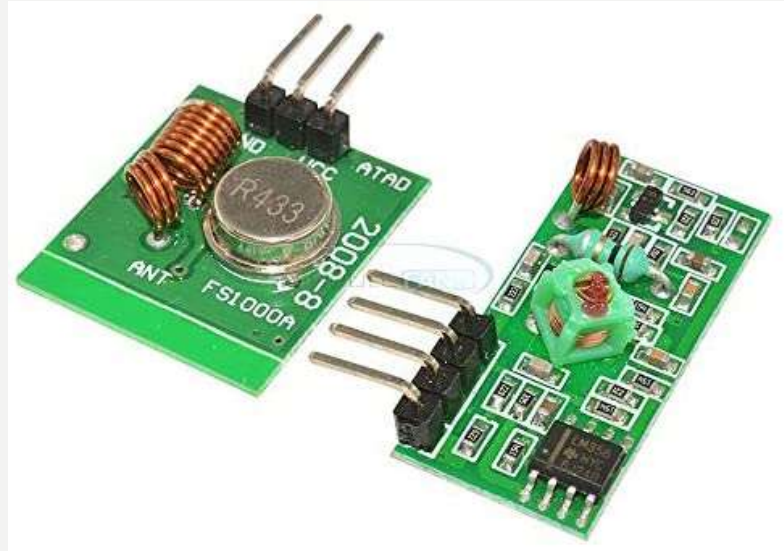


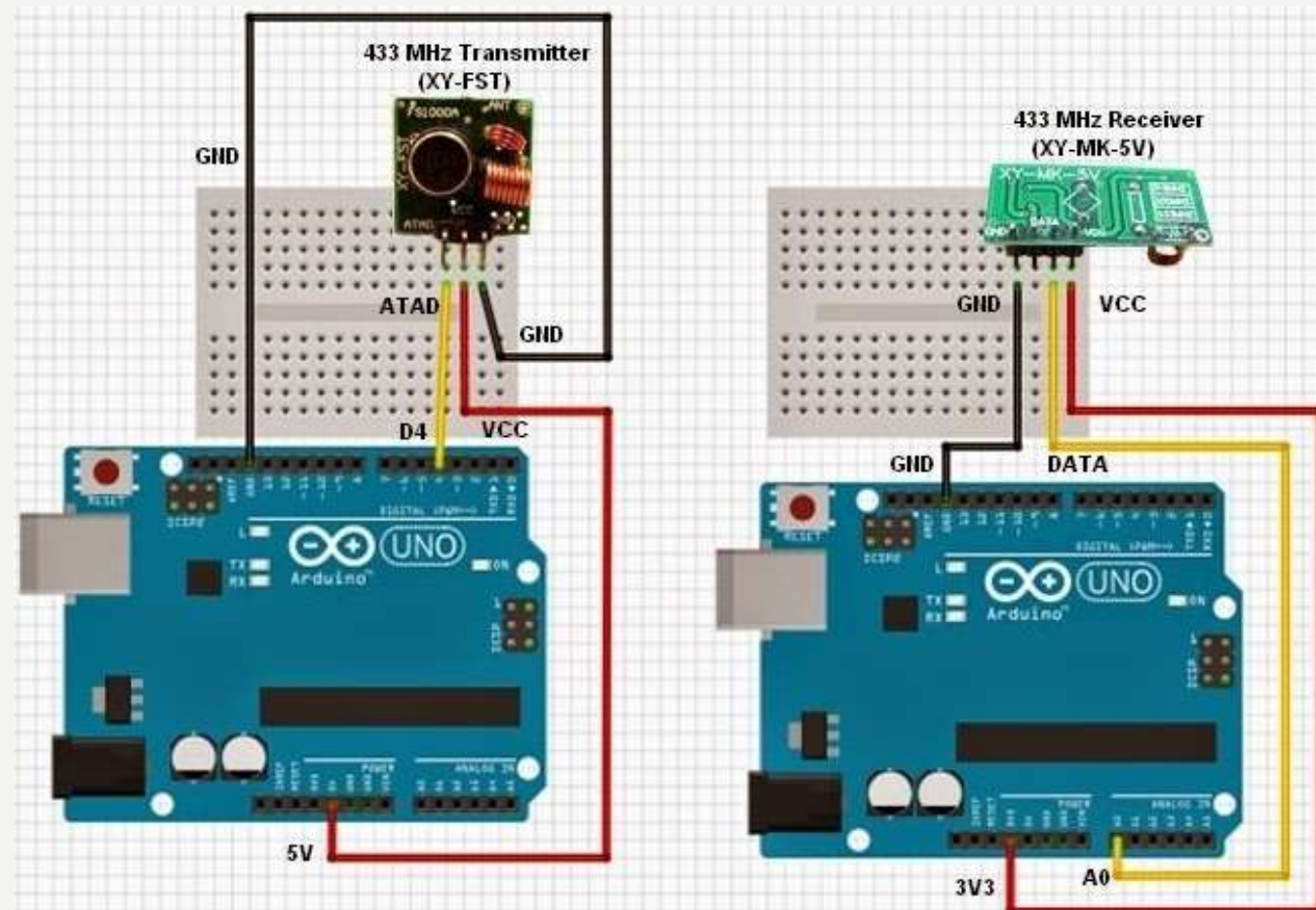
# 433 MHz



# RF MODULI

# 433MHz RF MODULI

- dodavanje mogućnosti bežičnog povezivanja Arduino uređaju.



# KRATAK OPIS HARDVERA -PRIJEMNIK

Ovaj modul je prijemnik. Jednostavan je, kao i što izgleda.

Srce modula je SAW (Surface Acoustic Wave) rezonator, podešen na 433MHz.

