

## 2. domaća zadaća

# Komunikacija Micro:bita

Aktivnost je zamišljena za učenike 7. razreda osnovne škole u sklopu nastavnog sadržaja „Mrežna komunikacija“ (Ishod A.7.1 ). Aktivnost može poslužiti za uvođenje ili utvrđivanje novih pojmova (Mrežno povezivanje, IP adrese, protokoli, ...).

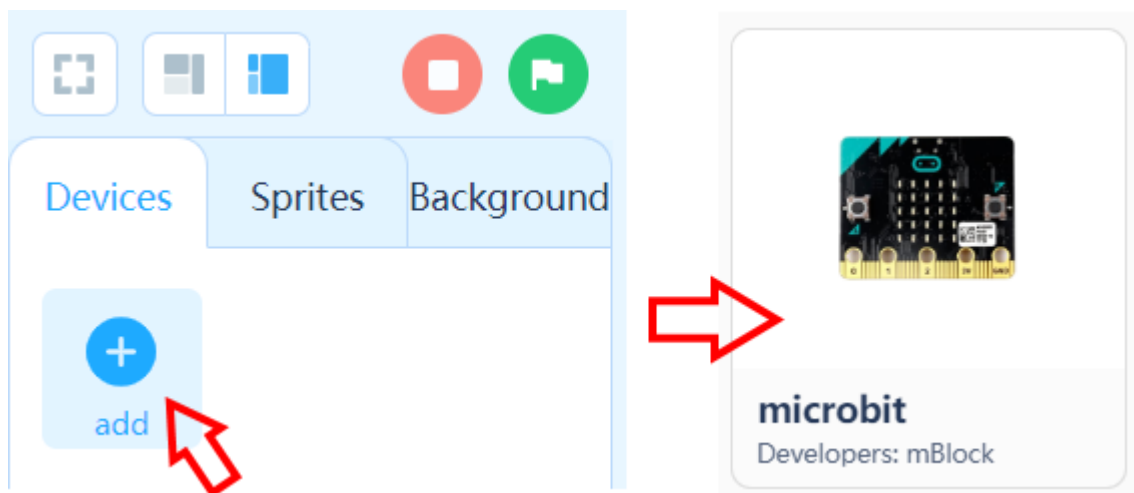
Aktivnost uključuje i rad s jednostavnim tipovima podataka (Ishod B.7.1).

Predviđeno potrebno vrijeme: 30 minuta

Za ovu aktivnost potrebna su minimalno 2 *Micro:bita*. Aktivnost ima smisla za 3 i više. Teoretski maksimalan broj je 25 *Micro:bita*. Te instaliran *mBlock 5* (v5.1.0).

## Priprema

1. Spojimo *Micro:bit* s računalom pomoću microUSB kabela.
2. Pokrenemo *mBlock 5* (v5.1.0).
3. Dodamo novi uređaj i odaberemo *Micro:bit*.



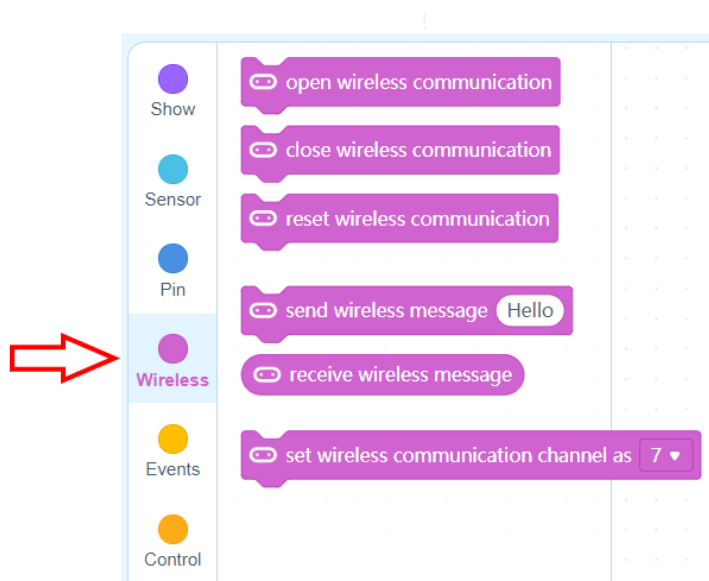
4. Proći kroz aktivnosti iz moje prve zadaće.

