

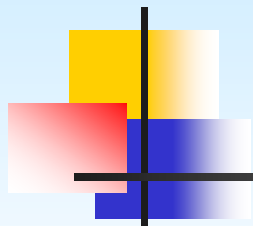
ПРИЛОЖЕН СЕМИНАР ПО ЕЛЕКТРОНИКА И  
СОФТУЕР

РУСЕНСКИ УНИВЕРСИТЕТ „АНГЕЛ КЪНЧЕВ”  
2015 г.

## **Управление на манипулатор РОБКО 01**

автор: Симеон Иванов

гр.Русе, 2015



## ЦЕЛ

---

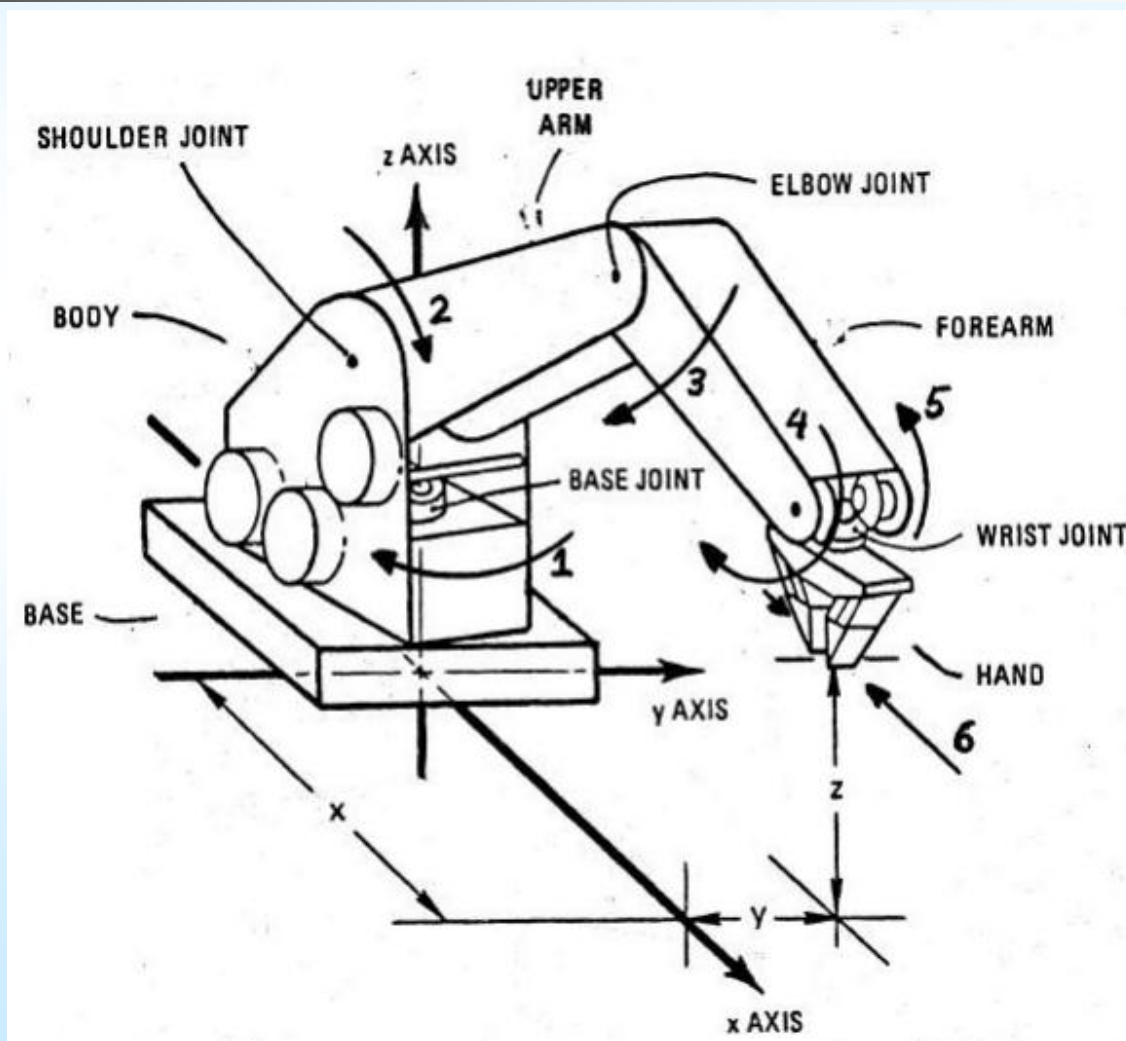
Презентацията представя роботизирания манипулатор **РОБКО 01** и неговото управление, реконструирано със съвременни технически средства

