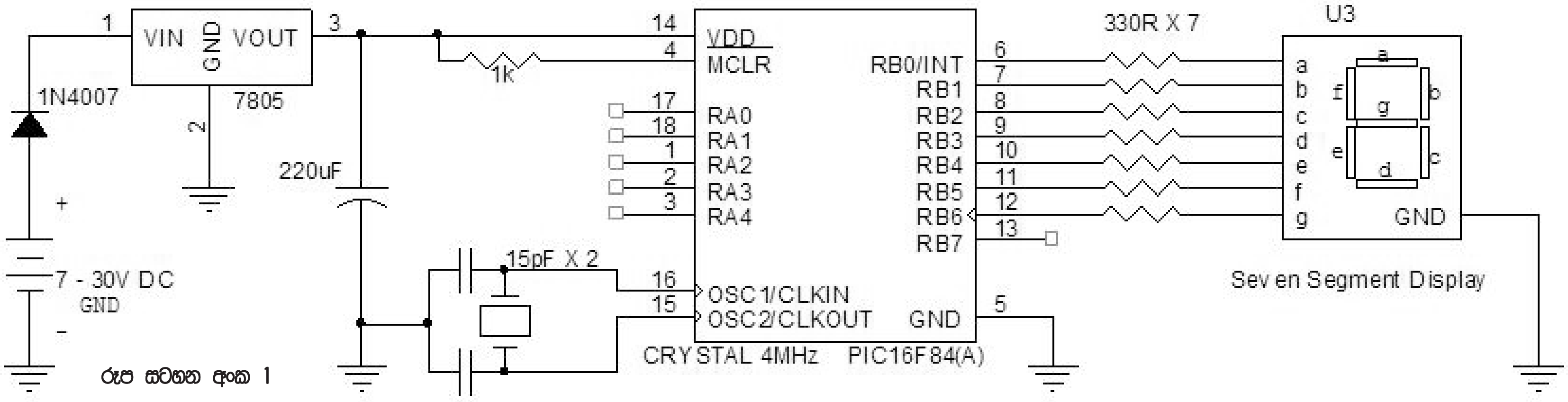
# MICROCONTROLLERS

14 ලිපිය - 5 කොටස

## සප්ත ඛණ්ඩ ප්‍රදර්ශක හැසිරවීම

මෙම ලිපිය තුළින් අප ඔබ වෙත ගෙන ඒමට බලාපොරොත්තු වනුයේ සප්තඛණ්ඩ ප්‍රදර්ශක (Seven Segment Display) සම්බන්ධ නිර්මාණයකි. සය වන ලිපියේ පළමු කොටසින් දැක්වුණු පරිපථය ම මෙහි දී ද යොදා ගැනේ. රූප සටහන අංක 1 බලන්න. එහි PIC 16F84(A) මයික්‍රොකොන්ට්‍රෝලරයේ B තොටුපළට සප්තඛණ්ඩ ප්‍රදර්ශකය සවි කර ඇත. සප්තඛණ්ඩ ප්‍රදර්ශකය පොදු කැතෝඩ වර්ගයේ එකක් විය යුතු ය. එනම් එක් එක් ඛණ්ඩ සඳහා යොදා ඇති LEDවල කැතෝඩ එකට සම්බන්ධ කර GND අග්‍රය හරහා පිටතට ගෙන ඇත. එම ඛණ්ඩ විවිධ සංයෝජනයන් ගෙන් දැල්වීම මගින් ඉලක්කම් ප්‍රදර්ශනය කරවා ගත හැකි ය. එම ඛණ්ඩක නම් කිරීම සඳහා සම්මුතියක් තිබේ. ඒ අනුව රූප සටහන අංක 1හි තිබෙන සප්ත ඛණ්ඩ ප්‍රදර්ශකයේ ඛණ්ඩක a b c d e f සහ g ලෙස නම් කර තිබේ.