## Uvodna aktivnost „Pošalji mi poruku“

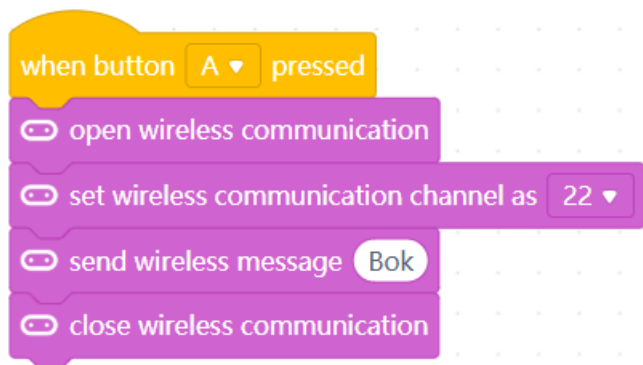
U ovoj aktivnosti ćemo poslati poruku s jednog *Micro:bita* na drugi. Pritiskom na tipku *A* prvi *Micro:bit* će poslati poruku koju će primiti drugi *Micro:bit* te ju prikazati pomoću mreže LED lampica.

Učenici ovu aktivnost mogu provesti u parovima, tako da jedan učenik prima, a drugi šalje poruku.

Blokovi za komunikaciju:



**1.** Kako bi uspješno poslali poruku pritiskom na tipku *A* trebamo prvo otvoriti bežičnu komunikaciju te postaviti kanal na kojem ćemo slati poruku. Nakon toga možemo poslati poruku i zatvoriti bežičnu komunikaciju.



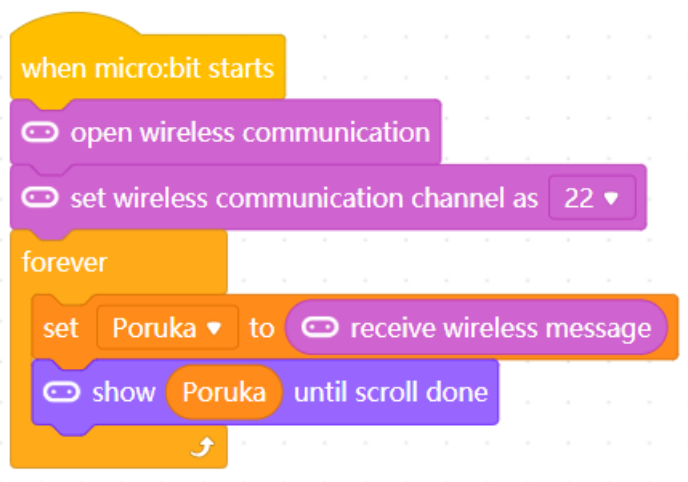
**2.** Kako bi uspješno primili poruku na drugom *Micro:bitu* također trebamo otvoriti bežičnu komunikaciju te odabrati **isti kanal** kao i *Micro:bit* koji nam šalje poruku. Za razliku od *Micro:bita* koji nam šalje poruku mi ne znamo kada će nam poruka stići. Zbog toga trebamo poruku primiti u beskonačnoj petlji.



Uploadamo program te isprobamo slanje poruke.

Primijetimo da *Micro:bit* kada nije primio poruku prikazuje *None*. Kada primi poruku prikazuje poruku. Prikazivanje poruke *None* ulazi u jedna od problema koji su detaljno opisano na kraju ovog dokumenta.