Tu je još i prekidački tranzistor i nekoliko pasivnih komponenti.



Kada je logičko HIGH na DATA ulazu, oscilator se pokreće i prizvodi konstantni RF nosilac i kada je logičko LOW da DATA ulazu oscilator se zaustavlja.

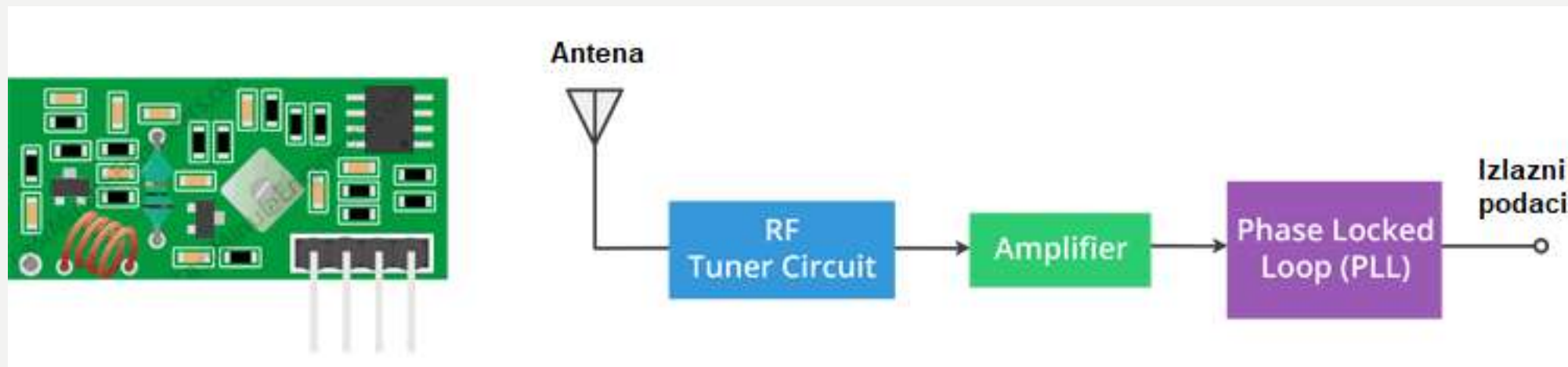
Ova tehnika se naziva ASK (Amplitude Shift Keying)

# KRATAK OPIS HARDVERA - PREDAJNIK

Ovaj modul je prijemnik..

