |
| **Acuratete** |  | ± 1 °C |  | ± 2 °C |
| **Plaja masuratorii** |  | 0 °C |  | 50 °C |
| **Timp de raspuns (Secunde)** | 1/e(63%) | 6 S |  | 30 S |

**2.1.6 Modul bluetooth HC-05**

Bluetooth-ul este un standard de tehnologie wireless folosit pentru trimiterea datelor la distanțe scurte folosind lungime de unda scurtă intre 2.4 și 2.485 GHz. Trimiterea datelor se face de la dispozitive fixe, mobile și clădiri PAN (Personal Area Networks). A fost inventat de vendorul telecom Ericsson in 1994, și original a fost conceput ca o alternativa a cablurilor de date RS-232. Se poate conecta la mai multe dispozitive, evitând probleme de sincronizare.



Bluetooth este administrat de Bluetooth Special Interest Group (SIG) care are 25000 de companii membre in ariile de telecomunicare, computing, networking și electronice de consum.

Operează la frecvențe intre 2402 și 2480 MHz, sau 2400 și 2483.5 MHz incluzând benzile de guard de 2 MHz. Folosește o tehnologie radion numită saltul frecvenței in lațimea spectrului.

Bluetooth-ul divide datele transmise in pachete și transmite fiecare pachet intr-unul din cele 79 de canale. Fiecare canal are lățimea de bandă de 1 MHz. De obicei face 800 de salturi pe secundă cu saltul în frecvența adaptiv pornit.

Un dispozitiv bluetooth master poate comunica cu maximum 7 dispozitive. Dispozitivele pot schimba rolurile, prin înțelegere și slave-ul poate deveni master. În orice moment, date pot fi transferate între master și orice alt dispozitiv.

HC-05 este un modul SPP (Serial Port Protocol) ușor de folosit, făcut pentru comunicatie wireless serială. Modulul este echipat cu Bluetooth V2.0 + EDR (Enhanced Data Rate).

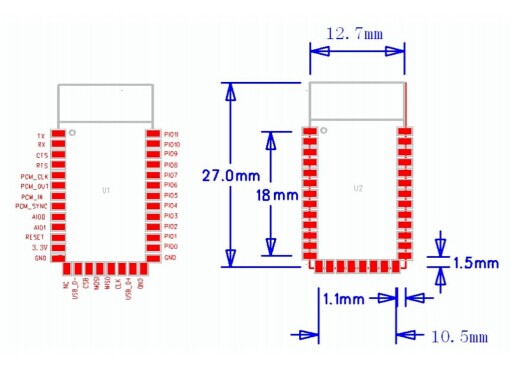
Modulul are doua moduri de operare, order-response și automatic connection. Sunt trei roluri de lucru, master, slave și loopback. Când modulul este in setarea automatic connection va urma starea default pentru transmisia datelor.

Specificații hardware:

* Sensibilitate 80 Dbm
* Pana la +4dBm
* Mod de funcționare in consum redus la 1.8 V
* Interfațare UART cu baud rate programabil
* Are antenă integrată

Specificații software:

* Baud rate standard 38400, 8 biți de date, 1 bit de stio, fara paritate. Suportă baud rate de 9600,19200,38400,57600,115200,230400,460800.
* Se întrerupe comunicarea la puls crescător primit in PIO0
* La pornire se auto conectează la ultimul dispozitiv conectat.



**2.1.7 Acumulator LiPo GENS ACE 11.1 V**

Reprezintă generația ce inlocuieste NiCd și NiMH și se folosesc în aplicatii unde masa acumulatorului este importanta. Principiul de funcționare este același ca și la alte acumulatoare lithium-ion, ioni pozitivi și negativi se intercalează de la un material electrod pozitiv la un material electrod negativ, cu electrolitul lichid oferind un mediu conductiv. Pentru a prevenii atingerea directa a electrozilor, se folosește un separator microporis care permite doar ionilor să treacă dintr-o parte in alta.

Celulele LiPo ofera producatorilor avantaje serioase. Pot produce acumulatori de aproape orice forma și au foarte o rată de descărcare in așteptare foarte scăzută (5% pe lună).

