

activity_main:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:id="@+id/main"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     android:orientation="vertical"
9     tools:context=".MainActivity"
10    android:padding="50dp"
11    android:background="#2D142C">
12    <Chronometer
13        android:id="@+id/textTime"
14        android:layout_width="wrap_content"
15        android:layout_height="wrap_content"
16        android:textSize="80sp"
17        android:layout_gravity="center"
18        android:textColor="#C72C41"/>
19
20    <androidx.appcompat.widget.AppCompatButton
21        android:id="@+id/btnStart"
22        android:layout_width="wrap_content"
23        android:layout_height="wrap_content"
24        android:text="Start"
25        android:layout_gravity="center"
26        android:layout_margin="25dp"
27        android:background="#EE4540"/>
28
29    <androidx.appcompat.widget.AppCompatButton
30        android:id="@+id/btnPause"
31        android:layout_width="wrap_content"
32        android:layout_height="wrap_content"
33        android:text="Pause"
34        android:layout_gravity="center"
35        android:background="#EE4540"/>
36
37    <androidx.appcompat.widget.AppCompatButton
38        android:id="@+id/btnReset"
39        android:layout_width="wrap_content"
40        android:layout_height="wrap_content"
41        android:text="Reset"
42        android:layout_gravity="center"
43        android:layout_margin="25dp"
44        android:background="#EE4540"/>
45</LinearLayout>
```

MainActivity:

```
12 class MainActivity : AppCompatActivity() {
13     lateinit var chronometer: Chronometer
14     var running: Boolean = false
15     var offset: Long = 0
16
17     val OFFSET_KEY = "offset"
18     val RUNNING_KEY = "running"
19     val BASE_KEY = "base_key"
```

```

31     chronometer = findViewById(R.id.textTime)
32     val btnStart: Button = findViewById(R.id.btnStart)
33     val btnPause: Button = findViewById(R.id.btnPause)
34     val btnReset: Button = findViewById(R.id.btnReset)
35
36     btnStart.setOnClickListener {
37         if (!running) {
38             setBaseTime()
39             chronometer.start()
40             running = true
41         }
42     }
43
44     btnPause.setOnClickListener {
45         if (running) {
46             saveOffset()
47             chronometer.stop()
48             running = false
49         }
50     }
51
52     btnReset.setOnClickListener {
53         offset = 0
54         setBaseTime()
55         running = false
56     }
57
58     if (savedInstanceState != null) {
59         offset = savedInstanceState.getLong(OFFSET_KEY)
60         running = savedInstanceState.getBoolean(RUNNING_KEY)
61
62         if (running) {
63             chronometer.base = savedInstanceState.getLong(BASE_KEY)
64             chronometer.start()
65         } else setBaseTime()
66     }
67 }

```

```
69  override fun onSaveInstanceState(savedInstanceState: Bundle) {  
70      super.onSaveInstanceState(savedInstanceState)  
71  
72      savedInstanceState.putLong("offset",offset)  
73      savedInstanceState.putBoolean("running",running)  
74      savedInstanceState.putLong("base_key",chronometer.base)  
75  }  
76  
77  private fun setBaseTime() {  
78      chronometer.base = SystemClock.elapsedRealtime() - offset  
79  }  
80  
81  private fun saveOffset() {  
82      offset = SystemClock.elapsedRealtime() - chronometer.base  
83  }  
84  }
```