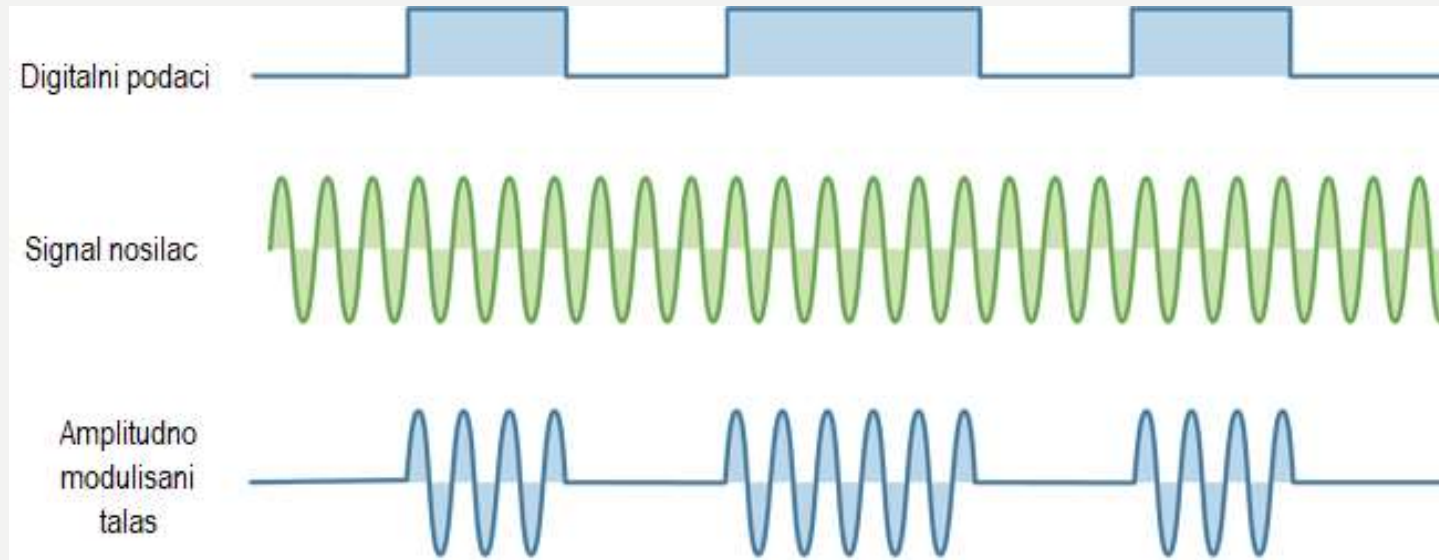
Sastoji se iz RF tjunera, par OP pojačavača za pojačanje primljenog signala nosioca.

Pojačani signal se dalje vodi do PLL-a, koje omogućuje dekoderu da se „zaključa“ na tok digitalnih bitova, što daje bolji dekodirani izlaz i otpornost na šum.