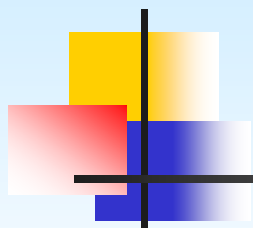
Представени са управляващата печатна платка и софтуер, позволяващи изпълнение на базови **G-команди**

# РОБКО 01 – Минало и настоящее

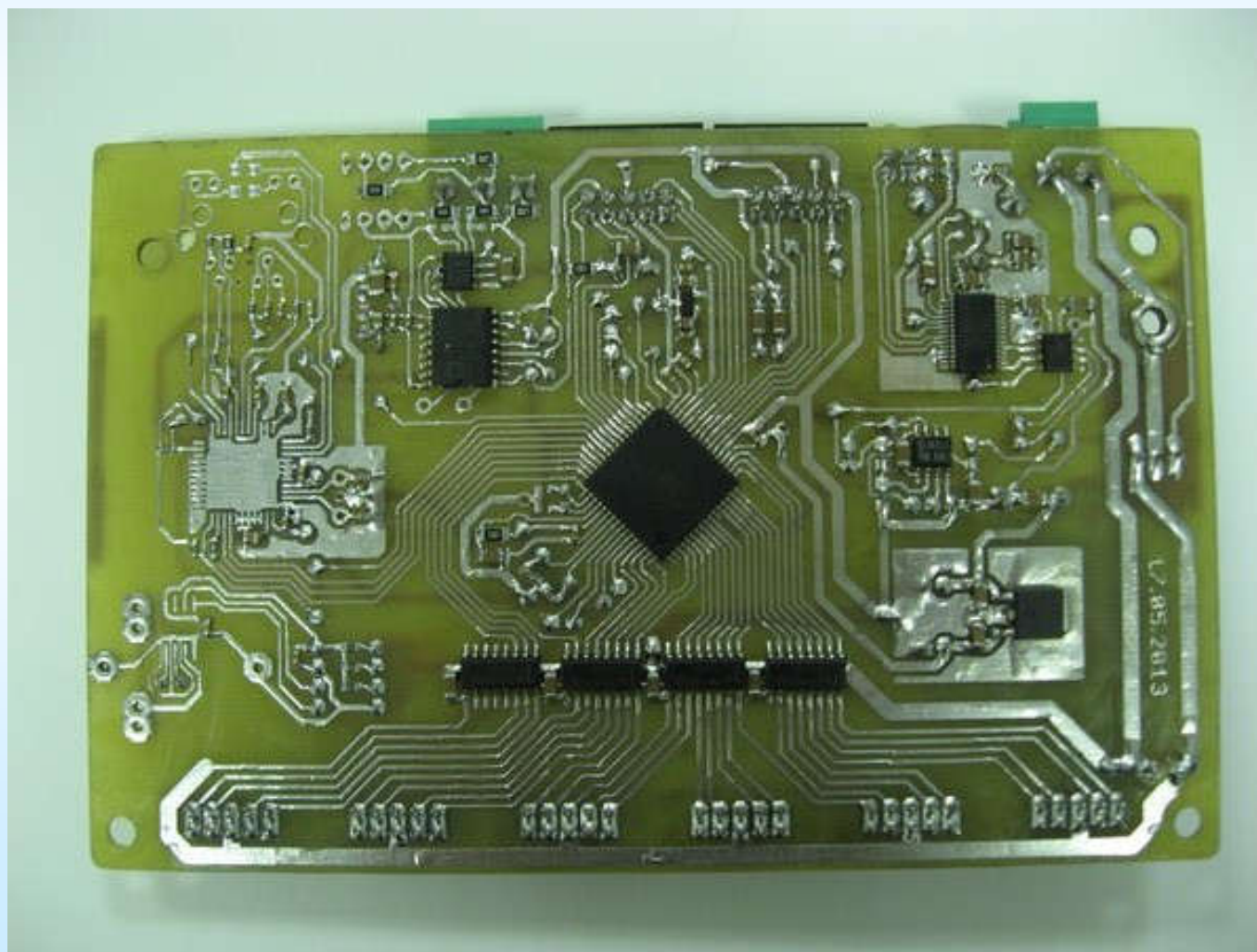


# Движения на РОБКО

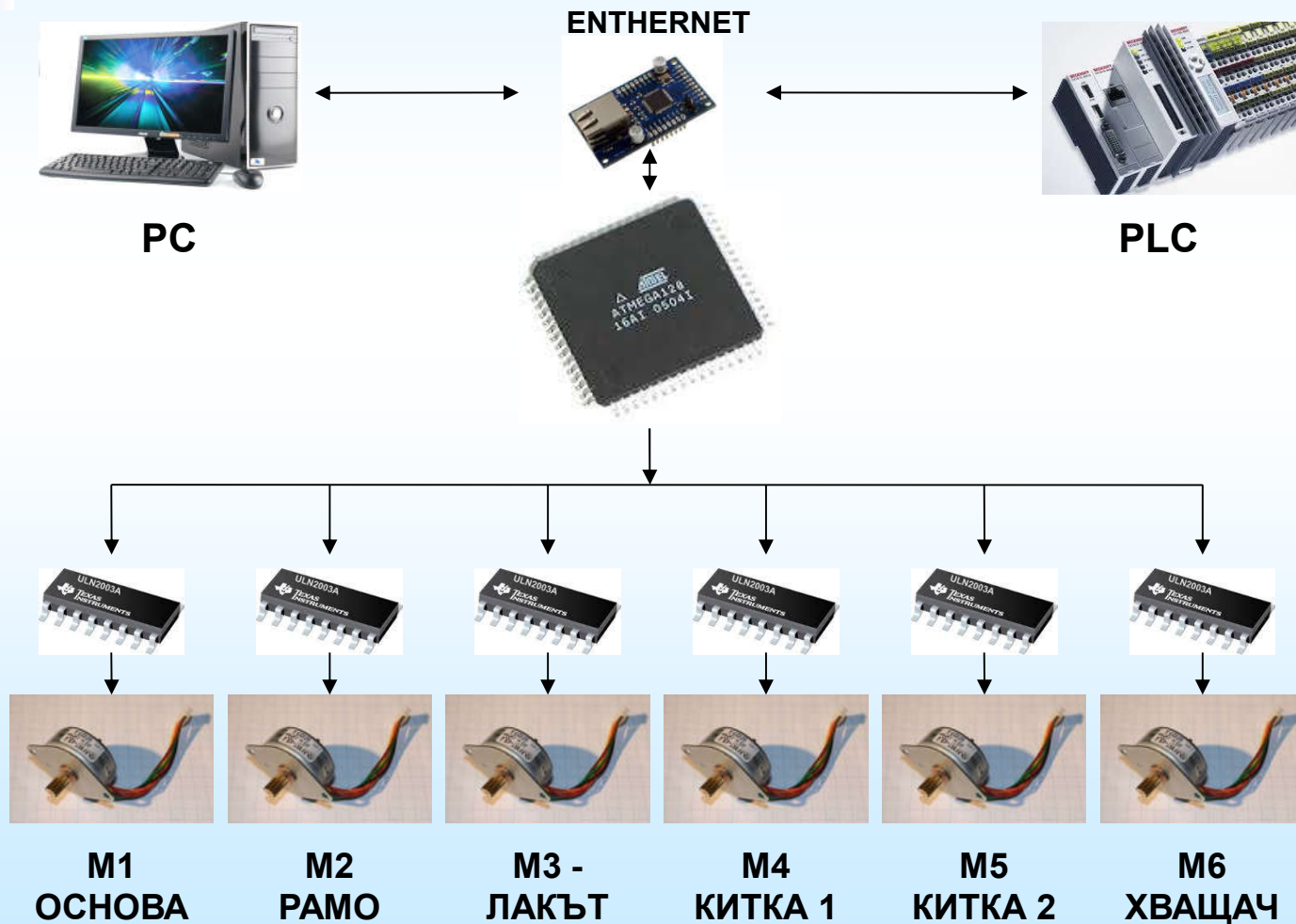


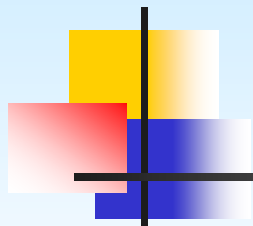


## Управляваща платка - общ изглед



# Управляваща платка – структурна диаграма



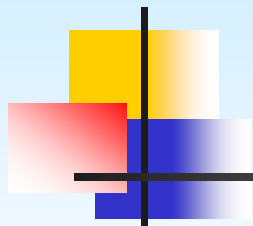


## Потребителски софтуер

---

- Графичен софтуер за персонален компютър (PC).
- Комуникира с робота чрез **ModBUS RTU / TCP**
- Осигурява възможност за управление в ръчен и програмен режим режим
- Интерпретира програма, съставена от **G кодове**





# Диалогов прозорец на РС софтуера

ROBKO 01 - ModBUS Control Center

Connection ROBKO 01 Help

Inverse Kinematics

X:  Y:  Z:

P:  R:  G:

G01 (PC) G01 Set Run

Forward Kinematics

X: Static Y: Static Z: Static

P: Static R: Static G: Static

Motor 1

Forward

Run

Motor 2

Forward

Run

Motor 3

Forward

Run

Motor 4

Forward

Run

Motor 5

Forward

Run

Motor 6

Forward

Run

Edit G Code:

```
G01 Z71.000  
  
G01 X350.036 Y53.528  
G01 Z50.280000 F150.0  
G01 X362.880 Y20.500 F150.0  
G01 X339.027 Y20.500 F150.0  
G01 X350.036 Y53.528 F150.0  
G01 Z71.000  
  
G01 X332.712 Y1.555  
G01 Z50.280000 F150.0
```

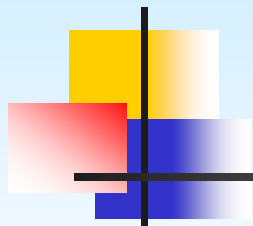
Run G Code

Run All

EMS

Static

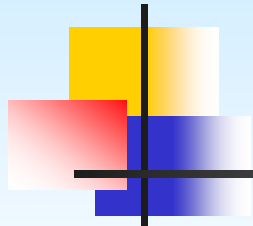




## Управляващ софтуер

---

- Изпълнява се от **8 битов** микроконтролер
- Комуникация с **PLC** или **PC**
- Формира движенията на работа в пространството на базата подадени команди
- Генерира управляващи импулси за стъпковите мотори



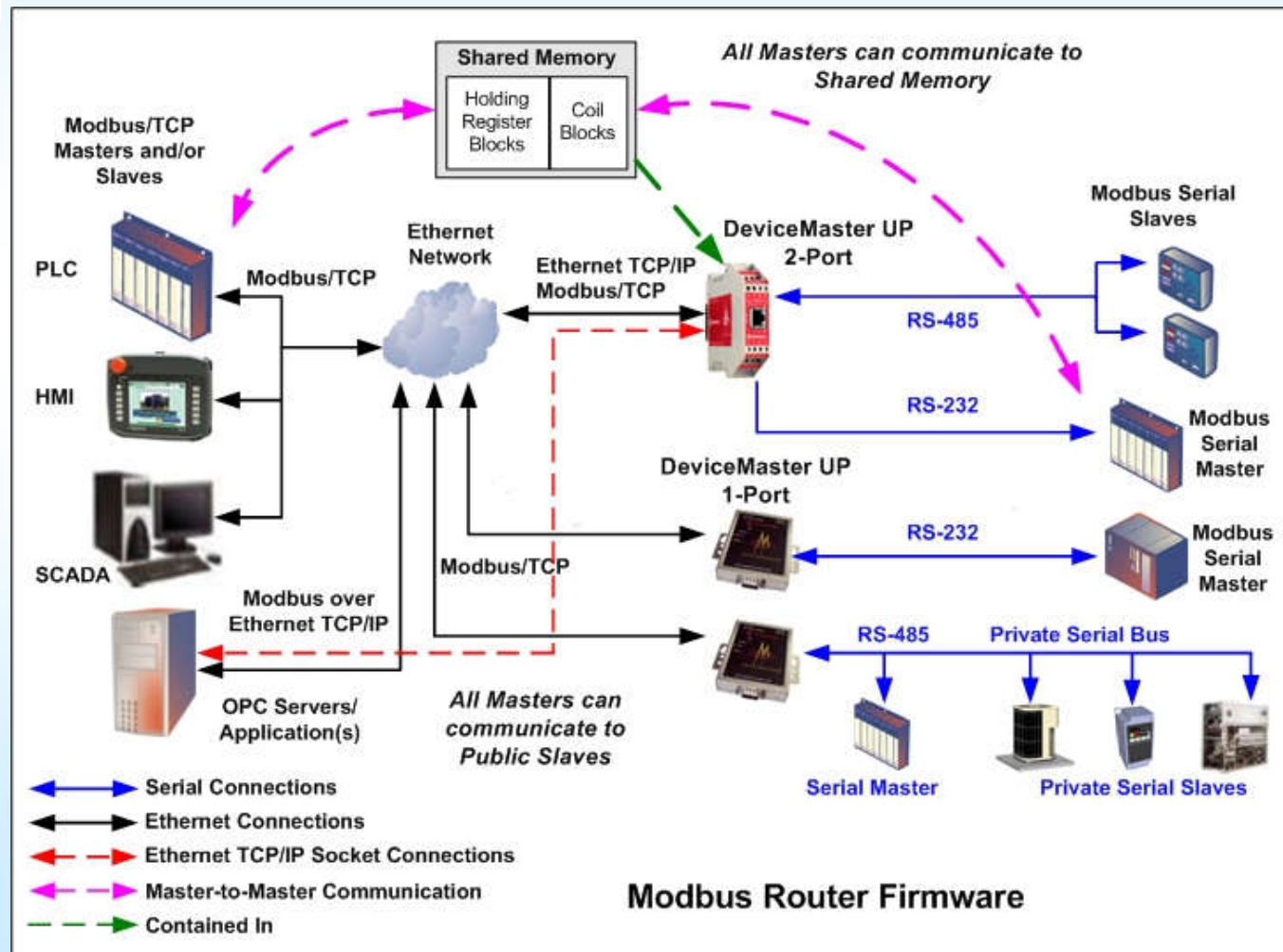
## Софтуер – поглед от ниско ниво

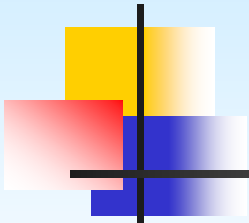
---

- Софтуера на работа е съставен от три основни нива:
  1. Комуникация – **ModBUS** протокол
  2. Интерполация – движение по права
  3. Права и обратна кинематика:
    - Права кинематика: задава се позиция и се изчисляват ъглите на ставите
    - Обратна кинематика: по текущите ъгли на ставите се изчислява позицията на манипулатора в пространството

# Комуникация – ModBUS TCP/RTU

## Общ преглед



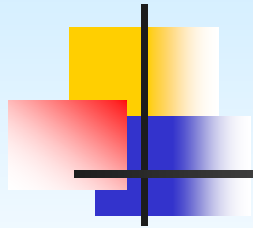


# Комуникация – ModBUS TCP/RTU

## Протоколно ниво

---

- **ModBUS** е базиран на **master – slave** концепция
- **Master** – изпраща съобщение към подчиненото устройство и очаква отговор
- **Slave** – връща отговор към главното устройство
- **Предимства:** лесен за реализация протокол; стандарт в промишлената автоматизация
- **Недостатъци:** липсва **point to point** комуникация; относително бавен



# Интерполация

## Движение по права линия

```
float a = 0.0f, delta = 0.001f; // Подготовка за движение по права. Задава се минимална дискрета.
```

```
// dx, dy, dz, dP, dR, Gn – линейни и ъглови изменения в позицията на отделните стави
```

```
while( a <= ( 1.0f + delta ) ) { // Обхожда се правата в интервала 0.0f до 1.0f (единичен вектор a )
```

```
    float Xn = X1 + a * dx; // dx = X1 – X0; - Движение по X
```

```
    float Yn = Y1 + a * dy; // dy = Y1 – Y0; - Движение по Y
```

```
    float Zn = Z1 + a * dz; // dz = Z1 – Z0; - Движение по Z
```

```
    float Pn = P1 + a * dP; // dP = P1 – P0; - Ъгъл на китката спрямо вертикалата
```

```
    float Rn = R1 + a * dR; // dR = R1 – R0; - Движение по X
```

```
    float Gn = G1 + a * dG; // dG = G1 – G0; - Отваряне/затваряне на хващача
```

```
int motor_pos_in_step[6];
```

```
Robko01_Inverse_Kinematics(
```

```
    Xn, Yn, Zn,
```

```
    (float)Pn,
```

```
    (float)Rn,
```

```
    (float)Gn,
```

```
    0.0f,
```

```
    motor_pos_in_step
```

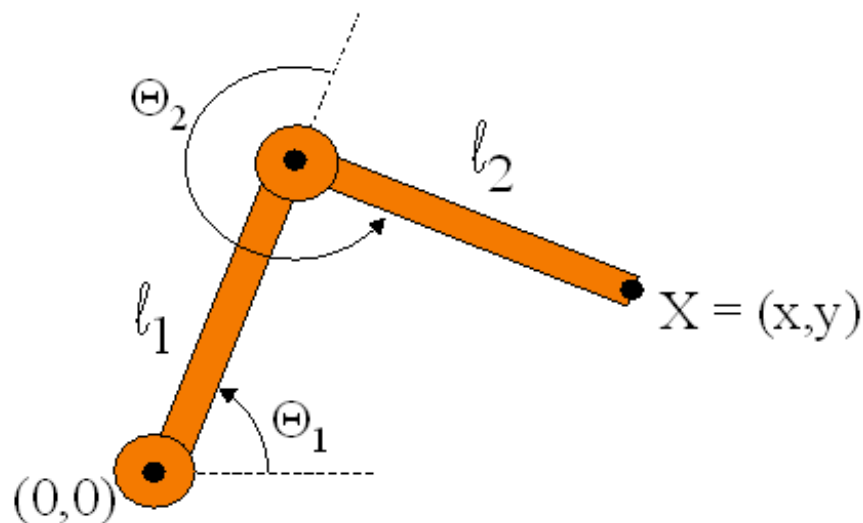
```
);
```

```
    a = a + delta; // Нова позиция в единичен вектор a
```

```
}
```

# Права кинематика

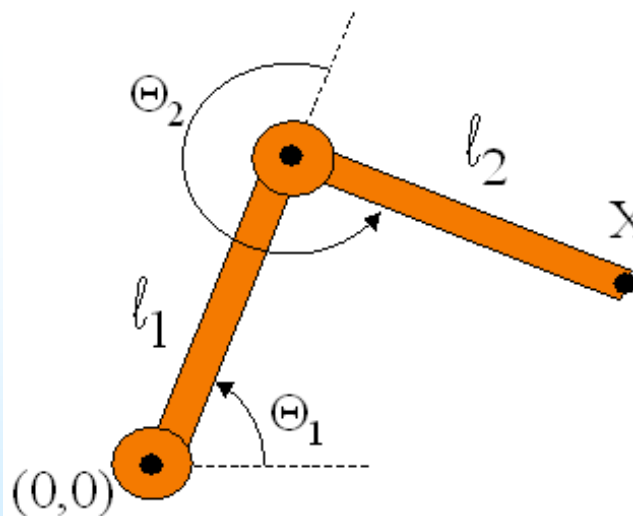
- Интересува ни позицията  $X = (x, y)$
- На базата на ъглите  $\Theta_n$  компютъра пресмята координатите  $(x, y)$  на  $X$



$$X = (l_1 \cos \Theta_1 + l_2 \cos(\Theta_1 + \Theta_2), l_1 \sin \Theta_1 + l_2 \sin(\Theta_1 + \Theta_2))$$

# Обратна кинематика

- Интересуват ни позицията ъглите  $\Theta_2$  и  $\Theta_1$
- На базата на позиция  $X = (x, y)$  компютъра пресмята ъглите  $\Theta_2$  и  $\Theta_1$



$$\Theta_2 = \cos^{-1} \left( \frac{x^2 + y^2 - l_1^2 - l_2^2}{2l_1l_2} \right)$$

$$\Theta_1 = \frac{-(l_2 \sin(\Theta_2)x + (l_1 + l_2 \cos(\Theta_2))y)}{(l_2 \sin(\Theta_2))y + (l_1 + l_2 \cos(\Theta_2))x}$$





**Благодаря за вниманието!**

---