Din punctul de vedere al siguranței celulele LiPo sunt afectate de aceleași probleme ca și celelalte celule lithium-ion. Asta inseamnă că supra încărcarea, supra descărcarea, supra incălzirea, scurt circuitarea, lovirea pot produce defecțiuni catastrofale, incluzând umflarea, scurgerea, explozia și focul.

Aceasta tehnologie oferă o mare densitate a energiei, greutate mică, și durată de exploatare convenabilă comparativ cu alte baterii. Necesită încărcarea cu încărcător special pentru Litiu Polymer și egalizor.

http://www.sierra.ro/images/spacer.gif



**Specificatii:**  
Tensiune: 11.1 V  
Capacitate: 3300 mA  
Rata descarcare: 25 C  
Curent maxim de incarcare: 16.5 A (5C)  
Curent de incarcare recomandat: 3.3 - 9.9 A  
Curent maxim de descarcare: 82.5 A  
Dimensiuni: 135 x 42 x 24 mm  
Masa: 285 g  
Conector: DEANS  
Conector egalizare: JST-XH

**2.1.8 Motoare de curent continuu fără perii DJI 2312**

Motoarele de curent continuu fără perii , cunoscute și ca motoare comutate electronic sunt motoare sincron care sunt alimentate de o sursa electrică DC printr-un invertor sau alternator care produce un semnal electric de curent alternativ pentru a conduce motorul. In acest context curentul alternativ nu este de formă de undă sinusoidală ci un curent bi-direcțional fara restrictie la forma de undă.Senzori adiționali și electronice controleaza output-ul în amplitudine al invertorului, deci viteza rotorului.

Partea rotor a motorului fără perii este de obicei un motor sincro cu magnet permanent, dar poate fi și un motor comutator.

Motoarele fără perii pot fi descrise ca motoare pas cu pas, dar termenul pas cu pas este rezervat pentru motoare care sunt operate intr-un mod care implică oprirea frecventă a motorului intr-un unghi bine definit.

Motoarele cu perii se gasesc incă de la mijlocul secolului 19, dar motoare electrice fără perii sunt o apariție relativ recentă. Un prim pas a fost atins in anii 60 odata cu evoluția tehnologiei cu stare solidă, cu alte imbunătățiri ce au venit in anii 80 mulțumită materialelor mai bune din care se produc magneți permanenți.

Motoarele fără perii au câteva avantaje față de motoarele cu perii, incluzând raportul mare intre cuplu și greutate, mai mult cuplu pe watt cea ce inseamnă o eficiență mai mare, sunt mai fiabile, mai silențioase și rezistă pentru mai mult timp. Din cauză că nu au bobinaj in în rotor, nu sunt supuse forțelor centriuge și pentru că bobinajul este in stator, pot fi răcite prin conducție, nefiind nevoie de flux de aer în interiorul motorului. Asta inseamă ca interiorul motorului poate fi închis, protejându-l in totalitate de praf sau altă materie.

Comutația motoarelor fără perii poate fi implementată folosind un microcontroller sau un microprocesor. Comutarea cu electronice in locul periilor permite flexibilitate mai mare și capabilități ce motoarele cu perii nu le au.

Din cauză ca rotația este direcționata de un controller, controllerul trebuie să știe poziția rotorului relativ cu statorul. Unele folosesc senzorii cu efect Hall sau codificator de rotație pentru a măsura direct pozitia rotorului.

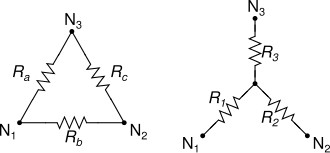
Intr-un motor fără perii două perii sunt energizate într-o perioadă cu polaritate egală și opusă, una înpinge rotorul și cealaltă îl atrage. Asta crește capacitatea cuplului motorul și efectul senzorului Hall sau codificatorul determină care bobine au fost energizate pentru a atinge strategia respectivă.

Un controller tipic conține 3 ieșiri bi-direcționale care sunt controlate de un circuit logic. Controllerele simple folosesc comparatoare pentru a determina cand trebuie avansată faza de iesire, pe când controllerele mai avanssate folosesc un microcontroller pentru a controla acceleratia, viteza și setarea fina a eficienței.