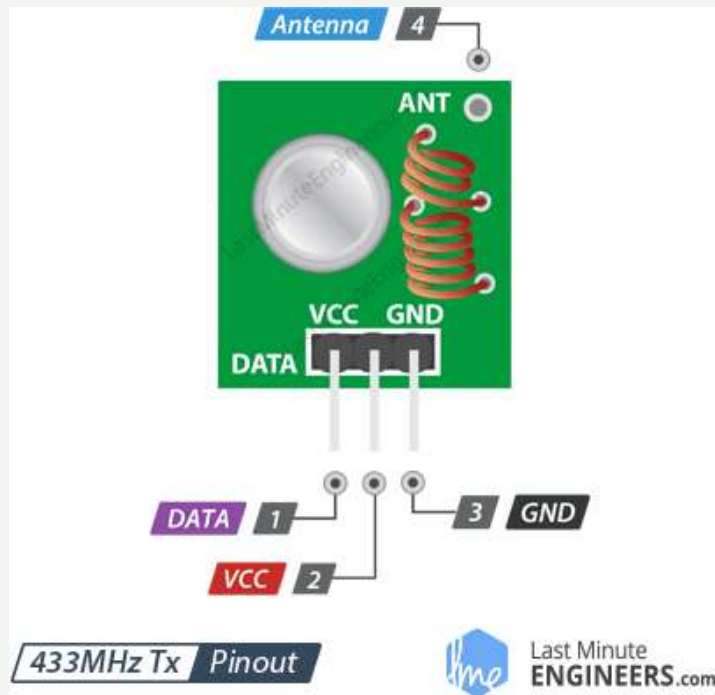
# ASK – AMPLITUDE SHIFT KEYING

- ASK je tehnika kojom se amplituda signala nosioca mijenja, kao odgovor na signal ulaznih podataka
- Naziva se binarni ASK



- Prednost ASK je jednostavna implementacija, kao i manji propusni opseg, nego kod drugih modulacionih tehnika kao FSK (Frequency Shift Keying).
- Nedostatak ASK je podložnost smatnjama drugih radio uređaja i pozadinskoj buci. Ipak sve dok se prenose podaci relativno niske brzine, može pouzdano raditi u većini okruženja.

# 433MHZ RF PREDAJNIK – PRIKAZ IZVODA



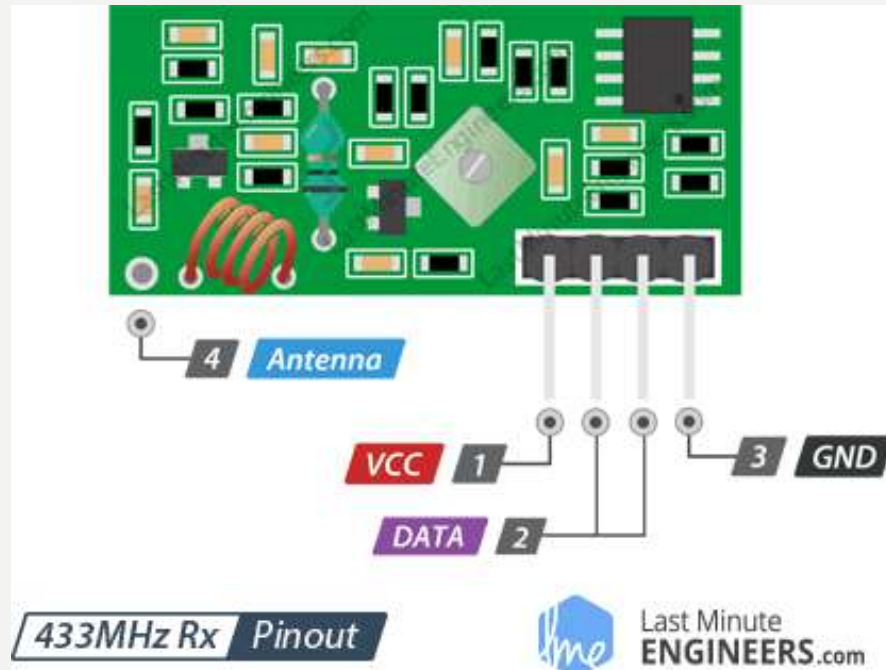
**DATA** - pin na koji se dovode podaci koji se žele prenijeti.

**VCC** - Napon napajanja predajnika. Može biti bilo koji pozitivni DC napon između 3.5V i 12V. Veći napon napajanja, veći domet.