Dobro je poruku zapamtiti pomoću varijable:



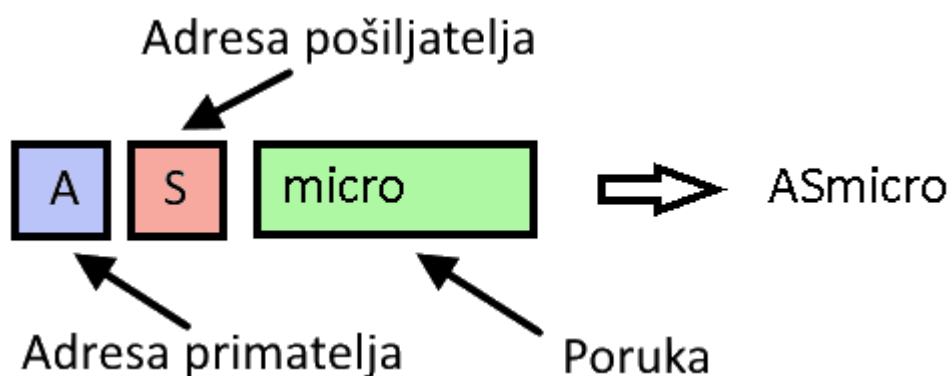
## Glavna aktivnost „Ping“

U ovoj aktivnosti nastavnikov *Micro:bit* ima ulogu servera, te se na njemu nalazi unaprijed pripremljeni program. Zadatak učenika je poslati i primiti poruku od servera te izmjeriti vrijeme koje je prošlo između slanja poruke i primanja odgovora.

Budući da će u ovoj aktivnosti, za razliku od prethodne, svi učenici imati mogućnost komunikacije s serverom na istom kanalu prije početka aktivnosti trebamo dogovoriti protokol za slanje i primanje poruka.

### Protokol:

1. Komuniciramo na kanalu 22.
2. Svaki od učenika treba odabrati svoju adresu. Adresa se sastoji od jednog slova engleske abecede. (A - Z) Sve adrese trebaju biti međusobno različite. Adresa servera je slovo S, dakle nitko drugi ne može odabrati slovo S kao svoju adresu.
3. Poruka (Paket) koji šaljemo treba biti sljedećeg oblika:



Možemo usporediti s shemom stvarnog paketa.

ZAGLAVLJE	PODATCI	PROVJERA
Adresa primatelja Adresa pošiljatelja	Podatci u binarnom obliku	Provjera duljine paketa, provjera prijenosa, itd.

4. Kada server na adresi *S* primi poruku „micro“ on će na nju odgovoriti porukom „bit“. (s odgovarajućim formatom zaglavlja)

### Mjerenje vremena:

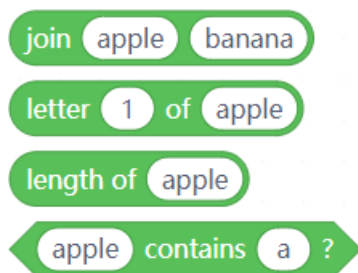
Micro:bit mjeri vrijeme koje je proteklo od njegovog uključivanja u milisekundama. Tom podatku možemo pristupiti pomoću bloka:

A blue Scratch block with a clock icon and the text "running time".

Ako neposredno prije slanja poruke spremimo vrijeme koje je prošlo od uključivanja u varijablu. Te isto napravimo neposredno nakon primanja poruke. Oduzimanjem druge varijable od prve dobivamo proteklo vrijeme u milisekundama.

### Rad sa stringovima:

Pomoću blokova za rad sa stringovima:



Možemo sastavljati poruku koju ćemo slati, odnosno ispitivati sadržaj primljene poruke.

### Zadatak 1.

Pošalji poruku serveru (pritiskom na tipku A) na adresi *S* prema gore dogovorenom protokolu. Izmjerite vrijeme proteklo između slanja poruke i primanja odgovora te ga prikažite pomoću mreže LED lampica. Na poruku „micro“ s odgovarajućim zaglavljem server će odgovoriti s „bit“. Ako odgovor ne primimo unutar 5 sekundi od slanja poruke smatramo da je poruka izgubljena.

