uPLC example programs - Instruction List (IL)

In order to write or edit the program please use the uPLC editor inside the NiLAB Starter software :https://www.nilab.at/dokuwiki/doku.php?id=nilab_starter:uplc_editor

Example 1 - Copy digital input to digital output

; IN0 (bit0 of reg 4620) → OUT0 (bit0 of reg 4621)

LD 4620, 0 ; load bit0 of digital input register
OUT 4621, 0 ; write result to bit0 of digital output register
END

Example 2 — Inverted input

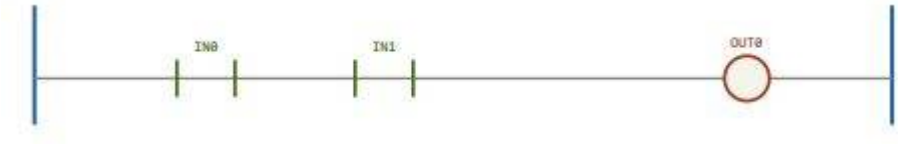
; OUT1 is ON when IN0 is OFF (NC contact logic)

LDN 4620, 0 ; load NOT(bit0 of digital input)
OUT 4621, 1 ; write result to bit1 of digital output register
END

Example 3 — AND logic between two inputs

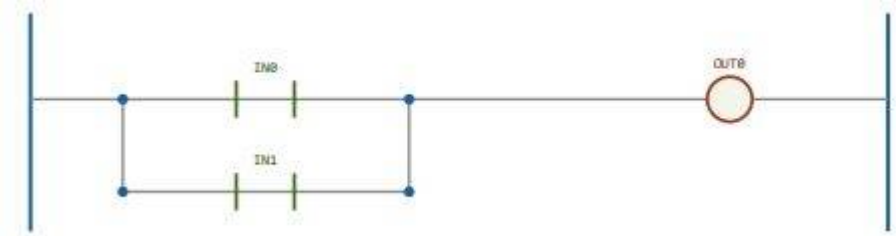
; OUT0 = IN0 AND IN1

LD 4620, 0 ; load IN0
AND 4620, 1 ; AND with IN1
OUT 4621, 0 ; write result to OUT0
END



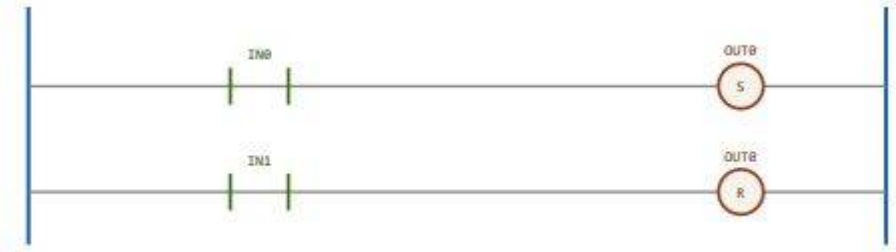
Example 4 — OR logic between two inputs

```
; OUT0 = IN0 OR IN1

LD 4620, 0 ; load IN0
OR 4620, 1 ; OR with IN1
OUT 4621, 0 ; write result to OUT0
END
```



Example 5 — SET/RESET latch

```
; SET: IN0 sets OUT0
; RESET: IN1 resets OUT0
; output holds its state when both inputs are OFF

LD 4620, 0 ; load IN0
SET 4621, 0 ; if stack=1, set OUT0 (latched ON)

LD 4620, 1 ; load IN1
RES 4621, 0 ; if stack=1, reset OUT0 (latched OFF)
END
```

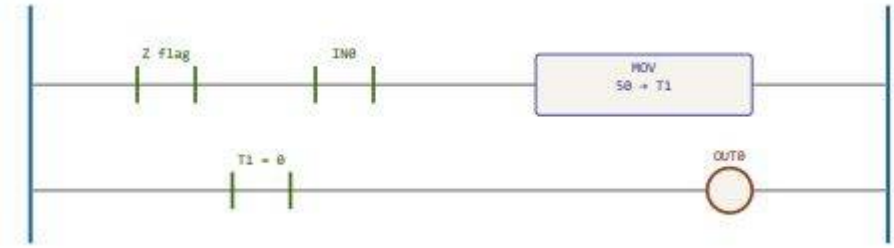


Example 6 — On-delay timer

```
; When IN0 goes high, load Timer1 with 50 scan cycles (~300ms)
; OUT0 turns ON when the timer expires

; Reload timer only when IN0 is high and timer has expired
LD 8671, 3 ; load Z flag: set when Timer1 == 0 (expired)
AND 4620, 0 ; AND with IN0
MOV 8678, 8028 ; reload Timer1 with constant 10 (reg 8678 = 10) ; replace 8678 with a user register
```

for longer delays

LD 8671, 0 ; load bit0 of status register: Timer1 expired
OUT 4621, 0 ; OUT0 ON when timer has expired
END