**GND** - je pin na koji se povezuje masa.

**Antena** - je pin za povezivanje eksterne antene. U cilju poboljšanja dometa, može se povezati komad pune žice, dužine 17.3cm

# 433MHZ RF PRIJEMNIK – PRIKAZ IZVODA



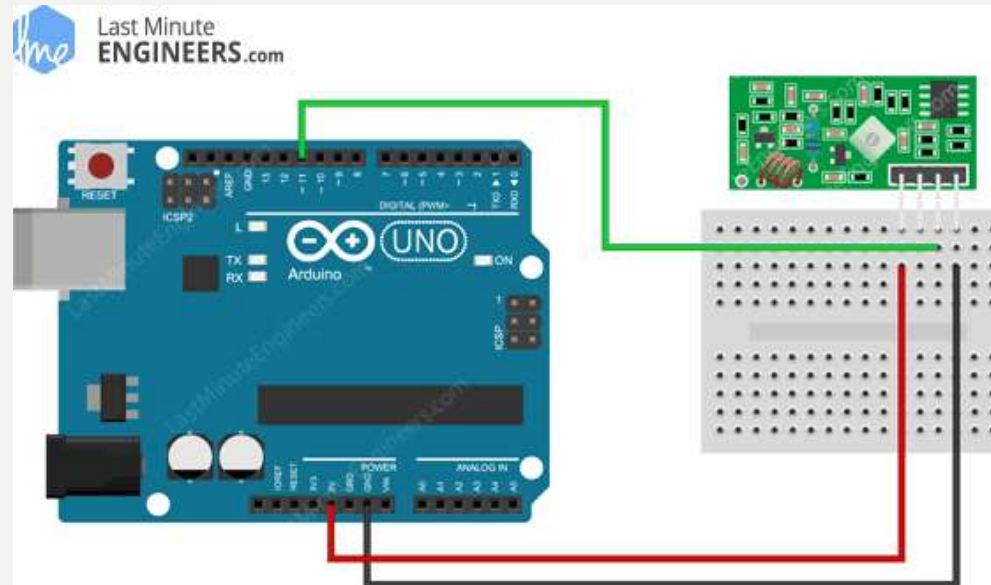
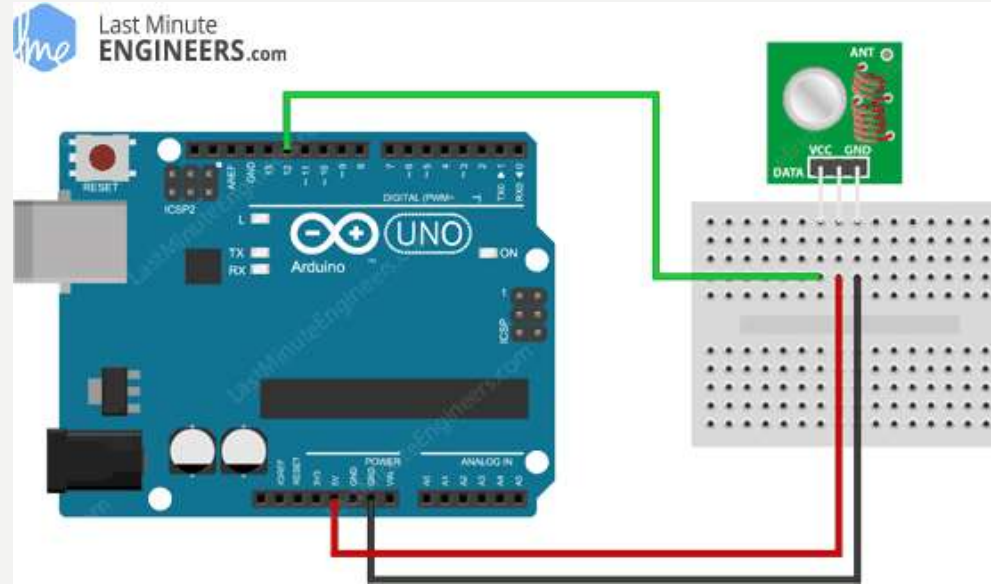
**DATA** – izlazni pinovi, na koji se pojavljuju primljeni podaci.