Controllerele care detectează poziția rotorului bazandu-se pe EMF au provocări in plus în inițierea mișcării din cauză ca nu se detectează nimic atunci când motoarele sunt staționare. Asta e de obicei rezolvat prin inceperea rotație dintr-o faza arbitrară dupa care sarindu-se la faza corectă. Asta poate provoca motorul să facă un pas in spate, adaugănd mai multă complexitate la secvența de pornire. Alte controllere fără senzori sunt capabile să măsoare saturația bobinei cauzată de poziția magneților pentru a detecta poziția rotorului.

Motoarele fără perii pot fi construite în mai multe configurații fizice. Configurația convențională, cunoscută și ca inrunner este aceea in care magneții permanenți fac parte din rotor. Trei bobine sunt infășurate in jurul rotorului in configurația outrunner și relația radială intre bobine și magneți este inversată, bobinele statorului de la centrul motorului in timp ce magneții permanenți se rotesc cu rotorul.

Sunt doua configurații comune de bobinare, configurația delta care conectează trei bobine una de alta in serie intr-un circuit triunghi, câteodată numit stea și la care puterea este aplicată la fiecare conexiune. Mai este configurația wye in formă de Y care conecteaza toate bobinele la punctul central, un circuit paralel la care puterea este aplicată la capatul rămas de la fiecare bobinare.

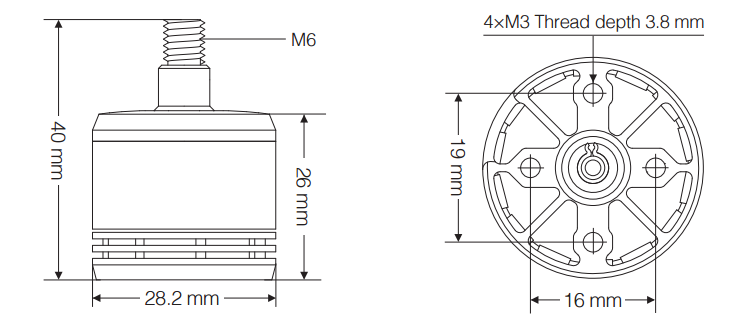


Un motor cu bobinaj in configurația delta ofera cuplu mic la viteză mică, dar poate oferii o viteză maximă mai mare pe cănd configurația wye oferă cuplu mare la viteză mică, dar nu ajunge la o viteză maximă la fel de mare.

Cu toate că eficiența este afectată mult din punctul de vedere al construcției motorului, bobinajul wye este de obicei mai eficient. În bobinajul conectat delta, jumătate din voltaj este aplicat intre bobinele adiacente provocând pierderi din cauza rezistivitătii crescute.

**Motorul fără perii DJI 2313** este un proiectat să fie folosit intr-un sistem multi-rotor care cântărește intre 1 și 2.5 kg. Motorul oferă un designe electromagnetic optimizat care are o eficiența ridicată, are stabilitate la cuplu la viteze mici, disipație mică de caldura la bază.

|  |  |  |
| --- | --- | --- |
| **Specificații** | |  |
| **1** | Baterie recomandată | 4S LiPo |
| **2** | Împingere maximă | 850 f. Rotor (14.8 V, Nivelul mării) |
| **3** | Greutate la ridicare (Recomandată) | 350 ~ 400 g/rotor (Nivelul mării) |
| **4** | Temperatură de funcționare | -10 ~ 40 Grade Celsius |
|  | **ESC** |  |
| **1** | Voltaj maxim permis | 17.4 V |
| **2** | Curent maxim permis (persistent) | 20 A |
| **3** | Curent maxim permis (3 secunde) | 30 A |
| **4** | Nivel semnal PWM de intrare | 3.3 V / 5 V |
| **5** | Frecvență semnal | 30 ~ 450 Hz |
| **6** | Baterie recomandată | 3S ~ 4S LiPo |
| **7** | Greutate (fără cablu) | 12.5 g |
| **8** | Greutate (cu cablu) | 27 g |
|  | **Motor** |  |
| **1** | Dimensiune sator | 23 x 12 mm |
| **2** | KV | 800 rpm / V |
| **3** | Greutate | 56 g |
|  | **Elice** |  |
| **1** | Diametru | 24 x 12.7 cm |
| **2** | Greutate | 13 g |

****

**2.2 Echipamente software**

**2.2.1 Utilitar de dezvoltare Arduino 1.6.9**

Utilitarul de dezvoltare Arduino sau - Arduino Integrated Development Environment – contine un editor de text pentru scrierea codului, un spatiu de mesaje, o consola de text, un toolbar cu butoane pentru functionalitati comune și o serie de meniuri. Functioneaza conectanduse la placa Arduino prin upload de programe sau comunicare serială.