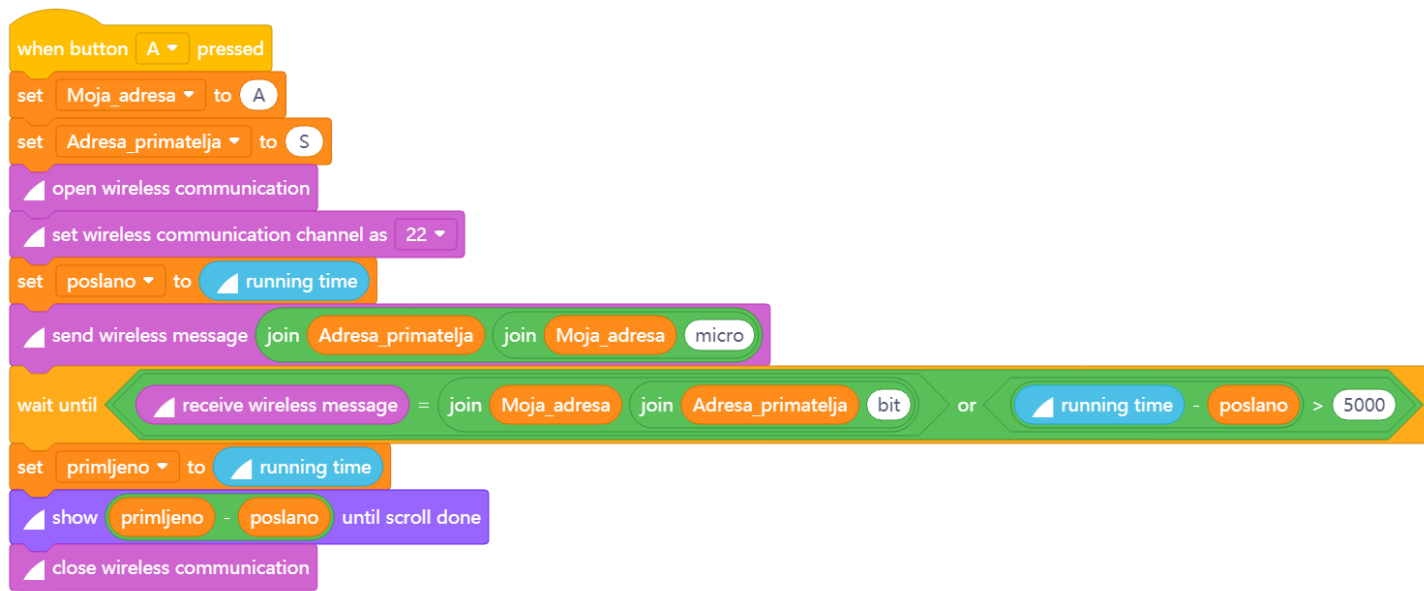
### Rješenje:

Odaberemo svoju adresu. Uvjerimo se da je naša adresa jedinstvena. Svoju adresu možemo pohraniti u posebnoj varijabli. Pretpostavimo da smo odabrali adresu „A“.

Pritiskom na tipku A otvorimo bežičnu komunikaciju, odaberemo kanal 22, zapamtimo vrijeme slanja i pošaljemo poruku oblika „SAmicro“.

Čekamo dok ne primimo odgovor odgovarajućeg formata („ASbit“) ili dok ne prođe 5 sekundi od slanja. Zapamtimo vrijeme primanja.