**VCC** - Napon napajanja predajnika. Za razliku od predajnika prijemu je potrebno dovesti 5V

**GND** - je pin na koji se povezuje masa.

**Antena** - je pin za povezivanje eksterne antene. U cilju poboljšanja dometa, može se povezati komad pune žice, dužine 17.3cm

# POVEZIVANJE PREDAJNIKA I PRIJENIKA SA ARDUINO UNO





# RADIOHEAD LIBRARY

Prije početka kodiranja, RadioHead biblioteka treba biti instalirana u Arduino IDE. Ona omogućuje znatno jednostavnije kodiranje.

RadioHead biblioteka omogućuje jednostavnu razmjenu podataka između Arduino ploče.

Može se koristiti za pogon svih vrsta radiokomunikacionih uređaja, uključujući 433MHz module.

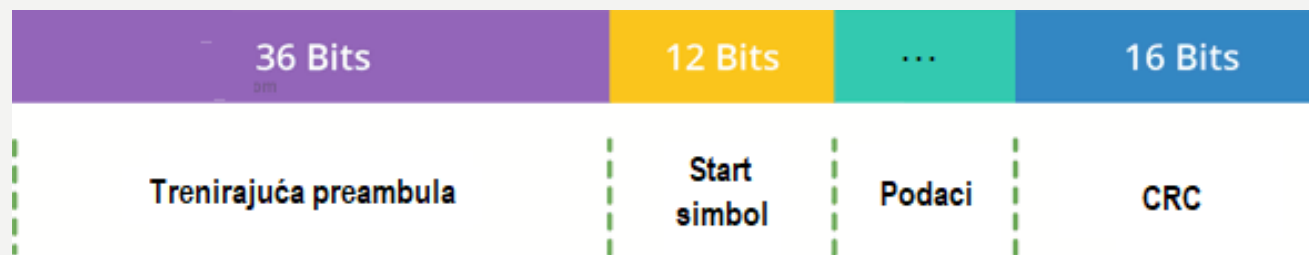
RadioHead biblioteka inkapsulira naše podatke u paket podataka koji uključuje CRC (Cyclic Redundancy Check), zatim ih s neophodnom preambulom i zaglavljem šalje ka drugom Arduino uređaju.

Ako su podaci ispravno primljeni, Arduino dobija informaciju da su podaci dostupni, te nastavlja s dekodiranjem i postupanjem.

RadioHead paket se sastoji iz sljedećih cjelina:

- 36-bitni niz parova jedinica i nula, koji se naziva „Trenirajuća preambula“ i šalje se na početku svakog prenosa. Ovi bitovi su potrebni prijemniku da se sinhroniše, prije nego što počene prihvatati stvarne podatke.
- Nakon toga dodaje se 12-bitni „Start simbol“.
- Zatim stvarni podaci.
- Sekvenca za provjeru integriteta podataka, dodaje se na kraju paketa. CRC podatke RadioHead ponovo izračunava na prijemnoj strani. Ako je provjera ispravna, prijemni uređaj se upozorava. Ako provjera CRC-a ne uspije, paket se odbacuje.

Sruktura kompletnog paketa prikazana je na slici:



Biblioteku možete preuzeti sa adrese:

<http://www.airspayce.com/mikem/arduino/RadioHead/>

Na ovoj adresi možete naći i detaljnije opis biblioteke

# ARDUINO KOD – ZA 433MHZ PREDAJNIK

Skeč počinje sa uključivanjem RadioHead ASK biblioteke.

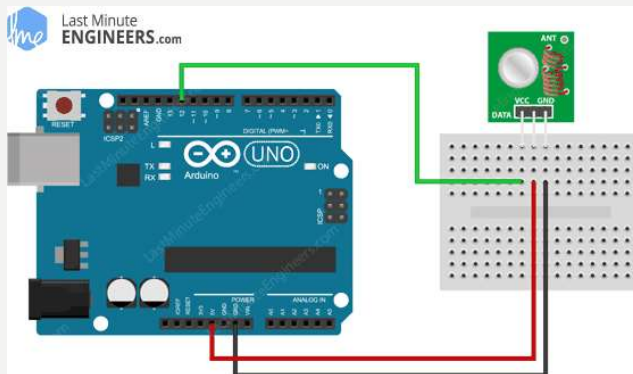
Potrebno je uključiti i [Arduino SPI Library](#) biblioteke, jer RadioHead se oslanja na nju.