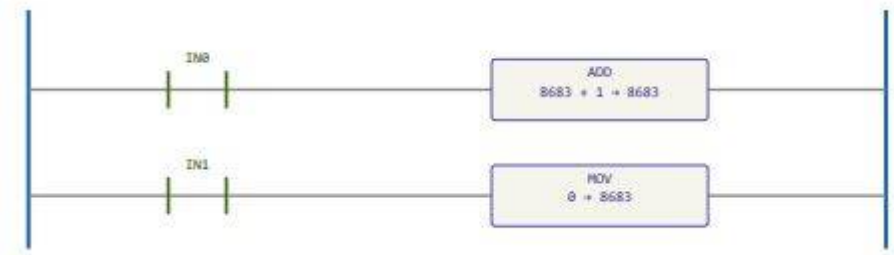


Example 7 — Cycle counter

; Increment user register 8683 each scan cycle when IN0 is active

LD 4620, 0 ; condition: IN0 active
ADD 8683, 8676, 8683 ; 8683 = 8683 + 1 (constant 1 stored in reg 8676)

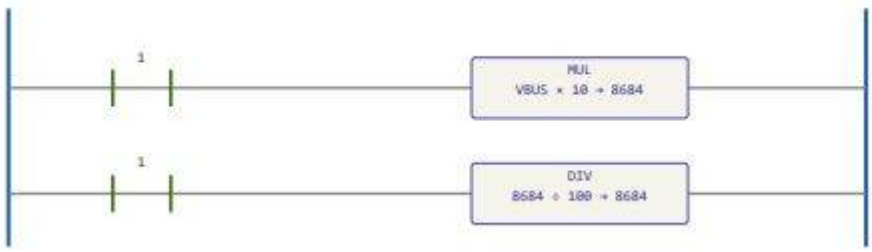
; Reset counter when IN1 is active
LD 4620, 1 ; load IN1
MOV 8675, 8683 ; 8683 = 0 (constant 0 stored in reg 8675)
END



Example 8 — Analog value scaling

; Scale VBUS voltage (reg 4631) by factor 1/10
; formula: 8684 = 4631 * 10 / 100

LD 8676, 0 ; stack = 1 (always true, constant 1 in reg 8676)
MUL 4631, 8678, 8684 ; 8684 = VBUS * 10 (constant 10 in reg 8678)
DIV 8684, 8679, 8684 ; 8684 = (VBUS * 10) / 100 (constant 100 in reg 8679)
END

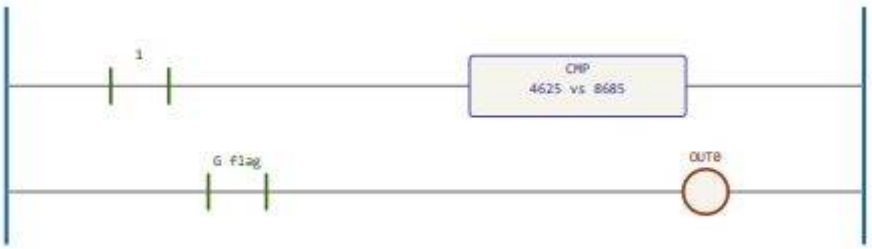


Example 9 — CMP: output when position exceeds threshold

; OUT0 turns ON when actual motor position (reg 4625, low word)
; exceeds the threshold value stored in user register 8685

LD 8676, 0 ; stack = 1 (always true)
CMP 4625, 8685 ; compare position vs threshold sets bit4 (G flag) in reg 8671 if 4625 > 8685

LD 8671, 4 ; load G flag (bit4): 1 if position > threshold
OUT 4621, 0 ; OUT0 ON when position exceeds threshold
END