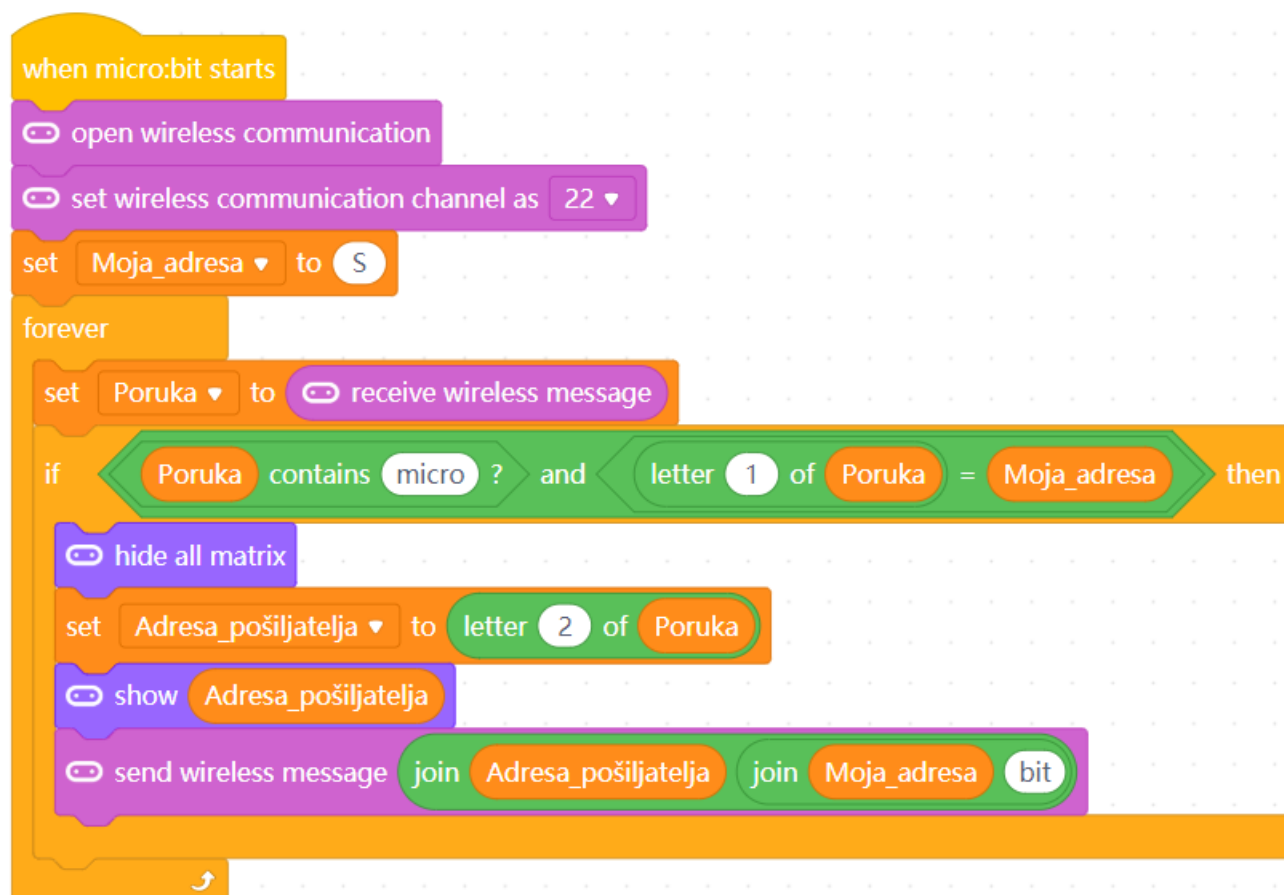
Izračunamo i prikažemo izmjereno vrijeme.



### Analiza programa za server:

Nakon što učenici uspiju izmjeriti vrijeme između slanja i primanja poruke (*latency*), zajedno s učenicima proanaliziramo program koji se nalazio na serveru.

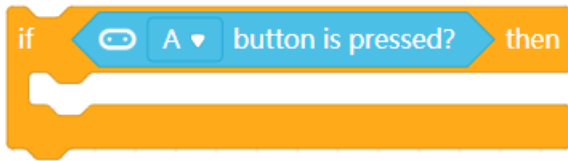
Server prima poruke u beskonačnoj petlji sve dok ne primi poruku koja sadržava „micro“ te ima adresu servera na prvom mjestu. Nakon toga prikazuje adresu pošiljatelja te generira i šalje odgovarajući odgovor.