Zatim je potrebno kreirati ASK objekat, da bi se pristupilo specijalnim funkcijama iz RadioHead ASK biblioteke.

U setup funkciji, potrebno je inicijalizovati ASK objekat.

U loop funkciji, startuje se pripremanjem poruke. To je jednostavni tekst niz smješten u karakter pokazivaču nazvanom msg. Poruka može biti bilo što ali ne smije biti duža od 27 karaktera (zbog boljih performansi).

Poruka se zatim šalje upotrebom funkcije send(). Iza funkcije send(), obično slijedi funkcija waitPacketSent(), koja čeka dok se kompletno ne odašalje predhodno poslani paket.



```
// Include RadioHead Amplitude Shift Keying Library
#include <RH_ASK.h>
```

```
// Include dependant SPI Library
```

```
#include <SPI.h>
```

```
// Create Amplitude Shift Keying Object
```

```
RH_ASK rf_driver;
```

```
void setup()
```

```
{
```

```
    // Initialize ASK Object
```

```
    rf_driver.init();
```

```
}
```

```
void loop() {
```

```
    const char *msg = "Hello World";
```

```
    rf_driver.send((uint8_t *)msg, strlen(msg));
```

```
    rf_driver.waitPacketSent();
```

```
    delay(1000);
```

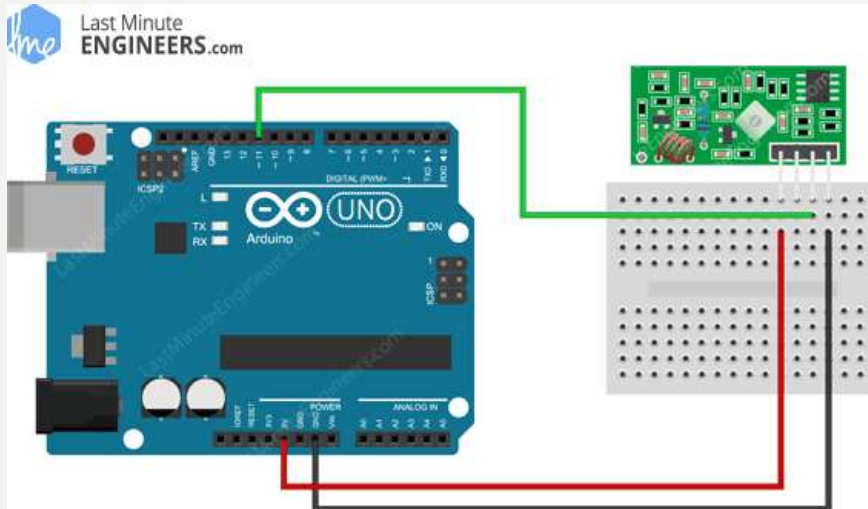
```
}
```

# ARDUINO KOD – ZA 433MHZ PRIJEMNIK

Sve do `loop()` funkcije kod prijemnika je gotovo identičan sa kodom predajnika.

U loop funkciji, kreira se bffer iste veličine kao polata poruka.