Scrierea de programe, sau schite (sketches) este facuta în editorul de text și sunt salvate cu extensia “.ino”. Editorul are optiuni de cut, paste și search, replace. Zona de mesaj ofera feedback în timpul salvarii și exportarii și arata erorile. Consola arata outputul text al programelor incluzand mesajele de eroare în detaliu și alte informatii. Coltul din dreapta jos al ferestrei afiseaza placa configurada și portul serial. Butoanele din bara permit verificarea, uploadearea, creare, deschiderea, salvarea schitelor precum și deschiderea monitorului serial.

Comenzi aditionale se gasesc în cele cinci meniuri: File, Edit, Sketch, Tools, Help. Meniurile sunt sensitive la context, ceea ce inseamna ca doar obectele relevante muncii curente sunt disponibile.

Meniul “File” contine optiunile: new, open, open recent, sketchbook, examples, close, save, save as, page setup, print, preferences, quit.

Meniul “Edit” contine optiunile: undo/redo, cut, copy, copy for forum, copy as html, paste, select all, comment/uncomment, increas/decreas indent, find, find next, find previous.

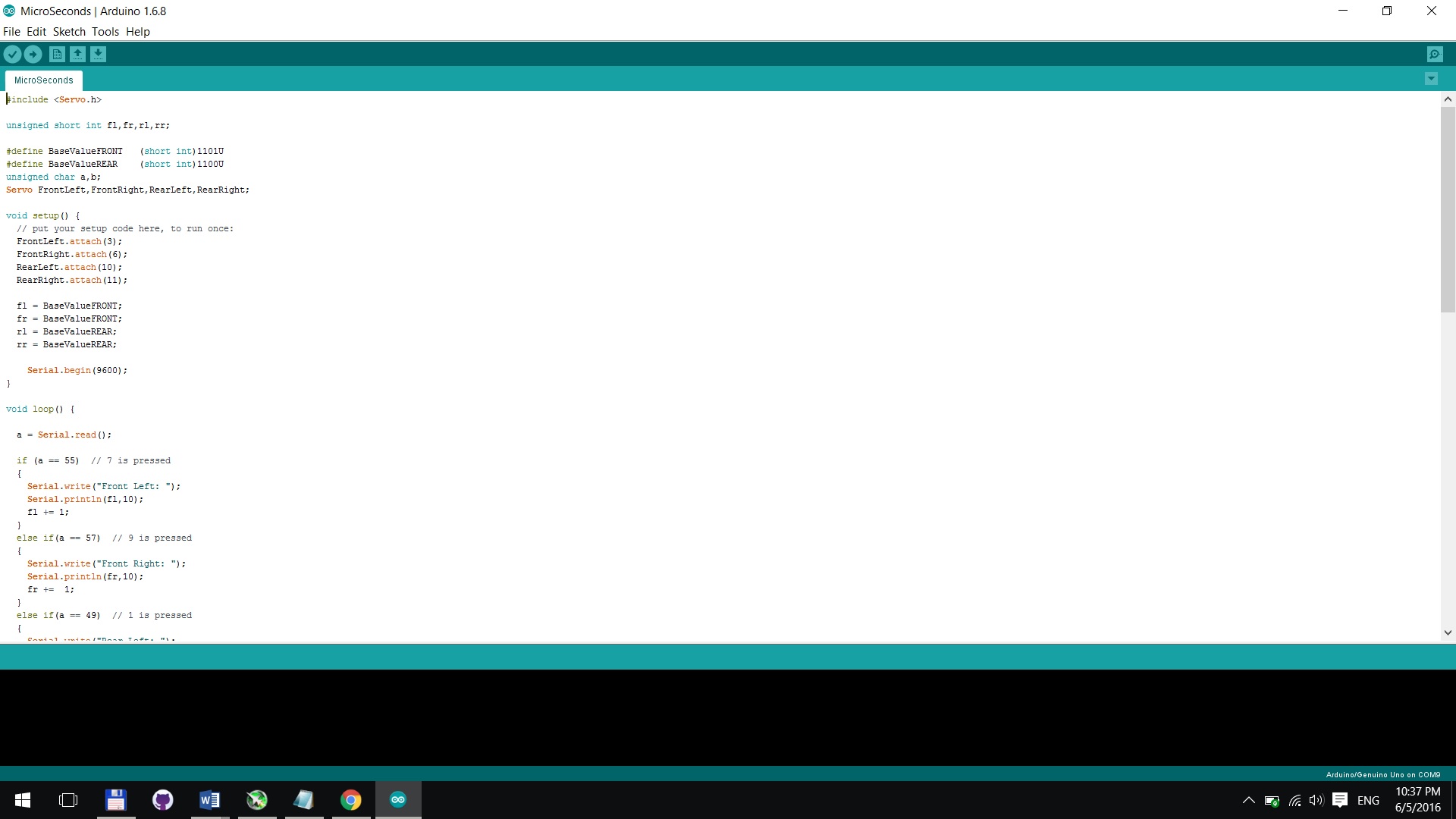
Meniul “Sketch” contine optiunile: verify/compile, upload, upload using programmer, export compiled binary, show sketch folder, include library, add file.