## Zadatak 2.

Pokušajte poslati poruku jedni drugima prema dogovorenom protokolu. Za to vaš *Micro:bit* treba biti u mogućnosti slati i primati poruke. Proučite primjer servera iz prethodnog zadatka.

Savjet je koristiti ovaj blok za slanje poruke:



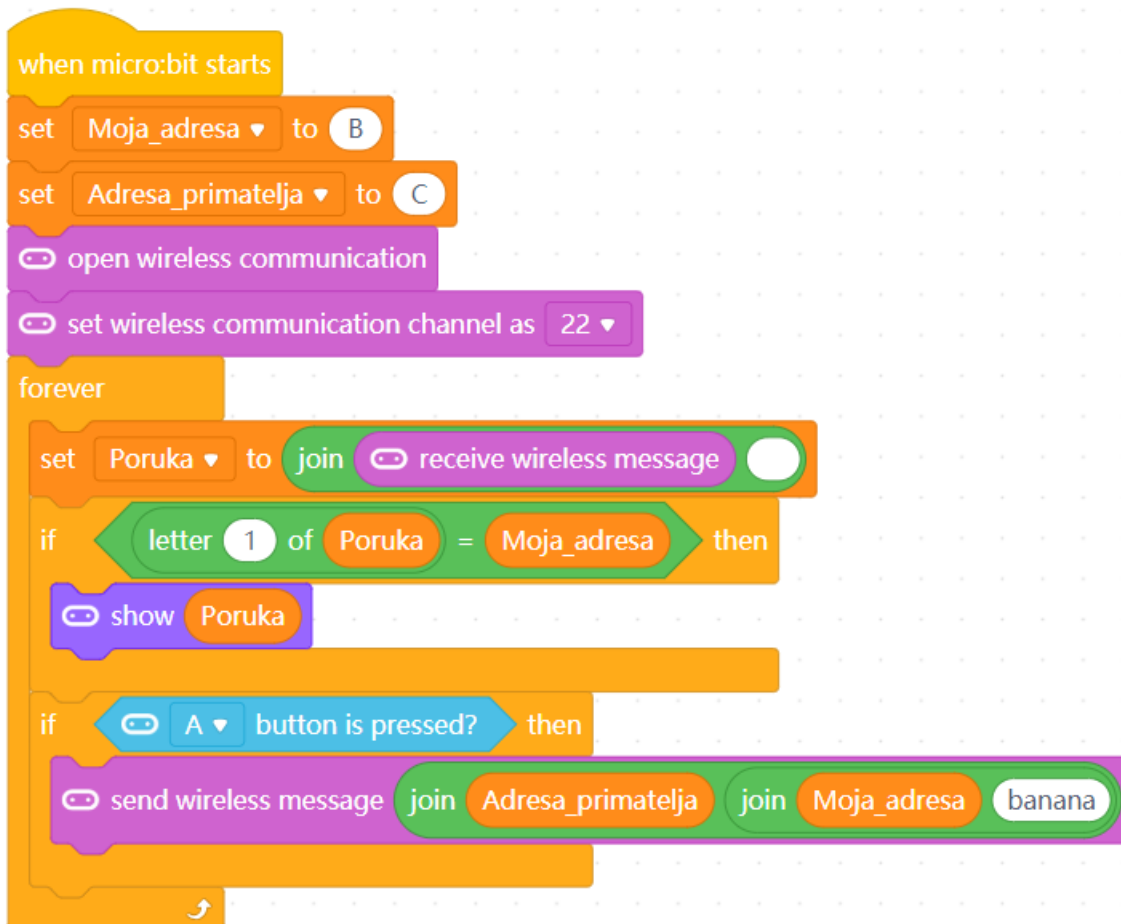
## Rješenje:

Možemo kreirati posebne varijable za adresu primatelja i pošiljatelja. Otvaramo bežičnu komunikaciju, postavljamo kanal 22, i u beskonačnoj petlji primamo poruke. Budući da kada nema poruka primamo poruku tipa *None* što može uzrokovati probleme pri radu sa stringovima možemo se poslužiti sljedećim trikom:



Provjerimo je li primljena poruka poslana nama. Ako je prikazemo poruku.