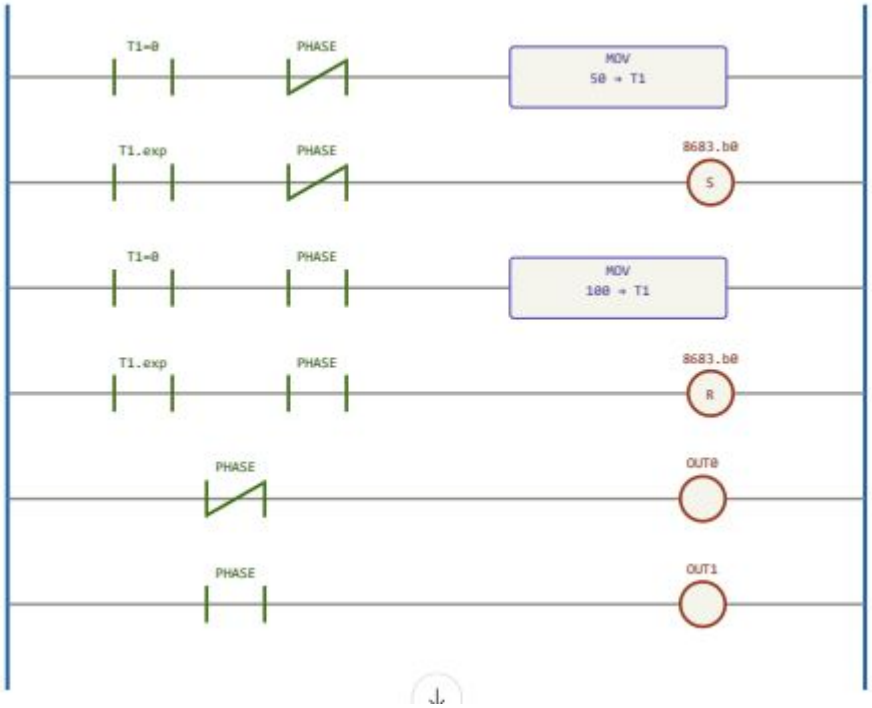
Example 10 — Two-phase sequencer with timers

; Automatic two-phase cycle:
; Phase 1: OUT0 ON for 50 scan cycles (~300ms)
; Phase 2: OUT1 ON for 100 scan cycles (~600ms)
; Sequence repeats continuously
;
; Variables used:
; reg 8683 bit0 = current phase flag (0=phase1, 1=phase2)
; Timer1 (reg 8028) controls both phase durations

; — Reload timer when expired and we are in phase 1 —
LD 8671, 3 ; Z flag: Timer1 == 0 (expired)
ANDN 8683, 0 ; AND NOT phase flag (we are in phase 1)
MOV 8678, 8028 ; reload Timer1 with 10 cycles ; replace 8678 with a user register set to 50 in production

; — Switch to phase 2 when Timer1 expires during phase 1 —
LD 8671, 0 ; Timer1 expired (bit0 of status reg)

```
ANDN 8683, 0 ; we are currently in phase 1
SET 8683, 0 ; set phase flag → enter phase 2

; — Reload timer for phase 2 duration —
LD 8671, 3 ; Timer1 expired
AND 8683, 0 ; we are currently in phase 2
MOV 8679, 8028 ; reload Timer1 with 100 cycles (constant 100 in reg 8679)

; — Switch back to phase 1 when Timer1 expires during phase 2 —
LD 8671, 0 ; Timer1 expired
AND 8683, 0 ; we are currently in phase 2
RES 8683, 0 ; clear phase flag → return to phase 1

; — Drive outputs based on current phase —
LDN 8683, 0 ; NOT phase flag = we are in phase 1
OUT 4621, 0 ; OUT0 ON during phase 1

LD 8683, 0 ; phase flag set = we are in phase 2
OUT 4621, 1 ; OUT1 ON during phase 2

END
```

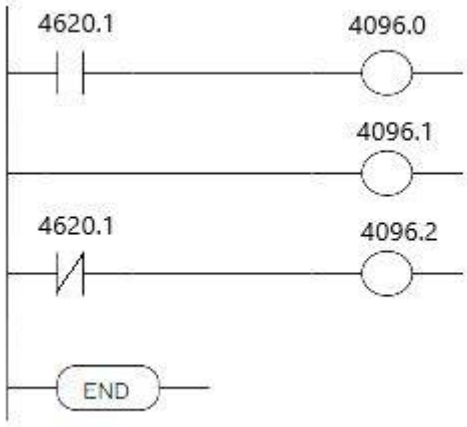


Example 11 - Enable the motor using digital input 1

This is a simple program to enable the motor with digital input 1. Parameter 4620 is the status of the digital inputs, Parameter 4096 is the command word of the drive (3⇒motor enable, 4⇒ motor disable)

```
LD 4620.1
OUT 4096.0
OUT 4096.1
```

```
LDn 4620.1
OUT 4096.2
END
```



Example 12 - Enable the motor after 6 seconds from plc start

This program set the Timer 1 to 1000 (using constant parameter 8680) and a flag bit (8683.0) to start counting 6 x 1000 = 6 secs.
After 6 seconds Digital output 0 is set to 1 and the motor is enabled using control word (Parameter 4096).

```
LD 8671.0
ANDn 8683.0
SET 8683.0
ADD 8028 8680 8028
LD 8671.0
AND 8683.0
OUT 4621.0
LD 4621.0
OUT 4096.0
END
```