රූප සටහන අංක 1

රූප සටහන අංක 2න් දැක්වෙන වගුව 0 සිට 9 දක්වා වූ එක් එක් ඉලක්කම ප්‍රදර්ශනය කිරීම සඳහා දැල්විය යුතු ඛණ්ඩක සහ ඒ සඳහා B තොටුපළට ලිවිය යුතු තාර්කික අගයන් නිරූපණය කෙරේ. උදාහරණයක් ලෙස "0" ප්‍රදර්ශනය කළ යුතු නම් ඒ සඳහා a b c d e සහ f යන ඛණ්ඩක දැල්විය යුතු ය. එනම් B තොටුපළෙහි RB0, RB1, RB2, RB3, RB4 සහ RB5 අග්‍ර තාර්කික 1 බවට පත් කළ යුතු වේ. ඒ සඳහා B තොටුපළට හෙවත් PORT B රෙජිස්ටරයට ලිවිය යුතු බිටු සැකැස්ම 00111111 වේ.

රූප සටහන අංක 3න් අදාළ ක්‍රමලේඛය දැක්වේ. එය C භාෂාව යොදාගෙන ගොඩනංවා ඇත. දැන් අපි එම ක්‍රමලේඛයේ එන එක් එක් උපදෙස තේරුම් ගැනීමට උත්සාහ කරමු. එම ක්‍රමලේඛයේ මුලින් ම දක්නට ලැබෙන Void ssdecode (int i); මගින් ssdecode නමැති උපදෙස් ගොනුවක් ඉදිරියේ දී දක්නට ලැබෙන බව හඟවයි. එම උපදෙස් ගොනුව භාවිත කරන විට එයට යම්කිසි අගයක් ද ලබා දිය යුතු ය. එම අගය integer වර්ගයේ එකක් බවත් එය අදාළ උපදෙස් ගොනුව මගින් i ලෙස නම් කරගන්නා බවත් හඟවයි. integer යනු බිටු 16කින් දක්වන ඉලක්කමකි. එමගින් 0 සිට $2^{16}$ (65536) දක්වා පූර්ණ අගයන් නිරූපණය කළ හැකි වේ. මෙම උපදෙස් ගොනුව පහළ දක්නට ලැබේ. එම උපදෙස් ගොනුව මගින් එක් එක් ඉලක්කමට අදාළව දැල්විය යුතු ඛණ්ඩක දල්වනු ලබයි. ඒ පිළිබඳ විස්තරයක් මෙම ලිපියේ අවසානයට යෙදේ.

int number; යනු number නමැති විචල්‍යය හඳුන්වා දීමයි. අප විසින් යම්කිසි විචල්‍යයක් භාවිත කෙරෙන්නේ නම් එය මුලින් ම හඳුන්වා දිය යුතු ය. ඒ අනුව මෙම number නමැති විචල්‍යය interger වර්ගයේ එකකි. ඉන්පසුව main නමැති ප්‍රධාන උපදෙස් ගොනුව දක්නට ලැබේ. එහි දී මුලින් ම TRISB රෙජිස්ටරයට "0" ලියා B තොටුපළ ප්‍රතිදාන ලෙස සකසා ඇත. ඉන්පසුව number විචල්‍යය "0" බවට පත් කොට ඇත.

එසේ සිදු කොට ඇත්තේ මුලින් ම "0" ප්‍රදර්ශනය කිරීම සඳහා ය. ඉන්පසුව while(1) හෙවත් නිරන්තරයෙන් ධාවනය වන උපදෙස් ගොනුව දක්නට ලැබේ. ඒ තුළ තිබෙන ssdecode (number); උපදෙස මගින් කියවෙනුයේ ssdecode නමැති උපදෙස් ගොනුවට ගොස් number නමැති විචල්‍යයෙන් දැක්වෙන අගය (මේ අවස්ථාවේ දී නම් "0") ප්‍රදර්ශනය කරන ලෙසයි. එසේ ප්‍රදර්ශනය කිරීමෙන් පසුව නැවත ප්‍රධාන උපදෙස් ගොනුවේ තිබෙන ඊළඟ උපදෙස වන number, mumber + 1; උපදෙස මගින් mumber විචල්‍යයේ අගය 1කින් වැඩි කෙරේ. එනම් කලින් තිබූ අගය 0 නම් දැන් එය 1 බවට පත් වේ. ඊළඟ අවස්ථාවේ දී ප්‍රදර්ශනය කෙරෙනුයේ මෙම නව අගයයි. එලෙස වාර කිහිපයක් යාමේ දී එම විචල්‍යයේ අගය 10 විය හැකි ය. එක් සප්තඛණ්ඩ ප්‍රදර්ශකයක් මගින් දහය නිරූපණය කළ නොහැකි බැවින් නැවත "0" සිට පටන්ගැනීම සඳහා if(number>9) number=0; උපදෙස යොදාගෙන ඇත. ඉන් පසුව තත්පරයක කාල පමාවක් ලබාගැනීම සඳහා Delay_ms (1000); උපදෙස යොදාගෙන ඇත.

දැන් අපි ssdecode නමැති උපදෙස් ගොනුව ක්‍රියාත්මක වන්නේ කෙසේ දැයි බලමු. මෙම උපදෙස් ගොනුව භාවිත කිරීමේ දී එයට අගයක් ලබා දිය යුතු බව කලින් සඳහන් විය. එලෙස ම එම උපදෙස් ගොනුවට ප්‍රවේශ වීම සඳහා ssdecode(number); උපදෙස ප්‍රධාන ගොනුවේ දී භාවිත විය.

එහි number යනු අප විසින් ssdeccde උපදෙස් ගොනුවට සැපයෙන අගයයි. එම අගය 0 සිට 9 දක්වා වෙනස් විය හැකි ය. එලෙස ලැබෙන අගය ssdecode උපදෙස් ගොනුව තුළ දී i නමැති විචල්‍යයකට මාරු කෙරේ. ඉන් පසුව එම අගය අනුව switch(i) යටතේ තිබෙන Case අවස්ථාවලින් ගැලපෙන අවස්ථාව තෝරාගැනීමට භාවිත වේ. උදාහරණයක් ලෙස number විචල්‍යයේ අගය 5 නම් එම අගය i විචල්‍යයට ගෙන switch(i) යටතේ එන Case 5 නමැති ස්ථානයට ගමන් කෙරේ. ඉන් පසුව Case 3 යටතේ එන PORT B = ab01101101; උපදෙස මගින් පහේ ඉලක්කම ප්‍රදර්ශනය කිරීමට අදාළ බිටු සැකැස්ම PORTB රෙජිස්ටරයට ලියනු ලබයි. ඉන් පසුව තිබෙන break; උපදෙසට අනුව මෙම switch කොටසින් සහ ssdecode උපදෙස් ගොනුවෙන් ඉවත් වී ආපසු ප්‍රධාන උපදෙස් ගොනුවේ ගමන් කරයි. මෙලෙස යම් කිසි උපදෙස් ගොනුවකට අගයන් යැවීම Parameter passing for functions ලෙස හැඳින්වේ. මෙය 6.1 ලිපියේ දී විස්තර කළ lookup table වැනි ම ක්‍රියාවකි.

| Number | Segments to be ON | Value on PORTB |
|---|---|---|
| 0 | a b c d e f | 00111111 |
| 1 | b c | 00000110 |
| 2 | a b d e g | 01011011 |
| 3 | a b c d g | 01001111 |
| 4 | b c f g | 01100110 |
| 5 | a c d f g | 01101101 |
| 6 | a c d e f g | 01111101 |
| 7 | a b c | 00000111 |
| 8 | a b c d e f g | 01111111 |
| 9 | a b c f g | 01100111 |

රූප සටහන අංක 2

```
void ssdecode(int i);
int number;

void main()
{
  TRISB  = 0;   //PORTB output
  number = 0;   //Start from 0

 while (1)
 {    ssdecode(number);
      number = number +1;
        if(number>9) number=0;
      Delay_ms(1000);
  }
}

void ssdecode(int i)
{
     switch (i)
     {
     case 0: PORTB = 0b00111111; break;
     case 1: PORTB = 0b00000110; break;
     case 2: PORTB = 0b01011011; break;
     case 3: PORTB = 0b01001111; break;
     case 4: PORTB = 0b01100110; break;
     case 5: PORTB = 0b01101101; break;
     case 6: PORTB = 0b01111101; break;
     case 7: PORTB = 0b00000111; break;
     case 8: PORTB = 0b01111111; break;
     case 9: PORTB = 0b01101111; break;
     }
}
```

රූප සටහන අංක 3

මොරටුව විශ්වවිද්‍යාලයයේ විද්‍යුත් හා විදුලි සංදේශ අංශයේ
ගාමිණී ජයසිංහ
කෝලිත ධර්මප්‍රිය