Isto tako na pritisak tipke A šaljemo poruku na adresu primatelja u odgovarajućem formatu.



## Moguće nadogradnje ove aktivnosti

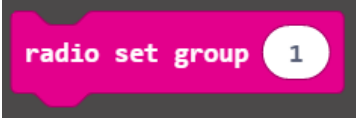

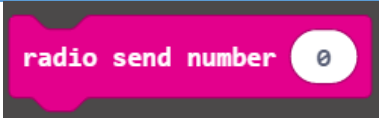

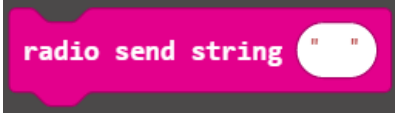
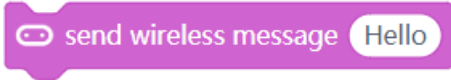
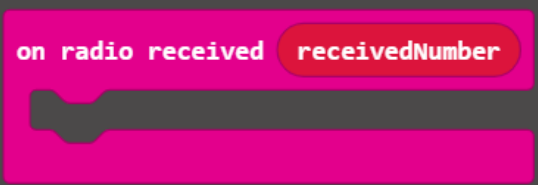
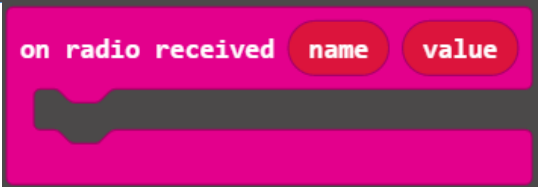
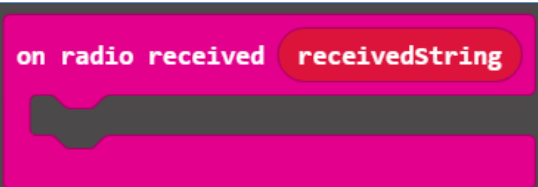
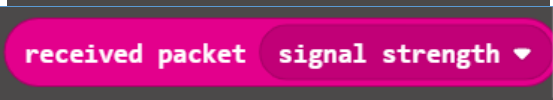

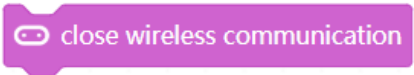
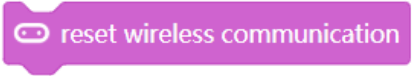

Dodati mehanizam za provjeru primanja poruke i ponovno slanje poruke u slučaju izgubljene poruke.

Napraviti blok za enkripciju poruke, na primjer, Cezarovom šifrom. Pokušati presresti šifriranu poruku koja se šalje između dva *Micro:bita* te napraviti dešifriranje.

Osmisliti sustav komunikacije Morseovom abecedom i tipkama A za slanje točke i B za slanje crtice.

## Detaljan opis problema

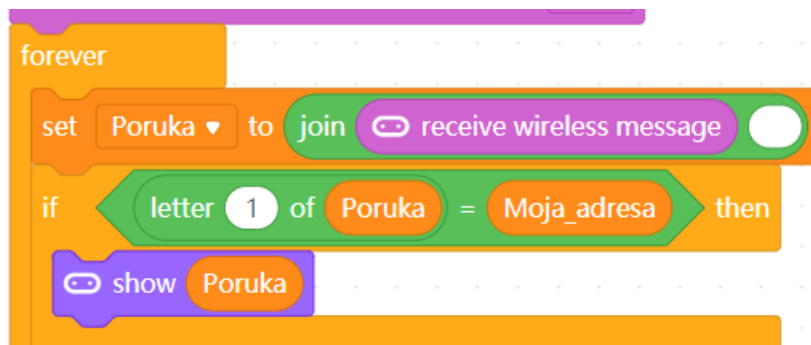
1. Glavni problem je nedostatak blokova u *Mblocku 5* za bežičnu komunikaciju. Na primjer *MakeCode* sadrži sljedeće blokove:

MakeCode	Mblock5
	 Ali nema mogućnost promjene kanala pomoću varijable
	Nema
	Nema
	
	Nema
	Nema
	Nema
	Nema
Nema, ali i ne treba	  
Nema, ali postiže istu funkcionalnost drugim blokovima	

To onemogućava korištenje kanala kao adrese u aktivnosti te smo primorani koristiti način komunikacije u kojem svi primaju sve poruke (možemo povezati s vrstom LAN mreže u kojoj sva računala komuniciraju pomoću sabirnice). Dolazi do potrebe za improvizacijom kod primanja poruke kako bi se izbjeglo da poruka tipa *None* uđe u operacije rada sa stringovima što će uzrokovati grešku.



```
while True:
    Poruka = radio.receive()
    if str(Poruka).find(str('micro')) > -1 and Poruka[0] == Moja_adresa:
```



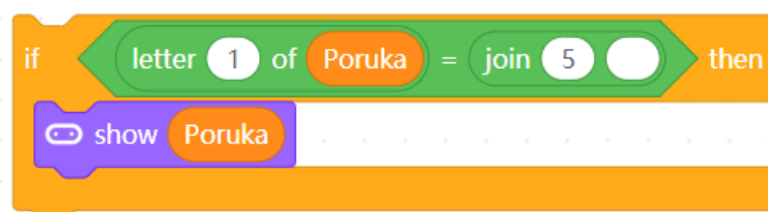
```
while True:
    Poruka = str(radio.receive()) + str('')
    if Poruka[0] == Moja_adresa:
```

2. Mješanje brojeva i stringova nije dobra ideja ako bi na primjer htjeli da su adrese iz aktivnosti cijeli brojevi („54micro“). Ovo će dati grešku:



```
if Poruka[0] == 5:
    display.scroll(str(Poruka), wait=False, loop=False)
```

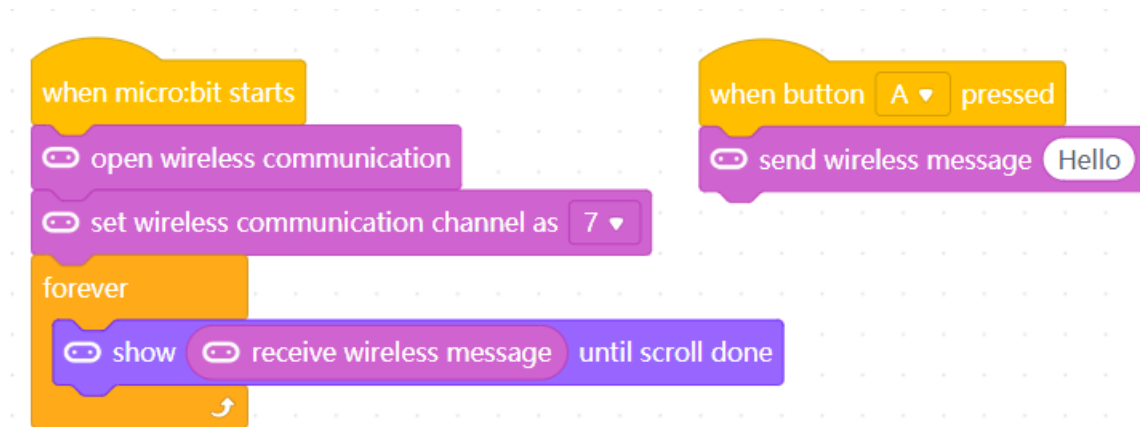
Ovo neće dati grešku.



```
if Poruka[0] == str('5') + str(''):
    display.scroll(str(Poruka), wait=False, loop=False)
```



3. Prema mojem testiranju, koje nije bilo temeljito, ovo ne radi:



Dok ovo radi:

