

💡 Common Branch Instructions (with Examples + বাংলা Explanation)

🟩 1 BEQ — Branch if Equal (Zero flag = 1)

👉 যখন দুইটা সংখ্যা সমান হয় তখন **jump** করবে

Example:

```
MOV r0, #5
MOV r1, #5
CMP r0, r1      @ 5 == 5 ? → হ্যাঁ, Zero flag = 1
BEQ equal_label @ তাই jump করবে equal_label এ

not_equal:
    B stop
equal_label:
    MOV r2, #100 @ এখানে আসবে কারণ r0 == r1
stop:
    B stop
```

🧠 বাংলা ব্যাখ্যা:

CMP রেজাল্ট সমান হলে (Zero flag সেট হলে), **BEQ** ব্রাঞ্চ নেয়।
অর্থাৎ, “equal” হলে লাফ দাও।

🟨 2 BNE — Branch if Not Equal (Zero flag = 0)

👉 দুইটা সংখ্যা সমান না হলে **jump** করবে

Example:

```
MOV r0, #7
MOV r1, #9
CMP r0, r1      @ 7 != 9 → Zero flag = 0
BNE not_equal   @ তাই jump করবে not_equal এ

equal:
```

```

    MOV r2, #0
    B stop
not_equal:
    MOV r2, #1      @ এখানে আসবে কারণ অসমান
stop:
    B stop

```

🧠 বাংলা ব্যাখ্যা:

CMP এর পরে যদি সমান না হয়, BNE কাজ করে।

3 BGT — Branch if Greater Than (Signed)

👉 $r0 > r1$ হলে jump করবে

Example:

```

    MOV r0, #8
    MOV r1, #5
    CMP r0, r1      @ 8 > 5
    BGT bigger

not_bigger:
    MOV r2, #0
    B stop
bigger:
    MOV r2, #1      @ এখানে jump হবে কারণ  $r0 > r1$ 
stop:
    B stop

```

🧠 বাংলা ব্যাখ্যা:

Signed comparison অনুযায়ী, $r0$ বড় হলে BGT লাফ দেয়।

4 BLT — Branch if Less Than (Signed)

👉 $r0 < r1$ হলে jump করবে

Example:

```

MOV r0, #2
MOV r1, #9
CMP r0, r1      @ 2 < 9
BLT smaller

not_smaller:
MOV r2, #0
B stop
smaller:
MOV r2, #1      @ এখানে আসবে কারণ ছোট
stop:
B stop

```

🧠 বাংলা ব্যাখ্যা:
যখন `r0` ছোট, তখন `BLT` jump করে।

📦 5 BGE — Branch if Greater or Equal (Signed)

👉 `r0 ≥ r1` হলে jump করবে

Example:

```

MOV r0, #5
MOV r1, #5
CMP r0, r1      @ 5 >= 5
BGE ok_label

fail:
MOV r2, #0
B stop
ok_label:
MOV r2, #1      @ jump করবে কারণ সমান
stop:
B stop

```

🧠 বাংলা ব্যাখ্যা:
যখন `r0` বড় বা সমান, তখন `BGE` ব্রাঞ্চ নেয়।

6 BLE — Branch if Less or Equal (Signed)

👉 $r0 \leq r1$ হলে jump করবে

Example:

```
MOV r0, #4
MOV r1, #6
CMP r0, r1      @ 4 <= 6
BLE less_or_eq
```

greater:

```
MOV r2, #0
B stop
```

less_or_eq:

```
MOV r2, #1
```

stop:

```
B stop
```

🧠 বাংলা ব্যাখ্যা:

ছোট বা সমান হলে BLE jump করে।

7 BHI — Branch if Higher (Unsigned)

👉 Unsigned comparison এ, $r0 > r1$ হলে jump করবে

Example:

```
MOV r0, #250
MOV r1, #200
CMP r0, r1      @ 250 > 200 (unsigned)
BHI high_label
```

low_label:

```
MOV r2, #0
B stop
```

high_label:

```
MOV r2, #1      @ unsigned বড় → jump
```

```
stop:
    B stop
```

🧠 বাংলা ব্যাখ্যা:

Unsigned সংখ্যায় বড় হলে **BHI** কাজ করে (যেমন $250 > 200$, এমনকি byte overflow হলেও)।

📌 8 BLS — Branch if Lower or Same (Unsigned)

👉 Unsigned comparison এ, $r0 \leq r1$ হলে jump করবে

Example:

```
MOV r0, #3
MOV r1, #8
CMP r0, r1
BLS lower_label

higher_label:
    MOV r2, #0
    B stop
lower_label:
    MOV r2, #1      @ unsigned ছোট → jump
stop:
    B stop
```

🧠 বাংলা ব্যাখ্যা:

Unsigned সংখ্যায় ছোট বা সমান হলে **BLS** jump করে।

📌 9 B — Unconditional Branch

👉 সবসময় jump করে

Example:

```
start:
    MOV r0, #0
    ADD r0, r0, #1
    B start      @ সবসময় start এ ফিরে যায় (infinite loop)
```

✖ উদাহরণ: BX LR in ARM Cortex-M3 Assembly

```
.syntax unified
.cpu cortex-m3
.thumb
.global main

main:
    MOV r0, #10          @ r0 = 10
    BL add_five          @ add_five() ফাংশনে যাও (return address LR-এ
সংরক্ষিত হবে)

    @ ফাংশন থেকে ফিরে এলে এখানে আসবে
    @ এখন r0 = 15 (কারণ ফাংশনে 5 যোগ করা হয়েছিল)
stop:
    B stop              @ প্রোগ্রাম থামিয়ে রাখো

add_five:
    ADD r0, r0, #5       @ r0 = r0 + 5
    BX lr               @ ফাংশন থেকে ফিরে যাও (return to main)
```

🧠 বাংলা ব্যাখ্যা:

B মানে unconditional jump — মানে যাই হোক, লাফ দাও ওই label এ।

🔪 Summary Table

Instruction	কাজ করে যখন	Condition flag ভিত্তি
BEQ	সমান (Equal)	Z = 1
BNE	অসমান (Not Equal)	Z = 0
BGT	বড় (Signed)	Z=0, N=V

BLT	ছোট (Signed)	$N \neq V$
BGE	বড় বা সমান (Signed)	$N=V$
BLE	ছোট বা সমান (Signed)	$Z=1$ বা $N \neq V$
BHI	বড় (Unsigned)	$C=1, Z=0$
BLS	ছোট বা সমান (Unsigned)	$C=0$ বা $Z=1$
B	সবসময় Jump	—