Meniul “Tools” contine optiunile: auto format, archive sketch, fix encoding, serial monitor, board, port, programmer, burn bootloader.

Utilitarul de dezvoltare Arduino utilizeaza conceptul de sketchbook: un loc standard pentru pastrarea programelor. Programele sau – schitele –pot fi deschise din butonul de open din bara. La prima rularea a utilitarului Arduino, acesta va crea un director pentru programe.

Inainte de uploadarea programului trebuie selectat portul corect de USB în tools. Dupa selectarea portului se apasa butonul de upload din bara. La fiecare folosirea a modului upload se folosește bootloaderul arduino, un porgram mic ce a fost incarcat în microcontroller. Acesta permite uploadarea codului fara hardware aditional. Bootloaderul este activ pentru cateva secunde și la reset, dupa care porneste codul care este incarcat pe microcontroller.

Librariile ofera functionalitate extra pentru programe, de exemplu manipularea datelor. Pentru folosirea librariilor trebuie selectat din Sketch > Import Library.



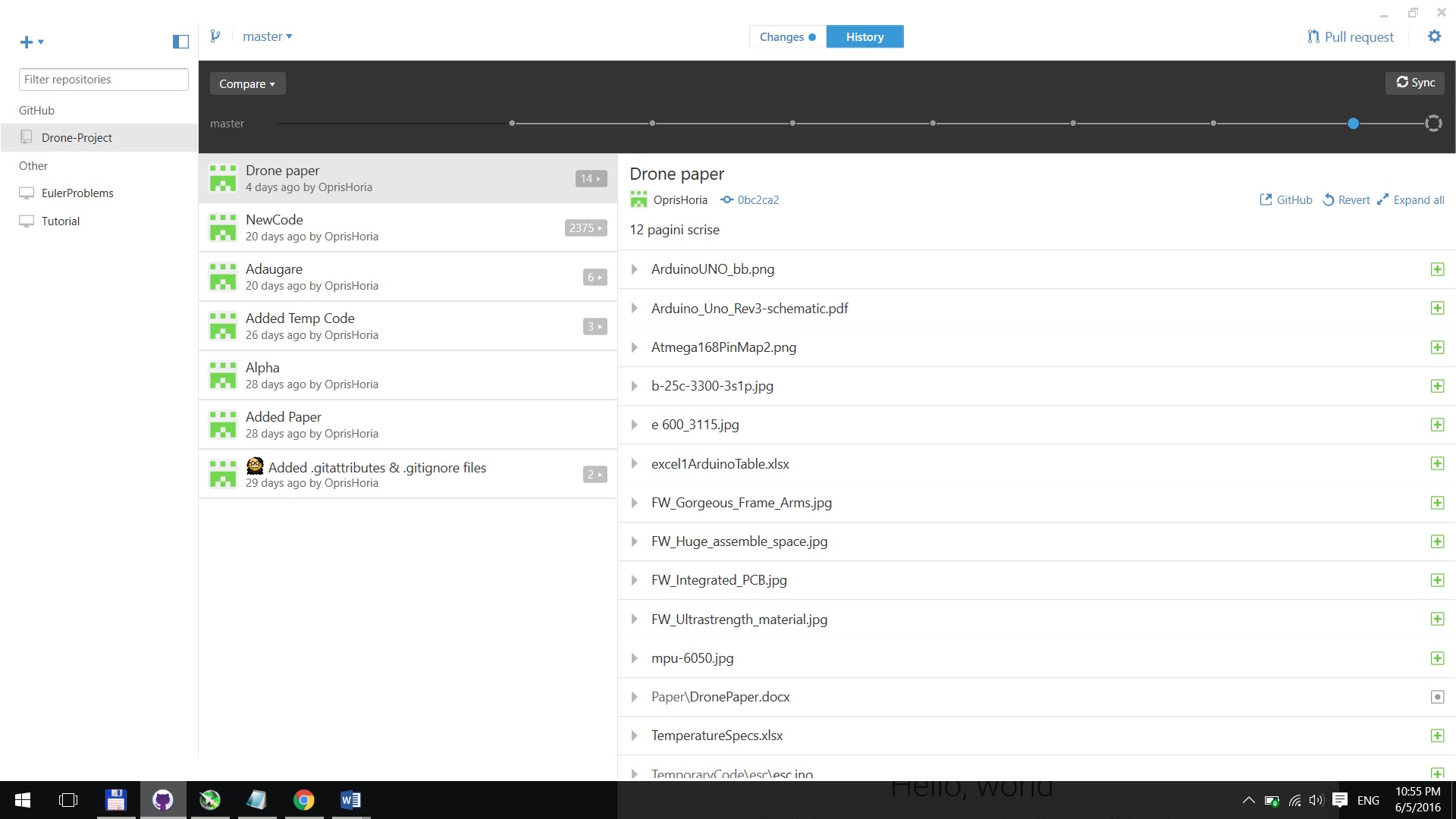
**2.2.2 Sistem de versionare GitHub**

Git hub este un sistem de verionare online. Ofera toate functionalitatile de control, revizie și source code management, adaugand în plus optiunile proprii. Deosebirea fata de GIT care este un tool controlabil doar din linie de comanda, gitHub ofera o interfata grafica Web și integrare desktop și mobila. Ofera și control la acces și alte optiuni de colaborare cum ar fi bug tracking, feature request, task management și wiki pentru fiecare proiet.

Proiectele din Git hub pot fi accesate și manipulate folosind interfata standard din linie de comanda, dar și din interfata web. Permite userilor inregistrati sau neinregistrati sa urmareasca proiectele publice de pe site. Site-ul ofera și functii de socializare precum feeduri sau followeri.

Git hub este folosit în special pentru cod, dar în aditie suporta și folosirea urmatoarelor formate:

* Documentatie, inclusiv fisiere readme generate automat.
* Urmarirea editiilor cu etichete,
* Cereri de pull cu review la cod și comentarii
* Istoria commit-urilor
* Grafice
* Director de integrare
* Notificari email



**2.2.3 Utilitar dezvoltare Visual Studio**

Visual Studio este un IDE creat de Microsoft. Este folosit pentru dezvoltarea programelor pentru Microsoft Windows și pentru site-uri web, aplicații web și servicii web. Folosește platforme de dezvoltare software Microsoft cum ar fi, Windows Api, Windows Forms, Windows store.

Visual Studio include un editor de cod ce suportă Intellisense (o implementare de completare de cod dezvoltată de Microsoft). Debuggerul integrat funcționează atat ca debugger la nivel de cod sursă, precum și ca debugger la nivel de procesor. Alte tool-uri incluse sunt form-urile pentru designul de aplicații cu interfață. Editorul suportă și compilare in background, numită și compilare incrementală.

Debuggerul folosit de Visual Studio permite setarea de breakpoint-uri (permit oprirea execuției intr-o anumită poziție) și de watch-uri (monitorizează valoarea variabilelor in timpul execuției). Breakpoint-urile pot fi condiționale, insemnănd ca pot fi puse in funcție de anumite condiții.

Designer-ul windows forms, folosit și pentru aplicația de control a quadcopterului este folosit pentru design-ul aplicațiilor grafice. Controale ce gestioneaza informații pot fi conectate la baze de date. Interfața user este legată folosing cod generat in C#.

In următoarea figură este prezentată fereastra in care se dezvoltă aplicații windows.

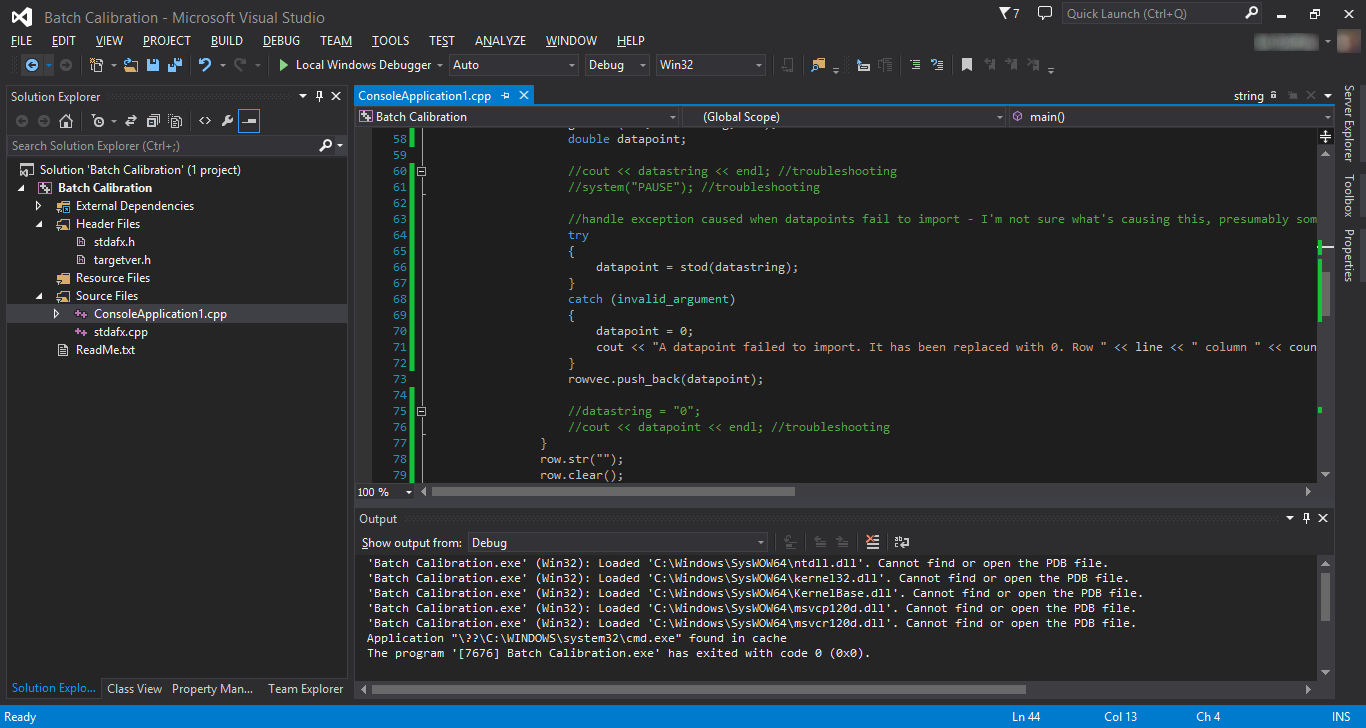


Fig \_\_

**Capitolul 3. Arhitectura sistemului**

**3.1 Arhitectura hardware**

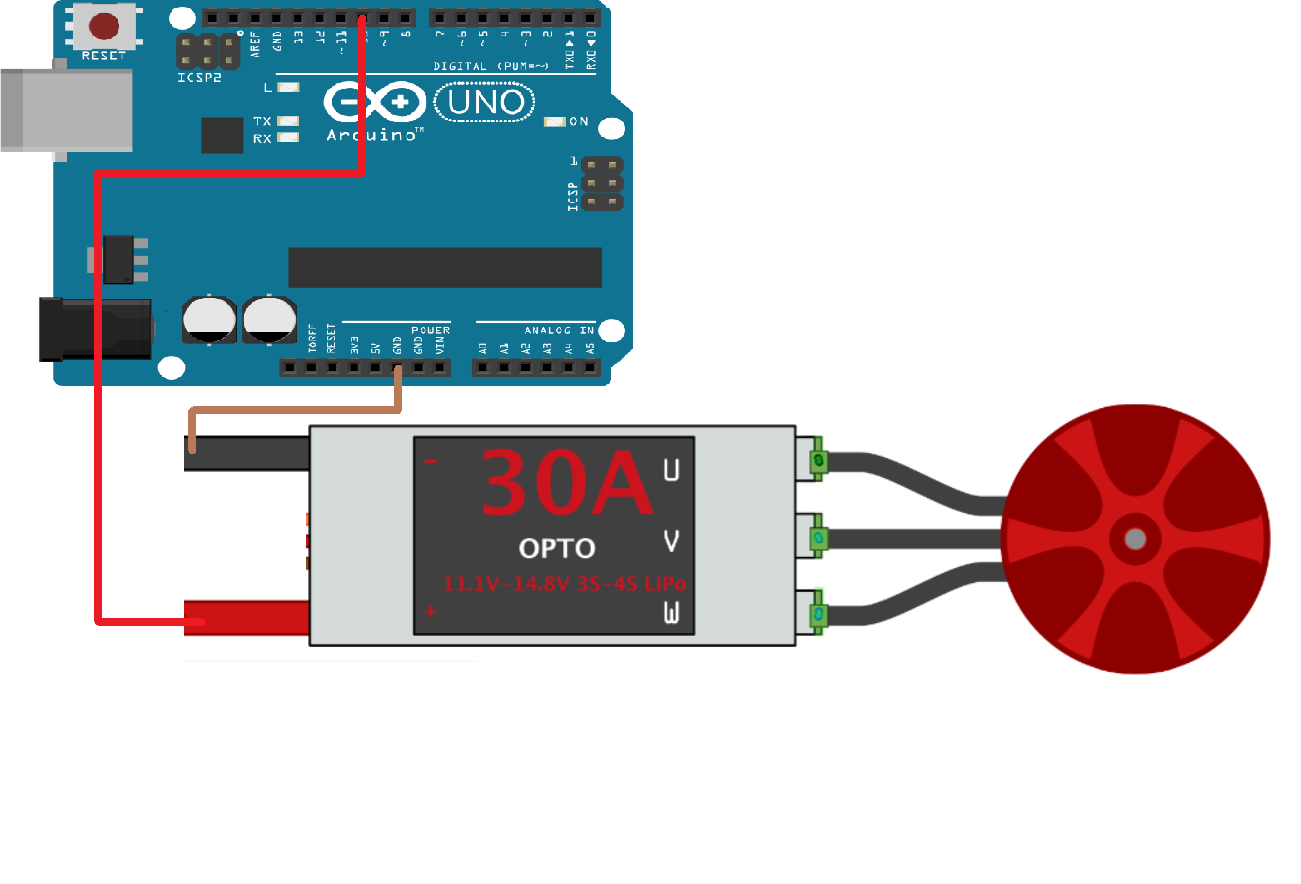
**3.1.1 Conexiune Arduino spre ESC și spre motoare**

Pentru controlul motoarelor se folosesc regulatoare electronice de viteza. Fiecare motor are un regulator propriu. Fiecare regulator are un fir de semnal și un fir de masă pentru intrare și 3 fire de iesire pentru motoare.

Firele de masă au fost conectate intre ele și la pinul GND al Arduino.

Fiecare fir de semnal a fost conectat la pinii (3,6,10,11) PWM ai Arduino.

Firele de ieșire ale regulatorului de viteză se conecteză in orice ordine la motorul electric făra perii. Dacă se dorește inversarea sensului motorului, se schimbă oricare doua fire între ele.



**3.1.2 Conexiune Arduino spre giroscop**

**3.1.3 Conexiune Arduino spre bluetooth**

**3.1.4 Conexiune Arduino spre aplicație mobilă**

**3.2 Arhitectura software**

**3.3 Arhitectura sistemului**

**3.3.1 Descrierea functiilor**

**Capitolul 4.**

**4.1 Implementare hardware**

**4.2 Implementare software**

**4.2.1 Descrierea modulelor**