Zatim se poziva funkcija `recv()`. Ova funkcija uključuje prijemnik, ukoliko već nije uključen. Ako postoji validna pristigla poruka, `recv()` funkcija kopira poruku u svoj prvi parametar buf.



```
// Include RadioHead Amplitude Shift Keying Library
#include <RH_ASK.h>
// Include dependant SPI Library
#include <SPI.h>
// Create Amplitude Shift Keying Object
RH_ASK rf_driver;
```

```
void setup() {
    // Initialize ASK Object
    rf_driver.init();
    // Setup Serial Monitor
    Serial.begin(9600);
}
```

```
void loop() {
    // Set buffer to size of expected message
    uint8_t buf[11];
    uint8_t buflen = sizeof(buf);

    // Check if received packet is correct size
    if (rf_driver.recv(buf, &buflen))
    {
        // Message received with valid checksum
        Serial.print("Message Received: ");
        Serial.println((char*)buf);
    }
}
```

# PRIMJER ZA VJEŽBU

Pomoću analognog joystick-a upravljati dizalicom. Pomjeranjem ručice joystick-a tako da se mijenja vrijednost po Y osi, upravljati kolicima po dizalici naprijed nazad (jedan servo motor). Pomjeranjem po X osi pomerati dizalicu lijevo desno za maksimalan ugao  $\pm 90^\circ$  (drugi servo motor).

Spuštanje i podizanje tereta kontrolisati glasovnim komandama. Dvije glasovne komande unutar 2 sekunde, izazivaju spuštanje tereta, jedna glasovna komanda unutar dvije sekunde izaziva podizanje tereta.

Spuštanje i podizanje tereta predstaviti sa 4 LED. Za spuštanje tereta, uključivati LED s lijeva na desno (odozgo na dolje), a za podizanje tereta isključivati LED s desna na lijevo (odozdo na gore).

Joystick i detektor zvuka povezani su na Arduino ploču A. 4 LED i servo motori posta povezani su sa pločom B.

**(4-3-2).**

# PRIMJER ZA VJEŽBU

Upotrijebiti sensor pokreta za automatsko aktiviranje stepenišne rasvjete u dvospratnoj zdradi.

Stepenišnu rasvjetu predstaviti sa dvije LED. Jedna za svaki sprat.

Na svakom spratu postaviti po jedan detektor pokreta, i povezani su na Arduino ploču A. Kada se na nekom detektuje pokret uključiti svjetlo na tom spratu. Svjetlo predstavljaju LED, povezane na Arduino ploču B.

Ukoliko između dvije detekcije pokreta prođe više od zadatog vremena isključiti svjetlo na datom spratu. Vrijeme zadavati putem serijskog monitora (predefinisati 10sec).

Stepenišno svjetlo reaguje na detetekciju pokreta samo ako je mračno, odnosno ako se pomoću fotootpornika detektuje nizak nivo svjetlosti. (Pogodno za stepeništa koja osvjetljava dnevno svjetlo)

Svaki put kada se svjetlo uključi ili isključi na nekom spratu prema serijskom monitoru poslati poruku uključeno svjetlo na spratu [br], odnosno isključeno svjetlo na spratu [br].

**(4-3-2 boda).**

# PRIMJER ZA VJEŽBU

Naplatni punkt na autoputu.

Senzor pokreta raguje dok prilazi vozilo i omogućuje uključenje rasvjete ispred naplatnog šaltera i postavljen je na Arduino ploči A. Rasvjetu predstaviti sa dvije istobojne LED i postavljene su na Arduino ploču B. Rasvjetu uključivati samo ako je mračno, odnosno ako se pomoću fotootpornika detektuje nizak nivo svjetlosti.

Ultrasonični sensor, na ploči A, je postavljen iznad vozila na naplatnom punktu i mjeri minimalno rastojanje od vozila koje prolazi. Na osnovu tog podatka vrši klasifikaciju vozila na putničke automobile i teretna vozila (autobuse).

Rasvjetu držati uključenu sve dok vozilo ne prođe ispod ultrazvučnog senzora ili ne protekne maksimalno vrijeme predviđeno za prolazak (ne koristiti delay!!). Maximalno vrijeme zadavati putem seriskog monitora (predefinisati 60sek).

Informaciju o broju putničkih i teretnih vozila koja su prošla od početka nadzora, ispisati na LCD-u i serijskom monitoru, koji se nalazi na ploči B.

**(5-4-3 boda)**